

CNLS **(Complex Nonlinear Least Squares)**

**Immittance, Inversion, and Simulation Fitting
Programs
for
WINDOWS and MS-DOS**

LEVM / LEVMW Manual

Version 8.12
(June 2013)

Ross Macdonald

Copyright: James Ross Macdonald

PRE-PREFACE

NEW WINDOWS VERSION OF LEVM: LEVMW

A. General

The free LEVM computer program for complex nonlinear-least-squares data fitting, simulation, and inversion, has now been updated, extended, and corrected since the issuance of Version 7.11 in 1999 and earlier versions. Of most importance is the addition of a powerful Windows front end, developed by Andriy Gorkovenko of Kiev, UA, working closely with me. The newest Windows version of LEVM, V.8.11: LEVMW, includes all of the original LEVM program, now brought up to date and itself still completely operable, as in the past, using MS-DOS. *In the following, I shall often use LEVM to mean either LEVMW or the original LEVM program.*

The wide generality of the original LEVM program has been maintained and coupled with ease of operation using Windows; the result is the new LEVMW package. It involves LEVMWIN.EXE, a powerful fitting engine; and the front-end graphic interface programs, LEVMRUN.EXE, a powerful and comprehensive plotting program for LEVMW outputs and residuals, and LEVMVIEW.EXE, as well as a detailed help file, LEVM.HLP. LEVMW works directly with any Windows operating system from W-95 to W-7 and beyond and is integrated with the present LEVM manual, now updated and corrected. Corrections to the manual are now usually indicated by ** at the beginning of changed or added parts. Note that some of the manual and files apply only to the MS-DOS LEVM version and are unneeded by LEVMW.

For those new to LEVM, note that it uses a specific type of input file containing a top part listing fitting choices, followed by a bottom part containing three data columns (see the AZC example on p. 1-4 of the manual and two how-to input-file-preparation descriptions, separately included as read files). The data files may contain either frequencies and the complex parts of one of the four immittances (admittance, impedance, complex modulus, and complex capacitance/dielectric constant), or time and temporal response.

The input file must specify which one of the many available circuits of the program are to be used for fitting or simulation. These circuits are shown graphically and are described in detail in Section 5 of the manual. In addition, they may be viewed directly from LEVMW. A

powerful feature of these circuits is that although they each often include up to a maximum of 42 possible parameters, only those parameters that are given non-zero values in the input file are used in fitting, thus providing many thousands of possible specific fitting structures.

Further, in addition to allowing circuit elements to be taken as resistances and capacitances (or resistivities and dielectric constants), LEVM provides **37** distinct distributed complex circuit elements (DCEs or DEs), such as the constant-phase-element (CPE), to be used in the circuits, thus greatly expanding their generality. See the list on p. 4-2 of the manual. In addition, the LEVMW DE-examples choice in the Help facility provides short on-line descriptions of each of the DCEs, as well as identifications of LEVM input files using them. These example files may be run directly there. Once a fitting circuit has been selected, one must enter initial values for those of its free parameter to be used for the fit, but very rough choices are generally sufficient.

Besides the information in this manual, the website: <http://jrossmacdonald.com> contains much related information, including a listing of all of Macdonald's scientific and other publications, nearly all of them available for downloading in pdf format.

B. Some details

There are two compressed folders available for downloading, LEVMMANUAL.PDF and LEVMWL.ZIP. The LEVMW/LEVM compressed folder, LEVMWL.ZIP, should be unzipped into a folder with a name of your choice. It will contain README and other files and folders named LEVMW, LEVMMISC, and FITTESTS, some of which contain subfolders.

The LEVMW folder contains all the LEVMW.EXE files mentioned above and help files; the LEVMMISC one contains subdirectories and files used in previous versions of LEVM; and FITTESTS, replacing the original LEVM subdirectory TESTS, contains fifteen circuit CKT.X subfolders, where X is A, B, ...T. These individual folders each contain a typical top-part file for the circuit X and usually several full test files for that circuit, ready to be run. They are included in order to provide guidance for preparing an input file using your data and that circuit. Circuits A, D, and E are particularly appropriate for electrochemical data, while H, K, O, and R are of particular value for solid materials such as glasses, disordered materials, single-crystals, and some polymer and liquid materials.

Note that for Windows operation, it is usually sufficient to have available just the files of the LEVMW and FITTESTS folders, while for MS-DOS operation one should use LEVM.EXE and the files in LEVMMISC and FITTESTS. The LEVMMISC folder contains the .FOR program files that may be used to recompile or recompile the main LEVM MS-DOS executable file LEVM.EXE or the Windows version, LEVMWIN.EXE. Two empty subroutines, SSUB and TSUB, for circuits S and T, are provided to allow the user to include user-written Fortran programs if desired. Note that the large LEVMWIN.EXE file has been compiled using Compaq Visual Fortran.

** Finally, note that it is useful to set LEVMW so that it will take the new estimates of the parameters of a fit model after a fit and make them available in the current fit file so that another fit can be started from the results of the first one. When starting a run, go to File at the top of a run file and click Options. Then ensure that the choice “Load OUTIN file after fitting stops” is checked. See the discussion in Section E on p. 0-5 of this manual.

C. Starting and running LEVMW

An initial test of LEVM or LEVMW may be made using the AZC full input file contained in the CKT.A folder. A discussion of such a test and its results appears in Section 1 of the manual. Particularly for MS-DOS operation, the Preface should also be read. For either type of operation, it is essential that the reader familiarize him/herself with the descriptions of the top three lines of the input file and the fitting choices described therein. See especially pp. 3-3 through 3-13 of the manual. Also read the README.TXT files before continuing.

For running AZC under LEVMW, invoke LEVMRUN, leading to the main Window of LEVMW. Although full instructions appear in the HELP files that are accessible from this Window, a brief outline is provided here. At the top line of the page click File, then its subitem Load TST. Move to the FITTESTS folder and then to CKT.A. Select Open and then select AZC. Again select Open. All of the parameter values of the top part of the AZC file will now appear in the main page. On this page, push the button Save and Run. The program will run and some of its final output will appear in a console window. Push the Enter key to leave the console window and return to the main page.

If the choice Load OUTIN file after fitting stops, which appears in the File, Options

menu, is selected (this is the default and the usual choice), then the main page will show the converged parameter values of the fit, and the fit file will be named OUTIN. This full input file may be accessed, edited, and saved with its name changed, if desired, by activating the Advanced Editor choice, but saving with a different name and changes in the top part may also be made directly on the main page. Finally, one may select Run on the top line of the page and then Launch LEVMVIEW to show graphics output.

D. Making an input file

Input file preparation in MS-DOS is described particularly on pp. 1-17 and 1-18. I recommend that one combine the top part of a file and the bottom data-only part using the batch file ONEMKE.BAT. Another choice, which combines a top part of a full file with the bottom part of another full file is to use CONLFL.BAT described on p. 7-3.

Preparation of LEVMW input files in Windows is very easy and straightforward. Here is an example. Invoke LEVMRUN and select File, Load TMP. Then navigate to FITTESTS and then CKT.A. Select AZC.T, the top part of the AZC input file. Then select Data on the top line of the main page and click on Load data from ASCII. Next select AZCDAT.B, the bottom part of the full AZC file, and then Open. The two parts will be combined and the main page will then contain a full AZC input file that can then be run. There are several other important choices accessed from Data. For example, LEVMW allows one to use any one of several impedance data sets obtained directly from a Solartron FRA measurement file for combination with a top-part LEVMW fitting input file without the need for any user manipulation or separate data transformation.

E. Frequency types and IRE choices

It is nearly always appropriate to set LEVM so that at the end of a fit it makes the output file OUTIN, one that contains converged values of the free parameters of the fit, also shown in the file PNTOUTL. To invoke OUTIN, the input-choice parameter IRE must be < 0 , usually taken as -10 or -11; see below. The user input file is renamed INFL by LEVM, and LEVM fits are always made with the frequency in radians/s. Since most experimental data involves frequency values in Hz, however, it is important to ensure that the frequency columns

of both the input and output files are in Hz for such data. If the input-choice parameter FREEQ is set to 'F', then LEVM converts an input frequency column in Hz to one in radians/s before fitting, and after fitting it ensures that the frequency column in OUTIN is again in Hz. No such conversion is carried out if FREEQ is left blank.

For some theoretical and simulation work, it is convenient to work with rad/s (or time) throughout. The input value of FREEQ must be selected to be consistent with the dimensions of the frequency column of the data, and it should always be blank for temporal-domain data. The table below shows the results of various IRE and FREEQ choices, where the frequency type of the input data and of INFL are always the same. Some of the output files involved in the last column are AUXPNTL1, LINOUT, LVOUT, and LINOTRL. In contrast, the frequency type of OUTIN and AUXPNTL is independent of the value of IRE. See the table below for choices and results.

DATA TYPE	FREEQ	IRE	PLOTS and some OUTPUT FILES
Hz	F	-10	rad/s
Hz	F	-11	Hz
rad/s	---	-10	rad/s
rad/s	---	-11	Hz

F. Immittance matters

Note that although the present LEVM/LEVMW manual deals primarily with the four complex immittance quantities, impedance (Z), modulus (M), admittance (Y), and complex capacitance or dielectric constant (E), when the data are in specific form, these complex quantities become resistivity (ρ , ρ), modulus (M), conductivity (σ , σ), and dielectric constant (ϵ , ϵ). For such data, Z is replaced by ρ , etc. See, for example, pp. O-16 and 2-9.

.

Some features of CNLS Program LEVM/LEVMW

- Millions of built-in circuit possibilities available - new ones can be added
- > 10 different weighting choices are available
- > 30 different distributed circuit elements (DCE's) available for circuits
- Complex-, real-, or imaginary-part fitting is possible
- A complex-fit optimization procedure is available
- Input data may be in Z, Y, E (complex epsilon), or M form
- Fitting or simulation may be at Z, Y, E, or M level
- Input/output may be in rectangular, polar, or log-polar form
- Applicable for conductive, dielectric, or mixed systems
- Fitting of transient-response data is possible for many response models
- Powerful 2- and 3-D plotting programs included
- Subtraction of the effects of selected circuit elements possible
- Measurement-model fitting and Kronig-Kramers testing available
- Estimation of distributions is possible by powerful methods of inversion of frequency or transient response data
- Many types of conductive-system dispersion models available for fitting
- Accurate stretched-exponential (KWW) fitting models included
- New generalized-exponential-distribution fitting model available
- Ordinary or anomalous diffusion Poisson-Nernst-Planck models available

LEVM Preface

The present LEVM Version-8 manual contains the following sections:

0. Preface
1. Introduction
2. LEVM Files (Descriptions And Functions)
3. LEVM Documentation (Input Choices And Definitions)
4. Distributed Circuit Elements (DCE'S or DE'S)
5. Circuit Models (A - K, O, and R - T)
6. Amarasinghe Plot Program: CTD and INLEV Program
7. Barsoukov MKLIO Program (Prepare LEVM input file, etc.)
8. Copyright, history and other matters

PLEASE READ THE 'README' FILES, if they exist, before proceeding. If you want to test LEVM immediately, see the QUICK START heading below. Otherwise, read the manual first, especially Sections 1, 4, and 5.

The LEVM program includes four main functions. First, it may be used to fit frequency-response or transient-response data by complex nonlinear least squares (CNLS) or nonlinear least squares (NLS). It may also be used for simulation of circuit and other model response functions, and for Kronig-Kramers testing. Finally, it may be used to estimate distributions of relaxation times (DRT) or activation energies (DAE) from frequency or transient response data. All output results may be plotted in 2- or 3-D form at any immittance level. Some of the main features of LEVM are summarized on the preceding page. A list of principal **acronyms** and names, and important LEVM hints, are included at the end of this section.

Notation/Abbreviations:

In this manual, notation is not entirely consistent. The following points are worth noting. First, the name of a parameter such as C_P will sometimes be given in this form and sometimes as CP. The infinity symbol is small when used as a subscript, so, for example, R sub infinity will often be written as R_{INF} or even as RINF. The usage 5-7 indicates page 7 of Section 5. Finally, Greek letters will often be written out: e.g., ξ as Xi.

PREFACE HEADINGS:

	Page No.
A. OVERVIEW: The LEVM Program	9
B. Description of the Contents of the Distribution Disks	11
C. Quick Start	11
D. Preliminary Information on Running the Program	12
E. Graphics Information	13

F. Initial Weighting Information	
General	13
Frequency response analyzer (FRA) weighting:	14
G. Subtraction Possibilities	15
H. Additional Information	
Analysis of high-resistivity materials and dielectrics	16
Miscellaneous matters	17
I. APPENDIX 1: Conductive and Dielectric System Dispersion	18
J. APPENDIX 2: Principal Acronyms and Names	21
K. APPENDIX 3: Tips and Fitting Hints	22

A. OVERVIEW: The LEVM Program

This preface contains some important introductory information concerning the LEVM complex nonlinear least squares (CNLS) fitting program. See the first pages of the manual (0-2 through 0-8) for initial discussion of the LEVM/LEVMW windows version of the program.

Much of the following material is explicitly concerned with earlier versions of LEVM rather than directly with LEVMW and some of it may thus be outdated. Nevertheless, a lot of the material in this Chapter is directly relevant to LEVMW as well, so it should all be read. For initial information on LEVMW, see pp. 0-2 to 0-6 of the manual, where downloaded files are identified.

The LEVM program runs on a PC or clone under MS-DOS, but since all source code is provided, it may be compiled for any other computer. Here, the program has been compiled using the Microsoft FORTRAN Powerstation V.1 compiler, yielding the executable file LEVM.EXE, supplied. It has also been compiled on V. 4.0, but the result only runs on Windows 95 and is thus not included. Very little free lower memory is required for execution of the program.

LEVM.EXE presently allows a maximum of 1001 data values and up to 42 free parameters. It is a simple matter to alter the LEVM source program code to provide different limits; one merely needs to edit the file SIZE.INC before recompiling. LEVM.EXE only runs on a PC that has some extended memory. Its limits are therefore only set by the amount of such memory available. PCs with several MB of RAM or more, as necessary for Windows or Windows95, are appropriate. LEVM.EXE requires the presence, in its subdirectory or path, of the MS-DOS extender file DOSXMSF.EXE. This is supplied with LEVM. With a 133MHz Pentium machine having 16MB of extended memory, there is no problem in running the present version of LEVM.EXE (with a maximum of 1001 data values) directly in DOS, in Windows 3.11, or in Windows95, provided the CONFIG file contains the line: FILES=40 (or more). For Windows95, it is best to run LEVM.EXE directly in Windows95 MS-DOS mode rather than in a MS-DOS Prompt box. LEVM.EXE establishes a stub of about 20KB in lower memory and requires at least 3MB of extended memory for proper operation, but less is required if it is compiled for fewer possible data points.

A simple test whether LEVM.EXE will work on your machine is to make a bootable floppy disk with a CONFIG.SYS file containing only: FILES=40. Copy the LEVM.EXE file to this disk, as well as the files DOSXMSF.EXE, AZC, and RNL.BAT (all these files are in Subdirectory TESTRUN). Then issue the command RNL AZC, and the program should begin and soon converge. Incidentally, for timing benchmark results see the input file OTSTIME.CD. The larger the number of data values and the larger the number of free parameters in a fit, the longer it will take for the program to converge to a final least squares solution.

The LEVM program code has been compiled with two different UNIX machines, with LAHEY FORTRAN, and with FORTRAN compilations running in the protected mode of a 386, 486, or Pentium machine. Although a numerical co-processor is not necessary, it is strongly recommended. Recompile may require changing the .FOR extensions to .F. Then, one should have no trouble using LEVM, after compilation, for any such machines. It has also been recompiled and run on Macintosh machines. There, it has been found appropriate to use in the MAKE file for LS FORTRAN the commands FmakeOptions =-opt=1 -u -r -i2 -saveall and FLinkOptions = -w -f -srt -ad 4. Note that with NDP FORTRAN, a UNIX machine, or with the Microsoft FORTRAN Powerstation, one avoids the 640KB limitation of DOS; this is the situation for the present LEVM.EXE file.

Please note that the current version of LEVM is much changed and generalized as compared to earlier versions (see the LCHANGES part of Section 8 of the Manual). For this reason, some input files used for earlier versions of LEVM and LEVM-OLSON may need to be modified for use with the present version. A unique and powerful feature of LEVM is its ability to determine parameters of the error variance model, i.e. weighting parameters, dynamically - using the data themselves.

As you will see from studying the manual, LEVM incorporates a large number of choices, contributing greatly to its flexibility and power. But it can also be used very easily and straightforwardly in even minimal mode, (e.g., change the AZC input test file to N=32 and IRE=0) (see Section 1). It allows you to fit complex, real, or imaginary frequency response data and, in addition, many mathematical functions (real data - see Circuit G in Section 5), including several popular transient response functions. It allows many weighting possibilities to be used in fitting, in order to obtain best parameter estimates, and it provides much statistics concerned with a fit.

A special strength of LEVM is its easy selection of a fitting model (see Section 5). Any smaller piece of the circuits provided can be used to construct the model and the scope of such a model is greatly extended by the provision of more than 30 different distributed elements (DE's) which can be used in the fitting circuit.

You will find details of using LEVM in the various sections of this manual. In particular, it will be very helpful for you to read the descriptions in Section 2 of the various files of the LEVM package in order to learn some of the many potentialities of LEVM.

NOTE: In addition, a short discussion of LEVM tips and suggestions now appears at the end of this section. It will repay reading.

B. Description of the Contents of the Distribution Disks: LEVM/LEVMW

Some of the following material is explicitly concerned with earlier versions of LEVM rather than directly with LEVMW. For explicit information on the latter, see pp. 0-2 to 0-6 of the manual, where downloaded files are identified.

****** See pp. 0-21 and 0-22 for acronyms used herein, for the procedure for naming composite fitting models, and for a list of important individual models with references. The important **MEISP** IS computer model, an extension of LEVMW, is briefly described on p.8-1 and may be downloaded from there.

Note that the **Randles** test files mentioned below are now present in the new, enlarged LEVMW folder of input file examples, FITTESTS; open the Ckt.B and Ckt. H sections of the FITTESTS folder to access them

Relevant LEVM file folders are: LEVMMISC and FITTESTS. The LEVMMISC folder contains: LVFOR, TESTRUN, LEVMISC, CTDPLOT, AND MKLIO.

LVFOR contains the FORTRAN source code for the LEVM program itself and a few related files. LEVMISC contains many ancillary executable and batch files. TESTRUN contains all the files needed to verify that LEVM will run and yield correct answers on the installed machine, including LEVM.EXE. Finally, FITTESTS contains files, such as BTMP, which may be used in constructing input files to run LEVM, and complete input files, such as JTST, which illustrate the use of all the 15 basic circuit models of LEVM (see Section 5).

****** Note particularly the inclusion of some RANDES test files. They are included because Randles behavior: electrode-reaction, double-layer and possibly diffusion effects, are particularly common in electrochemical Impedance Spectroscopy response. They thus may be helpful to the user faced with fitting data of this character. See Figs. 4.3.17 on p. 308 and 4.5.7 on p. 446 of the IS book (cited here as Ref. 49, on p.1-25) for Randles equivalent circuits.

****** To ensure that Randles circuit fitting includes ordinary or anomalous Warburg response (see pp. 101-102 of Ref. 49 (cited on p.1-25 herein and the discussion on pp. 4-9 and 4-10), one sets the distributed element (DE) NELEM (p. 4-2) parameter to the fixed value of 9 and the UDE1=P(8) of the DE1 element of Circuit B, for example, to 1 fixed, and its P(18) parameter to the fixed value of zero. Other, often more physically appropriate, diffusion-model possibilities are discussed on pp. 4-9 and 4-10, and in Circuit H on p. 5-15 and following pages.

Section 6 of the manual is devoted to information and instructions for using the files contained in CTDPLOT for 2- and 3-D plotting, including the executable file INLEV which may be used for preparing and modifying LEVM input files. But the MKLIO files of Section 7 are even more helpful in preparing full LEVM input files and transforming them. Initially see p. 1-7 of the Manual for discussion of the uses of several MKLIO files.

C. Quick Start

An important first step in using LEVM is verifying that it will work correctly on your machine. To do so, first transfer TESTRUN to a new folder, Then, read the discussion in Section 1 on Verification of the program using the AZC input file. Follow the instructions there. If there are no problems, run your own LEVM input file (see the discussions of input file preparation in Sections 1 and 7 and in

Part D of the present section).

You will only be able to get the most out of LEVM if you devote some time to familiarizing yourself with the manual and with the many options and possibilities inherent in LEVM and in its ancillary programs. But for many applications only a few of the options need be considered. There are really only two main points to keep in mind: (a) your data must be combined with a LEVM-type input header file to make a fully executable input file (see examples in Section 1, Part c), and (b) you must choose an appropriate fitting circuit/model from the possibilities described in Section 5; this circuit may involve one or more of the distributed circuit elements described in Section 4.

Preliminary Information on Running the Program

To use LEVM for CNLS fitting, one must have available an input file containing two parts (see the discussions in Section 1). The top (header) part of the input file (10 to 13 lines which are described in detail in Section 3; see also the last few pages in Section 1) is used to set choices which determine how LEVM analyzes the data, and the bottom part contains the data themselves.

A very important choice is the selection of a fitting circuit to represent the data (see the introduction to Section 5). The program is purposely not fully interactive; thus the input file must be created before invoking LEVM, although some changes in it may be made during operation of LEVM. Since most of the extensive possible modes of operation of LEVM are selected by the choices made in the header lines of the input file (especially the choices of lines 2 and 3), the discussion in Section 3 associated with each input line should be carefully examined. Some program possibilities are only discussed there and nowhere else.

The process of producing an input file (ASCII form) has been partly automated by means of batch and other programs (see GINPUT, DINPUT, and MKINP in Section 2, INLEV in Section 1, and MKLIO in Section 7). In addition, see the specific examples included on p. 1-7. Your data should be arranged in three columns: frequency, real part, imaginary part, (or frequency, magnitude and phase), but see GINPUT for greater flexibility. The numbers in a data line may be separated by spaces or commas and their positioning is unimportant. By the use of MKINP or MKLIO, the top part of an input file and your data are combined into the standard input-file form such as that of AZC. If only real data are to be fit, they should appear in the second column (with zeros in the third column), but either real or imaginary fits can be carried out separately with fully complex data when fitting of both the real and the imaginary parts simultaneously (the usual approach) is undesirable.

When data are input into LEVM, the form of the data (i.e. impedance, admittance, etc.) as they appear in the input file can be altered by an input choice so that the actual fitting will be carried out at any of the four immittance levels. See discussion of input line 2 of Section 3. In addition, fitting can be carried out for data in either polar or rectangular form. All data transformations and fitting types are possible. Also, as mentioned above, LEVM allows the direct nonlinear least squares (NLLS) fitting of real or imaginary parts of a data set separately as well as combined (CNLS). It includes code for fitting Arrhenius temperature-dependence data (see discussion of LINOTCON in Section 2, MKLTP, RNTEMP, and especially the NELEM=33 choice in Section 4) and, as well, several important transient response functions, particularly useful for dielectric materials.

Note that although LEVM produces screen output during operation (see Section 1), this is purposely incomplete, and the main output appears in the files PNTOUTL (described in Section 1) and AUXPNTL, with only the first usually needed. The latest PNTOUTL file may be scrolled on the screen by typing RLVP; it may be printed by typing PLV. A particularly useful optional output file is OUTIN (see Section 2). It is recommended that it be used extensively, especially in initial fitting work and in simulation studies.

E. Graphics Information

A very useful addition to LEVM is the provision of the general and versatile menu-driven 2- and 3-D plotting program, CTD.EXE, written by Dr. S. Amarasinghe (see Section 6). With a color monitor, its results are in color; it can carry out all needed data transformations; and it can even continuously rotate 3-D plots if so desired. CTD can plot both direct LEVM output of any type and, as well, any other appropriate data file. To use CTD with LEVM, place it and the enclosed file EGAVGA.BGI in the same subdirectory that contains the executable LEVM files.

Note that although one can use the CPTRAN program to transform data and fit results (see also the discussion in Section 3 of the choices $MAXFEV < 3$) and then plot the results with CTD, it is also possible, given a set of complex data in any of 16 modes, to use CTD directly to transform the data to any other of the 16 modes, save the results in a file, and plot them directly. See also the important MK4LF.EXE program.

F. Initial Weighting Information

General

Great weighting flexibility is a very important feature of LEVM. I give a brief description of LEVM weighting here, with more details in Section 1. The main types of weighting are selected by giving the input parameter IRCH (see Section 3 definitions and explications) a value between 0 and 6. But for $|IRCH| > 1$, positive IRCH values lead to weighting that involves the data values, and negative IRCH values to weighting that involves the model, i.e., calculated values of the fitting function. This very important type of weighting may also be selected by setting the input parameter IFP non-zero (weighting involving the model then occurs for a second fit, one following immediately after the first one, that involving the value of IRCH selected).

Thus, one can easily produce two sequential fits, with, for example, the weighting of the first involving the data (e.g. $IRCH = 2$) and that of the second the model (as if $IRCH = -2$). Two other sequential fits involving different weighting are selected by setting $IRE > 0$ or $IOPT > 0$. By simultaneous use of several of these choices, a LEVM run can involve up to three separate, sequential fits, each with different weighting.

Also very important in weighting when $|IRCH| > 1$ are the error variance model parameters $U=P(31)$, and $Xi=P(32)$. They may both be fixed, but when one or both of these parameters are free to vary, one has dynamic weighting, where, during least squares fitting the data themselves determine the most appropriate weighting.

Choosing $IRCH = 3$ (only appropriate for data in rectangular form) leads to modulus weighting (MWT) when Xi is fixed at 1 and $U=0$. That is, the uncertainty for both the real and the imaginary parts of each data point is equal to the square root of the sum of the squares of the real and imaginary parts for that point. I do not recommend it (see comparison of weighting results in Section 1).

Besides generally leading to worse fitting parameter estimates than $|IRCH| = 2$, it can be particularly confusing because it leads to an estimate of the overall standard deviation of the fit ($S_F = SIGMAF$), a measure of the goodness of fit, lower than the proper value, which is well estimated when weighting with $|IRCH| = 2$ is used. Thus, it appears to give a better fit, when, in fact, this is not the case.

The two weightings I believe to be most useful and appropriate under many conditions are unit weighting (UWT), produced by the choice $IRCH = 1$, and proportional weighting: $|IRCH| = 2$ (with $X_i=1$ fixed and $U=0$). But dynamic weighting, a new function of LEVM (see Sections 1 and 3), includes these and many more possibilities. Further, it makes fitting less subjective since the data themselves determine the most appropriate weighting.

Unit weighting is in effect no weighting: all data are weighted equally. This weighting is produced by choosing $IRCH=1$, and is most appropriate when the data range (Max/Min) is limited to the order of 1 to 8 or less - not often the case in the Impedance Spectroscopy (IS) field.

Proportional weighting means that the uncertainty of the real component of the data is proportional to its magnitude, and the uncertainty of the imaginary component is proportional to its magnitude. This weighting, again involving $X_i=1$, fixed, and $U=0$, is produced by choosing $IRCH=+2$ (PWT) or -2 (FPWT).

Especially important, I believe, is **function**, rather than **data** weighting, i.e. **FPWT** rather than **PWT**. See the discussion of this weighting in the papers listed under the heading of "Weighting" in the reference list which will be found at the end of Section 1. Of particular value should be the paper marked "Reference A" in this list. Also see Sections 1 and 3 of the present manual for further discussion of these and other important weighting choices.

Note that OPWT, for example, means that PWT fitting results have been optimized by setting the input parameter $IOPT > 0$ (say 9) and thus invoking a built-in weighting optimization routine (not the same as dynamic weighting but both may be used during the same fit). Optimization can be a very helpful aid in achieving the best possible parameter estimates from the data, the true aim of CNLS fitting.

Frequency Response Analyzer (FRA) Weighting:

There are three separate weighting choices available in LEVM which are particularly tailored for use with data obtained using a FRA. The first is invoked by $|IRCH| = 4$, and its provenance is discussed in the article by Spinolo et al. in the reference list at the end of Section 1. This weighting may be less appropriate than those which follow from the choices $|IRCH| = 5$ and 6. The user may want to carry out an $IRCH = 1$ or $|IRCH| = 2$ fit (with $IRE < -9$) before using $|IRCH| = 5$ or 6, but the latter have been particularly constructed for FRA-data situations and are likely to be more appropriate (for more information and limitations, see Section 3-9 and the paper by Orazem et al. under "Weighting" in the reference list).

G. Subtraction Possibilities

Do not follow the suggestions here until you are well familiar with both input and output LEVM files and operations. LEVM now allows one to carry out several kinds of (possibly) useful subtraction effects. For specificity, assume that a good fit of some data has been achieved using the parameters CP, RP, R1, C2, R2, and DE2 of Circuit B (Section 5). Two subtraction approaches, A and B below, will now be described. Assume that the fitting was carried out with IRE=-11, so that the file OUTIN has been made.

- A. First set MAXFEV=0 in OUTIN and do RNL OUTIN. The result will be a file whose "data" is just that which follows exactly from the parameters of the model and their interconnections. Now suppose that one wants to see what sort of response is obtained if one (or more) of these parameters is/are removed from the circuit. To do so, just set the value of the desired parameter(s) in Lines 4 (Section 3) of OUTIN to zero, set MAXFEV=0 again and do RNL OUTIN or, equivalently, RNLO. The resulting OUTIN file can then be run with RNL OUTIN, and the result shown on the screen or plotted.
- B. Again start with the OUTIN file made by the original successful fit. Read the discussion of MAXFEV=2 in Section 3 and follow the procedure outlined there. But notice that with this method, the effect of a given parameter is subtracted from the actual input data itself, not from data calculated from the fit as in A.

Subtraction of the present type must be carried out with care and circumspection. Suppose that one wants to subtract the effects of CP and RP from the original data. Such subtraction must be accomplished during a run at the Y level, DFIT=Y. So set DFIT=Y, MAXFEV=2, and set NFREE=3 for both P(28) and P(29) - positions 28 and 29 in the NFREE line. Then do RNL OUTIN. The OUTIN result will have the input values of RP and CP set to zero and their effect subtracted from the data. Further, if desired, one can then do RNL OUTIN again (to produce an AUXPNTL file) to see how good the fit will be and to examine the results with CTD.

The above process can, of course, be continued until the effects of all the parameters have been removed from the original data and only the residuals (random and possibly systematic errors) are left. For example, to subtract the effect of R1 from the result obtained above, one would next need to set DFIT=Z, NFREE=3 for P(1), MAXFEV=2, and do RNL OUTIN. Next, to eliminate the effect of C2, one would have to follow the above procedure with DFIT=Y. Then, with DFIT=Z one could eliminate R2 or DE2 or both. As far as parameter estimation is concerned, the parameter values obtained before subtraction will generally be more accurate than those obtained after one or more subtractions of the present kind.

Another subtraction possibility is operative only for the O Circuit.

H. Additional Information

Analysis of high-resistivity materials and dielectrics

Earlier versions of LEVM were designed to be particularly appropriate for analyzing data from electrochemical impedance measurements, with emphasis on electrode reaction, adsorption, and diffusion effects. The present version retains all such capabilities, but, in addition, it now includes a greatly extended ability to analyze bulk dispersion effects, required for the adequate interpretation of dielectric and very-high-resistivity data. The present version of LEVM is thus an **immittance**, rather than impedance, spectroscopy program.

LEVM has special features which make the analysis of high-resistivity data, such as those arising from disordered materials, glasses, polymers, and dielectrics, particularly simple. An important distinction is that between purely dielectric dispersion response data (dielectric-system dispersion: DSD) for which DC conductivity is negligible (or at least unconnected to the dielectric response), and conductive-system dispersion, CSD, where the DC conductivity is just the low-frequency limiting value of the real part of the complex conductivity associated with the dominant dispersion process. These matters are discussed in Appendix 1 of this Section and in detail in Refs. 40-43 listed at the end of the INTRODUCTION section of this manual.

Analysis is often carried out with the data in specific form. Let $CC = A/L$, the cell constant for a sample with plane, parallel electrodes of effective area A and separation L . Define the dielectric permittivity of vacuum as ϵ_V , so that the empty cell capacitance is $C_c = CC * \epsilon_V$. (See the definition of CELCAP in Section 3). Then if ρ and σ are the complex resistivity and complex conductivity, respectively, $\rho = CC * Z$ and $\sigma = Y/CC$. Finally, the complex dielectric constant, ϵ , usually written as E here, is $E = Y/(i * \omega * C_c) = \sigma/(i * \omega * \epsilon_V)$. Note that the file RNDIN.BAT may be used to transform raw data so that it is in specific form and to carry out other often-needed data transformations. In LEVM and in the Manual, **Gaussian units**, rather than SI ones, are generally used. Thus the value of ϵ_V is taken as 8.854188×10^{-14} F/cm.

The most appropriate way to analyze CSD-type data, DSD-type data, or data involving both kinds of dispersion processes in the measured frequency range, as well as possible electrode-impedance effects, is to use the O Circuit (see Section 5). But a few other possibilities are worth mentioning. Suppose that one is dealing with pure dielectric data and wishes to fit it to a distributed element response model such as that of Cole and Cole (Section 4). One could pretend that the E-level data are actually of Z (or ρ) type and fit the data to the model at the Z level. Another possibility is to use the NELEM < 0 choice described near the beginning of Section 4. But since even simple dielectric response is rarely the only contribution to the actual full response data, it is usually more appropriate to use the O Circuit, where other effects can be included and, through the power of CNLS fitting, quantified and separated.

One often uses LEVM to fit data to an appropriate response model which contains several temperature-dependent parameters. By carrying out such fitting for a range of temperatures, one can investigate the detailed temperature dependence of such parameters, such as Arrhenius response in a thermally activated situation. For further information about LEVM's capabilities in this area see the discussions under LINOTCON in Section 2, NELEM=33 in Section 4, the G-Circuit of Section 5, and EXTRLVM in Section 7.

Miscellaneous matters

If the signs of imaginary impedance data are entered as all positive, they can be changed globally by the use of the NEG parameter of the input file, line 2 (see Section 3). - The program **won't** fit properly unless this is done for this case! - Be sure to leave NEG blank, however, if the imaginary parts are entered all negative. Similarly, if the data involve frequency rather than angular frequency, one needs to tell the program this by entering F at column 11 of line 2 of the input. This must be right to get meaningful parameter estimates. It is essential for proper operation of LEVM that the actual number of frequency values in the data be entered for M, line 3 of the input file UNLESS one of the procedures discussed in Section 7, which use MKLIO, are employed.

Finally, in addition to its main output (on file PNTOUTL), LEVM produces a secondary output file AUXPNTL which contains a short version of the main output, which lists the parameter inputs and includes a list of the data, the fitted results, and the fitting residuals of each point. This output is in the proper form for inputting into the CTD graphics program. See also the discussions of the files LINOUT, LVOUT, LINOTR, and LINOTL in Section 2.

Because of the variety of transformations of the data available with the program, it has not been practical to make the headings of some of the data columns in the program outputs reflect the exact form of the data being fitted. Thus such headings do not indicate the specific type of immittance function being fitted, whether the data are in rectangular or polar form, etc., but information at the beginning of the LEVM output does indicate the exact choices used (e.g., ZR, ZM) and should be employed to interpret the output data and fitting lists.

NOTE: When you use LEVM to obtain fitting results which are to be published, please consider mentioning in your paper that they were obtained using LEVM. If you should need further help, beyond that in this manual, in using or interpreting LEVM, see the information in Section 8.

Good luck and good fitting!

Appendix 1 - Conductive and Dielectric System Dispersion

The frequency response parts of LEVM allow one to fit conductive-system dispersion data (CSD), dielectric-system dispersion data (DSD), or the combination of both within the measured frequency range. See especially the O-circuit discussion in Section 5 of this manual and Refs. 40-43 listed at the end of Section 1. Because of the great importance of these possibilities and the importance of properly making distinctions between them, pertinent definitions and analysis are provided here so that the many possible LEVM fitting choices will be fully understandable without the need to consult other sources. There are two main types of CSD fitting models, termed CSD0 ($n=0$) and CSD1 ($n=1$). In addition, the CSD1 fits may be of one of two kinds, CSD1a and CSD1b, depending on the available knowledge of the distribution of relaxation times (DRT) or activation energies (DAE) associated with the response model.

Let p_n represent all the shape parameters of a fitting model, where n distinguishes different but related models. In addition, there is always present a characteristic relaxation time (or equivalent), τ_o , which we will designate as τ_{on} . Consider now a dimensionless fitting model, $I_n(\omega, p_n)$, involving a single dispersion process described by the dimensionless DRT $G_n(x, p_n)$, where $x \equiv \tau/\tau_{on}$. Then

$$\frac{U_n(\Omega_n, p_n) - U_n(\infty, p_n)}{U_n(0, p_n) - U_n(\infty, p_n)} \equiv I_n(\Omega_n, p_n) = \int_0^\infty \frac{G_n(x, p_n) dx}{[1 + i\Omega_n x]}, \quad (1)$$

where U is a measured or model quantity of interest, and ω is the angular frequency. If we set $n = D$, then $U_D(\Omega) = \epsilon(\Omega)$, the complex dielectric constant, while for $n = C$, $U_C(\Omega) = Z(\Omega)$, the impedance, or $\rho(\Omega)$, the complex resistivity. Alternatively and hereafter, we shall set $n = 0$ for both dielectric-level DSD response and impedance-level CSD response (i.e., CSD0) involving the same formal DRT. In addition, the choice $n = 1$ will be used to denote the second type of CSD response, CSD1, one which involves a DRT that is proportional to the original CSD0 DRT multiplied by τ . It is inappropriate to use a CSD1 response model to fit DSD behavior. Although a CSD0 model may be used to fit CSD data, it is usually more appropriate to use a CSD1 one. Note that because of the relations between the four immittance levels, CSD should properly be described by a distribution of resistivity relaxation times (not conductivity relaxation times), and DSD by a distribution of dielectric relaxation times, equivalent to a distribution of conductivity relaxation times. Even when dispersion is not actually associated with a physical process involving a DRT, it may still be formally expressed in terms of one, as above.

In recent work of the author, it has been shown that a great deal of data for conducting-system glasses and single crystals is well fitted by a modified CSD1 Kohlrausch model, the K1, involving a shape parameter β_1 fixed at 1/3 (see, for example, downloadable papers #'s 228, 235, and 237 in the JRM home page, <http://jgrossmacdonald.com>).

The full response model is, from Eq. (1),

$$U_n(\Omega_n, p_n) = U_n(\infty, p_n) + \Delta U_n I_n(\Omega_n, p_n),$$

where

$$\Delta U_n = U_n(0, p_n) - U_n(\infty, p_n).$$

For CSD response, the Z-level model is thus

$$Z_n(\Omega_n, p_n) = Z_n(\infty, p_n) + \Delta Z_n I_n(\Omega_n, p_n)$$

where $Z_n(\infty, p_n)$ is often zero or entirely negligible for a dispersion response model.

Now the important question is: given knowledge of $I_0(\Omega_0, p_0)$ and possibly $G_0(x, p_0)$, what is the expression for $I_1(\Omega_1, p_1)$? Note that, in general, p_0 and p_1 here represent the same set of parameters but designate different values of them since fitting involving different, but related, DRT's will lead to different parameter estimates. First, it is useful to define the dimensionless moments of the $G_n(x)$, distribution as

$$\langle x^m \rangle_n \equiv \int_0^\infty x^m G_n(x, p_n) dx,$$

quantities which clearly depend on the shape of the distribution but not on τ_{on} . Since the distributions are taken normalized, it is necessary that

$$\langle x^0 \rangle_0 = \langle x^0 \rangle_1 = 1.$$

It follows from the definition of G_1 that

$$G_1(x, p) \equiv [x / \langle x \rangle_0] G_0(x, p), \quad (2)$$

where $p_0 = p_1 = p$ has been used. But it is worth reiterating that fits of a set of data with CSD0 and CSD1 approaches will lead to different estimated parameter values, so fit estimates of p_0 and p_1 for the same data will always differ. These results lead to

$$\langle x^m \rangle_1 = \langle x^{m+1} \rangle_0 / \langle x \rangle_0,$$

Therefore, $\langle x^{-1} \rangle_1 = 1 / \langle x \rangle_0$ when p_0 and p_1 involve the same values. Now the use of Eqs. (1) and (2) for $n = 0$ and 1 allows us to write for CSD1 response

$$I_1(\Omega, p_1) = \int_0^\infty \frac{G_1(x, p_1) dx}{[1 + i\Omega x]} = [i\Omega \langle x \rangle_0]^{-1} \int_0^\infty \frac{i\Omega x G_0(x, p_1) dx}{[1 + i\Omega x]}$$

It follows immediately, after rearrangement and using the normalization condition, that

$$I_1(\Omega, p_1) = [1 - I_0(\Omega, p_1)] / [i\Omega \langle x \rangle_0].$$

For a physically realizable model, $\text{Re}[I_1(0, p_1)]$ will equal unity as it should.

For concreteness, assume that the $I_n(\Omega, p_n)$ quantities involve specific response, i.e., the complex quantities $\rho(\omega)$ or $\epsilon(\omega)$. Consider now the following DSD (subscript D) and CSD (subscript C) complex conductivity and modulus responses:

$$\sigma(\Omega, p_0) = i\omega\epsilon_v \left[\epsilon_{D\infty} + \Delta \epsilon_D I_0(\Omega, p_0) \right] \text{ and } M(\Omega, p_n) = i\omega\epsilon_v \left[\rho_{C\infty} + \Delta\rho_C I_n(\Omega, p_n) \right]$$

where $\epsilon_{C\infty}$ and $\rho_{C\infty}$ are the CSD high-frequency limiting responses and ϵ_v is the permittivity of vacuum. Further, let us ignore any small differences between such quantities as $(\rho_{C\infty})_0$ and $(\rho_{C\infty})_1$.

Then the full responses are

$$\text{DSD} \quad \sigma(\Omega, p_0) = i\omega\epsilon_v \left[\epsilon_{D\infty} + \Delta \epsilon_D I_0(\Omega, p_0) \right]$$

$$\text{CSD0} \quad M(\Omega, p_0) = M_{C0}(\Omega, p_0) = i\omega\epsilon_v \left[\rho_{C\infty} + \Delta\rho_C I_0(\Omega, p_0) \right]$$

$$\begin{aligned} \text{CSD1a} \quad & \text{The moment } \langle x \rangle_0 = (\langle x^{-1} \rangle_1)^{-1} \text{ is known. Then} \\ & M_{C1a}(\Omega, p_1) - i\omega\epsilon_v \rho_{C\infty} = i\omega\epsilon_v \Delta\rho_C I_1(\Omega, p_1) \\ & = \epsilon_v \Delta\rho_C \langle x^{-1} \rangle_1 \left[1 - I_0(\Omega, p_1) \right] / \tau_{01} \end{aligned} \quad (3)$$

$$\begin{aligned} \text{CSD1b} \quad & \text{The moment } \langle x^{-1} \rangle_1 \text{ is unknown. Use the relation (Refs. 40, 43)} \\ & \langle x^{-1} \rangle_1 = \tau_{01} \Delta\rho_{C1} / [\epsilon_v (\epsilon_{C\infty})_1 (\rho_{C0})^2] \end{aligned} \quad (4)$$

in Eq. (3). Then,

$$M_{C1b}(\Omega, p_1) - i\omega\epsilon_v \rho_{C\infty} = (\Delta\rho_{C1} / \rho_{C01})^2 [1 - I_0(\Omega, p_1)] / (\epsilon_{C\infty})_1.$$

Note that when $\rho_{C\infty} = 0$, $\Delta\rho_C = \rho_{C0}$. For fitting with the CSD1a, the parameters are $\rho_{C\infty}$, $\Delta\rho_C$, τ_{01} , and p_1 . For the CSD1b, where we shall take $\rho_{C\infty} = 0$, the remaining parameters are τ_{01} , p_1 , and $(\epsilon_{C\infty})_1$. In CSD1b fitting, the $(\epsilon_{C\infty})_1$ parameter replaces the $\Delta\rho_C$ parameter otherwise present in the LEVM input file.

Finally, note that $\epsilon_{\infty} \equiv \epsilon_{D\infty} + \epsilon_{C\infty}$ and that $\epsilon_{D\infty}$ always exceeds unity for the usual immittance spectroscopy frequency range. In contrast, $(\epsilon_{C\infty})_1$ may range from very small to very large, as indicated by Eq. (4) when one remembers that $\langle x^{-1} \rangle_1$ is infinite for physically unrealizable distributions such as those leading to Havriliac-Negami or Cole-Cole responses without cut-off.

See also Refs. 45 – 47 (pp. 1-21 and 1-22)

Appendix 2 - Principal Acronyms, Names, and Models

a. General

BCD	Blocking-conductive distributed element
C	Subscript denoting conductive
CELCAP	Capacitance of empty measurement cell: See pp.0-16, 3-4
CD	Cole-Davidson distributed response
CNLS	Complex nonlinear least squares
CPE	Constant-phase distributed circuit element
CSD	Conductive-system dispersion
CSD0	A type of CSD response model involving the same formal DRT as DSD response
CSD1	A CSD response model involving a physically reasonable modification of a CSD0 model. There are two CSD1 types, CSD1a and CSD1b. See Appendix 1 of this section.
CV	Continuous distribution, variable (free) τ DRT fitting method
D	Subscript denoting dielectric
DCE	Distributed circuit element (same as DE)
DE	Distributed response element
DEC	Dispersive element - conductive
DED	Dispersive element - dielectric
DF	Discrete distribution, fixed (constant) τ DRT fitting method
DRT	Distribution of relaxation times
DSD	Dielectric-system dispersion
DV	Discrete distribution, variable (free) τ DRT fitting method
EDAE	Exponential distribution of activation energies
EDRT	Exponential distribution of relaxation times
ERM	Electrode response model
FPWT	Function-proportional weighting in least squares fitting; uses model values for weighting
GED	Generalized EDAE response model; includes Gaussian distributions
IS	Immittance (or impedance) spectroscopy
LEVM	The original CNLS fitting program described herein
LEVMW	The Windows oriented version of LEVM
MODE	An important fixed input parameter; see Section 3-10
NELEM	A fixed input parameter (also designated as NDE) for selecting a DE response model
NLS	Nonlinear least squares
PWT	Proportional weighting in least squares fitting; uses data values for weighting
SF \equiv S_F	Standard deviation of the relative residuals of a NLS or CNLS fit; also SIGMAF
UWT	Unity weighting in least squares fitting; equivalent to no weighting

b. Some fitting models: names, identification, and abbreviations

Models are usually defined and expressed at either the dielectric or impedance immittance level. Therefore, in model names listed here, let the last letter of a name abbreviation be denoted by J, where J=C or D, and C stands for conductive and D for dielectric as above. Many of the models are provided as DCEs, listed on p.4-2, and may be added to a LEVMW

circuit, such as Circuits B (p. 5-7) or 0 (p. 5-28).in one of the DE positions in the circuit. For example, DCD represents a Davidson-Cole model expressed at the dielectric immittance level. CDCD is a DCD with a capacitance in parallel with it, and DCD*R is one with a resistance R in series with it. Composite model names are often written with a • instead of * to indicate the presence of a component in series.

Some important fitting models are listed below. References to the Macdonald publication list (available for reading and free downloading at <http://jrosssmacdonald.com>) are added as appropriate, to provide model information and relevant references, and are expressed as either #xxx or Mxxx, such as M252 for paper #252. Other fitting models not mentioned here have often been found to be less appropriate, either physically or in their fitting ability, than those listed here, but this list is not meant to be comprehensive.

CCJ	Cole-Cole dispersive model
DCJ	Davidson-Cole dispersive model M246
DebJ	Debye relaxation (single relaxation time)
HNJ	Havriliak-Negami distributed response M252, Eq. A.5. HNJ includes CC, Deb, and DC models.
KWW	Kohlrausch-Williams-Watts stretched-exponential model M229,M239,M248
K0, K1	Important CSD0 and CSD1 Kohlrausch response models M234, M239, M245, M249
OMF	<u>O</u> riginal <u>M</u> odulus <u>E</u> ormalism response M229, M230
PNP	Poisson-Nernst-Planck ordinary diffusion model M11, M92, M124, M183, M252, M254, M256
PNPA	Poisson-Nernst-Planck anomalous diffusion model M255, M256
RANDLES	Models.. See p. 0-11
ZC	or ZARC: Cole-Cole complex DSD response model used for CSD response M221

Appendix 3 - LEVM Tips, Cautions, and Suggestions

Because LEVM includes so many possibilities, here I summarize some useful information which may help you get properly started after you have read this section and Section 1. See pp. 8-2-8-5 for a short list of some of the changes made in LEVM V. 7.01 and 7.1 to yield the present V. 7.11.

Data preparation and transformation

Read the Quick Start instructions on p. 0-11. Verify that you can run the input test file AZC (see pp. 1-3 through 1-8). Further, you can learn a lot about the possibilities of LEVM by examining and running some of the many other test files provided with LEVM in the subdirectory TESTS.

To fit your own data to a model (see Sections 4 and 5), you must pick an appropriate one, prepare the top part of an input file and combine it with your data to make the final LEVM input file. See p. 1-7 for a very easy way to carry out this procedure.

You may find it easiest to first pick one of the test files provided with LEVM which already incorporates the main fitting choices you will want to carry out on your data. For example, if your data is of Randles charge-transfer form, go to FITTESTS, Circuit B or O and pick a file that includes RAND in its name, and verify that it runs properly. Then use the ONEMKE batch file as described on p. 1-7 to combine the top of a RANGLES input file with your own data set. Finally, make whatever changes are needed in the top part of the resulting input file, MYDAT, to properly accommodate the specifics of your data/model situation and then run MYDAT to carry out the fit. In particular, you will need to decide which model parameters should be used, and which should be free or fixed. Provide rough guesses for their initial values.

If your frequency data is in Hz, set FREEQ in line 2 of the top part of the input file to F; leave blank if data already involve angular frequency.

If the imaginary part of the data (at the impedance or dielectric constant level) is positive, it needs to be made negative in the final input file. To do so, set NEG in line 2 of the input file to N; otherwise, leave blank.

If any of your free fit parameters either are or might need to be negative, on line "5" of the input file set the NFREE index number for that parameter to 2 rather than 1 (see p. 3-13).

To avoid lengthy convergence times when fitting experimental data, set the parameter IOPT on line 2 of the input file to a number greater than -9, say -4 or -5. For data with very small errors, use a value such as -8 or -9.

If your data are not in the proper form to run directly on LEVM, the batch programs DINPUT and GINPUT (pp. 2-5 through 2-7) provide a wide variety of transformation choices, even including the possibility of addition of various kinds of random data to exact simulated data.

Run-time problems

If a fit run is very slow in converging and one of the free parameter estimates gets smaller and smaller, repeat the run with the NFREE index number for that parameter set to 2 so that it is free to become negative.

If a fit run almost immediately stops iterating and gives an error message, it is likely that the initial

values of one or more of the free parameters are negative but their NFREE values are set to 1 rather than 2. Changing to 2 will correct the problem.

General

Note that many fit models allow time-domain fitting as well as frequency-domain fits. Thus one can generate simulated transient response or fit experimental transient response data to a response model. For models which allow both types of fitting, it is unnecessary and usually inappropriate to use Fourier transform techniques to change the data from one domain to the other. Instead, if the data are available in only one domain, just obtain a good fit for the data in that domain and use the fit parameters and the model to generate the "data" which corresponds to this fit in the other domain: set MAXFEV=0 in the input run file.

Besides screen output, ordinary frequency response fitting produces several output files. Perhaps the most important are PNTOUTL and OUTIN. In addition, there are situations where the files AUXPNTL, AUXPNTL1, LINOTRL.DAT, LINOTR, and LINOTL are also useful. Other output files are generated when one is using the K or R circuits and when log-log slopes of data are calculated using the MK4LF batch file (see p. 2-8).

Finally, when using LEVM in the MS-DOS environment, I have found it exceptionally useful to do so after invoking a DOS Shell, one which allows one to keep easy track of all input, output, and run files, and to make quick changes in input files for subsequent fitting. The one I use and find particularly satisfactory is 1DIRPLUS (WONDER PLUS) from Bourbaki of Boise, ID, 208/342-5849. Unfortunately, it is no longer available.

Introduction to the LEVM Complex Nonlinear Least Squares Fitting Program

This section contains important information on the CNLS fitting program LEVM. After a brief introduction to least squares fitting, it discusses several important aspects of using the program. Detailed discussion of several different input file preparation methods and of various weighting possibilities have been put in the appendices to this section. They provide background information on the various options available and are primarily of reference value. By following the examples in Part 3 of this section, the user should immediately be able to use LEVM with user data.

CONTENTS:

1. Introduction to CNLS Fitting	1-2
2. Verification of the Program	1-3
3. Examples of Input File Preparation and Use	
(a) Header and separate data file	1-7
(b) Full input file and separate data file	1-7
(c) Two full input files: data exchange	1-7
4. Discussion of Some Output Quantities	
(a) The PNTOUTL file	1-7
(b) Gradient outputs	1-8
(c) Discussion of important extra final output quantities	1-9
5. Some Fitting Hints	1-10
6. Run-time Errors	1-12
7. Fitting of Typical Data: Comparison of Different AZC Fits	
(a) Main weighting choices	1-14
(b) Consideration of some other AZC run choices	1-16
Appendix 1 - Input-file Preparation Methods	
(a) The three methods	1-17
(b) The batchfile/editor method	1-17
(c) The INLEV approach	1-18
(d) MKLIO methods	1-18
Appendix 2 - Weighting Choices	
(a) Old weighting possibilities	1-19
(b) New weighting possibilities	1-20
REFERENCES	1-22

Introduction to CNLS fitting

A list of general and specific references pertinent to CNLS and LEVM appears on the last pages of this section. Much general and specific information on IS and IS data analysis, including graphics and CNLS fitting, may be found in the book, Impedance Spectroscopy, J. R. Macdonald, editor, Wiley-Interscience, 1987 and especially in its enlarged Second Edition, 2005, edited by E. Barsoukov and J. R. Macdonald.

The aim of least squares fitting procedures is to find a set of parameters P which will minimize the sum:

$$S(P) = \sum w_j [Y_j - YC_j(P)]^2.$$

The sum is taken over 1...M, where M is the total number of data points. w_j is the weight associated with the jth point, Y_j is the jth data point value (at any of the four immittance levels) to be fitted, and $YC_j(P)$ is the corresponding value of the calculated fitting function involving the set of parameters P. Complex data can be fit using this technique by simply stringing out both the real and imaginary data in a single array which is twice the size of the number of data points. Thus if there are M data points, the 1st element of the data array would contain the real component of the first data point while the M+1th element would contain the imaginary component of the first data point. Note that when the fitting function is nonlinear in any of its parameters (NLSQ), fitting requires iteration. This is virtually always the case for models of interest to IS.

The weighting factor w_j is calculated from the uncertainty of the jth data point, $RW(j)$, as follows:

$$w_j = 1/[RW(j)]^2.$$

LEVM provides a very large variety of ways of computing $RW(j)$. It can be set to unity; it can be set to the value of the data point (or calculated value - i.e., function weighting) adjusted to some arbitrary power (using input parameter $\xi \equiv X_i = P(32)$); it can be input separately; and it can be computed dynamically. For more details on weighting, consult the discussion on weighting in the Preface, the IRCH choices listed in Section 3, the two pages on CNLS weighting definitions at the end of that section, and the weighting discussion presented in Appendix 2 of this section. A reference concerned with the specific minimization procedure used in LEVM is:

Levenberg-Marquardt algorithm: see J. J. More in G. A. Watson (ed.), Lecture Notes in Mathematics: Vol. 630, Numerical Analysis, Springer-Verlag, Berlin, 1978, pp. 105-116.

Verification of the Program

In the subdirectory TESTRUN, find the LEVM.EXE file (see the Preface section of this manual). Then issue the command RNL AZC to run LEVM with the input file AZC. You should not receive the reply, "Program too big to fit in memory," with the current version of LEVM.EXE mostly running in extended memory. The program should work and produce at least the output files PNTOUTL and AUXPNTL. The PNTOUTL file should be compared in detail with the PNTOUTL output file listing appearing on several pages immediately below. If your results are essentially the same as those shown, you have verified the proper operation of LEVM on your machine. See Part 4 below for a description of the contents of PNTOUTL.

Note that during the working of the program a sequence of number pairs appears on the screen (provided that $P(34) \geq 0$). The first measures the cumulative number of calls to the model and calculations of the gradient; the second identifies the part of LEVM used during fitting iteration. These are: 1 - main fit part and ROE; 2 - IFP not zero; 3 - IRE > 0; and 4 - IOPT > 0 and operative. When $P(34) < 0$, appreciably more iterative screen output appears.

If you want to see local values of the free parameters during iteration, set NPRINT = 1 or larger in the input file used. Also, if you want to see values of the gradient associated with each parameter at each step in the least-squares iteration, set the input parameter P(34) negative (see further discussion later in this section). After convergence, a subset of the PNTOUTL output (see Section 1.4 below) appears on the screen. Use the PAUSE key to temporarily stop its scrolling. Alternatively, when the program finishes, issue the command RLVP to show PNTOUTL on the screen page by page, or PLV to print it out. On laser printers, a manual Form Feed command may be necessary after the use of PLV to eject the last page.

AZC LEVM INPUT FILE

AZC A CKT EXACT PAR. VALUES: ZC(2D6,1D0,0.3), 1D6,1D-12. 8/11/96

```

0 ZZRR A .1000D+01C 0.0000D+00 0 -11
27 40 91 0 2 0 0 1 2.0000D+00
.91000000D+06 .00000000D+00 .00000000D+00 .00000000D+00 .00000000D+00
.12000000D+07 .14100000D+01 .00000000D+00 .38400000D+00 .50000000D+01
.00000000D+00 .00000000D+00 .00000000D+00 .00000000D+00 .00000000D+00
.00000000D+00 .00000000D+00 .00000000D+00 .00000000D+00 .00000000D+00
.00000000D+00 .00000000D+00 .00000000D+00 .00000000D+00 .00000000D+00
.00000000D+00 .00000000D+00 .00000000D+00 .13000000D-11 .00000000D+00
.00000000D+00 .10000000D+01 .00000000D+00- .10000000D+00 .00000000D+00
.00000000D+00 .00000000D+00 .00000000D+00 .00000000D+00 .00000000D+00
1000011010000000000000000000000100000000000
 1  .1000000000000D-03  .2900000000000D+07  -.5100000000000D+05
 2  .3162278000000D-03  .2900000000000D+07  -.6900000000000D+05
 3  .1000000000000D-02  .2800000000000D+07  -.9200000000000D+05
 4  .3162278000000D-02  .2700000000000D+07  -.1200000000000D+06
 5  .1000000000000D-01  .2600000000000D+07  -.1500000000000D+06
 6  .3162277000000D-01  .2500000000000D+07  -.1800000000000D+06
 7  .1000000000000D+00  .2300000000000D+07  -.2100000000000D+06
 8  .3162278000000D+00  .2200000000000D+07  -.2300000000000D+06
 9  .1000000000000D+01  .2000000000000D+07  -.2400000000000D+06
10  .3162278000000D+01  .1800000000000D+07  -.2300000000000D+06
11  .1000000000000D+02  .1700000000000D+07  -.2100000000000D+06
12  .3162278000000D+02  .1500000000000D+07  -.1800000000000D+06
13  .1000000000000D+03  .1400000000000D+07  -.1500000000000D+06
14  .3162278000000D+03  .1300000000000D+07  -.1200000000000D+06
15  .1000000000000D+04  .1200000000000D+07  -.9400000000000D+05
16  .3162278000000D+04  .1100000000000D+07  -.7400000000000D+05
17  .1000000000000D+05  .1100000000000D+07  -.6400000000000D+05
18  .3162278000000D+05  .1100000000000D+07  -.7400000000000D+05
19  .1000000000000D+06  .1000000000000D+07  -.1400000000000D+06
20  .3162278000000D+06  .9300000000000D+06  -.3200000000000D+06
21  .1000000000000D+07  .4900000000000D+06  -.5100000000000D+06
22  .3162278000000D+07  .8900000000000D+05  -.2900000000000D+06
23  .1000000000000D+08  .9800000000000D+04  -.9900000000000D+05
24  .3162278000000D+08  .9900000000000D+03  -.3200000000000D+05
25  .1000000000000D+09  .9900000000000D+02  -.1000000000000D+05
26  .3162278000000D+09  .9900000000000D+01  -.3200000000000D+04
27  .1000000000000D+10 .1000000000000D+01 -.1000000000000D+04

```

ω Z' Z''

The spacing of data columns above is closer than that in an actual data file.

AZC PNTOUTL FILE

LEVM: Complex Nonlinear Least Squares Immittance Data Fitting Program

AZC A-CKT EXACT PAR. VALUES: ZC(2D6,1D0,0.3), 1D6,1D-12; 8/11/96

Data entered in ZR format to be used in ZR fit;
Circuit Model: A

Fit of Complex Data

- # of data points = 27 weight: IRCH = 2 # of free parameters = 5
- prints every 0 iterations max. # iterations = 91 main WT uses: DATA

Weights involve the magnitudes of data or function values raised to the power of 1.00000D+00

Initial parameter guesses and fixed (0) or free (1 or 2) status:

P(1) = 9.100000D+05	1	P(17) = 0.000000D+00	0
P(2) = 0.000000D+00	0	P(18) = 0.000000D+00	0
P(3) = 0.000000D+00	0	P(19) = 0.000000D+00	0
P(4) = 0.000000D+00	0	P(20) = 0.000000D+00	0
P(5) = 0.000000D+00	0	P(21) = 0.000000D+00	0
P(6) = 1.200000D+06	1	P(22) = 0.000000D+00	0
P(7) = 1.410000D+00	1	P(23) = 0.000000D+00	0
P(8) = 0.000000D+00	0	P(24) = 0.000000D+00	0
P(9) = 3.840000D-01	1	P(25) = 0.000000D+00	0
P(10) = 5.000000D+00	0	P(26) = 0.000000D+00	0
P(11) = 0.000000D+00	0	P(27) = 0.000000D+00	0
P(12) = 0.000000D+00	0	P(28) = 0.000000D+00	0
P(13) = 0.000000D+00	0	P(29) = 1.300000D-12	1
P(14) = 0.000000D+00	0	P(30) = 0.000000D+00	0
P(15) = 0.000000D+00	0	P(31) = 0.000000D+00	0
P(16) = 0.000000D+00	0	P(32) = 1.000000D+00	0

The following parameters are always fixed:

P(33) = 0.000000D+00	P(37) = 0.000000D+00
P(34) = -1.000000D-01	P(38) = 0.000000D+00
P(35) = 0.000000D+00	P(39) = 0.000000D+00
P(36) = 0.000000D+00	P(40) = 0.000000D+00

- No transformation made: data in ZR format

Algorithm terminates having estimated that the relative error in the sum of squares is at most FTOL

FINAL L2 NORM OF THE RESIDUALS 0.9050982D-01
 NUMBER OF FUNCTION EVALUATIONS 12
 NUMBER OF JACOBIAN EVALUATIONS 7

(Note: the numbers 12 and 7 above may be computer-dependent)

WEIGHTED SUM OF SQUARES, S: 8.1920273022224D-03
 SIGMAF SQUARDÆ ESTIMATE, XS: 1.6718423065760D-04
 SIGMAF ESTIMATE: 1.2929974116664D-02
 FIT QUALITY FACTOR: -2.494D+02
 NO. DEGREES OF FREEDOM: 49

ORIGINAL PARAMETER NUMBERS
 PARAMETER ESTIMATES
 ESTIMATED STD DEV OF PARAMETERS
 ESTIMATED RELATIVE STD DEV OF PARAMETERS

	1	6	7	9	29
9.9822D+05	1.9916D+06	9.8491D-01	2.9827D-01	9.9996D-13	
3.8262D+03	5.8606D+03	1.6213D-02	1.0461D-03	2.7956D-15	
3.8330D-03	2.9426D-03	1.6461D-02	3.5072D-03	2.7957D-03	

PDAV= 5.9079D-03 PDRMS= 7.9301D-03

ESTIMATED PARAMETER CORRELATION MATRIX

0.1000D+01
 -0.2554D+00 0.1000D+01
 0.2569D+00 0.2005D-01 0.1000D+01
 0.2901D+00 -0.7285D-02 0.2191D-01 0.1000D+01
 -0.5546D+00 0.1227D+00-01043D+00-0.7830D-01 0.1000D+01

AV ARR	SD WR	SD WI	SD WC	AV ARI	# IT
1.074D-02	1.728D-02	8.594D-03	1.293D-02	5.424D-03	0
AV RR	SD RR	SD RI	SD RC	SSE	# OPIT
4.444D-04	1.728D-02	8.594D-03	1.293D-02	3.397D-01	0

Examples of Input File Preparation and Use

Header and separate data file

This is the situation one would face on first fitting a new set of data. Let us use the AZCTP and AZCEDAT files, present in the TESTS subdirectory of LEVM. The first of these is the header portion of the full AZC file (involving truncated data) but here with M, the number of data points, set to zero in the third line and with the input parameter IRE set to -10 in this line (in order that the file OUTIN be produced on fitting). The second file is the data (without a numbering column) from the AZCE file, one which involves the original exact AZC data. Then if we enter: ONEMKE AZCTP AZCEDAT AZCTE, a new full input file, AZCTE, will be produced. Now if we enter: RNL AZCTE, the data will be fitted and the results will appear in PNTOUTL, and, in addition, the new file OUTIN, which contains the new fit parameter estimates, will be made. It can be run with the command RNLO.

Full input file and separate data file

It is frequently the case that a fit has been carried out under one set of conditions and then measurements are made with a slightly different set, perhaps a different temperature. It will then be convenient to use the original-fit parameters in a new input file involving the new data. A convenient way of doing this is described below. See also EXTRLVM in Section 7.

Here we shall use the full AZC file and the AZCEDAT data file. If we enter ONEMKE AZC AZCEDAT AZCNE, the new full input file AZCNE is produced with its M value determined by the number of data lines in AZCEDAT. It can then be run with RNL AZCNE.

Two full input files: data exchange

In this case, let us use the full input files AZC and AZCE. Because of the presence of a numbering column in the AZCE file, we must use the CONLFL batch file instead of just ONEMKE. In order to make a full input file which uses the header part of AZC and the data part of AZCE, enter CONLFL AZC AZCE AZCC. Then run the result with the command RNL AZCC.

Discussion of Some Output Quantities

The PNTOUTL file

First, examine the PNTOUTL file listing in Part 2 above, obtained by running AZC. You will see that much of the output is self-explanatory and tells you which input choices were made in the input run file. You will need to first find the circuit model used (here A) and then use the identification list for this circuit (see Section 5) in order to pair the parameter values with the actual elements in the A circuit used in the fitting (i.e., all non-zero parameters in the PNTOUTL initial listing). Only parameter values with a 1 or 2 after them are free during the fitting.

The quantity XS in the output is the chi-squared statistic for the fit; its square root is the estimated SD of the fit, SIGMAF. A fit quality factor (FQF) value is also listed to help in comparing which of two fits is most appropriate, especially when one doesn't use the utility FXITEST supplied with this program. The restrictions on applying FXITEST, listed in Section 2, also apply for FQF comparisons. This measure is discussed in L. Kirkup and J. Sutherland, *Comp. in Phys.* **2**(1988)64. The smaller (algebraically) FQF is, generally the better the fit, but the difference may or may not be statistically significant!

The next few lines in PNTOUTL are self explanatory. Note that the quantities, PRSDAV and PRSDRMS (discussed in Section 1-8 below) are particularly valuable in assessing the overall adequacy of a fit. If they exceed say 0.2 or 0.3, it is a poor fit. For a good fit, one would expect 0.03 or less.

Since the correlations, and the SD's of the parameters are approximations based on a final linearization of the nonlinear situation, they cannot be considered very accurate; of the two sets, generally the correlations are the least accurate.

The last two data lines involve mostly weighted quantities and relative ones. For PWT proportional weighting, their values should be the same. The first line uses RW(I) values (the uncertainties of the data points) for normalization, while the second uses the actual data values themselves for normalization. Thus, generally the first line is the more appropriate since it is based on the weights actually used in the fitting, but the second line is particularly useful for unity weighting (UWT) situations.

The quantity IT measures the number of IFP (< 0) iterations or the number of IRE (> 0) iterations actually used during the run. Similarly, OPIT lists the number of optimization iterations required for optimization convergence.

Finally, SSE is a negative entropy measure of goodness of fit. It, and some of the other outputs listed in PNTOUTL, are discussed in detail in the paper by Macdonald and Potter listed in the references at the end of this section.

Gradient outputs

In order to produce **screen output** of parameter gradients during fitting iteration, one should set $P(34) < 0$. Such output can be quite useful in examining the approach to convergence. It should be distinguished from the screen output produced with $NPRINT = 1$ where only the parameter estimates themselves are output.

When $P(34) < 0$, the **gradient output** first involves the L2 norm of the residuals, a measure of the adequacy of fit. Then for each gradient calculation, on each succeeding line one sees an iteration number, the free-parameter identification number, the free-parameter value, the free-parameter gradient, the product of these quantities, and the part of the gradient associated with weighting (it will be zero unless $IRCH < 0$ or IFP is nonzero). Ideally, as iteration progresses towards convergence, the free parameter gradients should approach zero (a least squares condition). Because of the use of finite-length arithmetic and numerical differentiation, one only gets an unchanging value of zero, however, when the parameter is a bad choice. One will generally see each gradient decrease in magnitude as the iterations progress, however. Note, however, that if the gradient for a parameter seems to be zero at every iteration, it usually means that this parameter is not appropriate (or at least not set free to vary) in the circuit being fitted. The remedy is obvious.

But the screen gradient output (unlike the fitting calculations themselves) is unscaled. Therefore, a more immediately instructive quantity is the product of a parameter estimate and its gradient estimate. Although, for example, when a resistor is of the order of 10^6 and a capacitor is of the order of 10^{-12} , there will generally be a very large difference in their final unscaled converged gradient values, there will be a much closer match between their parameter*gradient values. The important thing to look for is to see that all such products are appreciably smaller at the end of iteration than at the beginning, and usually less than 10^{-6} in magnitude. If one or more aren't, input guesses need improvement. This is also the case if one sees very little change in the gradient values as iteration progresses (unless, of course, the input parameters are already all at their best values!).

Discussion of important extra final output quantities

For some DAE fitting models (see Section 4) and when fitting using the O Circuit (Section 5), four lines of extra output may appear at the end of the regular output on the screen and at the end of PNTOUTL. The 16 CSD-related output quantities are associated with moments of the DRT used for fitting the data and with low- and high-frequency limiting CSD "dielectric" quantities. They are particularly useful for simulation studies and in connection with fitting of high-resistivity CSD data, especially when the data are in specific form. They do not usually appear for DSD fitting of data, and they only appear for CSD fitting choices for which the moments of the DRT are known or can be calculated. See Ref.(40) for a detailed description of these matters.

First, the quantities $XXM1$, $XX1$, $XX2$, and $XX3$ are the dimensionless moments, $\langle x^m \rangle$, of the DRT for, respectively, $m = -1, 1, 2$, and 3 . It is assumed that the distribution is normalized to unity. For a non-cutoff DSD distribution (except the Gaussian), $XXM1$ will be infinite, here approximated by $10^{**}100$. The quantity $AVTAU$ is the fit value of TAU multiplied by $XX1$. The quantities RN and AIN , when nonzero, are exact or approximate normalization values (See discussion in Circuit O).

The letters D and C designate dielectric or conductive-system quantities. Thus EDINF (a fit value) and ECINF (a value calculated from the DRT part of the total CSD fit model) are the high-frequency limiting values of the dielectric part of the total dielectric constant (denoted by EINF and calculated from the total fit model after convergence) and that of the conductive-system part. Similarly, the low-frequency limiting values are ECO (calculated in the same way as ECINF) and EO, calculated from the final fit results. For a consistent fit for data without any dielectric-system dispersion (so $ED0 = EDINF$), one finds $EINF = ECINF + EDINF$ and $EO = ECO + EDINF$.

The last of these relations does not apply when the fit model includes electrode parameters since their presence usually increases the effective zero-frequency dielectric constant, EO, very greatly. XRAT is the ratio between ECO and ECINF. Note that ECINF is actually zero for non-cutoff CSD0 fits. The values of EINF and EO are only correct for fully complex or real-part fits. In most situations, EINF and EO are calculated by extrapolation of the dispersive part of the fitting model used. But this is inappropriate for such models as the no-cutoff Havriliak-Negami one, which, in general, leads to physically unrealizable response in the limit of both high and low frequencies. In such cases, the limiting values shown are calculated from the above formulas.

The quantity RHO0 is the CSD resistive scale factor, R or ΔR ; TAO is the fit value of the relaxation time parameter of the fitting model; and ETAU (epsilon sub tau) is an effective dielectric constant related to it. It is defined as $\tau_0 \Delta P / [CELCAP * (R_0)^2]$, or $\tau_0 \Delta \rho / [\epsilon_V * (\rho_0)^2]$ for specific data. Here ϵ_V is the permittivity of vacuum, $\Delta R = R_{C0} - R_{CINF}$, and $R_{C0} = R_{DC}$. Values of both TAO and ETAU always appear, but when $|MODE| = 1$, fitting involves TAO, while the choice $|MODE| = 2$ involves direct fitting of ETAU and subsequent calculation of TAO. See Appendix 1 of the PREFACE for a discussion of types of CSD response. The present output can occur for either CSD1a or CSD1b fitting (see, e.g., Section 4, NELEM=6,7). But CSD1b, unlike CSD1a, is only appropriate when the high-frequency-limiting part of the CSD dispersion, ρ_{CINF} or R_{CINF} , is zero. If not, some of the final output quantities will be incorrect.

Some Fitting Hints

When beginning to fit new CNLS data for the first time, one always faces the problem of trying to pick reasonable initial values of the parameters of the fitting model. Although LEVM can converge when such parameter estimates are far from their correct values, the more free parameters in the fitting, the more difficult it is for the program to converge to the correct set of parameter estimates, ones best representing the data for the chosen fitting model. Some strategies to attack this problem are mentioned below.

The more free parameters there are in a fit, the more desirable it is that their initial values be reasonably close to their final converged values. An example of a many-parameter situation appears in the JTST file (in the subdirectory TESTS). A discussion of fitting these data is provided under Circuit J in Section 5 and is worth reading in the present context.

When one is dealing with a model involving many parameters of uncertain values, it is often desirable to keep most of them fixed at plausible values during initial fit exploration. If convergence is attained with the few remaining parameter(s) free, then their fit values can next be used fixed and other parameters set free for fitting. In this way one can often progressively approach good values of all the parameters, and a final fit can be made with all of them free. Further, in initial exploration, it is sometimes worthwhile to divide the data into two or more regions and try fitting each of them separately. The results can then be finally combined into a complete model and all parameters taken free at the same time.

In sequential approaches to fitting, it will be very helpful to produce and use the file OUTIN (see Section 2). For initial and difficult fits, it can also be very helpful to use the simulation capabilities of LEVM before much CNLS fitting, and/or to examine (directly or graphically) the data generated by the current choice of model and parameter values (see discussion of MAXFEV = 0 in Section 3). Finally, note that sometimes one or more parameters may need to be negative. If you do not allow this, convergence problems can occur or, at the least, a poor fit will be obtained. A sign of the need to allow a parameter to be negative is its progression during fitting to a very small positive value. You should suspect any parameter whose fit value becomes very small. Try another fit with its value of NFREE set to 2 in order to allow it to go negative if needed.

When many parameters are free to vary, say N of them, the CNLS program must try to find the absolute minimum of the weighted sum of squares in an N-dimensional space. The larger N, the more likely the program is to get stuck at a local (not absolute) minimum.

Convergence of a CNLS fitting run does not necessarily guarantee that the absolute minimum (**least** squares) has been found. A useful test of reaching at least a local minimum is to always run with IRE < -9, so the file OUTIN is automatically produced. Then, when a run is finished, just do RNL OUTIN (or RNLO) and see if the same results as before are obtained. To increase the chances that an absolute minimum has been found, change the input free parameter choices to values appreciably different from those originally used. Then do a fit and see if the same set of parameter estimates is obtained. If so, the chances are that the proper minimum has been found. If not, use the result with the lowest value of the standard deviation, SD, and perhaps run its OUTIN again.

Whenever fully complex data are available, they should be fitted with CNLS (DATTYP = C). But there are times when only one of the two parts of complex data is available, and it is also sometimes useful to compare real-part, imaginary-part, and complex fit results when fully complex data are available. But it is worth mentioning that such single-part fits cannot always estimate all the parameters of a complex model. For example, real-part admittance-level fitting cannot estimate the value of a parallel capacitance that may be present.

The present version of LEVM incorporates various alternate ways by which inductive effects can be included in the fitting model. A recommended way is to use a negative resistance and capacitance, as in the composite distributed circuit element (DE) of choice NELEM = 17 (see Section 4). Another is to use a specialized DE with NELEM = 2 or 3. NELEM is designated by NDE_x, where x is a number, in the circuit parameter choices of Section 5.

Run-time Errors

ERRORS!! If the LEVM program doesn't even start iterating (no, or at most only 1 or 2 numbers appearing sequentially on the screen after a minute or so and no change thereafter), it generally means that there is something wrong with the input data file. It is essential that all choices be made and be consistent. Check particularly to see if M is set to the actual number of frequency values and that N is set to the total number of parameters of the circuit used (a value between 32 and 50). Also check to see that the signs of the imaginary part of the data are consistent with the presence or absence of the letter N in the NEG input position. If trouble continues, set MAXFEV = 0 in Line 3 of the input file, do a run and examine the predicted data values in AUXPNL or in OUTIN if it is being made. This tests only the model predictions for your choice of parameters, no least squares iteration. If it doesn't work at all, some of your input choices must be wrong. If it does work, compare its results with the original input data to discover regions of very poor fit before changing your circuit and/or parameter guesses. Another possibly useful procedure is to compare your input file in detail, line by line, with AZC or with one of the comparable test files provided in the TESTS subdirectory.

It may happen that iteration continues for a while, then stops and the following message appears:

**NOTE: LARGE VALUE OF ONE OR MORE RELATIVE STANDARD DEVIATIONS
INDICATION OF A WRONG OR POOR INITIAL CHOICE OF A FREE PARAMETER.
CHECK FOR INCOMPLETE CONVERGENCE AND/OR SINGULAR MATRIX MESSAGES.**

PAUSE: To resume execution, type: go

When such a message appears, it may mean that convergence is just not complete and that further iteration may lead to convergence. Thus, it will always be worthwhile to type "go" (**NOT** "GO") and push "ENTER" at this stage. If OUTIN has been made, you might double the input value of MAXFEV in it and do RNL OUTIN to continue the run.

When the above approach works, fine; but it doesn't always work if problems are extreme. Instead of a normal completion occurring when "go" is entered, one may first get the following message:

!!!! BAD CORRELATION MATRIX: !!!!

When this happens, it means the matrix is singular, and you should see some (meaningless) values of zero and/or absolute values of less than 10^{-10} in the correlation matrix output. This is a worse error than that above but note the value of SD in the output. Then continue the run. If it stops after running out of iterations, compare the new SD value with the old one. If the new one is not significantly smaller, one needs to change input choices.

If the program gives a SINGULAR MATRIX message, it means that your choice of circuit and/or parameter guesses is usually inadequate. Then one or more of the parameter standard deviation estimates will usually be very much larger than the parameter estimate itself (or possibly zero instead). In your next run of this data, set $P(34) < 0$ so that parameter gradients will appear during iteration. By examining the gradient values, one can often get an idea of which one (or more) of the parameter input guesses is too far off. You should then work on changing parameter input guesses and/or model choice and structure so that convergence will occur. Sometimes the singular-matrix problem arises because a parameter is constrained to be positive but **SHOULD** be allowed to go negative (set NFREE for the parameter to 2 and see if this cures the error). Note that it is possible to obtain a "domain error" message if you initially set, by mistake, an exponent (for example, PHI in the KWW model) to a large value such as 10 or 100 when it must actually be limited to 1 or less.

For at least reasonably adequate data, I have always found it possible to change either the circuit and/or the initial parameter guesses so that the singular-matrix message disappears and all parameter SD estimates are reasonable. I strongly urge you not to publish fits obtained for which the SINGULAR MATRIX message is present. They are not "least squares" and are far from being best possible fits. (See, however, the discussion of Circuit R).

Finally, there are two distinct solecisms which sometimes appear in the literature. I suggest you abjure them! The first is the use of "complex impedance." Since impedance is complex by definition, the qualifier is nugatory and unnecessary. But such terms as "complex dielectric constant," "complex dielectric susceptibility," "complex resistivity," and "complex conductivity or conductance" are pertinent. The second is the designation "Nyquist plot" to describe a complex-plane plot of, for example, the real and imaginary parts of impedance data. Nyquist was concerned with two-port (three-terminal) feedback situations; in contrast, impedance measurements in their simplest realization essentially involve one-port (two terminals). Thus the use of "Nyquist" in the present context is improper.

Fitting of Typical Data: Comparison of Different AZC Fits

Main weighting choices

The AZC data set (formed by truncating exact data to two figures) may be used to illustrate some of the various weighting possibilities discussed in Appendix 2 of this section. Since the SSE fit measure and the AA and RA measures are relatively independent of U and Xi, they will be compared for various weightings. Results are summarized in the **table** following this page. There, U_0 and Xi_0 (written as ξ_0 in the table) are the initial values of U and Xi, and the latter are the final values of these error variance model parameters. When they are the same, they are, of course, fixed.

The first three IRCH values (without optimization) represent conventional fit results. Comparison of the RSD AA and RA results with the RE ones allows one to see (a) how well the relative standard deviation estimates approximate the true errors, and (b) how the latter depend on the weighting possibilities. It is clear from these results that MWT (IRCH=3) leads to the worst estimates, closely followed by UWT (IRCH=1). As expected, PWT (IRCH=2) yields the best of these results. It is, however, quite surprising how much fitting improvement is produced for these data by optimizing UWT results and how relatively little occurs when MWT results are optimized. The PWT results are already sufficiently good here that optimization does not improve them much.

The results in the table indicate that the RSD values are generally pretty good estimates of the true RE values. **For experimental data sets, MWT is sometimes more appropriate than PWT, but note that it only applies for complex fits. Try both before making a final choice.

Since the PWT results are already so good, the FPWT ones (IRCH=-2) cannot improve on them very much here. The next fits were obtained with Xi free (FGWT) and with both U and Xi free. Results such as **A|B** denote the estimated value of a quantity and its estimated relative standard deviation. It is clear that a meaningful value of U is not found when it is free and that thus it cannot be distinguished from zero. This is as it should be, since the errors produced by truncation should have little or no directly additive component and may be expected to be essentially proportional in character. Note that the actual additive errors associated with the value $U = 3.7$ are approximately given by $U \cdot \text{SIGMAF}$ (see Ref. A), here about 4×10^{-3} . Although the present results furnish no firm basis for concluding that Xi is different from unity, the last two lines in the table (which should be compared with the FPWT results) suggest that a slightly better fit is obtained when $Xi = 1.06$ rather than 1.

Table 1. Some fitting comparisons with different weightings for the AZC data.

IRCH	IOPT	U_0	ξ_0	U	ξ	SSEAA(RSD)		RA(RSD)	AA(RE)	RA(RE)
1	0	-	-	-	-	0.353	0.0294	0.0382	0.0375	0.0434
	9	-	-	-	-	0.307	0.0053	0.0072	0.0057	0.0063
2	0	0	1	0	1	0.340	0.0059	0.0079	0.0054	0.0075
	9	0	1	0	1	0.274	0.0039	0.0050	0.0054	0.0066
3	0	0	1	0	1	0.398	0.0347	0.0550	0.0213	0.0236
	9	0	1	0	1	0.292	0.0193	0.0316	0.0082	0.0091
-2	0	0	1	0	1	0.334	0.0059	0.0079	0.0052	0.0074
	9	0	1	0	1	0.268	0.0039	0.0050	0.0055	0.0065
-2	0	0	1	0	1.13 0.033	0.272	0.0049	0.0066	0.0053	0.0067
	9	0	1	0	1.06 0.010	0.236	0.0038	0.0050	0.0056	0.0065
-2	0	1	1	6.4 1.5	1.19 0.050	0.257	0.0046	0.0061	0.0053	0.0061
	9	1	1	3.7 2.3	1.17 0.039	0.245	0.0040	0.0053	0.0054	0.0062
-2	0	0	1.06	0	1.06	0.309	0.0053	0.0071	0.0050	0.0070
	9	0	1.06	0	1.06	0.255	0.0038	0.0049	0.0055	0.0065

Consideration of some other AZC run choices

Quickest results will be obtained when the input choices IOPT, ROE, IFP, and IRE (see Section 3) are all set to zero. Setting IFP > 0, should, in principle, yield better results than just using $IRCH \geq 2$, and it should often be used along with positive IRCH choices. By carrying out fits with ROE set between 1 or 2 and 0.1, it is often possible to obtain better parameter estimates than with ROE = 0 (equivalent to ROE = infinity). Values between 0.1 and 3 have often been found to yield superior results (pick the value which leads to the smallest value of S_F (SIGMAF) and the other fit measures. Generally, finding the best value of ROE will take time and is a trial and error procedure. The IRE parameter, which turns on residual-type weighting (RWT) when positive, is discussed in Section 3.

As the results of the Table show, optimization can lead to a much better fit than obtained without it. Thus, it is useful to explore its effects by setting IOPT = 7 or more (for complex data only). Of course this will add some more iteration steps. If it turns out that the values of SD WR and SD WI in PNTOUTL are close together, optimization won't help. If, however, their ratio is of the order of 2 to 10, it generally will help considerably. When the ratio is of the order of 10 or more, the optimization result will approximate to (but not be quite as good usually) as a separate fit of only the real or imaginary part alone, whichever one has the smaller SD. If time is available, I certainly recommend setting IOPT = 7 to 10 provided $|IRCH| < 5$.

Appendix 1: Input-file Preparation Methods

The following discussions are particularly appropriate for LEVM. But for LEVMW it is straightforward to combine the top part of an input file and the bottom data part, as well as to save the data separately. If needed, invoke the extensive LEVMW HELP file.

The three methods

An input file needed for LEVM CNLS data fitting consists of a top (header) part, where choices which guide the program are made, and a bottom part which contains the data (and, as well, weighting quantities when IRCH=0). LEVM allows input files to be made or changed by several different methods: one using batch files to construct the input file, say LINF, and a full-page editor to change it; another using the Amarasinghe program INLEV (see Section 6) to either construct or change LINF; and a third which uses the **ONEMKE.BAT** and other batch files discussed in Section 7. All of these methods have their advantages and disadvantages, and which one to use may depend on the ready availability of a full-page editor. They are described in more detail below.

INLEV is particularly easy to use if one starts with a data set and wants to make LINF. By contrast, I find that if a given LINF is already available and one or more of the input choices needs change, the use of an editor, which allows one to see all aspects of the file at once on the screen, is very helpful and easy to use. Note that MS-DOS and WINDOWS include a full-screen editor. I usually use WORDSTAR in non-document mode because it takes up so little disk space, but another choice I like and often use is the VIEW-EDIT facility of the very helpful DOS-Shell program: Wonder Plus (1dir+) available from Bourbaki, Inc. of Boise, Idaho.

If one only wants to change the initial guesses for fitting model parameters, there is another choice available as well as those above. To invoke it, set ICP=1 in the top part of the LEVM input file, and, on running LEVM, you will be prompted for new values.

The batchfile/editor method

To make a LEVM input file using this approach, the immittance data to be fit (it may be impedance (Z), admittance (Y), complex dielectric constant (E), or modulus (M)) should be entered into a ASCII text file in three columns: frequency, real component, imaginary component. It can be in any format. Give this file a name, say MYDAT. It will become the bottom of your input file.

In addition, the top part of the final input file is needed. On preparing for a new fit unrelated to old ones, it is convenient to use one of the TMP files provided in the LEVM TESTS subdirectory. Otherwise, copy an old related input file to a new name. If you plan to fit with Circuit A, for example, use ATMP, etc. For information on the available circuit models, consult Sections 4 and 5 of the Manual. Using an editor, the appropriate header information should be entered, and all other choices made (see the detailed discussion in Section 3 of the items which appear in the lines of the top part of the input file).

In many cases, only a few values will need changing, but you will certainly have to enter initial guesses for the values of free (and any fixed) parameters of the model in lines 4), and, in line 5, specify whether the parameters are free (positive only, or positive or negative) or fixed.

Once the top and bottom parts of the input file are complete, they may be combined into a final input file, say LINF, by using the batch file MKINP. The appropriate command for the situation discussed here is: MKINP ATMP MYDAT LINF. See the discussion of MKINP and DINPUT in Section 2 and especially information of ONEMKE.BAT in Section 7.

The INLEV approach

This method uses the executable file INLEV.EXE. It is fully menu driven, needs no editor, and uses the same data file, MYDAT, mentioned above but with two additional lines added at the top: an identification line and a line containing the number of data points (see Section 6).

In making a new file with INLEV, invoke it and, when asked, enter the name of your initial data file. Then INLEV allows you to enter/change any or all of the LEVM input choices. When you exit INLEV, you will have available a complete 40-parameter LEVM input file containing your data and ready to run.

INLEV can be used to change any or all elements of an existing LEVM input data run file. If you do not want to change the file name, just push "enter" when you exit INLEV.

INLEV does not use the ATMP, BTEMP, etc. files contained in the subdirectory, TESTS, but they are very convenient for the batchfile input file construction method since they contain a number of (usually) appropriate choices already made which have to be made separately in INLEV. Further, INLEV makes input files which have eight actual lines (designated Line 4) for parameter choices, a total of 40. In the simplest, minimal fitting cases, on the other hand, only 32 parameters need be specified (requiring seven lines in the input file). I have, however, elected to provide all the TMP files in a format which involves 40 parameters, so that P(34) is available and may be set negative to invoke gradient output, if desired.

For a fit with 32 to 35 possible parameters, one might therefore want to use the batchfile method of making a new input file, while for more complicated fits, especially those that require values of several FRA reference resistor(s) ($|IRCH|= 5$ or 6), it will be simplest to use INLEV or ONEMKE. Once an input file of either type is made, it can be changed by using either INLEV or a full-page editor.

MKLIO methods

See Section 7 for many details about ONEMKE, MKLIO.EXE, and other useful file-making and file transformation methods and Part 3 above for some examples of their use.

Appendix 2: Weighting Choices

Old Weighting Possibilities

The choice of weighting to use in CNLS fitting is usually very important and may even be crucial to the achievement of a good fit. For this reason, many weighting choices have been provided in LEVM. The seven choices of IRCH (see discussion in Section 3), involving values from 0 to 6, provide the main weighting type choices. But note that when $IRCH > 1$, the weighting uses the input data (unless $X_i = 0$, giving UWT). Monte Carlo studies that I have carried out [see Ref. A] show that such weighting leads to appreciable bias in the parameter estimates, even when the fitting model used is correct. Alternatively, for IRCH negative and less (algebraically) than -1, the calculated values of the model or fitting function are used in the weighting. When the fitting model is correct, this choice leads to very small residual bias in the parameter estimates, and is thus to be preferred.

There are actually two different ways the fitting model, rather than the data, may be used in the weighting. The first, as above, is to choose $IRCH < -1$. The second involves a sequential approach. No matter what value of IRCH is used, if IFP is set > 0 , the model will be used in the weighting after a first fitting is carried out. Thus if $[IRCH,IFP,X_i] = [2,9,1]$, for example, a PWT fit, involving the data, is first carried out, followed by one with weights involving the fitting model, FPWT. The two FPWT fit results would be exactly the same for the choice $[-2,9,1]$.

There are two more sequential-fit weighting possibilities - one involving $IRE > 0$, and the other $IOPT > 0$. For the IRE choice, see Section 3. Whatever the value and sign of IRCH, after carrying out a fit involving that choice, a last fit takes place involving weighting optimization when $IOPT > 0$. Such optimization, which is only appropriate for complex data, changes all the real weights together and all the imaginary weights together in such a way as to make the more accurate of the real or the imaginary measurements play a more important role in determining the parameter estimates (and vice versa). In this way, the full data set is used to optimize the determination of the parameter estimates.

An IOPT fit always comes last. Thus, if $[IRCH,IFP,IOPT,X_i] = [2,9,9,1]$, for example, a fit with $IRCH = 2$ would be followed by one with (effectively) $IRCH = -2$, and an optimization fit would conclude the run. Optimization works on the results of the preceding fit, that before the IOPT fit (except for $IRE > 0$ and robust estimation: $ROE > 0$; i.e. the final optimization, if invoked, is not robust but is CNLS). Thus, the choices $IRCH = 1$ and $IOPT = 9$ would lead to a UWT fit which would then be optimized (but still would be UWT).

Initial fitting explorations might well be carried out with UWT and PWT ($IRCH = 1$ and 2). How should one decide which give the better results? Because of the difference in the character of the weighting for these choices, the estimated standard deviations (SD's) of the fits, SIGMAF, and the fit quality factor, are not directly comparable for the two fittings. But, the SD's of the parameters, or, better, the relative SD's of the parameters (RSD), may be directly compared and so may SSE values (see No. 4 above). To make such comparison easier, the last two values in the PNTOUTL line containing these quantities are the average of the absolute values of the relative SD's, AA (termed PDAV in the output), and their RMS average, RA (termed PDRMS in output). [When $IPAR < 0$ and a comparison with exact parameter values is possible (as it is for the present AZC data), AA and RA type output appears for the actual relative errors (RE) as well: denoted PRSDAV and PRSDRMS].

Because we are primarily interested in the degree of fit of the fitting model parameters, AA and RA do not include the relative SD's of the variance model parameters U and X_i when they are free (see below), but only the relative SD's of the fitting model parameters. Usually the fit with the lowest value of either (or both) of the AA and RA values is to be preferred. If UWT is the better, then do UWT with optimization. If PWT is the better, it is then reasonable to use the OUTIN file as input for a succeeding fit with a choice like $[IRCH,IFP,IOPT,X_i] = [2,1,9,1]$. Incidentally, the value of IFP doesn't matter if it is greater than zero, but that of IOPT determines the maximum number of optimization iterations allowed.

New weighting possibilities

Consider now the cases where $|IRCH| > 1$. It is possible to combine unity weighting (involving the assumption of additive errors in the data and the input parameter $U = P(31)$) and fractional power weighting by picking fixed values of U and X_i [parameter P(32)] in the input file. Either of these quantities may be zero or both may be nonzero. Rather than take $|IRCH| = 2$, U nonzero, and $X_i = 0$, however, it is better to just make the UWT choice $IRCH = 1$. The choice $U = 0$ and X_i nonzero produces pure fractional power weighting, reducing to PWT or FPWT when $|IRCH| = 2$ and $X_i = 1$, fixed.

But the present version of LEVM goes much beyond these possibilities. It is possible to take $U = 0$ (fixed) and X_i free and let the least squares fitting procedure itself treat X_i as a parameter of the fit, yielding a best estimate of its value. Finally, both U and X_i may be taken free simultaneously, and the fitting will then determine their joint best values (U free and X_i fixed is not allowed). When both are free, the procedure may not always converge, however, especially when the errors have very little or no additive component. Experimentation is in order for these possibilities. Further, X_i free, or free X_i and U, work best when the final fit of the data turns out to be at least a reasonably good one, and thus the fitting model used is a good approximation for describing the data. Often a good final fitting choice for IS data might involve something like $[IRCH,IFP,IOPT,U,X_i] = [2,1,9,0,1 \text{ free}]$.

It is recommended that initial fitting for $|IRCH| < 5$ always be carried out with $U=0$ and $X_i=1$, both fixed. If a reasonable fit is obtained with $|IRCH| > 1$, one should then consider further fits with X_i and/or X_i and U free. If one of them should lead to a better fit with reasonably well determined values of X_i and/or U (i.e. with their relative standard deviations less than say 0.4), then carry out a final fit with X_i and/or U fixed at the values obtained when they were free; this should yield the best and most appropriate fit.

Although the FRA-data weighting choices $|IRCH| = 5$ and 6 have been added to LEVM, they are not suggested for most situations, largely because they involve several fixed parameters whose values are uncertain. In various studies, several different sets of values have been discussed but no unique set recommended. Nevertheless, it may sometimes be worthwhile to try this weighting and its present fixed parameter set in order to investigate its consequences.

REFERENCES

General

1. Armstrong, Bell, and Metcalfe, in *Electrochemistry*, Chem. Soc. Specialist Periodical Reports **6** (1978) 98.
2. Archer and Armstrong, in *Electrochemistry*, Chem. Soc. Specialist Periodical Reports **7** (1980) 157.
3. Gabrielli, Technical Report No. 004/83, Solartron Instruments, England.
4. Sluyters-Rehbach and Sluyters, in Comprehensive Treatise of Electrochemistry, Vol. **9**, pp. 177-292, Plenum Press, 1984.
5. Impedance Spectroscopy: Emphasizing Solid Materials and Systems, J. R. Macdonald, editor, Wiley-Interscience, 1987. See many references therein.

**Second edition: Impedance Spectroscopy: Theory, Experiment, and Applications, E. Barsoukov and J. R. Macdonald, editors, Wiley-Interscience, 2005.
6. Macdonald, J. Electroanal. Chem. **223** (1987) 25; **307** (1991).
7. Macdonald, *Electrochim. Acta* **37** (1992) 1007.
8. Macdonald, *Ann. of Biomed. Eng.* **20** (1992) 289. A general review.
9. Macdonald and Wang, *Solid State Ionics* **60** (1993) 319.

Complex Least Squares Fitting

10. Macdonald, Schoonman, and Lehnen, *Solid State Ionics* **5** (1981) 137; *J. Electroanal. Chem.* **131** (1982) 77.
11. Boukamp, *Solid State Ionics* **20** (1986) 31.
12. Macdonald and Potter, *Solid State Ionics* **23** (1987) 61.
13. Levenberg-Marquardt algorithm: see J. J. More in G. A. Watson (ed.), Lecture Notes in Mathematics: Vol. 630, Numerical Analysis, Springer-Verlag, Berlin, 1978, pp. 105-116.
14. Macdonald, *Solid State Ionics* **58** (1992) 97.

Weighting

15. Spinolo, Chiodelli, Magistris, and Tamburini, J. Electrochem. Soc. **135** (1988) 1419.
16. Macdonald, Electrochim. Acta **35** (1990) 1483. **Reference A**
17. Macdonald and Thompson, Commun. in Statistics - Sim. and Comp. **20** (1991) 843.
18. Orazem, Agarwal, Jansen, Wojcik, and Garcia-Rubio, Electrochim. Acta **38** (1993) 1903. See also Refs. 39 below.

Distributed Circuit Elements

19. Cole-Cole: J. Chem. Phys. **9** (1941) 341.
20. Davidson-Cole: J. Chem. Phys. **19** (1951) 1417.
21. Havriliak-Negami: Havriliak and Negami, J. Polym. Sci. **C14** (1966) 99.
22. Constant-Phase Element, Macdonald, Solid State Ionics **13** (1984) 147.
23. Dissado-Hill, Dissado and Hill, J. Chem. Soc. Faraday Trans. 2 **80** (1984) 291.
24. Kohlrausch-Williams-Watts: Macdonald and Hurt, J. Chem. Phys. **200** (1986) 69. Also see Refs. (40) and (43) below.
25. Distribution of Activation Energies, Macdonald, J. Appl. Phys. **61** (1987) 700, and references therein.
26. Gerisher Impedance: Makkus, Hemmes, and de Witt, Ber. Bunsenges. Phys. Chem. **94** (1990) 960.
27. General Diffusion: Franceschetti and Macdonald, J. Electroanal. Chem. **101** (1979) 307; Franceschetti, Macdonald, and Buck, J. Electrochem. Soc. **138** (1991) 1368.
28. Many DCE's: Hurt and Macdonald, Solid State Ionics **20** (1986) 111.

New, Miscellaneous Publications

29. New EIS Results; KK: Macdonald, *Electrochim. Acta* **38** (1993) 1883.
30. KK Results: Boukamp, *Solid State Ionics* **62** (1993) 131.
31. EIS and DRT results for NaCl: Macdonald, *J. Appl. Phys.* **75** (1994) 1059.
32. Effective-Medium DCE: Macdonald, *Phys. Rev. B* **49** (1994-II) 9428.
33. IS fitting results for disordered material: Macdonald, *Appl. Phys. A* **59** (1994) 181.
34. Unity power-law response, hidden bulk relaxation: Macdonald, *J. Electroanal. Chem.* **378** (1994) 17. The word "relation" in the title of this paper should be "relaxation."
35. KK and the DRT alternative: Boukamp and Macdonald, *Solid State Ionics* **74** (1994) 85.
36. DRT - Accurate inversion of relaxation data: Macdonald, *J. Chem. Phys.* **102** (1995) 6241.
37. DRT, early work: Macedo, Moynihan, and Bose, *Phys. Chem. of Glasses*, **13** (1972) 171 and Moynihan modulus approach. C. T. Moynihan, L. P. Boesch, and N. L. Laberge, *Phys. Chem. Glasses* **14** (1973) 122. See also, Macdonald, *J. Appl. Phys.* **58** (1985) 1955 and Refs. 40 and 43 below.
38. Simple KK alternative, Boukamp, *J. Electrochem. Soc.* **142** (1995) 1885. See Circuits K and R herein for this and other possibilities.
39. Measurement models, KK alternative, P. Agarwal, M. E. Orazem, and L. H. Garcia-Rubio, *J. Electrochem. Soc.*, **139** (1992) 1917. P. Agarwal, O. D. Crisalle, M. E. Orazem, and L. H. Garcia-Rubio, *J. Electrochem. Soc.* **142** (1995) 4149. See CKTs K and R and Ref. 5, p.181.
40. General discussion of EIS fitting and DRT analysis, especially for high-resistivity materials, Macdonald, *J. Non-Cryst. Solids* **197** (1996) 83-110. Important Erratum: *ibid*, **204** (1996) 309. Also, the quantity G_D in Eq. (A2) should be G_{CD} .
41. Gaussian response, Macdonald, *Physics Lett. A* **220** (1996) 351.
42. Combined CSD and DSD response; Nowick data for disordered material, Macdonald, *J. Non-Cryst. Solids*, **210** (1997) 70.

43. CSDO, CSD1, and DSD distinctions. Very accurate fitting using the stretched-exponential (KWW) response model for arbitrary exponents, beta: Macdonald, J. Non-Cryst. Solids, **212** (1997) 95.
44. Transformation of random noise in Kronig-Kramers integral transforms. Macdonald and Piterbarg, J. Electroanal. Chem. **428** (1997) 1.
45. Limiting slopes, discrimination between models, accurate fits. Macdonald, J. Appl. Phys., **82** (1997) 3962.
46. The Ngai and other coupling models and the cutoff model, comparisons, generalizations, and application to explain non-Arrhenius behavior in high-conductivity disordered materials. Macdonald, J. Appl. Phys. **84** (1998) 812.
47. Accurate discrimination between conductive and dielectric relaxation processes. Macdonald, Brazilian J. of Physics, **29** (1999) 332.
48. Examination of the Funke mismatch and relaxation dispersion response model. Macdonald, Solid State Ionics 124 (1999) 1.
49. Impedance Spectroscopy: Theory, Experiment, and Applications. Second Edition, E. Barsoukov and J. R. Macdonald, editors, Wiley-Interscience, 2005. Revised and greatly expanded.

All of the scientific publications of Ross Macdonald are categorized and listed in his WWW website, <http://jrossmacdonald.com>. In addition, **ALL** of his publications are available for **direct downloading** as pdf files. See the serial listing in the website.

Descriptions of Principal LEVM Files

The LEVM program files are provided by downloading or on a floppy disk in compressed (zipped) form. After expansion, the files reside in six subdirectories named: CTDPLOT, LEVMFOR, LEVMISC, MKLIO, TESTRUN, AND FITTESTS. Most of those files which are not explicitly discussed below are mentioned in the PREFACE or in Section 1. For LEVMW downloaded files, see the discussion on p. 0-3 at the beginning of this Manual.

Main LEVM Program Files

1. LEVM source files (CHANGE .FOR TO .F AS APPROPRIATE)
(NEW NAME, CORRESPONDING OLD NAME)
LV0.FOR OR LEV1.FOR
LV1.FOR OR LEV2A.FOR
LV2.FOR OR LEV2B.FOR
LV3.FOR OR LEV3.FOR
LV4.FOR OR LEVOLA.FOR
LV5.FOR OR LEVOLB.FOR
LV6.FOR OR OSUB.FOR
LV7.FOR OR RSUB.FOR
LV8.FOR OR SSUB.FOR
LV9.FOR OR TSUB.FOR
SIZE.INC (USED TO CHANGE MAX LIMITS OF LEVM)
2. Executable LEVM Program Files:
LEVM.EXE (USES EXTENDED MEMORY),
or for Windows: LEVMRUN.EXE

These files require an input file named INFL. Given an input file of any name, INFL is automatically produced in LEVMW and by the run batch files, such as RNL.BAT in LEVM. LEVM.EXE and LEVMRUN.EXE allow a maximum of 1001 frequencies.

3. Other Executable Files

CONLFL.BAT	(data exchange between input files)
CORDER.COM	(data transformation)
CPTRN.EXE	(data transformation and preparation)
CTD.EXE	(2 and 3D graphics)
DATATRAN.EXE	(data transformation)
DINPUT.EXE	(prepare, transform data file)
EXTRLVM.EXE	(see Section 7)
FXITEST.EXE	(statistics)

GINPUT.EXE	(preparation of input data file)
INLEV.EXE	(prep. and modification of input file)
LCTDIN.EXE	(convert LEVM input file to CTD format)
MK4LF.EXE	(for Y or Z data: make other data, get slopes)
MKLTP.EXE	(converts LINOTR file; see RNTEMP.BAT)
MKFREQ.EXE	(make input data file with set frequencies)
MKLIO.EXE	(make, structure, transform an input file)
ONEMKE.BAT	(make input run-file using MKLIO)

Output Files which can be made by LEVM when it is run

1. Files always made

PNTOUTL Main output for printing

AUXPNTL Auxiliary output (also used by CTD)

LASTIN This is a copy of the last input file used for LEVM. It is made automatically whenever RNL, RNLO, etc. is used to begin a fit.

2. Files made only for various input choices

AUXPNTL1 (appears only when MAXFEV > 2)

This file is produced if IPRINT ≥ 1 and is particularly useful if it is desired to plot residuals of different kinds using an external plotting program. It involves a column of frequency values followed by columns of Re and Im unweighted residuals, followed by columns of Re and Im normalized residuals. If IPAR = 0, the last two columns are normalized using corresponding model values; if IPAR > 0, the normalization involves the uncertainties used to form weights. Therefore, this last choice gives weighted residuals. With proportional weighting there is no difference in the results of these two choices. When real data only are being fitted, set IPAR to 0, then output columns are: normalized residuals, weighted residuals, and residuals normalized with model values. Note that when IPRINT in the input file is set to < 0, frequency values are changed to **log freq.** ones ** and also the last column becomes $[\log_{10}(|\text{last column}|) + \text{Mode}]$. In summary:

Col. #:	1	2	3	4	5	6	7
Row #	f or log(f)	Re res	Im res	norm. Re res	norm Im res	See above	

FILES ONLY MADE WHEN IRE < 0

OUTIN

This is a very useful file, particularly when one is trying to find the most appropriate circuit configuration and parameter values to fit a set of data. I recommend you use it extensively, usually with IRE = -11 or -10.

OUTIN is produced after a fitting run has been made. It is exactly like the particular input file (e.g., AZC) used in running LEVM, but it incorporates the final least squares values of the free fitting parameters from the run that produced it (if MAXFEV > 2 for this run. See discussion of MAXFEV in Section 3 for other important possibilities). Thus it can be used without change as input for a second LEVM fit. This is particularly helpful if the original run did not converge completely:

continue it by running OUTIN. If the parameter results in the new OUTIN are exactly the same as those in the input file or in the last OUTIN result, one has a necessary (but not sufficient) test of true least squares results. If the results are appreciably different, the solution is not necessarily least squares and either the model, the weighting, or the initial choices for the free parameters may have been poorly selected. See also RNLVOI.BAT.

If MAXFEV = 0 in the original LEVM input file, and OUTIN is produced, it has the original parameter values, but it incorporates the calculated data associated with the original parameter and frequency choices and will have MAXFEV = 91, so it can be run immediately. This is used in SIM.BAT.

The files below are only made when IRE < -9

LINOUT, LVOUT

LINOUT is in the form needed for direct usage by some 2 and 3-D plotting programs. The first part is the input data, followed by the fitting results. Thus a plot can readily be made which compares the original data and the final least squares fitting model predictions. LVOUT contains only the fitting results (freq.,log10(freq),real,-imag.). LINOUT is used by CPTRAN.BAT.

LINOTL, LINOTR, LINOTRL.DAT

The data columns in LINOTR are freq (as determined by IRE=-10 or -11), Re input data, |Im| input data, Re output data (fit results), and |Im| output data (slightly different for real or imaginary fits only). Those in LINOTL are the same except that all values are logarithms (base 10) of those in LINOTR (taking absolute values where necessary). Thus, these files may be used directly with external plotting programs (I use GRAPHER from Golden Software) or spreadsheets. See IPRINT < 0 for log10 transformation. Finally, LINOTRL.DAT produces freq, log(freq), Re input, log(Re input), |Im input|, log|Im input|, Re output, log(Re output), |Im output|, and log|Im output|.

Summary of LINOTRL.DAT file elements, where f is either frequency or ω :

Col. #:	1	2	3	4	5	6	7	8	9	10
	f	log(f)	Re input	log(#3)	Im input	log(#5)	Re out	log(#7)	Im out	log(#9)

LINOTCON

There are two forms of this file. If ATEMP = -2, DFIT = Y, and CELCAP is given its proper value (not unity), then LINOTCON contains columns of frequency (as determined by IRE=-10 or -11), input values of Re and Im conductivity, and output (fitted) values of these quantities.

Alternatively, an Arrhenius fit is carried out when the G Circuit is used (using appropriate parameter choices) with ATEMP = -3, and DATYP = R. Then the freq data are actually absolute temperatures, T, whether IRE=10 or -11. The output columns are T, 1000/T, log(Q), and relative errors, where Q is here the value of the real part of a low-frequency-limiting (DC) conductance or conductivity. The relative error is here defined as (QM-QD)/QD, where M and D denote model and data, respectively. More comprehensive temperature-dependence fitting possibilities are provided in the NELEM = 33 choice of Section 4.

OUTPARM, OUTDRT

The OUTPARM file contains distribution of relaxation times (DRT) information and is only produced when the K or R Circuit is used and P(40) is non-zero. OUTDRT is only produced for $|P(40)| > 1$. See the detailed discussions for the K and R Circuits (Section 5).

Auxiliary Files**AZC, AZC1, AZCE, AZC.40**

AZC is a stripped-down input test file. Its data have been truncated to two places. AZC1 is a file with the same data but with more fitting choices turned on. AZCE is a file with the original AZC data untruncated. These files involve the A circuit, a ZC distributed circuit element (DE) with a resistor in series with it, R_1 , and this combination in parallel with a capacitance C_p .

ATST, BTST, etc.

Miscellaneous complete input test files.

ATMP, BTMP, etc.

These files contain the top 11 (or more) lines of a LEVM input file, those that are needed to select the various run options. ATMP is set up for the A fitting circuit, etc. To produce a full input file, one must add appropriate data (see discussions of MKLIO, DINPUT, GINPUT, MKINP, and SIM below, and discussion in Section 1).

BEEEP.COM - used to produce a beep at the end of a .BAT file.

CONTROL.COM - used for condensed printing.

CORDER.COM, .FOR

This program can make various transformations on input data. It is run by the command: RNCORD DATAFILE, where DATAFILE is any file of IS data (it should end with -1). Run it to see the various choices possible. (See also CPTRAN.BAT.) Output of CORDER occurs in the file OUTFLC.

CPTRN.EXE - This is a transformation program used with CPTRAN.BAT

CTD.EXE and INLEV.EXE

For more information on these important programs, see Section 6. A few further points about CTD are discussed here. First, CTD can now plot results for complex, real, or imaginary fits and can readily produce **BODE** plots. When a run with any value of MAXFEV is made, CTD can use the resulting AUXPNTL output file to plot results. But note: when MAXFEV=0, only the model results are available for plotting (no residuals). Similarly, when MAXFEV=1 or 2, only the data are available for plotting (again, no residuals). Only when MAXFEV > 2 are the data, the model predictions, and residuals all available for plotting. The form of the AUXPNTL file is somewhat different for the different values of MAXFEV to allow the above CTD possibilities. Finally, note that in CTD when the option "Type" is called for, after a choice has been made, one must specify a Data Set Number. Here, the choice 1 specifies the data (green) and choice 2 specifies the model fit results (purple).

DATATRAN.COM, .FOR

This program can make various transformations on input data. It is run by the command: RNDTRAN DATAFILE, where DATAFILE is any file of IS data (it should end with -1). Run it to see the various choices possible; these are different from those of CORDER. DATATRAN output appears in the file OUTFLD.

DINPUT.EXE, .FOR; See also **GINPUT.EXE, FOR** and especially **MKLIO** in Section 7 for other means of preparing LEVM input files. DINPUT may be used to prepare the bottom (input data) part of a complete input data file for LEVM. It also includes data-scaling and transformation options. One way to run it is the command: RNDIN DATAFILE (where DATAFILE need not, but may, end with -1). DINPUT uses an actual input file named **INTRAN**. Given data in a file (say DATAIN) of three columns (frequency, real, imaginary) separated by spaces or commas, DINPUT produces an output file, **OUTTRAN**, with the data in the proper form for LEVM. Using a word processor or ONEMKE.BAT, it can be combined with an input template file (say ATMP) to produce a final input file for direct use with LEVM. Alternatively, the final file, say MYINPUT, may be produced by the MS-DOS concatenation command: COPY ATMP + OUTTRAN MYINPUT. To run DINPUT directly, use the batch file RNDIN (see below). DINPUT also allows one to scale the real and imaginary data values if desired, and allows the imaginary values to be multiplied by the corresponding input frequency (normal or angular frequency). In addition, it includes a large number of choices of ways to modify and/or transform a data file.

For data which requires no transformation, it is usually easier, however, to use the batch file **MKINP** or especially the Section-7 file **MKLIO** instead (see below and see SIM). When DINPUT is run, it will interactively ask for two separate inputs: first enter a number: usually 1 is appropriate, and then enter three numbers: e.g., 0,0,0 will skip any scaling or other transformations. The other choices are particularly helpful for transforming data columns for use in producing different kinds of plots. See SIM.BAT. DINPUT is also very helpful if one wants to use known, fixed weighting (IRCH = 0). Uncertainties (taken from a separate data file with frequency and uncertainty columns) can easily be appended at the end of the data of the input file using a choice presented by DINPUT.

FXITEST.EXE

This is a statistical test to help one to decide which of two fits is the more appropriate. Here XS is a value of chi-squared for a fit and ID is its number of degrees of freedom; their values appear in the summary report of a LEVMW fit. Consider two fits with differing numbers of free parameters. The needed values of ID1, XS1, ID2, and XS2 also appear in the output files, PNTOUTL, of the two runs. When one invokes FXITEST, by using, for example, the batch file FXI.BAT, these quantities should be entered as requested by the program. Then, a two-sided F test of the likelihood that XS1 and XS2 are associated with the same underlying distribution is carried out and yields the probability, PROB, that they are so associated. If PROB is less than about 0.1 or so, then they are likely to be associated with different distributions. If a one-sided F test is desired, use PROB/2.

This test should only be used for comparing runs with similar weighting (PWT and FPWT, but not UWT and PWT). It should not be applied for runs in which the weighting power (or exponent) XI is different. Besides being helpful in comparing runs for weightings such as PWT, FPWT, and OFPWT with the same XI value, the test is particularly useful in comparing runs in which only the number of degrees of freedom (ID) differs. For example, one run might have 5 free parameters and the other only 4. Particularly when the relative standard deviation of one of the parameters of the fit is large (say, 0.3 or more), it is often worthwhile to **eliminate** this parameter entirely from the model (remove it completely; do not just set it fixed and non-zero) and carry out a new fit for comparison. FXITEST will then tell you whether the resulting changes are significant or not (PROB <= 0.1).

GINPUT.FOR, **GINPUT1.EXE**, **GINPUT2.EXE**

****The first .exe file is likely to work on earlier computers running Windows XP, etc. The second one is appropriate for computers running such programs as Windows 7 involving 64-bit conditions.**

This is a new input-data-preparation program somewhat similar to DINPUT. It does not include as many possible transformations, but it does include several new valuable features. In particular, the input data may appear in any of up to nine columns in any order. In addition, the output form can be selected for direct use with LEVM (or as an input to DINPUT), as an input to the important LEVM-input-producing program INLEV, or in the proper form for use with separate CTD plotting. A comparison of DINPUT and GINPUT is shown below. When GINPUT is invoked, its various choices are shown on-screen. ** In using it, enter three numbers, return key, and then three more. GINPUT.FOR uses **INTRAN** for its input file and **OUTTRAN** for its output one.

RNGIN.BAT

The command RNGIN MYINPUT allows one to use GINPUT to make changes in MYINPUT (a file containing IS data) to produce a LEVM data input file, a file for input to DINPUT, or a file for plotting with CTD or a commercial program. For preparation of a complete LEVM input file, use the batch file MKGIN (similar to MKINP) or ONEMKE, or use the facilities of LEVMW.

Comparison of LEVM input batch files

	GINPUT	DINPUT
Maximum number of columns in a data file	9	3
Choice of columns for frequency, real, and imaginary data	Yes	No
Maximum number of initial descriptive lines	9	0
Make a descriptive line?	Yes	No
Transformations possible?	a) f to omega	a) very many

Output (in the file OUTTRAN)

	a) 3-column data only	a) 3-column data only
	b) Descriptive line and # of points (INLEV input form) DATA	
	c) Descriptive line and data type (CTD-plot input) DATA	
Typical input choices:	1 3 1 1 2 3	1 0 0 0
Batch files:	RNGIN MKGIN	RNDIN MKINP

MK4LF.EXE, .FOR

This program manipulates IS data in several useful ways. Although most of its work can be accomplished separately using various features of LEVM, it carries out several helpful operations automatically. Briefly, given data in rectangular form at a particular level, MK4LF first produces files of data at each of the four IS levels: PLZ.DAT, PLM.DAT, PLY.DAT, and PLE.DAT, which are particularly useful for plotting and direct examination. For each of the four possible levels, the pertinent file contains six columns: the frequency, f , (in Hz or radians) the log₁₀ of the frequency, the real part, the imaginary part (all positive), the log₁₀ of the real part, and the log₁₀ of the imaginary part. As discussed below, MK4LF also can calculate log-log slopes of response curves, with or without subtraction of a zero-frequency value. As supplied, MK4LF is limited to a maximum of 1001 points, but it can be recompiled for a smaller or larger limit.

MK4LF uses the LEVM-produced file **LINOTR** as its input. This file contains five columns: the frequency in Hz, the original real and imaginary parts, and the fit real and imaginary parts. Four input variable values will be requested during a run: IFC, CELCAP, ICH, and SCALE. The value of IFC selects the type of input and output frequency columns. If the CELCAP input value is set to zero, CELCAP is set at the permittivity of vacuum; otherwise, any appropriate value may be entered. The input variable ICH (± 1 or ± 2) is used to select the original data (Cols. 2 and 3) for transformation (± 1), or use ± 2 to transform the fit data. The input variable SCALE is usually set to ± 1 but may be used to multiply all data by an arbitrary constant.

Any input file of the above form (with the first three columns containing original data and, for example, the last two columns containing actual zeros) may also be used if it is named LINOTR. When using LEVM to produce LINOTR, one can set IRE to -11 in order to ensure that the first column of this file is in Hz rather than in radians. The LINOTR file may be produced without fitting by setting MAXFEV to 1 and running the LEVM input file which contains IS data of interest. Input choices in MK4LF allow one to use a LINOTR file with frequency in Hz or rad/s and produce output files with either type.

MK4LF also gives the user the possibility of subtracting the zero-frequency value of the real part of the data at the original level from the data. This is particularly useful in forming such results as a column of $[s(f) - s(0)]$, where s is the complex conductivity. If subtraction is not needed, enter zero for the requested Z(0) or Y(0) input choice. The output file PLYZSUB.DAT contains the following eight columns: frequency, log freq., subtracted real part, its log, real part, its log, ϵ_s'' , and its log. Here, $\epsilon_s'' = (\sigma - \sigma_0)/(\omega \epsilon_v)$. **These last two columns are of most interest for Y or σ data when one is interested in the dependence on frequency of the dielectric loss, but σ can also be Y or Z here when ϵ_v is replaced by Celcap.**

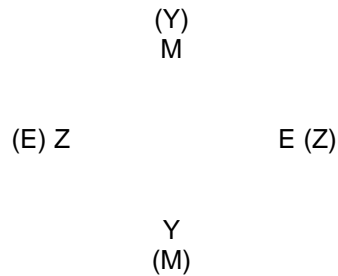
Also MK4LF calculates the logarithmic slopes of the real and imaginary parts of the chosen input data. One can either obtain slope values at given input frequencies sequentially or obtain a table of all slopes except those at the end points of the data. See input choice descriptions shown when running the program. To stop the program when entering sequential frequency choices, enter a value of zero. The slopes are calculated as numerical derivatives using points just before and after a given frequency. Thus, the closer together the frequency values, the more accurate will be the calculated slopes. A value of about 10 points per decade usually provides sufficient accuracy, but 20 points per decade is better.

The S slope values appear in the files PLSLOPES.DAT and PL3DSL.DAT. For PLSLOPES, the first value shown is the data (row) number (2,3,4...). The second is the value of Z(0) or Y(0) used, and the third is the value of the frequency (in Hz). The next column contains the log10 value of the frequency, and the next two are the slope of the subtracted real-part data and the slope of the non-subtracted real-part data, S. The last column contains the slope of the imaginary part of the data. For PL3DSL, the columns contain log freq., the subtracted-data slope, the non-subtracted real-part slope, the difference 1 - S, its log10 value, the imaginary-part slope, and the inputted value of the temperature.

Summaries:

Col. #	1	2	3	4	5	6	7
PLSLOPES:	row #	Y(0)	f	log(f)	S subt.	S	S im
PL3DSL:	log(f)	S subt.	S	1-S	log(#4)	S im	Temp

The input data level must be specified as Z or Y when running MK4LF, but it may actually be E (the complex dielectric constant) or M (the complex electrical modulus). This is because of the following useful formal correspondences (see Ref. 25):



The level symbols in parenthesis and those without parentheses form two circles showing possible correspondences (e.g., Z and (E)) and closest transformations (e.g., Y, Z, M, E, Y or Y, E, M, Z, Y).

This diagram shows that if one uses E-level data in LINOTR but tells the program that it is Z-level, then the PLZ, PLM, PLE, and PLY output files will actually contain not Z, M, E, and Y data but E, Y, Z, M data, respectively. Similarly, if one uses M-level input data, but tells the program that it is Y-level, then the PLY, PLZ, PLM, and PLE files will actually contain M, E, Y, Z data. Thus, the input file can involve data at any of the four levels. Since, however, LEVM allows one to transform from one level to another very easily, it is often most appropriate to use Z or Y.

MKFREQ.EXE, .FOR

This self-explanatory program asks for a few inputs and produces (in the file LVFREQS) a LEVM input data file consisting of a sequence of frequencies (selected by input choices: additive and multiplicative sequences possible) with all real parts set to unity and all imaginary parts set to zero. It is particularly valuable for simulation: i.e. when one wants to examine the predictions of a certain circuit model choice over a given set of frequencies (no CNLS). The full simulation procedure has been automated by means of the file SIM.BAT. See its discussion below.

OUTPARM

This output file can be produced when the K or R Circuits are used for fitting. It has been extensively revised to include more useful output data in simple form. See the K-Circuit writeup in Section V for a full description of OUTPARM.

OUTDRT (** logarithmic values when IPRINT < 0)

This is a DRT output file which contains information in appropriate form for plotting in different ways. In addition, the K and R Circuit models can now make files which allow one to plot estimated-standard-deviation rectangles around each DRT point. See the K-Circuit writeup for full information.

POPDIR.COM, PUSHDIR.COM (useful in LEVM, not LEVMW)

These programs may be used to run a program from any subdirectory (examine RNCORD.BAT for an example). A CD command to move to the subdirectory where the program to be run resides must be present in the batch file.

Definitions of some LEVM Batch Files

MKLVG MS Fortran LEVM MAKE batch file (uses the actual make file LEVMG)

MKGIN Same as MKINP, but for GINPUT

MKINP Used to help prepare an input file for LEVM. First, see discussion of DINPUT, above. Suppose that you have input data in three columns in a file DATAIN. The command: MKINP ATMP DATAIN MYINPUT uses DINPUT to produce the final input file MYINPUT. Note that after MYINPUT is made, the number of frequency values must be entered correctly in the M position, line 3, of the top part of the file (NOT necessary when using MKLIO), and such quantities as NEG,FREQ,N selected properly. See also **GINPUT and RGIN.BAT** which runs it.

ONEMKE, ONEFIT, ONEFIT2	See Section 7
PLV	Print PNTOUTL
RLVP	Scrolls PNTOUTL on screen
RNCORD %1	Run CORDER interactively. %1 is input file name
RNDIN %1	Run DINPUT. %1=datafile; e.g.: RNDIN DATFL
RNDTRAN %1	Run DATATRAN interactively; %1 is input dataset name
RNFOR %1 %2 %3	Run, time .exe pgm. %1=run file (no .exe); %2 = your inputfile; and %3 is name of input file required by the program (e.g., INFL for LEVM).
RNL %1	Run LEVM. %1= inputfile; e.g.: RNL MYINPUT
RNLO	Runs LEVM using the input file OUTIN
RNLVOI %1	Runs LEVM input file %1, then runs resulting OUTIN file, uses this for input and produces another OUTIN file. All three output results appear in PNTOUTL. Necessary that IRE < 0 in %1. Useful for checking consistency of fitting, especially long runs (e.g., use it for an overnight run)
RNTEMP	Runs MKLTP.EXE. See NELEM=33 choice in Section 4. When this is used, MKLTP transforms Cols. B,C, and D of the LINOTR output file to log10 form; useful for temperature-dependence plots.
RNTFOR %1	Run and time %1 .exe or .com pgm (no extension)
RPL %1	Run, time LEVM; print output

CPTRAN

This useful batch file invokes the program CPTRN.EXE and does several things. It uses the results of a LEVM fit run in the following ways. The LEVM run, which involves a particular choice of DFIT, must incorporate a value of IRE of -10 or (algebraically) less, so that both OUTIN and LINOUT are produced. When you start CPTRAN and the initial menu appears, choose A and do not enter a return. Then a choice is presented for transforming the data in LINOUT (both input data and fitting-result data) from their original level (Z, M, E, or Y) to any one of these, which we will designate as X (X includes the original level so choices below can even be used for it). Be sure to use only upper case letters in specifying your choice of X and enter a return after your choice.

Next, a choice for possible scaling of the transformed data is presented. If a value different from zero (gives unity scaling) is entered for the scaling factor, all real and imaginary data values are divided by this factor and a file CPNDFL.X is made. Even with an input choice of zero, this file is made and saved. The resulting file involves columns for frequency, input real and imaginary, and output real and imaginary, and is thus useful as input to a commercial plotting program. This option is particularly valuable for normalizing complex plane plots with, e.g., the DC resistance, so that the needed x-axis length is then unity.

Next, CPTRAN allows one the choice of producing a transformed output data file, CPDFL.X, where X is again your original choice of M,Z,Y, or E. This file is in the same format as LINOUT and may be used directly for plotting. Then, the files CTDP.X and a temporary AUXPNTL file for the transformed data are also made, and CTD is finally invoked, allowing 2-D and 3-D plotting. Enter the file name CTDP.X in CTD.

It is IMPORTANT to note that the new data files, CPNDFL.X and CPDFL.X, are erased at the beginning of each new CPTRAN run. Thus, when you are doing more than one such run, change the names of the files you want to keep after each run before the next such run (e.g.: do COPY CPNDFL.* MYDATA.* at the command line).

To summarize, if your original fit is, for example, at the impedance level, CPTRAN produces an output data file involving the same data (no new LEVM run), but with them now transformed to any other level (and possibly scaled), and then it carries out a plot of the transformed data (it plots both the original and the fitted data at the new level). You can then continue, if you wish, with another plot at a different level and scaling, etc. without having to leave CPTRAN (you can thus produce X = Z,Y,M and E saved output files in a single CPTRAN run). A better alternative for making data files at all levels simultaneously is to use MK4LF.

In fitting, it is always a good idea to examine the complex plane shape of the fit results and/or original data at all four levels. CPTRAN makes this easy. It allows both complex plane plots and Bode plots, but not residual plots. To obtain the latter, one would have to make a new LEVM run with a different choice of DFIT and then invoke CTD. CPTRAN also uses the programs COLOR2.BAT and GETLETR.COM. Use no return after the A or B choice in the first menu but use returns for the next choices. For E and/or M level plots, set CELCAP in the LEVM input, if known. Otherwise, the default value of unity will be used.

SIM

This file automates the process of circuit simulation (see MKFREQ above). Before running SIM, prepare the top part of a LEVM data input file by renaming ATMP or one of the others to SIM1. This file should contain your circuit choice, initial nonzero parameter values, number of frequency values, etc., and it must have MAXFEV = 0 and IRE < -9 (e.g. a value of -11). At least one value of NFREE in LINE 5 should be nonzero.

When SIM is invoked, it first runs MKFREQ; so inputs to define the set of frequencies must be entered, with the total number of frequencies the same as that in SIM1. Also SIM runs DINPUT, so its interactive input (e.g., 1 ENTER and then 0,0,0 ENTER) must be entered as requested. The output file SIM2 contains a LEVM input file still with MAXFEV = 0 and without final calculated data values), and the new OUTIN is the same with MAXFEV = 91 and with data values calculated from the input choices at the given frequencies. It is automatically run, leading to the files PNTOUTL and AUXPNL. Then, CTD is automatically invoked, allowing the shape of the output to be examined and plotted in various ways.

Through the use of SIM [followed by CPTRAN (or use MAXFEV=1 in the LEVM input file and new choices for DFIT to produce output data files at any level in LINOTR; or use MK4LF)], it is extremely easy to produce data files (if desired) and plots at the Z,M,E, and Y levels for any circuit-parameter combination within the scope of LEVM. Thus the shape of the response (in the complex plane or as Bode plots) can readily be explored at any level. Before running LEVM when the guesses for the circuit and/or parameter values are quite uncertain, it will usually be quite helpful to use SIM and, if needed, CPTRAN, to allow comparison of response shapes with the shape of the original data to be fitted. Note that SIM.BAT does lots of things while running and may produce some transient screen output. Ignore it except when you must push ENTER to continue.

Note that whether the input parameter FREEQ in SIM1 is set to F (frequency input in Hz) or left blank (frequency input in angular frequency), the final set of frequencies and data in OUTIN always involves angular frequencies, but IRE= -10 or -11 determines the frequency character in various output files.

LEVM INPUT FILE AND PARAMETERS

This section contains a full description of all of the LEVM input parameters and options. It provides a discussion of the inputs of each separate line in the input file.

Important run-time choices are invoked by the values used for the Line-2 parameters of the LEVM input file, particularly IOPT, CELCAP, ROE, IFP, and IRE. See the definitions of these quantities given below. The shortest and simplest fitting runs and output will be achieved with IOPT, CELCAP, ROE, IFP, and IRE all set to 0 and IPRINT = 0. I recommend IPRINT = 1 and IRE = -1 or -11, however, for most runs. Although the above choices are appropriate for initial fitting runs, after a reasonably good fit has been found one should try other values of the choices, first IOPT = 9 (although convergence usually occurs in 3 or 4 iterations). Many of the choices lead to sequential runs. In all such cases there will first be an ordinary CNLS run. If IFP = 1 and IOPT = 9, the results of the first CNLS run are used as inputs for a short FPWT run (usually better than PWT), and then a final optimization run will be carried out. The outputs for each of these runs will occur sequentially in the output file. No special iteration is needed when ROE is non-zero. Incidentally, the optimization procedure works by making the SD WR and SD WI quantities close to equal.

The ROE parameter deserves some further discussion. When it is non-zero, an approximate robust regression Huber M-estimator, instead of pure least squares, is invoked. See [Exploring Data Tables, Trends, and Shapes](#), by Hoaglin, Mosteller, and Tukey, Wiley 1985, pp. 293-301. The actual choice ROE = 0 is equivalent to ROE = infinity in the program and leads to ordinary least squares fitting. Any non-zero value of ROE yields a combination of least squares for the smaller residuals and least absolute value for the larger ones. When ROE is non-zero, values between say 0.1 and 2 or 3 are usually appropriate.

One problem with robust estimation is that statistical estimates are generally more inaccurate for it than for ordinary CNLS fitting. In particular, SIGMAF is no longer related to a pure sum of squares (although the SD values appearing at the end of the ROE output are). Thus, SIGMAF is no longer necessarily a good estimate for overall goodness of fit. But one is primarily interested in the best possible estimates of the parameters, and SIGMAF and parameter relative standard deviation estimates can still be used for comparing results with differing values of ROE.

Although I have improved the statistics for the M-estimator so that its estimates for parameter relative standard deviations are better, they still cannot be considered more than semi-quantitative approximations, generally erring on the optimistic side. Thus, when runs are made with various values of ROE, one should pick that value that yields the lowest value of PRSDAV or PRSDRMS in the ROE output. One can investigate this for oneself using the AZC data provided (with IPAR set to < 0). It leads to best results for ROE = 1.0 for PWT and FPWT and for ROE = 0.19 for UWT. I have not found, however, that ROE minimization as above always leads to results as good or better than those of IOPT optimization in complex data runs. Try both. ROE non-zero might be primarily used for fitting of real or of imaginary parts of the data alone, cases where IOPT does not apply. For fully complex data, the use of IOPT non-zero can lead to very substantial improvements in the parameter estimates, particularly for UWT.

It may sometimes be better to use generalized least squares (GLS - see Giltinan and Ruppert, J. Pharmacokin. Biopharm. 17(1989)601) than the ROE > 0 option. See the choices under CELCAP below. GLS involves an automatic sequence of fits: first a NLLS (Nonlinear least squares) or

CNLS fit with the power-law parameter, Xi (and the parameter U, if present) fixed; then follows a fit using the results of the first fit with all fixed parameters but the power-law parameter, Xi (and U if present), free. The next step involves a fit with the usual parameters free but Xi (and U) fixed at the values obtained in the prior fit, and so on until Xi (and U) no longer change. The final result involves a fit with Xi (and U, if present) fixed at their converged values. GLS is considerably slower than ELS (extended least squares, the usual LEVM method), and ordinary ELS (set CELCAP ≥ 0) is preferable except for very poor data.

INPUT PARAMETERS FOR LEVM

The input parameters for LEVM control a wide range of functions of the program, including data types, data conversions, fitting model choices, and other NLLS/CNLS fit parameters. The parameter names all refer to the actual variable names in LEVM. All input parameters are entered in formatted FORTRAN line images; thus their exact spacing and positioning are crucial to ensure correct results. Each line will be presented and described sequentially, with notes on the column numbers of each field. Any parameter noted to have a default value may be left BLANK. It will be helpful to study the typical data set inputs supplied.

As the data arrays are currently dimensioned, LEVM.EXE can accommodate up to 2002 data points (1001 real and 1001 imaginary components) and up to 42 free parameters. LEVM will require some redimensioning (see the SIZE.INC file) if any of these values are to be exceeded or if one wants to use a smaller LEVM.EXE file.

Note: the main output of LEVM (more than that which appears on screen) appears in the file PNTOUTL; some secondary output occurs in AUXPNTL. When MAXFEV is set to 0 in the input, exact model predictions are produced in these files, in the dataform used for the input file. No fitting is carried out for this choice.

BEGINNING OF TOP PART OF INPUT FILE

LINE1 [FORMAT(A80)]
DESCRIPTION OF RUN

COLS 1-80
ALPHA

ALPHA Alphanumeric description of run: This is printed as a header for the output files (may be left blank). It is essential, however, that this be the first line in the file. No blank lines are permitted above it.

LINE2 [FORMAT (I4, 2X, 7A1, D10. 4,A1, I10, 1D10.4, 2I6, 1X, D10.4, 2X, D10.4)]
DATA AND FIT TYPES

COLS	1-4	7	8	9	10	11	12	13	14-23	24
	IOPT	DINP	DFIT	PINP	PFIT	FREEQ	NEG	FUN	CELCAP	DATTYP
	25-34	35-44	45-50	51-56	59-67	69-77				
	IPAR	ROE	IFP	IRE	$\epsilon_{C1\infty}$ *	ϵ_{C10} *				

IOPT Turns on CNLS optimization iteration (only for complex fit)
 IOPT = 0 - No optimization
 IOPT > 0 - Sets maximum number of allowed optimization iterations (9 is reasonable)
 When IOPT > 0, screen output appears for each of its iterations: the output is: 2M, No. of free parameters, and the fit quality factor (FQF)
 When IOPT < 0, CNLS convergence accuracy can be controlled (no optimization). It is approximately $10^{**(-|IOPT|)}$. Thus values between -3 and -14 are appropriate, with -8 or -9 reasonable for data with noise. When IOPT = 0, the maximum number of convergence iterations are carried out.

* ** These quantities only appear in OUTIN for the IRE < 0 choice, the FUN = O circuit selection, and P(5) > 0. They also require the MODE = -1 choice.

- DINP** Input data system (same choices as in DFIT but can be different)
- DFIT** Fitting system
 Data may be entered in any of the following forms and may be converted to any of them for the fit (see CELCAP)
 Z - Impedance <default>
 Y - Admittance = $1/Z$
 M - Modulus = $i\omega Z$
 E - Epsilon (complex dielectric constant) = $1/M$
- PINP** Complex input format (same choices as in PFIT - but PINP and PFIT can be different)
- PFIT** Complex fit formats
 The complex data may be entered and then fit in any combination of the following formats:
 P - Polar (magnitude, angle in radians)
 D - Polar (magnitude, angle in degrees)
 L - Polar natural log: ($\ln(\text{magnitude})$, angle in radians)
 R - Rectangular (real, imaginary) <default>
- FREEQ** Input frequency format which provides the option to enter the frequency in Hertz. **See table on p. 0-6 of this manual.**
 All fitting is done with angular frequency ω (omega).
 F - Input frequency is in Hertz
 blank - Input frequency is angular <default>
- NEG** Selects sign change of 2nd column of input data (e.g. imaginary part)
 N - Multiplies Imag. data by -1 as inputted from file
 Blank - No action taken
- FUN** Selects model fitting function to be used. Valid models are A - K, O, R, S, and T. For more information, see Section 5 on model specifications.
- CELCAP** Capacitance of empty measuring cell (in Farads) - required when converting between the Z-M and E-Y systems. It is not necessary to provide a value for Z <--> Y conversions and fits, but one should be provided for Z,Y <--> M,E conversions and also for all fits in the M-E system, even if the data is in M or E format (since all the circuit models calculate Z and then transform it to the system being fitted). The default value is <1.0>. Set to permittivity of vacuum (8.854188×10^{-14} F/cm. in **Gaussian** units), when dealing with dielectric or conductivity data (use value of the input parameter, **M**, < 0; see line-3 information). See also p. 0-16.

The generalized least squares fitting option is selected if CELCAP < -2.

CELCAP = -1: Sequential ELS (IRCH = -2)
 CELCAP = -2: As in -1 choice but U also free
 CELCAP = -3: Sequential GLS (IRCH = 2)
 CELCAP = -4: As in -3 choice, but U also free

DATTYP Type of data fit desired: C, R, or I.

C - CNLS: combined fit of both immittance data columns simultaneously (real and imaginary or, e.g., magnitude and phase).

R - real fit of second input data column (first immittance data column: real or magnitude).

I - real fit of third input data column (second immittance data column: imaginary or phase).

IPAR When zero: The Residuals/Model output columns in AUXPNTL and AUXPNTL1 are calculated (normalized residuals). ** Residuals = data - model predictions.

When > 0, the output column: Residuals/Model in both are changed to Residuals/Uncertainty (i.e. weighted residuals) without changing column heading. Useful for plotting, especially when Xi is not unity. See further discussion in Section 2 under AUXPNTL1.

When < 0, an extra line of parameter relative standard deviations occurs in the output, based on the input values of the free parameters of the fit. If the fit yields estimates equal to the input values, the values in this line will be zero. This feature is useful when exact values are known (as in simulation studies). To invoke it for the AZC input file, set IPAR = -1 and set the parameter estimates to their exact values.

ROE; A/L When ROE > zero, turns on robust regression (after first NLS main fit is completed). A subsequent fit associated with non-zero IFP will use such regression, but IRE and IOPT choices do not. Reasonable range: 0.1 to 3.0. For LEVMW, set ROE on screen.

When ROE > 0, the PNTOUTL file lists the value of $RKE=ROE \cdot \text{SIGMAF}$ and of MN, the number of data points (out of M) which involve least squares minimization (the rest involve absolute value minimization). Also, **no correlation matrix** appears in screen output when ROE = 0. (Use PNTOUTL output or set ROE < 0 for screen view).

***When the input value of ROE is **negative**, it should be entered as the negative of the **A/L** cell constant with a dimension of cm (see p. 0-16) when the data are at the impedance level (ohms) and not of specific form (ohm-cm). Then, in the next fit run, the real and imaginary parts of the impedance are multiplied by A/L, resulting in specific format. For the next run, the -A/L value will be automatically set to zero and may be changed to a positive value if robust regression is desired. For specific data, it is appropriate to set CELCAP to the permittivity of vacuum so that dielectric level results are of relative dielectric constant character (called dielectric constant here).

IFP When zero: nothing new happens.

When > 0, activates direct function weighting (e.g., FPWT) after first NLS main fit is completed. Use any positive integer. Note that function weighting may alternatively be invoked in the first main fit section merely by setting IRCH < 0 (provided |IRCH|

> 1).

When < 0, turns on iterative function weighting (after first NLS main fit is completed) - like PWT but uses calculated function values for producing weights. The value of |IFP| sets the maximum number of iterations allowed. Reasonable values are 7 or 8, but IFP > 0, which doesn't involve a separate iteration sequence, is nearly always a better choice since it converges quicker (to the same results). Thus, there is little reason to use the IFP < 0 choice; it is included primarily for comparison purposes.

IRE When zero: nothing new happens.

When > 0, turns on residual-weighting (RWT) iteration (after first NLS main fit is completed). The value of IRE sets the maximum number of iterations allowed. Reasonable values are 2 or 3.

The use of residual weighting with IRE = 2 is sometimes useful in obtaining good parameter estimates. But the estimates of the standard deviations of the parameters provided by the program under these conditions are not as reliable as the ones obtained when IRE = 0. Thus, it is usually impossible to know for sure whether residual weighting has, in fact, given better estimates or not. Only when synthetic data for which the correct parameter estimates are known (such as AZC) are available can one make valid comparisons. This state of affairs, of course, severely limits the utility of residual weighting, and will do so until more reliable calculations of the standard deviations of the parameters are available.

- When < 0, no residual-weighting, but the output file OUTIN is produced when a fit is carried out. It contains the fit estimates of the parameters and may be used as an input file for the same data and model. Before making OUTIN, the program saves the input file used to make it as LASTIN. If one runs RIN, OUTIN is replaced by LASTIN. When IRE < -9, the files LINOTL, LINOTR, LINOTRL, LINOUT and LVOUT are also produced by a LEVM run. When also ATEMP = -2 or -3, the file LINOTCON is produced as well.
- When IRE > -11, all output files involve the angular frequency (or time) used in the fitting, but when IRE = -11, they all involve ordinary frequency values in Hz.

** Additional MAXFEV = 1 or 2 choices (in Line-3), then:

Scaling transformations of fitted or simulated data (data in rectangular form)

If P(39) > -10**17, Re and Im data values are multiplied by |P(39)| when P(39) not 0

If FUN = O (O circuit), then

If P(39) < -10**17, Re and Im data values are multiplied by P(5), where P(5) = ρ_0

If P(36) is not 0 and DFIT = YY, $\sigma' \rightarrow \sigma' - \sigma_0$.

If FUN not equal to R and P(38) not 0:

If P(38) < 10**17, multiply freq. column of data by P(38)

If FUN = O (O circuit), then

If P(38) > 10**17, multiply freq. column of data by P(6), where P(6) = τ_0 . *****

LINE3 [FORMAT(9I5, D10.4, 5X, D10.4)]
 DATA, WEIGHTING, AND FITTING SPECIFICATIONS

COLS	1-5	6-10	11-15	16-20	21-25	26-30	31-35	36-40	41-45
	M	N	MAXFEV	NPRINT	IRCH	MODE	ICP	IPRINT	IGACC
	46-55	auto	auto						
	ATEMP	SDWC	SDRC						

M Number of data points (frequencies). Maximum value: 1001
 Set M < 0 initially to make CELCAP = permittivity of vacuum. Program will reset M positive.

N Total number of model parameters (free + fixed). Minimum absolute value 34, maximum 50. When <0, P(33) and P(34) are used as reference parameters: see p. 5-2.

MAXFEV Maximum number of evaluations of the model allowed (approximately 1/2 the maximum number of iterations).

A reasonable value is 49 to 99, but it can be much larger. If convergence fails in the allowed number, a message will appear in PNTOUTL and on the screen.

The following three choices are most appropriate when IRE is set to -1 or less, causing the file OUTIN to be produced when a fitting run is carried out. The results of the choices below then appear in OUTIN (and in other output files if IRE<-9). After carrying out a run with one of the choices below, it is often appropriate to do a run using the OUTIN file produced, but results occur in the output files whether such a run is made or not.

When MAXFEV = 0, no fitting is performed but calculated model data corresponding to the parameter choices is written to the output files. Use MAXFEV = 0 to calculate model (circuit) predictions for given parameter values. Useful for plotting and preliminary investigations. See discussion of files OUTIN and SIM in Section 2. **Do not** set DATTYP=I when MAXFEV=0, and **you must** set NFREE(I)=0 or 2 for any negative parameter choices.

When MAXFEV = 1, no fitting is performed but the data are changed from their current level (set by DINP value) to a new level (set by DFIT). Thus if these quantities are ZZ, change them to ZY and run with MAXFEV = 1. Results, including the data in OUTIN, will be at YY level. Particularly useful for transforming data and producing new results for plotting in the output files generated when IRE<-9. ****See additional choices at bottom of p. 3-6. *******

When MAXFEV = 2, parameter **subtraction** is possible. See discussion of subtraction in the Preface. Use this choice after a fit has been made with such quantities as RP and CP free. Then with MAXFEV = 2 and IRE= -10 or -11, set NFREE = 3 for all such parameters whose effects are to be subtracted from the data (**also set NFREE = 3 for any NDE parameter associated with the parameters to be subtracted**) and make a run. The resulting new input file (OUTIN) will not involve those parameters whose NFREE values were set to 3, and their effects will have been subtracted from the data. Another fitting run can then be made using the changed data. See the correspondence between NFREE numbers and **word descriptors** (used in LEVMW) on p. 3-13.

NPRINT Causes intermediate fitting results (free parameter estimates) to be printed every NPRINT iterations. If NPRINT < 1, then there is no intermediate printing. ** For NPRINT < 0, e.g., NPRINT = -4, screen shows only every 4th line. Best to use even numbers.

IRCH Selects form of data weighting: (here "Weights" refers to the standard deviations, RW(I), used to calculate actual weights). The weighting variance corresponding to say REAL(RW(I)) is $v_i = \text{REAL}(\text{RW}(I)) * \text{REAL}(\text{RW}(I))$, and the corresponding weight is $1/v_i$. See weighting definition sheets at end of this Section.

- 0 Reads input weights (Lines 7 when present) into RW(I). Use DINPUT.BAT for data preparation. ** Also involves Unit weighting when **IRCH** 0 or 1.
- 1 Unit weighting (UWT): $\text{REAL}(\text{RW}(I)) = \text{IM}(\text{RW}(I)) = 1$
- 2 Weights assigned as shown on sheets at end of Section 3. Involves the data values Y(I) and the parameters U and Xi. (For proportional weighting, Xi = 1 and U = 0: PWT). CNLS results generally somewhat biased for this choice.
- 2 Weights involve YC(I), which is calculated from the model (the fit model prediction). (For proportional weighting, Xi = 1 and U = 0: FPWT). Recommended, less bias.
- 3 Modulus weighting using Y(I). (MWT when U=0 and Xi=1, fixed). See end sheets and J. Electroanal. Chem. **178**(1984)11.
- 3 Modulus weighting using YC(I). (FMWT when U=0 and Xi=1, fixed). Biased.
- 4 Special FRA weighting using Y(I). See article by Spinolo in Ref. list.
- 4 Same as IRCH=4 but uses YC(I).
- 5 FRA 1250,1286 weighting using Y(I) and reference resistors (P(33) etc., see discussion in Section 5). Also see article by Orazem et al. in Ref. list. Xi and U not used with $|\text{IRCH}| > 4$. Also, do not use Optimization (set IOPT=0 or negative).
- 5 Same as IRCH=5 but uses YC(I).
- 6 Same as IRCH=5 but weighting constants set for FRA 1250 and PAR 273 potentiostat.
- 6 Same as IRCH=6 but uses YC(I).

MODE This fixed parameter now plays several important roles. Let $MDE=|MODE|$. When $MODE \neq 4$, the scaling of the model parameters is computed internally. Otherwise, an external scaling procedure is called for during LEVM operation which requests scaling exponents). Not usually necessary.

Next, when $MDE = 0$ or 1 , the input T (tau) variable is treated as usual. BUT, when $MDE = 2$ (use for CSD response only), for many distributed circuit elements (DCE's, see next section), the input T quantity is redefined as an effective dielectric constant, ϵ_r , and T in the response formula is replaced by the new relaxation time TT , where then

$$TT = T * CELCAP * (R0 * R0) / R,$$

and $R = \Delta R = R0 - Rinf$, and $R0 = Rinf + \Delta R$. In most cases of interest, $Rinf = 0$.

In addition, when $MODE \geq 0$, fits are of either CSD=CSD0 or DSD type, **BUT**, for a few important response formulas, when $MODE < 0$, they are of only CSD1 type. See Appendix 1 of the Preface; Section 4, the NELEM = 6,7 choices; and the final Circuit-O discussion.

Finally, for a few DCE's the choice $MDE = 8$ leads to the calculation of transient rather than frequency response (e.g., NELEM = 14 or 34). Also see p. 5-33 for the $MODE = -6$ choice.

ICP - When non-zero, invokes request for new values of free parameters to be entered during LEVM operation. Leave zero unless you want to change them on next run.

IPRINT Selects level of output (especially in PNTOUTL).
 0 - minimum output
 1 - standard output. (Note that residuals only appear in AUXPNTL for this choice of IPRINT and also residuals/calculated results are output - but see IPAR)
 2 - more output - e.g., includes residuals and residuals/measured (in PNTOUTL) and residuals/calculated (in AUXPNTL - but see IPAR).
 3 - maximum output, includes iterative step output. Requires plenty of disk space for long output file, PNTOUTL.
 <0 - when IRE<-9 as well, OUTIN is also produced.

IPRINT<0 (|IPRINT| is actually used) causes the frequency column in AUXPNTL1 to be log (freq.), where the character of freq. is determined by IRE=-10 (angular) or -11 (Hz).

IGACC Selects relative accuracy of EDAE, GDAE, and other response functions (see especially O-circuit choices) requiring integration. (i.e., 2 gives 10^{-2} ; 6 gives 10^{-6} , etc.). For medium accurate data, use 2. If a reasonably good fit is found, then use its parameter estimates (in OUTIN) as inputs with IGACC now set to 3 or 4. Only use 4 to 7 (much slower) for very accurate data.

When $IGACC < 0$, $IGACC = |IGACC|$, but weight shaping is invoked. Then when LEVM is run, you will be requested to enter four corner frequencies, fl1, fl2, fh1, and fh2. The weighting, set by other inputs, is reduced proportional to f (frequency: units Hz or radians, whichever was used in input) between f_{min} and fl1, and is reduced proportional to $f^{0.5}$ between fl1 and fl2. Similarly, the weights are reduced proportional to $f^{0.5}$ between fh1 and fh2, and proportional to f^{-1} between fh2 and f_{max} . There is no effect in the region $fl2 \leq f \leq fh1$. To eliminate say the high frequency roll-off, just set fh1 and fh2 $> f_{max}$, and similarly for eliminating low-end roll-off.

These possibilities are useful when some extreme low and/or high end data values are appreciably less accurate than mid-range ones. By means of weight shaping, their effects can be reduced without needing to eliminate these data points, and yet they can still contribute, in reduced measure, to the fit.

ATEMP Absolute temperature (e.g., Circuit H) or choice variable.

<0 - In K, O, and R circuits causes appropriate parameters to be defined as dielectric constants rather than as capacitances. See O-Circuit discussion in Section 5.

-2 - With CELCAP not equal to unity, this choice causes the output file LINOTCON to be produced. It is particularly appropriate for complex conductance data fitting.

-3 - With DATTYP=R and fitting with the G circuit, again causes (a different version of) LINOTCON to be produced. If the independent variable is temperature and an Arrhenius plot is of interest, LINOTCON contains the needed results of the fit for plotting. But see OCKt. Also see ATEMP values set on pp. 4-7 and 4-8.

**SDWC,
SDRC**

These quantities are the prior fit values of the SD's shown as SD WC and SD RC in the LEVM outputs (e.g., screen and PNTOUTL). They only appear automatically in OUTIN after a fit; not entered in preparing input file.

Fitting Tolerances and Parameters

The following quantities are no longer set in the input but are given fixed values in the program itself. They have been selected to produce stringent convergence criteria, needed for low-error data and for some DRT estimation. See IOPT < 0.

- DRSS (Relative) Numerical derivative step size: Default = 1.0D-8
- FTOL Convergence parameter for sum of squares:
LEVM terminates when the relative error in the sum of squares is < FTOL.
Default = 1.0D-15. FTOL=XTOL= 10**(IOPT) when IOPT < 0.
- XTOL Convergence parameter for the vector of free parameters:
LEVM terminates when the relative error between the vector of the free parameters and its values for the last iteration is < XTOL. Default = 1.0D-28.
- GTOL Convergence parameter for the vector of free parameters: Default = 0.0.
- FACTOR Positive number used in determining the initial step bound. This bound is proportional to FACTOR*(L2 norm of P) where P is the vector of free parameters. Default = 100.

LINE4 [FORMAT(5D15.8)] MODEL PARAMETERS

This "line" is actually seven to ten real lines (each containing up to 5 parameter choices) long.

P(I), I = 1, N,

where N = total number of parameters (32 to 50)

P(I) is the input value of the Ith parameter for the model being used. Care should be taken in giving parameters values of zero since this is equivalent to removing an element from a circuit entirely, rather than having one with a value of zero. For further information, consult the section on circuit models. The number and function of parameters are determined by the model selected by the FUN parameter on line 2 of the input file.

FIXED PARAMETER CHOICES

Some P(I) values with $29 < I < 41$ may be set negative or non-zero in order to invoke a particular choice. When negative, $|P(I)|$ is actually used in any calculations with the input value.

Parameters P(33) up to possibly P(39) may be used for various weighting situations - see Section 5: Parallel, Reference, and Weighting Possibilities for a detailed discussion. Below are listed other orthogonal parameter choices.

- P(30) Used for different element types in Circuits K and O. See Section 5 of this manual and discussion of P(37) below.
- P(31) When fixed and set to -1,-2, or -3, used to invoke and define the plotting of uncertainty rectangles around DRT points (K circuit or R circuit only). See discussion of DRTLSD.DAT file in K-circuit write-up.
- P(33) > 0 Only operative for the R circuit. Used for DRT regularization-type inversion. P(33) is the value of the regularization parameter, lambda. See R-circuit discussion, p. 5-41. See also P(33) discussion in the O circuit material.
- P(34) < 0 Causes parameter gradients to appear on screen at each iteration. Valuable to show which parameters are well determined during fitting iteration. No gradient output for 0.
- P(34) > 0 Used in setting up LEVM input files for DRT fitting and inversion (Circuits K and R). See detailed discussion of P(34) > 0 choices in R-circuit discussion.
- P(35) > 0 Used in K and R circuit DRT calculations. Only needed when some of the DRT parameters are taken fixed rather than free. P(35) should then be set equal to the total number of DRT parameter pairs used in the fitting, but it can also be used for other DRT fitting situations as well. See R-circuit discussion.
- **P(36) May be used in setting up a LEVM DRT file. See the discussion under the P(34)=1 choice in Section 5, R circuit. Also used in the K-Circuit for CP definition and use.
- P(37) This parameter controls the character of the fitting parameter P(30) in the O circuit. If -4, P(30) is a series inductance; if -16, P(30) is a series capacitor; and if > 0, P(30) is a series resistance. Same for R circuit, but in addition, for P(37) = 0, P(30) is a parallel conductance.
- **P(38) Important DRT choice parameter. See the list of possibilities in the R-circuit discussion.
- **P(39) May be used in setting up a LEVM DRT file. See the discussion under the P(34)=1 choice in Section 5, R circuit. P(39) is also used to set the regularization order when P(33)>0. ** If P(39) < 0 and MAXFEV = 1 or 2, $Y'(I) = |P(39)|*Y'(I)$ and $Y''(I) = |P(39)|*Y''(I)$.
- P(40).NE.0 Very important DRT choice parameter. Required to carry out calculations and produce the Distribution of Relaxation Times (DRT) file OUTPARM using the K, O, or R Circuits.). See the detailed list of P(40) choices in the R-circuit discussion.

** See additional choices when MAXFEV = 1 or 2 listed at the bottom of p. 3-6.

LINE5 [FORMAT(32I1)]
PARAMETER TYPES

NFREE(I), I = 1, 50

NFREE(I) determines whether the Ith parameter value is fixed or is free to vary to adjust the fit:

- 0 - (fixed) Parameter is fixed
- 1 - (Pos) Parameter is free, but must stay positive
- 2 - (any) Parameter is free, may be positive, zero, or negative
- 3 - (sub) The effect of this parameter will be subtracted from the data if a run with MAXFEV = 2 is carried out.

Care should be taken to fix all parameters except those specifically being allowed to vary to find the best fit. All circuit elements removed from circuit by setting them to zero and all the parameters which select options within a circuit but are not actual model elements should be FIXED (i.e., all parameters from P(33) through P(40).

Beginning of bottom part of input file

LINE6 [FORMAT(I5,3(5X,D20.13))]
DATA LINES

COLS	1-5	11-30	36-55	61-80
	IC	FREQ(I)	Y(I)	Y(I+M)

IC - Dummy counter

FREQ(I) Frequency (angular or Hertz) associated with Ith data point.

Y(I),Y(I+M) Data to be fitted in either rectangular form (real, imaginary) or polar form (magnitude or natural log of magnitude, and phase). Note that Y(I) is real part and Y(I+M) is imaginary part of data for the rectangular form. If imaginary values are entered all positive when they should be negative (as for usual IS impedance data) one must set NEG to N in Line 2 of the input. Number of lines = M.

LINE7 [FORMAT(I5,25X,2(5X,D20.13))]

DATA WEIGHTS (Only for the choice IRCH=0 on LINE3)

These are the last possible items for the bottom part of the LEVM input file (actually standard deviations or uncertainties used to calculate the weights) to be used in the fit. These can normally be calculated internally, but in special cases it is desirable to assign them separately (number of lines = M). Otherwise, enter nothing here. You can format and add weight data to your full input file using MKINP and the weight file; select choice 5 during MKINP operation.

I = 1, M (LINE3)

COLS	1-5	36-55	61-80
	IC	RW(I)	RW(I+M)

IC - Dummy counter (may not be omitted)

RW(I) Uncertainty associated with data point Y(I)

RW(I+M) Uncertainty associated with data point Y(I+M)

CNLS Fitting and Weighting

Immittance data: $i(\omega_j) = i_j \quad (j = 1, 2, 3, \dots, N)$

Immittance model: $l(\omega_j, \bar{\theta}) = l_j \quad (\theta_p: p = 1, 2, 3, \dots, P)$

Residuals of fit: $R_j = i_j - l_j = R_j' + i R_j''$

Weighting variances: $v_j = v_j' + i v_j''$

Objective function to minimize

$$S_N \equiv \sum_{j=1}^N \left(\frac{(R_j')^2}{v_j'} + \frac{(R_j'')^2}{v_j''} \right)$$

Error Variance Model

Variance accounts for resolution and power-law errors

$$v_j' = U^2 + \left| F'(\omega_j, \bar{\theta}) \right|^{2\xi}$$

$$v_j'' = U^2 + \left| F''(\omega_j, \bar{\theta}) \right|^{2\xi}$$

Weighting Choices

IRCH	Name	U	ξ	F'	F''
Old: Fixed Weighting					
1	UWT	1	0	0	0
2	PWT	0	1	i_j' (data)	i_j''
3	MWT	0	1	$ i_j $	$ i_j $
New: Variable Weighting					
-2	FPWT	0	1	$I'(\omega_j, \bar{\theta})$ (model)	$I''(\omega_j, \bar{\theta})$
-3	FMWT	0	1	$ I(\omega_j, \bar{\theta}) $	$ I(\omega_j, \bar{\theta}) $
2	GWT	Free	Free	i_j'	i_j''
2	GMWT	Free	Free	$ i_j $	$ i_j $
-2	FGWT	Free	Free	$I'(\omega_j, \bar{\theta})$	$I''(\omega_j, \bar{\theta})$
-3	FGMWT	Free	Free	$ I(\omega_j, \bar{\theta}) $	$ I(\omega_j, \bar{\theta}) $

Distributed Elements Multifunction Documentation

This section describes the different distributed circuit elements (DCE's or DE's) available for use in the fitting circuit models (See the next page for a listing). LEVM contains a function subroutine DISTEL which allows any of a variety of distributed elements to be substituted into the available circuit models. This feature simplifies coding of the individual model subroutines, permitting more flexibility in less space. In addition, new distributed elements may be added to the programs by recompiling a single function which makes the new element available in all of the existing circuit models. Note that although the specific DISTEL expressions given in this section are nearly all returned as impedances, DISTEL can also be invoked at the dielectric constant level by setting NELEM negative. (See also the important dielectric possibilities discussed in Circuit O).

The DISTEL function is passed a total of six parameters: four which are free parameters of the element; one which is the angular frequency at which the impedance is being calculated; and one which selects the particular element being used (NELEM; also called NDE). Each of these is described below:

- RDE = R. Free parameter, usually a resistance (BUT SEE BELOW)
Usually $R = R_0 - R_{inf} = \Delta R$, but will be $R = C_0 - C_{inf} = \Delta C$, or $\Delta \epsilon$ if NELEM < 0 or for DSD response in the O-ckt.
- TDE = T. Free parameter; usually a time constant/relaxation time: tau. (|MODE|=0 or 1).
- TDE = ϵ_τ . Sometimes a fit model may use **TT** (p.3-9), not = T, when |MODE|=2; see listing below and MODE discussion for more. TDE will then be an effective dielectric constant. The symbol TT in a response formula **includes** the T possibility but not *vice versa*.
- UDE = U. Free parameter [NOT the same as U=P(31)]
- PDE = PHI. Free parameter, typically an exponent
- OMEGA. Fixed parameter, represents frequency of AC signal; sometimes time or temperature
- NELEM. Fixed parameter; selects DE to be used (in the circuits, it is designated by NDE_x, where x is a number).

The R, T (or ϵ_τ), U, and PHI parameters may be free fitting parameters passed to the program, while OMEGA is passed by the model subroutine. Care must be taken to fix NELEM at a constant value, otherwise the fitting routines will try to adjust its value, producing unpredictable (though necessarily bad) results. The input quantity MODE selects between T and ϵ_τ for some of the DCE's, as described above. Also see the MODE discussion in Section 3. The choice MODE=-2 is only available for CSD1a fitting.

NOTE:

There is a special input option on the NELEM parameter. If this parameter is set to be negative, the program returns the impedance of a dielectric element, calculated by taking the quantity calculated by the function (with R taken as $C_0 - C_\infty$) to be a complex dielectric constant (or capacitance) and converting it to an impedance. Specifically, if this option is selected, the function takes the calculated "impedance" of the element then takes the reciprocal and divides by $i*\omega$:

$$Z_{\text{diel}} = 1 / (i*\omega*Z) = E' - i E'' ,$$

where $E = \text{Epsilon}$ and $\omega = \text{celcap}*\omega = C_c\omega$. See the discussion in paper #12. ***BUT it is usually much better to do dielectric data fitting using the DSD part of the O Circuit *****.

Below is a list of the currently available DCEs, followed by abbreviated individual descriptions. See the following pages for full information on them.

NELEM	NELEM
0 Short circuit	20 PLE #1 Power-law DCE
1 R and C in parallel	21 PLE #2 E-level
2 CPE, etc., #1	22 PLE #3
3 CPE, etc., #2	23 Ladder network #1
4 ZC #1 Cole-Cole (Z-level)	24 Ladder network #2
5 *ZC #2	25 Effective medium using PCPE E-level
6 ^*HN #1 Havriliak-Negami, etc.	26 Series CPE: SCPE <i>Z-level</i>
7 #^*CD #2 Use for CD model only.	27 Parallel CPE: PCPE <i>E-level</i>
8 *HN #3	28 PLE E-level DSD O-circuit only
9 ^*Generalized Finite Warburgs	29 ^*Generalized Davidson-Cole
10 #^*Accurate KWW responses; no cutoff	30 *EDAE1 exact model: $\phi = 0$ or 1
11 Jonscher response	31 Effective-Medium BDM equation
12 *EDAE1: Exponential DRT (See CktO)	32 #^*KWW BETA=0.5, exact
13 *EDAE2: Exponential DRT	33 Arrhenius and other thermal fits
14 ^*GDAE2: Gaussian DRT; incl. trans. resp.	34 ^*Simplified, generalized EDAE model
15 General Diffusion #1	35 ^* KWW BETA=1/3, exact
16 General Diffusion #2	36 #^* KWW arb. BETA, uses DRT
17 Four ideal elements (including L)	37 Calculates KWW DRT, arb. β
18 Dissado-Hill, Z level	38 Not used
19 Dissado-Hill, E level	

All NELEM choices except 33 and 37 include DSD or CSD0 response when $\text{MODE} \geq 0$.

* For these elements, the input quantity $T=\tau$ is replaced in the fit formula by TT when $|\text{MODE}|$ is set to 2, and the input T is then an effective dielectric constant. $\text{TT} = T$ for other values of MODE. Useful for analysis of bulk dispersion and for aiding in comparing the activation energies of R and T for a thermally activated situation. No change when $|\text{MODE}| = 0$ or 1.

- ^ This symbol indicates that the DE so marked allows, in addition to CSD0 (or DSD) response, CSD1 response for $\text{MODE} < 0$. See Preface, Appendix 1, and specific DE's for details, and the O-Circuit discussion for other CSD fitting possibilities. CSD1a response is available for $\text{NELEM} = 6$ and 7 (CD model only), 10, 14, 32, and 34-36. CSD1b response is available for $\text{NELEM} = 6$ or 7 (HN model not including CD), 9, 10 and 29. See also the CSD1a $\text{MODE}=-6$ choice of p. 5-33.
- # This symbol indicates that the DE so marked produces extra final output involving limiting dielectric quantities and moments of the relevant DRT. Such output is particularly useful when the data being fitted are in specific form: e.g., complex conductivity rather than complex conductance. This output is described in detail in Part 4c of Section 1. It is usually only present for CSD fits. ** Further, it may require use of the O-circuit and the $\text{Mode} = -1$ choice. When more than one DE is present in the O-circuit input file, in order to obtain full extra results, after convergence deactivate all DE's and their line-5 free-parameter settings (except the P(10) CSD DE), set $\text{MAXFEV}=0$ and run OUTIN, then fit OUTIN once more: only proper CSD values will then appear at the end of the file.

Use $\text{CELCAP} = C_c$ for Z or Y data, or $\text{CELCAP} = \epsilon_v$, the permittivity of vacuum, for rho (complex resistivity) or sigma (complex conductivity) data.

Note that setting $\text{NELEM} = 0$ removes all of the effects of the associated DE parameters even when they remain unchanged and non-zero.

In addition to the DE's listed above, the O Circuit contains several important additional response models, as well as allowing all the above elements to be used. Particularly useful are the new GEDAE (generalized exponential distribution), and Havriliak-Negami and Cole-Davidson models with cutoff, making them physically realizable.

Unfortunately, fitting of a given model requires different parameter assignments for some different fitting circuits. Such differences are illustrated for the HN $\text{NELEM}=6,7$ model below.

Discrete Ideal Elements

In addition to the various distributed elements provided, DISTEL offers several combinations of ideal circuit elements (see $\text{NELEM} = 0-3$, and 17). These allow more flexibility in the model circuits.

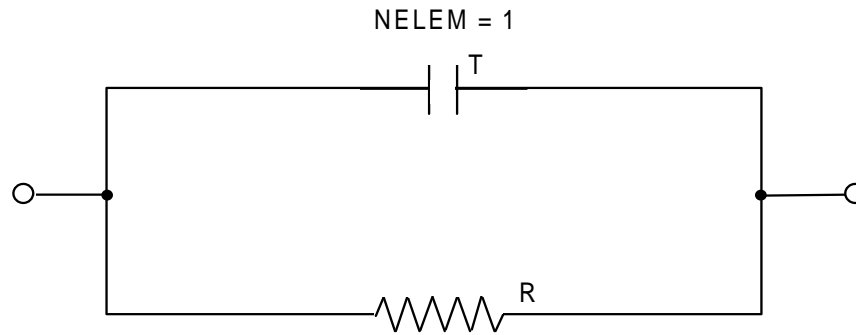
Short Circuit

$\text{NELEM} = 0$

In this option, none of the free parameters are used. Instead, a value of 0 is returned for the impedance. Note that the model subroutine will never actually call DISTEL if this option is used, since an element value of zero is interpreted as a command to remove the element from the circuit. For more information on the zero-handling conventions of these sub-routines, consult the section on MODELS.

R - C Parallel Combination

NELEM = 1



U, PHI - Not Used

This option substitutes a simple R-C parallel combination into the circuit. If either the resistor or capacitor (here T) is assigned a value of zero, that element is ignored, NOT just set to zero. The formulae used are given below:

If $R=0$ and $T \neq 0$

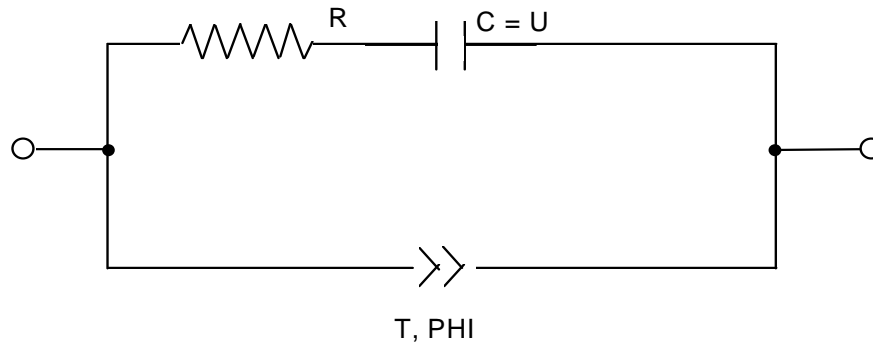
$$Z = 1 / (i * T * \text{OMEGA})$$

Otherwise

$$Z = R / (1 + i * R * T * \text{OMEGA})$$

Constant Phase Element (CPE), and/or Series RC in parallel

NELEM = 2, 3

**Model 1 (NELEM=2)**

$$Z(\text{CPE1}) = 1 / [T(i*\text{OMEGA})^{\text{PHI}}]$$

Model 2 (NELEM=3)

$$Z(\text{CPE2}) = R / [i*R*T*\text{OMEGA}]^{\text{PHI}}$$

The constant phase element (CPE: shown as >> in the diagram) is a simple distributed element which produces an impedance having a constant phase angle in the complex plane. The exponent PHI determines this angle (which is $(\pi/2)\text{PHI}$, with $-1 \leq \text{PHI} \leq 1$). In the special case of $\text{PHI}=1$, the CPE acts like a capacitor with T equal to the capacitance. The CPE can also yield an inductance ($\text{PHI} = -1$, or a resistance: $\text{PHI} = 0$). Two different (though mathematically equivalent) versions of the CPE are provided here. The two are included because there can be different correlations between the parameters, which may be of some interest in certain applications. See discussion in the NELEM = 6 section.

See Solid State Ionics **13**(1984)147.

NELEM=2:

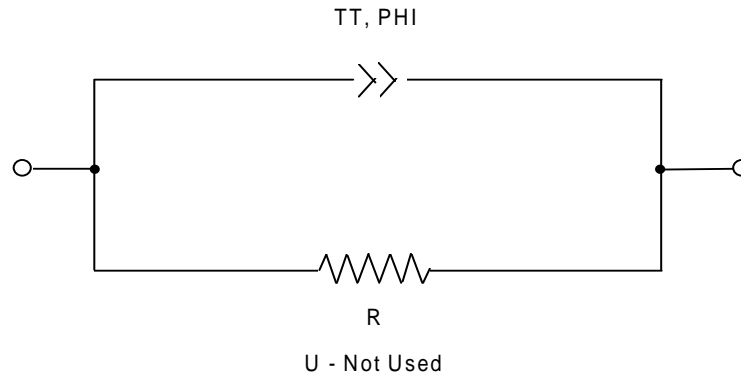
May use R,T,U, and PHI. For CPE only, with NO parallel RC elements: set R=0, U=0 in input. If T is set zero in the input, the overall DE with NELEM = 2 is just a R and C (set by U) in series. If either of these is set zero in the input, only the other contributes to the impedance. Note that T is not a capacitance here except when $\text{PHI} = 1$.

NELEM=3:

Uses only R, T, and PHI. No parallel branch here. When NELEM = 3, R or T must be fixed and the other taken free. For this choice, T is a capacitance.

ZARC - Cole (ZC) Element

NELEM = 4, 5



The ZARC - Cole element (the ZC, Model 2 below) is a common distributed element which can be thought of as representing a Constant Phase Element in parallel with a resistor, as illustrated above. The ZARC produces a complex plane impedance curve which forms an arc of a circle (hence ZARC) with a center displaced from the real axis. The smaller PHI is, the more the center is displaced (for PHI = 1, a semicircle results, just as for an R-C parallel combination). For more information and a critique of the ZC, see *Solid State Ionics* **133** (2000) 79.

Two different versions of the ZC are included here. Also see the HN response on the next page. In spite of the “U – not used” statement above, Model 1 has been modified to use it, especially with the values U=0 and U=1. This model allows T to be negative, even when Phi<1, sometimes needed to represent electrode adsorption and reaction effects (See *J. Electroanal. Chem.* **82 (1977) 271). Note that when R is held fixed at a large value such as 10**30 and U is fixed at 1, the Model-1 circuit reduces to CPE form, but one where T can be negative (as also in the NELEM= 26 AND 27 choices), unlike the Model-2 response below with Phi<1. *****

**Model 1 (NELEM=4)

$$Z = R / (1 + (R**U)*T*(i*OMEGA)^{PHI})$$

Model 2 (NELEM=5)

$$Z = R / (1 + (i*TT*OMEGA)^{PHI})$$

See *Solid State Ionics* **15**(1985)159 and **20**(1986)111. The latter reference gives more information on many of the present distributed circuit elements. For DSD behavior (see Circuit 0), the ZC then describes long-known Cole-Cole dielectric response.

Havriliak-Negami (HN) and other Elements

NELEM = 6, 7, and 8

The empirical Havriliak-Negami (HN) element reduces to the symmetric ZC form when PHI is taken fixed at unity, and to asymmetric Cole-Davidson (CD) response when the exponent U is taken fixed at unity. It thus encompasses both possibilities. CD response leads to an asymmetric complex plane plot (except in the limit of PHI=1). See also the O circuit for HN with cutoffs: CSD, DSD, and information on CSD0 and CSD1: CSD1a, and CSD1b response in the Preface, App. 1.

The best CNLS fits are obtained when the correlations between parameters are as low as possible. The correlations themselves depend on how the model is parameterized, with one set of parameters often yielding lower correlations than another. Thus, it is useful to have available several different parameterizations for the same model. Here, two different choices are provided.

The forms used for the HN impedance are:

NELEM=6: MODE = 0,1, or 2: CSD0, or DSD (when used in the DSD part of Circuit O)

NELEM=6: MODE = -1 or -2: U = 1 fixed, CD response: CSD1a (see below).

NELEM=6: MODE = -1: U ≠ 1, fixed or free. HN response: CSD1b (see below).

NELEM=7 Use only for CD and MODE= -1 situation. Same as NELEM=6 but has extra dielectric and DRT-moments output

$$Z_{HN} = R / (1 + [i*TT*OMEGA]^U)^{PHI} = R * I(\omega, p_0) \quad (\text{See Preface, Appendix 1}).$$

For NELEM=6, (|MODE| = 0, 1), T is a relaxation time, while for NELEM=6 (|MODE|=2), T is an effective dielectric constant. The NELEM=6,7, MODE=-1, CSD1b response redefines R so that this input parameter becomes ϵ_{CINF} , but in the NELEM=7 extra final output, estimates of both the resistance R and ϵ_{CINF} appear, as they also do for CSD1a response.

$$\text{NELEM=8: } Z_{HN} = R * \sin[(\text{Pi}/2)*U] / (1 + [i*TT*OMEGA]^U)^{PHI}$$

**It is sometimes desirable to use two conductive-system DCEs in the O-circuit, one representing bulk response and the other electrode effects. The first one might be the KWW1 response model (NELEM = 10 or 36), set for CSD1a response with the input value of Mode (e.g., -1), and the other (the DE3 element in the O-circuit), possibly a HN element. If Mode = -1, the HN element would ordinarily be of CSD1a type. But usually in this situation, it should instead be of CSD0 character. To accomplish this when the input value of Mode is negative, set |TEMP| > 900.

The HN without cutoff yields non-physical power-law response at extreme low and high frequencies. This causes its CSD1 response to be particularly inadequate at the frequency extremes. The low-frequency-limiting frequency power-law exponent and log-log slope is U, and the high-frequency limiting slope is -U*Phi at the Z level. The final output produced for NELEM=7 uses the HN or CD DRT for calculation, rather than the above NELEM=6,7 formula, with a fixed value of 15 for the range of integration parameter, also called U for several models.

See Ref. 21 in Section 1; Wang and Bates, Solid State Ionics, **50**(1992)75; and Macdonald and Wang, Ref. 9.

When NELEM=6 or 7, U=1, fixed, MODE =1 or 2, and PHI=0.5, fixed, the resulting Cole-Davidson

response (Ref. 20) at the Z level is fully equivalent to the **Gerisher Impedance** (see Ref. 26). When it is applicable, however, it provides specific formulas for R and $\text{Tau}=T$ in terms of diffusion constants and other microscopic quantities (Ref. 26).

Because of the difference of parameter assignments under different fitting conditions, an example is included below for the HN model and EDINF, the limiting high-frequency dielectric constant associated with dielectric, not conductive, processes. Its role is played by C_g for the first-column situation below. If $\text{MODE} < 0$, CSD1b response is involved here. For the O circuit (Section 5), $P(35) = \text{NCH1}$, denoted NCH below. Further, parameter $P(\text{xx})$ will be denoted as Pxx below. The first and second columns involve fitting without DRT cutoff but the third involves such cutoff.

PARAMETER ASSIGNMENTS

Not O circuit		O circuit, NCH=4	O circuit, NCH= 3
P2	0	EDINF	EDINF
P3	0	0	U = GAMMA
P5	0	R	R
P6	R	T	T
P7	T	U =GAMMA	U (range)
P8	U = GAMMA	0	0
P9	PHI	PHI	PHI
P10	7	7	0
P29	C_g	0	0
P35	0	4	3

****CAUTION:** Multiple DE's in the same input file

When the O-circuit is used with MODE set to -1 for the CSD DEC model defined by the value of NELEM in parameter $P(10)$, fitting may often require the presence of one or two additional DE's in the input file, typically with values of NELEM appearing in $P(20)$ and/or $P(25)$. It is usually desirable, however, that the $\text{MODE} = -1$ choice only affect the response associated with $P(10)$, not that of the other DE model responses.

To ensure the $\text{MODE} = 1$ condition for $P(25) = 6$ or 7 , just set $\text{ATEMP} < -900$. This applies for circuits B, D, E, J, and O, all involving a DE in $P(21) - P(25)$.

For the O-circuit, when the value of $P(23)$ is set to < -900 **or** when ATEMP is < -900 , Mode is set to 1 for the DCE3 distributed element involving $P(25)$.

For the O-circuit the DED dielectric-response models used with $P(20)$ greater than zero, e.g., $\text{NELEM} = 12, 25, \text{ or } 27$, are independent of the choice of MODE , so this issue does not arise for them.

Generalized Finite Warburg (GFW) Elements and PNP and PNPA models

NELEM = 9 (See also NELEM=15 and 16)

The Generalized Finite Warburg is an extension of another more common element: the Finite Length Warburg (FLW). The FLW is the solution of the one-dimensional diffusion equation of a particle, one which is completely analogous to wave transmission in a finite-length shorted RC transmission line. See JRM papers #'s 91, 92,124,183,188, 252 (1973 to 2010) for relevant studies of more complicated FLW-type situations. The simple FLW impedance expression is:

$$Z_{FLW} = R * \tanh(s) / s, \quad (1)$$

where $s^2 \equiv i\omega\tau$; $\tau \equiv L^2 / D$; L is the effective diffusion thickness; and D is the diffusion coefficient of a diffusing particle. Here $\omega \equiv 2\pi\nu$ is the radial frequency, often designated as OMEGA in the LEVM Fortran program. Further, τ will be taken as a free fitting parameter having the dimension of time. Eq.1 with either tanh or ctnh may also be used for the NELEM =15 or 16 choices with PD = 0.

The previous NELEM=9 LEVM DCE was extended in 2009 to include some new types of distributed circuit elements for fully dissociated situations. Since the usual DCE format in LEVM involves only the parameters R , τ , U , ω , and $\psi \equiv \text{PHI}$, when NELEM=9 it is **necessary***** to use the parameters, UDE (e.g., UDE=P(8)=P8 for the DE1 choice in Circuit B) and P(18) to select various choices. The MODE=1 (CSD0 or DSD) choice is appropriate for the equations below, but when MODE = -1, CSD1b response is also available (see p. 0-18), only appropriate for conductive-system dispersion, but usually MODE=1 should be used here for all these choices.

With the choices P18=0 and UDE=1 fixed, the quantity $\tanh(x)$, with x complex, is used in Eq. 2 below. For UDE=2, however, the $\tanh(x)$ of Eq. (2) is replaced by $\text{ctnh}(x)$. So two different forms of these GFW equations are thereby made available. The tanh one choice involves shorted or absorbing electrodes, while the ctnh one leads to open circuit completely blocking ones.

When UDE=1 (or 2), an anomalous diffusion form of Eq. 1 is selected, one that involves changing its exponent of 0.5 to $0.5 * \text{PHI}$ with $0 < \text{PHI} \leq 1$; this is equivalent to changing s to the CPE $s^\psi \equiv (i\omega\tau)^\psi / 2$, and thus generalizing FLW response to GFW. The PHI=1 choice reduces to ordinary finite-length diffusion. For UDE=1, the anomalous-diffusion version of Eq. 1 becomes

$$Z_{GFW} = R * \tanh(s^\psi) / s^\psi, \quad (2)$$

an expression that was perhaps first suggested in my paper #169 of 1985 and was first included in this section of LEVM in the early 1990's. It was later derived by Bisquet and Compte, J. Electroanal. Chem. **499** (2001) **112**, Eq. 35, along with other related equations, such as one involving ctnh rather than tanh (see their Eq. 32), thus then involving an open circuit (a reflecting boundary) instead of shorted conditions at the electrodes. The parameter U (=UC1) is not used in Eq. (2). Note that when $\text{PHI} < 1$, the number of diffusing particles is not conserved for this expression. Here Z and R (usually designated ρ_0 for specific situations) may be either full or specific quantities but they will be assumed specific here. See the Randles circuit examples in the FITTESTS folder, Circuits B and O for examples where the FLW or GFW is included as part of the composite fitting model (see also p. 0-11).

The P8=3-7 choices below (set P18=0) involve a more complex diffusion expression, (PNP or PNPA: see also the Circuit H discussion of even more complicated PNP/PNPA models)) one for a uni-univalent material of length L with equal diffusion coefficients of the charge carriers and completely blocking plane-parallel electrodes. Here U (=U1) is taken as a free parameter and

defined as $M = L / (2L_D)$, the number of Debye lengths in a half cell. Next, define the quantity $p^{2a} \equiv 1 + s^{2\psi}$. Then for P8=3 or 6 the anomalous “interface” impedance may be written (see p.1619 of paper #124 and also the LV5.FOR part of LEVM),

$$Z_i = (R * p^{2a} / s^{2\psi}) / [(M * p^a) * \text{ctnh}(M * p^a) - 1], \quad (3)$$

consistent with satisfaction of the Poisson equation for PHI=1. When P8=4, 5, or 7, one obtains the total impedance, Z_T (see Eq. 24' in #124, isomorphic with Eq. 1 of paper #92 when PHI=1; the P8=5 choice). Also, for PHI=1, as discussed on p. 1620 of #124, Eq. 3 with a different value of M is appropriate for the one-mobile situation. The parameter P(6) input quantity is τ ($= R * \epsilon_V * \epsilon_{D\infty}$) for choices 3, 4, and 5, but for 6 and 7 it is just the bulk dielectric constant, $\epsilon_{D\infty}$, where the geometric specific capacitance is $\epsilon_V * \epsilon_{D\infty} / 4\pi L$. One would expect less correlation between R and $\epsilon_{D\infty}$ than between R and τ , with the former yielding better convergence and perhaps smaller estimated parameter uncertainties in most fits.

It is worth mentioning that JRM PNP papers #'s 124 and 183 relax the equal valence, equal mobility, and full dissociation conditions applying to Eq. (3). Further #183, dealing with blocking electrodes, contains results for very many choices for various conditions. Finally, for fitting experimental data, a separate free high-frequency-limiting bulk capacitance parameter must be included in parallel in models where it is not intrinsic, unlike the present Z_T model (see, e.g., paper #248). See pp. 5-16 through 5-18 for analyses of arbitrary generation-recombination.

For further discussion of important PNP and PNPA responses see p. 4-26.

Kohlrausch-Williams-Watts (KWW) Stretched Exponential

NELEM = 10 (see also NELEM = 32,35-37)

This distributed element provides four approximations to the important Kohlrausch-Williams Watts fractional exponential response in the frequency domain. Fitting of temporal KWW data is readily carried out using Circuits G or O of Section 5. Only approximate frequency responses are provided here but some of the approximations are excellent. Also note that the NELEM = 32 and 35 choices provide very accurate KWW response for PHI = 0.5 or 1/3, respectively, for which the relevant DRT is known in closed form. The four approximations (in order of increasing accuracy) are: (1) The approximate Macdonald-Hurt method described in Ref. 24 and used in earlier versions of LEVM. Set ATEMP > 0. It functions as a good first fit for ordinary DSD and CSD0 data, but yields unsatisfactory results, particularly at low frequencies, for CSD1-type data fitting. When ATEMP > 0 and < 1000, and Circuit O is used, the parameter P(2) is the geometrical capacitance CINF= C_{inf}, while the choice ATEMP >= 1000 causes this parameter to be the equivalent dielectric constant (make sure that the CELCAP input parameter is set to an appropriate value, e.g., the permittivity of vacuum for data in specific form, such as conductivity rather than conductance, etc. (If you wish, set the input parameter M < 0 (initially) in order to automatically obtain CELCAP = 0.8854..x10⁻¹³ Farads/cm (Gaussian units), the vacuum permittivity, EPSV). See NELEM = 36 for KWW response with cutoff: see Ref. 46 listed on p. 1-25 for details about cutoff.

For the remaining three responses, **set ATEMP < 0**. These responses all involve series summation (with the epsilon algorithm (see Ref. 43) used for part of the frequency range in order to extend the range of applicability of the two types of series employed). Again, when the O-Circuit is used, P(2) will be the dielectric constant corresponding to CINF. (2) Set U >= 10. For (3), set U>0 and < 10 (e.g., 1) to obtain at least an order of magnitude increase in accuracy. Finally, **for (4), set U <= 0**, for at least another order of magnitude accuracy increase. Note that (2) will not always converge faster than (3) or (4) because of its relatively low accuracy. It is sometimes appropriate to start fitting using (2) or (3), and use the fit result (e.g., OUTIN) as input for a final fit with the choice (4). When this choice is used to fit accurate PHI = 0.5 data, one obtains parameter estimates with relative standard deviations of less than 10⁻⁷ and most relative

fit residuals less than 10^{-6} when $0.2 \leq \text{PHI} \leq 0.999$. The accuracy of all the above approximations tends to decrease for $\text{PHI} < 0.3$, but Approx. (4) is still quite accurate down to at least $\text{PHI} = 0.1$ for $W = \omega\tau \geq 10^{-6}$.

When the KWW element is used in the O-Circuit or elsewhere, several additional lines of output are produced dealing with moments of the appropriate DRT and with dielectric parameters (see Part 4c of Section 1 of the manual). All the present approximations (and moments) are for KWW response WITHOUT any cutoffs, but the NELEM = 32, 35, and 36 response-function choices involve possible cutoffs. See Ref. 43 for more information about the present algorithm and the effect of cutoff, and Refs. 45-47 for accurate fitting with the KWW model with possible cutoff.

R - Resistive ($R_0 - R_{inf}$), or Capacitive scaling factor ($C_0 - C_{inf}$).

T - KWW Time constant or ϵ_τ . See discussions of MODE choices and TT.

PHI - KWW Exponent (often defined as BETA)

U - Used here to set accuracy; see above

DSD (**KD**), CSD0 (**K0**), and CSD1 (**K1**) responses are available here.
For MDE = -1: CSD1a, and for MDE = -3: CSD1b.

See NELEM = 36 for KWW response with or without cutoff. Use NELEM = 10 when no cutoff and increased convergence speed but somewhat reduced accuracy is desired.

With the O-circuit, use **MODE=MDE=1** for KD or K0 response, and **MODE= -1** for K1-model response. Use **NELEM=35** for the K1 model with beta1 fixed at 1/3: the quasi-universal **UN model**. For more information, see reprints #'s 226-227, 233-238 and later ones in the serial publication list of the Macdonald home page.

When the O-circuit is used with two conductive-system DCEs (see the discussion on pp. 4-6 and 4-7) and the second one is of KWW form, with the first one of CSD1a character (e.g., Mode = -1), then if NELEM = 10 and one wants the second one to be **KWW0** rather than **KWW1**, set P(23) to < -900.

Jonscher Response

NELEM = 11

This model is one of the Jonscher 'Universal Dielectric Response' functions - the GJ2.

In terms of the input parameters, its impedance (NELEM=11) is

$$Z = R * [(S^U) \{ (1/\text{TAN}(P1) - i) \} + (S^{-PHI}) \{ (1/\text{TAN}(P2) - i) \}],$$

where $S = \text{OMEGA} * \text{TAU} = T * \text{OMEGA} = \text{TOMEGA}$, $P1 = (\text{PI} * U / 2)$, $P2 = (\text{PI} * \text{PHI} / 2)$. This GJ2 is, more or less, two related CPE's in series.

On the other hand, when NELEM= -11, R must be interpreted as a capacitance, $C_0 - C_{inf}$, and at the admittance level, we have the parallel combination of two related CPE's.

See J. Appl. Phys. **58**(1985)1971.

Exponential Distribution of Activation Energies

(FORM 1, ASYMMETRIC: EDAE1)

NELEM = 12

The important Exponential Distribution of Activation Energies element utilizes all four of the potentially free parameters. Here R is the element's scaled resistance; PHI is an exponent factor; the character of T (and TT) depends on the value of MODE; and |U| is a range scaling factor which must be non-zero (U = 0 degenerates to a simple Debye response). **Generally, $U = \ln(\tau_{\min} / \tau_0)$, where T or TT is τ_0 , the characteristic relaxation time of the model and τ_{\min} is the minimum value of τ for the equivalent distribution of relaxation times.

The expression for the complex EDAE impedance is not expressible in simple closed form in the general case, though for specific values of PHI it can be. Instead, the EDAE distribution is expressed as an integral. For its explicit form, see J. Appl. Phys. **58**(1985)1955-1970. The limits of the integration are 0 and |U|, where U is here a logarithmic range variable (try |U|=30 initially). The input parameter PHI is the PHI of the EDAE1 function, which also involves the inputs R and T (tau). Here U may be < or > than 0. No cutoff effects if |U| > 40.

The integral is broken into real and imaginary parts as usual. The resulting integrals are evaluated as discussed under NELEM = 14. One must ensure that a value of IGACC, the integration accuracy parameter, is set in the input file. For transient response, use NELEM=34.

See: J. Appl. Phys. **58**(1985)1955-1970 and J. Appl. Phys., **61**(1987)700, and NELEM = 30 herein. The present NELEM = 12 EDAE model does not calculate moments of the distribution. One that does and may, in addition, involve a more general distribution is available in the O circuit, choice 1 (GEDAE); see also NELEM = 34. The O-circuit model also produces full extra (dielectric-constant-oriented) output at the end of the regular output.

Finally, when T is taken negative and |MODE| < 2, parameters are redefined. This type of EDAE1 fit is appropriate when the data are in specific form (e.g.: complex resistivity, etc.). The quantity |T| is the pre-exponential factor in tau: taua, and R is the activation energy E. U and PHI have their usual meanings and |ATEMP| is the temperature. See algorithm in LV5.FOR for more details. **Note that the value of MODE is not relevant for this DE.

Exponential Distribution of Activation Energies

FORM 2, SYMMETRIC: EDAE2)

NELEM = 13

This form of the EDAE uses two exponential terms (one equal to the negative of the other), and the range of integration is broken into two pieces with U here being half the U of the EDAE1. The other parameters still have the same roles as for the EDAE1. This function will yield a symmetric response curve. Inputs: R, TT, U, PHI, and NELEM. When U is very large, no cutoff effects appear.

A six-parameter general form of the EDAE is available in Circuits D (DSUB) and O (OSUB). It involves Phi1 and Phi2. When the |Phis| are not much larger than 0.5, the limiting (no cutoff) low-frequency power-law exponent (log-log slope) is about -Phi2, while the limiting high-frequency slope is approximately -Phi1. If Phi1>0 and Phi2= -Phi1, one obtains the present EDAE2 symmetric response. The range of a Phi is -infinity < Phi < infinity.

See: J. Appl. Phys. **58**(1985)1955,1971 and J. Appl. Phys., **61**(1987)700.

Gaussian Distribution of Activation Energies

(FORM 2, SYMMETRIC: GDAE2)

NELEM = 14

The GDAE2 involves integration, as does the EDAE. A Romberg integration routine is used for all EDAE and GDAE calculations (QROMB). Its accuracy can be set by the choice of IGACC in the input of LEVM. The smaller this parameter, the faster and less accurate the integration. An initial value of 2 is reasonable, but for fitting to reasonably accurate data one should carry out a final fit with IGACC=3 or 4.

A full description of the GDAE approach occurs in J. Appl. Phys. **61**(1987)700. See also Ref. 41 (end, Section 1). Further information can be gleaned by consulting the present source code, that under NELEM=14 in DISTEL, and that under circuit D in DSUB. The GDAE approach in DSUB involves the full, possibly asymmetric model. Here, the limits of integration are plus and minus U. In addition, PHI=XIGDAE=XI, a width parameter of the model. The quantity THGDAE of the full GDAE model is here taken identically zero. For MODE = -1, CSD1b response is available, and for MODE=8, transient response fitting and simulation is invoked.

Note: the GDAE2 response is a special case of the general GEDAE generalized exponential response function of the O circuit, but the present DE uses a PHI which is proportional to the inverse of that used there in order to make it consistent with EDAE response as well.

Finally, the present input parameters are R, T, U, and PHI, with $0 < U < 60$, and MODE=8 yields transient response.

General Diffusion DCE

NELEM = 15 and 16

This function represents homogeneous diffusion with a general boundary condition. In its full form it is also appropriate for diffusion of a neutral species in an electrode. Its mathematical representation is essentially the same as that of a homogeneous transmission line with arbitrary terminating impedance. See J. Electroanal. Chem. **101**(1979)307 for definitions and further discussion. Because diffusion is so important in IS, several general diffusion (GD) possibilities are included here. It is worth noting, however, that for unsupported binary conduction, as in most solid materials, the special choices associated with the BCD element of Circuit H should be used.

If NELEM = 15, there are three different parameterization possibilities available (determined by the value of U = UDE selected) for the three macroscopic GD parameters. Alternatively, if NELEM = 16, four microscopic parameters are introduced, as discussed below. Finally, as discussed in the section on the B circuit, if one of the above NELEM values is used for circuit element DE₃ but U is taken negative, correlation present between the GD impedance and some other circuit elements is taken into account in the fitting.

A. NELEM = 15

In order to define the GD impedance, let $s = \text{OMEGA} \cdot \text{TD}$, a normalized frequency variable. We always set $\text{TD} = \text{TDE}$ here, where TD is the diffusional time constant. Further, define $X = (i \cdot s)^{0.5}$, and $\text{XTX} = X \cdot \tanh(X)$. Then, the impedance for general diffusion is, in terms of the three parameters, QD, PD, and TD:

$$Z_D = \text{QD} \cdot (1 + \text{PD} \cdot \text{XTX} / (i \cdot s)) / (\text{PD} + \text{XTX}).$$

- (a) When $|\mathbf{U}| = 1$ (and fixed), QD = RDE (dimensions of resistance) and PD = PDE, all in terms of the DE input parameters of a given circuit.

Note: that PD is a normalized (dimensionless) reaction rate constant here. When PD is very large (for example, set it at 1.D30 fixed) the above impedance reduces to that of the ordinary finite-length Warburg, that obtained from the NELEM = 9 case with PHI = 0.5 there. By contrast, when PD = 0, the present expression reduces to that of an open-circuit (restricted) diffusion situation, one which leads to

$$Z_D = \text{QD} \cdot \text{ctnh}(X) / X$$

(see J. Electrochem. Soc. **129**(1982)1754), JRM paper #154. The general expression with PD disposable spans all conditions between these two extremes.

- (b) When $|\mathbf{U}| = 2$, CD = RDE and has the dimensions of a capacitance, PHI = PDE, QD = TD/CD, and PD = TD*PHI. Here, PHI is the quotient of an unnormalized rate constant and the thickness of the available diffusion region.
- (c) Finally, when $|\mathbf{U}| = 3$, QD = RDE and PD = TD*PHI.

Sometimes, fitting with $|U| = 1, 2,$ or 3 leads to appreciably different correlations between the parameters. That parameterization which involves the lowest correlations is generally best. But also note the differences in standard deviation estimates for the results of the various U choices.

Finally, note that when the data only encompass an infinite-length Warburg response region (impedance proportional to $1/X$), only one free parameter can be obtained from them, and one must set one of the pair QD,TD ($|U| = 1$) or CD,TD ($|U| = 2$), as well as PDE, fixed, not free.

B. NELEM = 16

For this choice, the (partly) microscopic parameters $QDM=RDE$, $k=PDE$, $d=|U| = |UDE|$, and $D=TDE$ are involved. Here, k is a reaction rate constant, d is the thickness of the available diffusion region, and D is the effective diffusion constant of the diffusing entity (uncharged, or charged for a fully supported situation). The PD parameter of (A) above is equal to $k*d/D$. Further, the QD parameter above is related to the present QDM scale factor by $QD = QDM*d/D$. The diffusion impedance is

$$Z_D = QDM*(1 + (k*d/D)*XTX/(i*s))/(k + (D/d)*XTX),$$

where here $X = d*(i*\omega/D)^{0.5}$. Thus, TD in (A) equals d^2/D . When k becomes very large, this expression reduces to the usual finite-length Warburg form: $Z_D = QD*[tanh(X)/X]$ and to the above open-circuit result when $k = 0$. Clearly, one must keep d fixed when fitting with either of these k choices. Furthermore, when NELEM = 16, only three of the four parameters of Z_D may be simultaneously free. Thus, when the extent of the diffusion region is known (e.g., the electrode separation), it is appropriate to fix d .

Ideal Elements: RX, RY, C, and LL

NELEM = 17

R - Parallel resistor: RX

T - Parallel capacitor: C

PHI - Inductance: LL (in series with RY)

U - Resistor: RY (in series with LL)

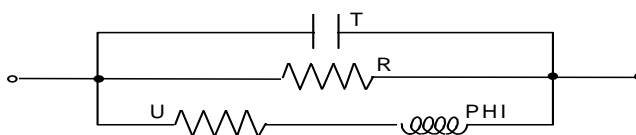
This DCE, which is actually entirely composed of ideal elements, involves RX and C in parallel, and the result in parallel with the series combination of RY and LL.

Although I feel strongly that a circuit which is used to model inductive effects [$\text{Im}(Z) > 0$], probably arising from an adsorption-reaction process, should use a negative differential capacitance and a negative differential resistance [See discussion and rationale in J. Electroanal. Chem. **82**(1977)271], I have provided an alternative in this DCE. Thus, if one wanted to obtain an inductive-like semicircle (below the real axis in a conventional impedance plane plot), one could use, for example, either

A. Circuit D: C4 negative [NFREE(C4) =2]
R4 negative [NFREE(R4) =2]
and NDE4 = 0, or

B. Circuit D: C4 and R4 not included (set to zero) and NDE4 = 14. Use positive initial values of RX and LL (both free) and omit RY and C (set T and U to zero).

Note: With NELEM=17, it is possible to include inductive effects in any of the circuits of the present program wherever a DE element appears in the circuit.



Dissado-Hill Response Element (DHE)

NELEM = 18 and 19

The Dissado-Hill frequency-response distributed circuit element (Dissado, Nigmatullin, and Hill, Adv. Chem. Phys. 63 (1985) 253) yields response similar to, but still different from, the HN and the EDAE (general double exponential distribution of activation energies, termed EDAE3 or DEDAE). The DHE is too complex to list its impedance expression here. It is calculated by a series expansion, mostly as given in the above reference. But the original program did not converge adequately near its peak. In the present realization of the response function, a powerful convergence accelerating technique is incorporated which yields rapid convergence even at the peak.

The DH leads to a limiting low-frequency power-law slope of U and a limiting high-frequency power-law slope of $-\Phi$ where these input parameters are limited to the ranges $-1 < U \leq 1$ and $0 < \Phi \leq 1$. The DH also uses the usual inputs R ($R_0 - R_{inf}$) and T (τ), or ϵ_r for NELEM=18. Since it is primarily defined as a dielectric response function, its use directly as an impedance requires an appropriate value of CELCAP in the input file.

NELEM=18:

DH defined at the Z level (MODE = 0,1, or 2)

NELEM=19:

DH defined at the complex dielectric constant level.

Power Law Related Elements

NELEM = 20 through 28

It is sometimes found that conductance involves power-law frequency response with a log-log slope, or exponent, of unity or possibly even greater. Since such behavior cannot be described by the CPE, a few different fitting possibilities are provided here. When the real-part has an exponent of unity or greater, the associated imaginary part does not exactly follow power-law behavior. But usually it can be reasonably well approximated with a power law whose exponent is somewhat smaller than that of the real part. Several Power-Law response elements (PLE) are provided for fitting to data which exhibit such behavior. Here R, U, and T are just fitting parameters. When U=0, note that #22 reduces to the CPE. Here $Y=1/Z$ is complex conductance.

NELEM=20:

$$Y = R*(\text{OMEGA})^{**T} + i*U*(\text{OMEGA})^{**PHI}. \text{ For NCL response, set } U=PHI=0$$

NELEM=21:

E LEVEL. See LV5.FOR for details of response, section 2100.

NELEM=22:

$$Y = R*(i*\text{OMEGA})^{**T} + i*U*(\text{OMEGA})^{**PHI}$$

**NELEM=23:

$$Y=(U/2)*(i*T*\text{OMEGA})*[1 + \{1 + 4/(i*R*T*\text{OMEGA})^{**PHI}\}]. \text{ Ladder when } PHI=0.5$$

**NELEM=24:

$$Y = (U/2R)*[1 + \{1 + i*4*R*T*\text{OMEGA}\}^{**PHI}]. \quad \text{Ladder \#2}$$

**The NELEM=25-28 DE's should be used with specific data, such as complex conductivity, and should only be used in the O-Circuit, with DED: P(20) = 25 or 27, and DC3: P(25) = 26.

**NELEM=25:

$$\begin{aligned} YP &= T*(i*\text{OMEGA})^{**(-PHI)} \text{ Effective medium response using } \mathbf{PCPE} \text{ (\#27)} \\ YC &= 3*U*[YP - R] / [YP + 2R - U*(YP - R)] \\ E &= R*(1+YC) \text{ Dielectric level. In O-circuit, set } P(40)=4 \end{aligned}$$

**NELEM=26

SCPE; Series element in O-circuit. $SIGMA = U*EV*(i*\text{OMEGA})^{**PHI} = 1/\text{RESISTIVITY}$

EV = PERMITTIVITY OF VACUUM (Uses CELCAP). In O-circuit, set P(35)=4

**NELEM=27

PCPE; EPSILON LEVEL $E=U*(i*\text{OMEGA})^{**(-PHI)}$. In O-circuit this is a parallel element; set P(40)=4

NELEM=28

$$E = R*(\text{OMEGA})^{**T} - U*(i*\text{OMEGA})^{**PHI}$$

NELEM=29

MODIFIED DAVIDSON-COLE RESPONSE
 $Z=R*(1+U) / [1+ U*(1 + i*TT*\text{OMEGA})^{**PHI}]$

CSD1b response available for MODE = -1. Note that this model approaches the CD as U approaches infinity.

Exponential Distribution of Activation Energies

(Special fixed values of PHI: EDAE1)

NELEM=30

This is a simplified version of the NELEM=12 version of EDAE1. It uses closed form expressions for EDAE1 response for the special FIXED values PHI=1 (possibly appropriate for conducting system response) or PHI=0 (possibly appropriate for dielectric system response: see O circuit, and Refs. 33, 34, 45, and 47). Fitting is thus more accurate and much quicker for these fixed values since integration is not necessary.

Here, R, T, and U may be free, with PHI fixed at 1 or zero. For initial fitting, take U=-30, fixed, and after convergence try U free as well. When PHI < 0, the quantity $G=1/R$ is subtracted from the EDAE expression at the admittance level and the result then converted to the impedance level. To do this, which is sometimes useful in theoretical studies, use PHI= -1 (for the PHI=1 case with subtraction) or PHI= -0.01 (in place of the PHI=0 case).

The BDM Effective-Medium Response Function

NELEM = 31

This DCE, which is only appropriate for a conducting system, is an approximate version of a more exact, but complicated and implicit, effective-medium hopping response solution (Ref. 32). The BDM is discussed and used in Refs. 32-34 and has been suggested to be of wide applicability by Dyre. In using the present response function to fit actual data, it is necessary to provide an appropriate value of CELCAP. Here, the input parameters R, T, and U, which may be free to vary, can play several different roles (see below), but the value of PHI is always fixed.

Microscopic (PHI < 0) and macroscopic (PHI > 0) versions of the BDM are available. Define E_c as the effective activation energy (in e.V.) for a thermally activated system, and $x_c = E_c/k_B T_a$. When U > 0, $E_c = U$ (and the absolute value of the input quantity ATEMP is the absolute temperature, T_a); otherwise $x_c = |U|$.

When PHI is fixed at a negative value such as -1, the input quantity R is the quantity τ_a , the minimum relaxation time for a thermally activated system; it may be very small - of the order of $10^{**(-20)}$ or less. Here the input quantity T is not used when T is zero or positive. When it is negative, it may be taken free to vary, and its absolute value multiplies the microscopic intrinsic relaxation time, τ , which is calculated internally using CELCAP and other input quantities. (For more details, see the Fortran source code in LV5.FOR for this response function). Because there is some uncertainty in a constant used in the microscopic calculation, this use of T allows more flexibility in fitting.

When PHI is positive and fixed at 2, 4, or 8, the macroscopic version is invoked. Here R is the strength of the relaxation at the impedance level ($R_0 - R_{inf}$). If PHI = 2, T is the low-frequency-limiting value of the dielectric constant increment associated with the conducting system. Then, the intrinsic relaxation time, τ , is just $R*(CELCAP*T)$. If PHI = 4 or 8, however, T is the intrinsic relaxation time itself. If PHI = 8, particularly appropriate for simulations and theoretical calculations, the calculated response is like that with PHI = 4 but the zero-frequency limiting value of the admittance has been subtracted out before conversion back to the impedance level.

Accurate KWW Response Function for BETA = 0.5

NELEM = 32

"Exact" Williams-Watts (stretched exponential) response: BETA = 0.5 responses calculated from exact DRT with possible τ_{amin} cutoff. See Refs. 40, 46 for details of cutoff procedure.

If MODE < 0, calculate CSD1a = KWW1=K1 form directly. Example: Mode = -1 or -2.

If MODE >= 0, then calculate CSD=CSD0=KWW0=K0 or DSD=KD form directly.

MODE= 8: Transient Response - CSD0 or DSD USE ZZ levels and real data

MODE= -8: Transient Response - CSD1

MODE=16: Coupling Model frequency response; use U= -40 to eliminate cutoff effects

Cutoff determined by value of U; *** it will usually be negative here. When $U < -40$ or so, there is no effect in any practical measured frequency range. $U = y_{min} = \ln(\tau_{amin}/\tau_0)$. Cutoff: $x_{min} = \exp(U)$. When U is positive and large, response approaches Debye; see Ref. 46.

Free parameters are R and $T = \tau$ or e_{τ} , and possibly U. The input value of PHI is immaterial here but must be fixed. Also, set a fixed value of integration accuracy, IGACC, such as 5 or 6.

Temperature-dependence fitting formulas

NELEM = 33

This choice provides a wide variety of fitting formulas for temperature-dependent quantities. In particular, various Arrhenius and Vogel-Fulcher forms are available with and without pre-exponential temperature-dependent factors. Thus, temperature-dependent values of τ , Rho0 (the low-frequency limiting (DC) value of Rho), or σ_0 ($=1/\text{Rho0}$) data may be fit with many different choices. The data file uses absolute temperature in the first column (OMEGA) and the value of the dependent variable in the second column. The third column is initially arbitrary. For plotting purposes, the output file contains $1000/\text{temperature}$ in the imaginary output column (See LINOTR, etc. files, and use RNTMP.BAT). Since the fit is of real-part only, the presence of values in the imaginary column does not affect the fit results. For simplicity, definitions and fitting formulas are listed below in approximately their Fortran form. See Refs. 42 and 43 for more details concerning appropriate fitting formulas.

As an example, suppose we want to fit τ data to an Arrhenius form: $\tau = \text{taua} \cdot \exp(E/k \cdot \text{TEMP})$. Here taua is the inverse of a barrier attempt frequency, is usually less than $10^{**}(-12)$ s, and is taken temperature independent. Then choose $\text{ATEMP} = -4$, $\text{CELCAP} = 1$, $U = 0$, $\text{PHI} = 0$, $T = \text{taua}$, and $R = E$, with R and T taken free. Alternatively, suppose we want to fit data to $\text{Rho0} = (\text{taua}/\epsilon_v)(k \cdot \text{TEMP}/E) \cdot \exp(E/k \cdot \text{TEMP})$. Then take $\text{ATEMP} = 0$, $\text{CELCAP} = \epsilon_v$, $U = 0$, $\text{PHI} = 1$, $T = \text{taua}$, and $R = E$, with R and T again taken free to vary. For σ_0 fitting, one might use $\text{ATEMP} = 4$.

ATEMP SELECTS FORM; DATTYP **MUST** BE REAL ***
 REAL FITS ONLY. CELCAP may be set to unity when appropriate.
 TEMP = OMEGA = TEMPERATURE (K)
 USUALLY, $R=E$ (eV), and R , PHI , and U are possible input parameters
 THE QUANTITY $1000/\text{TEMP}$ IS OUTPUT IN IMAG. COL. OF DATA, SO
 $\text{CTEMP} = 1.D3/\text{OMEGA}$. Equivalent to $10^{**}3/\text{TEMPERATURE}$
 IF $\text{CELCAP} = C_g$, THEN USE R_0 OR Y_0 ; IF $\text{CELCAP} = \epsilon_v$, THEN USE RHO0 , OR
 SIGMA0 FOR DISTEL (distributed-element) OUTPUT

DEFINITIONS AND FITTING FORMS FOLLOW BELOW:

TEMPU	=	TEMP - U	! (T - T0)
EK	=	R * AKINV	! E/k
EE	=	EK / TEMP	! E/(kT)
EEU	=	EK / TEMPU	! E/[k(T - T0)]
TAV	=	T / CELCAP	! Tau / CELCAP

EXT	=	EXP(EE)
EXTU	=	EXP(EEU).

VALUE OF FITTING FORMULA

ATEMP:

$$-64 \quad \text{DISTEL} = \text{TAV} * (\text{OMEGA} ** \text{PHI}) * \text{EXTU} + \text{i} * \text{CTEMP}$$

$$-4 \quad \text{DISTEL} = \text{PHI} + \text{T} * \text{EXTU} + \text{i} * \text{CTEMP}$$

$$-2 \quad \text{DISTEL} = \text{PHI} + \text{TAV} * \text{EXTU} + \text{i} * \text{CTEMP}$$

$$-1 \quad \text{DISTEL} = \text{TAV} * (\text{OMEGA} ** \text{PHI}) * \text{EXTU} + \text{i} * \text{CTEMP}$$

**In conventional notation, with $\text{DISTEL} = \rho$, $R = E$, and $U = T_0$, this response is:

$$\rho(T) = (\rho_a / \text{CELCAP}) T^\phi \exp[(E / k_B) / (T - T_0)] + i(10^3 / T)$$

Usually take ρ_a and E free to vary, and $\phi = 0$ or ± 1 , but it may be taken variable. See the example file OMFTMP1 in FITTESTS, Circuit O

$$0 \quad \text{DISTEL} = \text{TAV} * (\text{EE} ** (-\text{PHI})) * \text{EXTU} + \text{i} * \text{CTEMP}$$

$$1 \quad \text{DISTEL} = \text{TAV} * (\text{EE} ** (-\text{PHI})) * (\text{EXT} - 1.\text{D0}) + \text{U} + \text{i} * \text{CTEMP}$$

$$2 \quad \text{DISTEL} = 1.\text{D0} / (\text{TAV} * (\text{OMEGA} ** \text{PHI}) * \text{EXTU}) + \text{i} * \text{CTEMP}$$

$$4 \quad \text{DISTEL} = 1.\text{D0} / (\text{TAV} * (\text{EE} ** (-\text{PHI})) * \text{EXTU}) + \text{i} * \text{CTEMP}$$

$$8 \quad \text{DISTEL} = \text{U} + 1.\text{D0} / (\text{TAV} * (\text{EE} ** (-\text{PHI})) * (\text{EXT} - 1.\text{D0})) + \text{i} * \text{CTEMP}$$

$$16 \quad \text{DISTEL} = -\text{R} * \text{DLOG}(\text{TEMP}) - \text{PHI} * (1.\text{D0} - \text{TEMP}) + \text{i} * \text{CTEMP}$$

$$>16 \quad \text{DISTEL} = \text{U} + \text{R} * (\text{TEMP} ** \text{PHI}) + \text{i} * \text{CTEMP}$$

Simplified, *generalized* exponential DAE response model

NELEM = 34

This is a simplified version of the full GED response model implemented in Circuit O. See its description there. Here, only the parameters, R,T (or ϵ_τ), U, and PHI are used, as well as GAMMA1 [set by the value of P(13)]. When $U > 0$, the model is symmetric, while for $U < 0$, it is antisymmetric. It reduces to the exponential DAE models selected by NELEM = 12 and 13 when GAMMA1 is fixed at 1. When $|\text{MODE}| = 8$, the model yields transient rather than frequency response. This DCE is not necessarily needed with the O circuit since that circuit contains the more general GEDAE response for the choice: NCH = 1.

Now define PHI1 as the value of PHI found for CSD1a fitting when $\text{MODE} < 0$, and PHI0 the value obtained for CSD0 or DSD fitting with $\text{MODE} \geq 0$. When $\text{GAMMA1}=1$, fitting of exact CSD0 frequency-response data (obtained, for example, with $\text{MODE}=1$) with the CSD1a approach (e.g., $\text{MODE}=-1$), or vice versa, yields identical fits with the same parameter values, except that $\text{PHI0} - \text{PHI1} = 1$, a requirement for such exponential DRTs. But when GAMMA1 is not fixed at unity, the resulting DRT is not simple-exponential in form and the above relation no longer applies. It is interesting to compare results with KWW fitting (e.g., $\text{NELEM}=36$; see also Refs. 45 and 47). There, one finds $\text{BETA0} + \text{BETA1} \cong 1$ instead of the above relation between PHIs. But comparison of CSD1a fits with DSD ones of the same data (use the O circuit with $\text{P}(40)=\text{NCH}=1$ for EDRT, EDAE situations), shows that $\text{PHI1} \cong \text{PHI0}$ and $\text{BETA1} \cong \text{BETA0}$.

Accurate KWW Response Function for $\text{BETA} = 1/3$

NELEM = 35

"Exact" Williams-Watts (stretched exponential) response: $\text{BETA} = 1/3$ responses calculated from exact DRT with cutoff. See Ref. 40 for details of cutoff procedure. **After a fit, when you go to "Launch Levview," **extra output** appears at the end of the Print, or View, Report choices."

If $\text{MODE} < 0$, calculate $\text{CSD1a} = \text{KWW1}=\text{K1}$ form directly "This choice is the **UN** model."
 If $\text{MODE} \geq 0$, then calculate $\text{CSD}=\text{CSD0}=\text{KWW0}=\text{K0}$ or $\text{DSD}=\text{KD}$ form directly

$\text{MODE} = 8$: Transient Response - CSD0 or DSD
 $\text{MODE} = -8$: Transient Response - CSD1
 $\text{MODE} = 16$: Coupling Model frequency response; use $U = -40$

Cutoff determined by value of U; *** it will usually be negative here. When $U < -40$ or so, there is no effect in any practical measured frequency range. $U = \ln(\text{taumin}/\text{tau0})$. Cutoff: $x_{\text{min}} = \exp(U)$.

Free parameters are R and $T=\text{tau}$ or ϵ_τ , and possibly U. The input value of PHI is immaterial here but must be fixed. Note that the present K1 response model is also denoted the **UN (universal model)**. A fitting example, input file OTSTMOY.UNS, using the CUNS model, is in the FITTESTS folder, O-circuit.

This function first accurately calculates a K0 or K1 distribution of relaxation times (DRT) from an exact expression using Airy functions; then it calculates the associated frequency or transient response by means of an integral over the DRT. The accuracy of the integration is set by the choice of IGACC and involves $10^{-\text{IGACC}}$. A value of **IGACC** of 6 or 7 is usually adequate for simulated data and a choice of 3 or 4 is usually large enough for ordinary experimental data.

KWW Response Function for arbitrary BETA

NELEM = 36

This approach uses series and an approximate DRT formula to first calculate the KWW DRT (See the NELEM = 37 choice), and then uses the DRT, with or without cutoff, to calculate KWW frequency or transient response to high accuracy. It uses the subroutines GDRT and KWWDRRT. It is most accurate for $0.2 \leq \text{BETA} \leq 0.8$ but can be used outside of this range. Convergence is very slow for $\text{BETA} > 0.8$, so the use of the NELEM=10 choice for $0.8 < \text{BETA} \leq 1$ is recommended; see below. See Refs. 45 and 47 for more details of the use of this important DCE.

If $\text{MODE} < 0$, calculate $\text{CSD1a} = \text{KWW1}=\mathbf{K1}$ form directly
 If $\text{MODE} > 0$ or = 0, then calculate $\text{CSD}=\text{CSD0}=\text{KWW0}=\mathbf{K0}$ or $\text{DSD}=\mathbf{KD}$ form directly

MODE = 8: Transient Response - CSD0 or DSD: **K0** or **KD** response

MODE = -8: Transient Response - CSD1 = $\text{KWW1}=\mathbf{K1}$ response

Comparison of transient results with the NELEM=35 ones, show good agreement until the response has dropped from its initial value by a factor of 10^{**6} , and worse agreement for smaller values. Transient calculations must be made at the Z immittance level.

Possibly free parameters: R, TAU, U, PHI (=BETA). ATEMP (always fixed) should usually be set to ≤ 0 , but see the actual NELEM =36 algorithm in LV5.FOR for other choices.

Since this algorithm is appreciably slower than that of NELEM=10 (no cutoff), it is often useful to fit data first with the latter choice, then with the present one, perhaps starting with a free value of U of -15 or -20. Further, for the present choice, it is unnecessary to set IGACC to > 4 , and a value of 2 or 3 will usually be appropriate for most experimental data. In the integration, the top limit (TOP) is automatically calculated and depends on the value of BETA. The bottom limit is BOT.

$U < -1000$ BOT = $-12/\text{BETA}$

$U = -1000$ BOT = $-6/\text{BETA}$

$U > -1000$ BOT = input value of U (may be positive); may be a free parameter.

The first two values of BOT may yield only minimal cutoff, and the first runs slower than the second. Set U fixed at less than -40 to eliminate cutoff effects.

At the beginning of a fit run, values of the following initial quantities used in the fit appear on the screen:

666, BETA, BOT, TOP

If BETA and U are free parameters, both BOT and TOP can change during a fit.

One of the series used in this work is Eq. 54 in Montroll and Bendler, J. Stat. Phys. 34 (1984) 129. Others, and the epsilon algorithm are also used.

The approximate DRT used in one range appears in E. Helfand, J. Chem. Phys. 78(1983)1931.

When the O-circuit is used with two conductive-system DCEs (see the discussion on p. 4-7) and the second one is of KWW form, with the first one of CSD1a character (e.g., Mode = -1), then if NELEM=36 and one wants the second one to be KWW0 rather than KWW1, set TEMP to < -900 .

KWW DRT for arbitrary BETA

NELEM = 37

This approach uses subroutine KWWDRT (which involves the combination of a series and an approximate DRT formula) to calculate very accurately the KWW DRT without cutoff for arbitrary BETA. (See the NELEM = 36 description for more details). Set MODE=1 or -1.

Possibly free parameters are : R, TAU (=TAU0), and BETA. U is not used.

The “frequency” column in the data must list TAU values, not frequency ones. It can range from very small to a maximum value of about $300 \cdot \text{TAU0} / \text{BETA}$. Set DATTYP=R (real) and DINP, DFIT, PINP, and PFIT to ZZRR. With IRCH=1, and R=1, the calculated KWW DRT values associated with each TAU value will appear in the Real-data column. With other values of IRCH and unknown DRT data in the Real column, such data can be fitted to the KWW DRT, yielding estimates of R, TAU0, and BETA as well as measures of the degree of fit.

It is convenient to use the O-circuit with P(35) or P(40) set to 4 for this calculation/fit.

**For the DRT at the dielectric level, use the DINP, DFIT inputs in Line 2 of the input file : EERR, not ZZRR. In an epsilon-data K0 or K1 fit using the O-circuit, set P(40)=4 and possibly P(20) = 27 or 20. See examples in the FITTESTS folder.

For the KWW0=K0** or **KD** (Mode=1) with BETA = 0.5, the exact DRT is alternately used, including cutoff if present. Set BETA = 0.5 fixed, and ATEMP < 900. The program also can calculate KWW1=**K1** results if MODE = -1. Cutoff is determined by the value of U, which may be a free variable when fitting experimental data. No cutoff effects appear when $U < -50$, since the program returns a value of 0 for the DRT for all values of $\text{TAU} < \exp(U)$. Cutoff effects are usually appreciable when $U > -10$. See the various example files in FITTESTS, Circuit-O.

Summary of some NELEM = 9 PNP and PNPA fitting models (See NELEM=9 and M252,253)

It is desirable to express PNPA frequency-response equations in terms of total normalized impedance ($Z_{TN} \equiv Z_T / R_\infty$), normalized interface impedance ($Z_{iN} \equiv Z_i / R_\infty$), and high-frequency-limiting parameters, R_∞ and ε_∞ . Then $\tau \equiv R_\infty C_\infty \equiv R_\infty (A/L) \varepsilon_V \varepsilon_\infty$; where A is the area of a plane-parallel electrode; L is the separation of the two electrodes; and ε_V is the permittivity of vacuum. Next define the new normalized parameters $S \equiv (i\omega\tau)$, $U \equiv S^\psi$, $P \equiv p^2 \equiv 1+U$, $q \equiv Mp$, and $Q \equiv \tanh(q)/q$, with $0 < \psi \leq 1$. For $\psi < 1$, PNPA anomalous diffusion is involved and PNP diffusion when $\psi = 1$. Note that for the PNP situation $\varepsilon_0 = \varepsilon_\infty / Q_0 \equiv \varepsilon_\infty [M \operatorname{ctnh}(M)]$, where usually $10 < M < 10^4$. JRM papers in <http://jrossmacdonald.com> are cited below as **Mxxx**.

The equation connecting the expression for a normalized impedance model and its interface part is (M124)

$$Z_{TN} = 1/[S + 1/(1 + Z_{iN})], \text{ and its inverse is } Z_{iN} = [(1 + S)Z_{TN} - 1]/[1 - SZ_{TN}].$$

We now define three distinct PNPA expressions **(A)** $Z_{TN} = (U + Q)/[U(1 + S) + Q(S - U)]$; **(B)** $Z_{TN} = (S + Q)/SP$; and **(C)** $Z_{TN} = (U + Q)/SP$. Equation (A) follows from the PNPA expression for Z_{iN} : Eq.A.11 in M252; Eq.(B) is Eq.A.12 and is an incorrect version of Eq.(A); and Eq.(C) is Eq.10 in J. Phys. Chem. B **113**, 11371 (2009) by Lenzi, Evangelista, and Barbero.

In the PNP limit, $\psi = 1$, all three equations reduce to (M92, Eq. 1): $Z_{TN} = (S + Q_1)/S(1 + S)$, where Q_1 is the $\psi = 1$ version of Q when p involves S rather than U . The corresponding PNPA Z_{iN} expressions, Ai, Bi, and Ci, are

$$Z_{iN} = PQ/[U(1 - Q)], \quad Z_{iN} = (1 + S)Q/[S(1 - Q)], \text{ and}$$

$$Z_{iN} = [(U - S) + (1 + S)Q]/[S(1 - Q)], \text{ respectively.}$$

Equation Ai is the same as Eq. 3 on p. 4-10. All three above equations reduce for the PNP situation to Eq. Bi with Q replaced by Q_1 . Note the minor difference between Eqs. B and C above. B follows from Bi, while C follows from Ci, not Ai. It is clear that only Ai involves purely interface anomalous diffusion quantities, a reasonable requirement, while Bi and Ci both involve the PNP quantity S .

These equations are available for fitting data with NELEM=9 and additional choices involving P18 below. If $9 \leq P18 \leq 11$, then the choice P18=9, 10, or 11 yields Z_T for Eqs. A, B, or C, respectively, while for -9,-10, or -11, Z_{iN} is produced. For P18=4 or 8, however, weighted sums involving S and U are used with Eq. Ai to calculate responses that involve both PNP and PNPA behavior simultaneously. In this case, let $A=P8$ and $(1-A)=AM$. Then for P18=4, $Z_{iN} = 1/Y_{iN}$ with $Y_{iN} = [(A/Y_{i1}) + (AM/Y_{i\psi})]$.

For P18=8, U in Eq. Ai is replaced by $S+U$. For P18=4 or 8 here, Z_T is output, and their negative values yield Z_{iN} . Results apply for either equal mobilities or only negative charges mobile (M252).

See also many PNP/PNPA models available in Circuit H.

Fitting Circuit Models Documentation

This section describes the circuit models used by the LEVM program to fit impedance or immittance, data. Included is a discussion of the special options of the models, schematics of each circuit, parameter assignments, and information on implementation. The circuit models are designated A through K, O, and R through T. For example, circuit model A appears in the subroutine ASUB.

For greatest flexibility in adding new circuits, two stand-alone subroutines, each containing a new fitting model, may be used as part of LEVM. These are designated as SUBROUTINE XSUB(M,FREQ,P,F), where X is S or T. They may be written as separate units, so one or more of them can be separately compiled and linked with LEVM during the final linkage stage. Then when changes are made in XSUB, only it will require recompiling. Dummy XSUB routines are provided but do nothing at present. Any new circuit desired can be added in XSUB (with up to a total of 50 circuit parameters) and linked during recompiling in order to provide almost unlimited expandability of the LEVM circuit possibilities.

Note that **by the use of special distributed element (DE) additions, the circuits discussed in this section can be greatly enlarged in scope** (See Section 4). Not only can DE circuit elements be used to provide inductance, and more complicated ideal-element subcircuits, but they may be used for many different types of distributed elements, including general diffusion and distribution of activation energies elements. Simplified versions of the latter are available as DE's, and a full-response version is included by the DAE element of Circuit D and in Circuit O. Also the BCE element of Circuit H provides exact, completely blocking response and more general partially-blocking response as well. The inductance, L, invoked by P(30) in these circuits, is provided to account, at least in part, for any wiring inductance present and of importance. Such inductance is usually negligible except at very high frequencies and may often be omitted. Of course, this L may be used for other reasons as appropriate.

Introduction

Part of the flexibility of the LEVM program is its ability to select among several different circuit models by using the FUN parameter. The program currently supports models A-T, though not all of these have been implemented (see above). Each call to the subroutine MODEL causes it to return an array of impedance values corresponding to the circuit being used, the array of frequencies at which each of the data points was read, and the current value of the model parameters. The actual subroutines which do the calculations for each of the circuit models return only the impedance. The subroutine MODEL then converts the impedance to whatever immittance system is being fitted (e.g. Y [admittance], M [modulus], or E [epsilon]).

Parallel, Reference, and Weighting Parameters

The elements CP and RP appear in most of the fitting circuits. They may be used directly as part of the fitting model (e.g. CP = Cg, the geometric capacitance of the electrode-material system); they may be used to account for the input impedance of the measuring instrument at the input to the experimental cell (see the **Spinolo et. al** paper in the reference list); or they may be used for the combination of both effects.

LEVM allows a maximum of 42 parameters to be free at the same time. If NP is the maximum number of free and fixed model parameters, NP is 50 for LEVM.EXE unless LEVM is recompiled. Parameters P(1) through P(32) may be free, but all above P(32) are automatically fixed at their input values, except for the R-Circuit where P(41) to P(50) may also be free (NP=50). In the simplest case, set the input number of free and fixed parameters, NP, to 32 (its minimum value), and no other parameter values need be considered. In this situation, there can be no more than 7 lines in the input file for parameter choices. But if NP > 35, one must have eight lines to ten lines available in the parameter-choice part of the input file.

When the input value of NP is set negative, fitting using a reference impedance (consisting of P(33)=RR and P(34)=CR) is invoked (see Eq. 2 in Spinola). The RR and CR elements are included for possible use as measuring reference elements (usually with a FRA). They are in parallel, must always be fixed, and have no effect unless RR is nonzero and a complex data fit is being run with ZZRR input choices. They are only appropriate for use when the input data involves the FRA input/output voltages V_x and V_y , not the impedance components (see discussion in Spinolo et. al). When this is the case, the use of known input values of RR and CR (may be taken zero if appropriate) allows the fitting to take place at the voltage level, possibly desirable under some conditions, rather than at the usual immittance level. If ZM is the impedance calculated by the fitting model and ZRF is the impedance of the reference elements, then invocation of this choice, as above, replaces the fitting model impedance by the quantity $ZM/[ZM + ZRF]$; this, in turn, is the quantity then compared with the voltage-ratio data by least squares, rather than ZM.

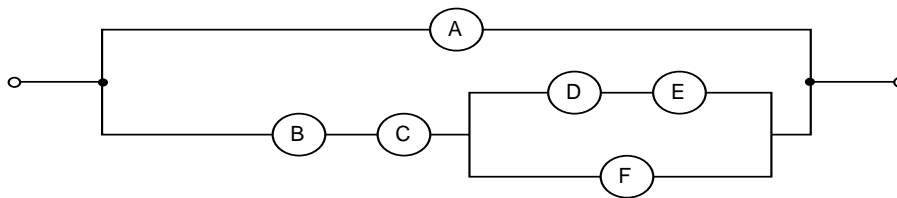
Alternatively, when NP is positive and set to 33 or more, one may use the weighting choices IRCH = 5 or -5. Then a special weighting appropriate for FRA data is invoked (See Ref. 18) and also the discussion under the IRCH=5 choice in Section 3). The parameters P(33), P(34), up to possibly P(39) are then: the first reference resistor value used, the number of the last frequency for which this resistor is used (e.g., 20, indicating that the data after the first twenty frequencies use a different reference resistor), the next reference resistor, the number of the last data point which uses this resistor, and so on up to P(39). There is no need to give a number for the last region. For example if there are 40 frequencies and the reference resistors 100, 10,000, and 1,000,000 ohms are used in order for the first 10, the next 10, and the last 20 frequencies, one would set P(33)=100, P(34)=10, P(35)=10000, P(36)=20, and P(37)= 1,000,000. Note that the input file would need to include eight lines for the parameters for these choices.

Removal of Circuit Elements

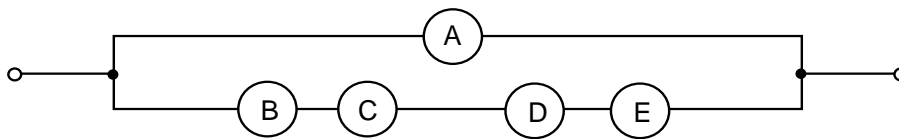
Since the present complete circuit models are quite intricate, it is virtually always desirable to use simplified versions of the circuits, as if some of the elements had not been incorporated into the circuit. It would be possible, though sometimes difficult, to fix the values of the extraneous elements at very small or very large values (whichever would be appropriate) to minimize their effect on the system. The circuits in LEVM make this much simpler to do, using a special convention: **Any element with a parameter value set to zero is assumed to be not present.** It is this feature that makes the present relatively few full circuits allow many thousands of smaller circuits to be defined and used in LEVM.

If a parameter set to zero is an element in series with another element, this amounts to an impedance of zero (or short circuit). If it is in parallel with another element, it is equivalent to an infinite impedance (or open circuit). This feature can greatly simplify the process of choosing parameter values, but the programmer must remember that setting a parameter to zero is NOT always equivalent to giving it a zero impedance. It will be useful to consider an example circuit to see how this process works.

Below is shown an example circuit with elements A-F.



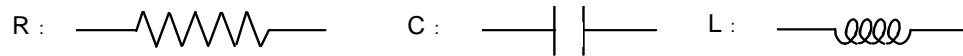
In this circuit, if parameter A were zero, the circuit would just be the lower limb of what is pictured. If F were set to zero instead, it would look like



In both of these cases, the impedances of the elements set to zero are infinite, as in an open circuit. Alternately, if B and D were set to zero, their impedances would be zero. But if instead D and E were both set to zero, B and C would be in series with F, as if D and E were an open circuit with infinite impedance. This behavior is quite straightforward and can simplify parameter choices but care is urged in its use.

Discrete Elements

The circuit models make extensive use of basic discrete elements: resistors (R), capacitors (C), and inductors (L). They are denoted in the model diagrams by the standard symbols:



The impedance of each element is as follows:

$$Z_R = R, \quad Z_C = 1 / (i\omega C), \quad Z_L = i\omega L,$$

where $i = \sqrt{-1}$ and $\omega =$ angular frequency of the AC signal.

Distributed Elements

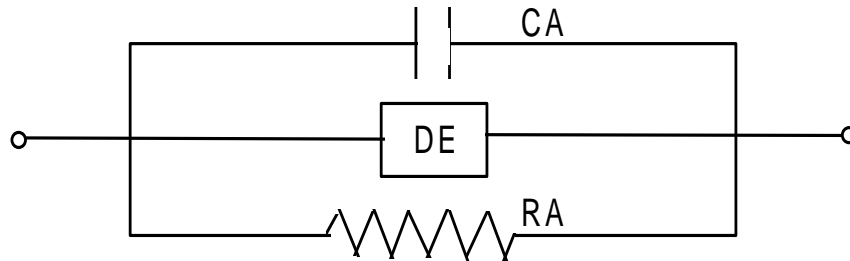
In addition to the standard discrete elements, there are many distributed elements available designed to be of use in Immittance Spectroscopy. The model circuits allow any of a number of these elements to be substituted into the circuit. Distributed elements are represented as impedances, Z_{DE} , and in a circuit diagram as:



Each distributed element has five associated parameters. NELEM is an integer which selects the distributed element being used; R, T, U, and PHI are all parameters which determine the characteristics of the element. Note that in circuit diagrams (as well as in the model subroutines) these parameters are usually identified as NDE, RDE, TDE, UDE, and PDE. If a distributed element is to be omitted from a circuit, NELEM must be set to zero - the other DE parameters will be ignored. Note that for NELEM = 1 a parallel R-C combination with resistance R and capacitance T is substituted for the element. Also, the DE for NELEM = 17 is made up of R, L, and C elements, and other R, L, and C combinations can be formed using NELEM = 2 to 5 with PHI fixed at -1, 0, or 1. For more information on the different distributed elements available, consult Section 4.

Augmented Distributed Elements

A special option exists in some of the circuits which allows a distributed element to be 'wired' in parallel with a resistor and a capacitor. This type of element is represented in circuits as:



It should be noted that in the present version of LEVM, any of these elements may be used with or without the other. When R_A or C_A appear separately or together but with no DE, it is a good idea to set the NELEM (NDE) parameter to zero but even this is unnecessary if the other DE elements are set to zero.

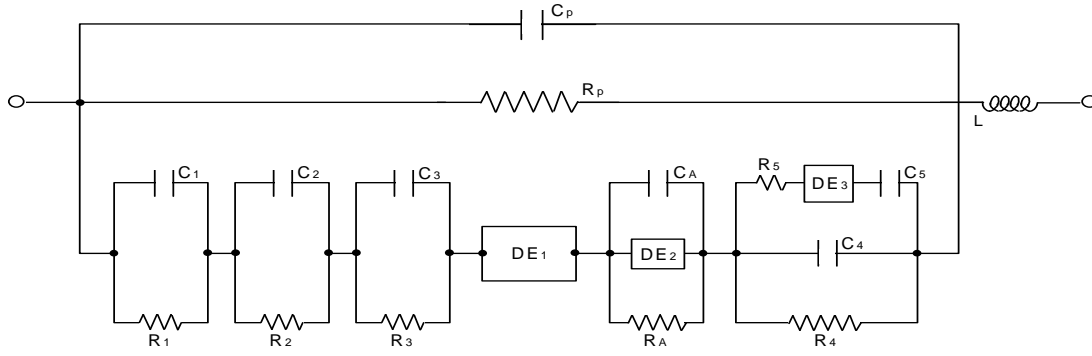
Augmented distributed elements have 6 associated parameters as well as NELEM: the four associated with distributed elements (R,T,U,PHI) and the values of the augmenting resistor and capacitor (R_A and C_A). Note that setting R_A and/or C_A to zero effectively removes them from the circuit, leaving an unaugmented distributed element still present if they are both zero.

The impedance of an augmented distributed element, Z_{ADE} , is calculated as follows when Z_{DE} is nonzero:

$$\text{If } R_A = 0, Z_{ADE} = 1 / (i\omega C_A + 1/Z_{DE})$$

$$\text{Otherwise } Z_{ADE} = R_A / (i\omega R_A C_A + R_A/Z_{DE} + 1).$$

Circuit A



(a)

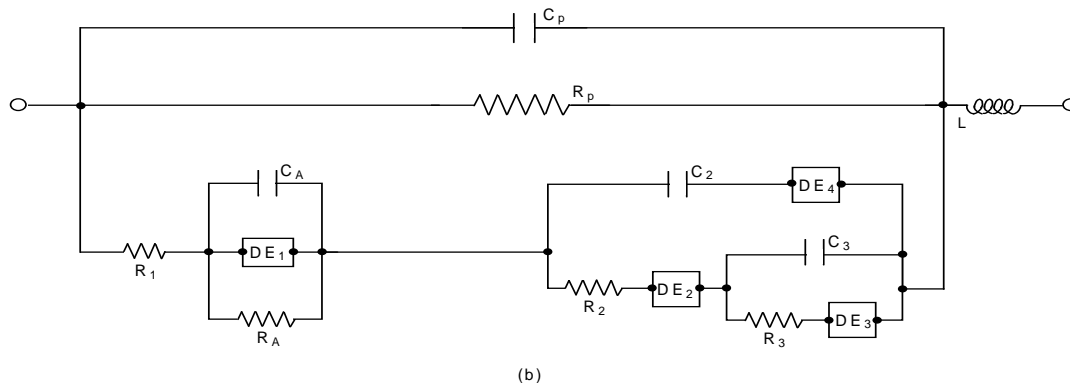
Parameter Assignments

R1 = P(1)	RDE2 = P(11)	C3 = P(21)
C1 = P(2)	TDE2 = P(12)	RA = P(22)
R2 = P(3)	UDE2 = P(13)	CA = P(23)
C2 = P(4)	PDE2 = P(14)	R4 = P(24)
R3 = P(5)	NDE2 = P(15)	C4 = P(25)
RDE1 = P(6)	RDE3 = P(16)	R5 = P(26)
TDE1 = P(7)	TDE3 = P(17)	C5 = P(27)
UDE1 = P(8)	UDE3 = P(18)	RP = P(28)
PDE1 = P(9)	PDE3 = P(19)	CP = P(29)
NDE1 = P(10)	NDE3 = P(20)	L = P(30)

POSSIBLY FREE, WEIGHTING PARAMETERS:

U = P(31)	Xi = P(32)
-----------	------------

Note that this circuit allows one to use 6 RC's in series if so desired. To do so, use nonzero values of R1-C1, R2-C2, R3-C3, R4-C4, RA-CA, and NDE1=2, RDE1=0, TDE1=T=1/R, UDE1=U=C, and PDE1=PHI=0 (see p. 4-3), or use NDE1=2, RDE1=R, TDE1=T=C, UDE1=U=0, and PDE1=PHI=1 (fixed, not free). See also Circuits K and R.

Circuit B**Parameter Assignments**

R1 = P(1)	RDE2 = P(11)	RDE4 = P(21)
RA = P(2)	TDE2 = P(12)	TDE4 = P(22)
CA = P(3)	UDE2 = P(13)	UDE4 = P(23)
R2 = P(4)	PDE2 = P(14)	PDE4 = P(24)
C2 = P(5)	NDE2 = P(15)	NDE4 = P(25)
RDE1 = P(6)	RDE3 = P(16)	R3 = P(26)
TDE1 = P(7)	TDE3 = P(17)	C3 = P(27)
UDE1 = P(8)	UDE3 = P(18)	RP = P(28)
PDE1 = P(9)	PDE3 = P(19)	CP = P(29)
NDE1 = P(10)	NDE3 = P(20)	L = P(30)

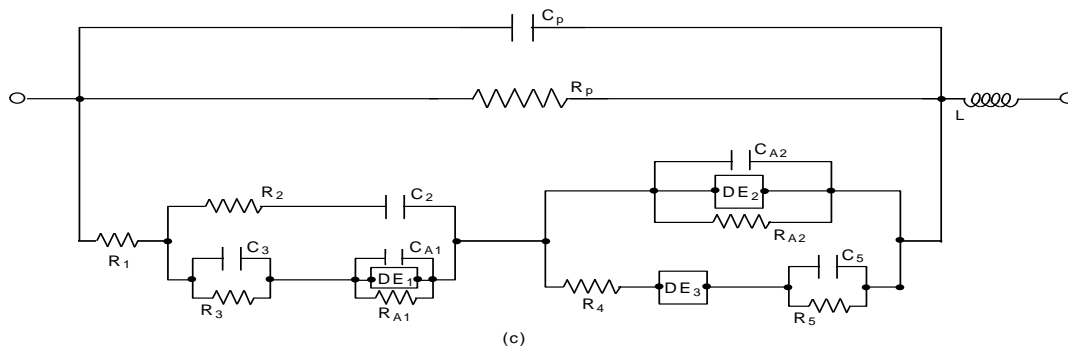
POSSIBLY FREE, WEIGHTING PARAMETERS:

$$U = P(31)$$

$$Xi = P(32)$$

For the special choice $UDE3 < 0$, some parameters are redefined. For $UDE3 < -9$, appropriate for an unsupported case, P(4) is $R2/R1$ and P(5) is $C2/CP$, both now normalized quantities. Here, R1 and CP must be non-zero, free or fixed. Alternatively, when $U = -1, -2, \text{ or } -3$ and $NDE3 = 13$ or 15 , appropriate for neutral species or the supported case, $P(16) = RDE3/R2$, $P(27) = C3/R2$, and $P(26) = R3/R2$. These changes allow cases to be better treated where the actual circuit elements, $R2 (\neq 0)$, $R3$, $C3$, and $DE3 = Z_D$, the general diffusion expression (see discussion of NELEM=15, Section 4) are all inter-related. See J. Electroanal. Chem. **101**(1979)307. To eliminate R3 and C3 effects, set $P(26) = 0$ and $P(27) = 10^{-30}$.

Circuit C



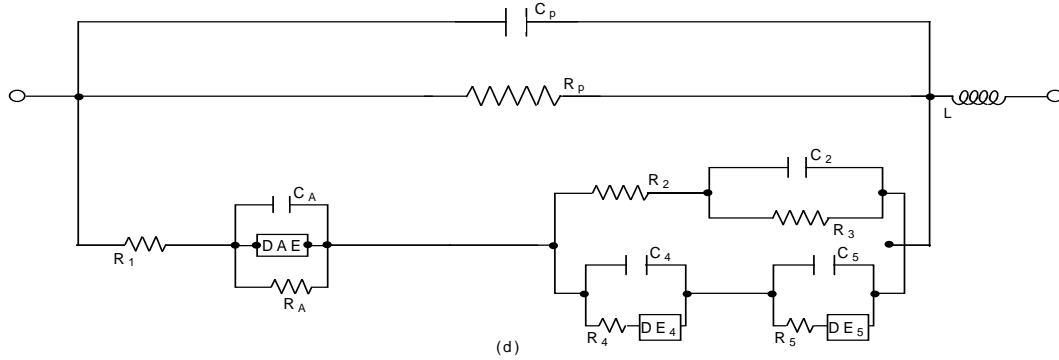
Parameter Assignments

R1 = P(1)	RDE2 = P(11)	RA1 = P(21)
R2 = P(2)	TDE2 = P(12)	CA1 = P(22)
C2 = P(3)	UDE2 = P(13)	RA2 = P(23)
R3 = P(4)	PDE2 = P(14)	CA2 = P(24)
C3 = P(5)	NDE2 = P(15)	R4 = P(25)
RDE1 = P(6)	RDE3 = P(16)	R5 = P(26)
TDE1 = P(7)	TDE3 = P(17)	C5 = P(27)
UDE1 = P(8)	UDE3 = P(18)	RP = P(28)
PDE1 = P(9)	PDE3 = P(19)	CP = P(29)
NDE1 = P(10)	NDE3 = P(20)	L = P(30)

POSSIBLY FREE, WEIGHTING PARAMETERS:

U = P(31)	Xi = P(32)
-----------	------------

Circuit D



Parameter Assignments

R1 = P(1)
 RA = P(2)
 CA = P(3)
 RDAE = P(4)
 TDAE = P(5)

C2 = P(11)
 R3 = P(12)
 R4 = P(13)
 C4 = P(14)
 DCH = P(15)

RDE5 = P(21)
 TDE5 = P(22)
 UDE5 = P(23)
 PDE5 = P(24)
 NDE5 = P(25)

U1 = P(6)
 U2 = P(7)
 PHI1 = P(8)
 PHI2 = P(9)
 R2 = P(10)

RDE4 = P(16)
 TDE4 = P(17)
 UDE4 = P(18)
 PDE4 = P(19)
 NDE4 = P(20)

R5 = P(26)
 C5 = P(27)
 RP = P(28)
 CP = P(29)
 L = P(30)

POSSIBLY FREE, WEIGHTING PARAMETERS:

U = P(31)

Xi = P(32)

When $DCH = 0$ (i.e. $P(15) = 0$), $P(4) - P(9)$ are used for the six parameters of the general exponential distribution of activation energies, EDAE. They are so designated above. $U1$ and $U2$ are the upper limits of integration in the integrals of the model, and $PHI1$ and $PHI2$ are exponents in it.

The accuracy of the calculation of the DAE element is determined by IGACC in the input file. Start with 2; after convergence, try 3 or 4 for IGACC.

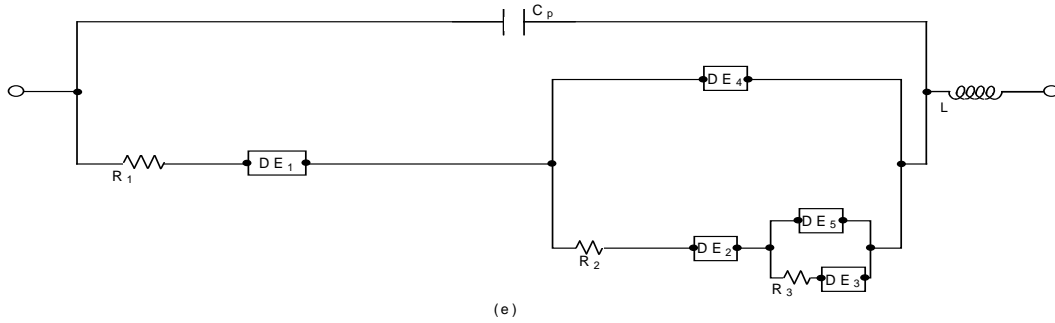
The following built-in EDAE choices are available: (a) fix $U2$ at >800 , then $U2=U1$ (with $U1$ taken free) in the integration; (b) fix $PHI2$ at >800 , then $PHI2=-PHI1$; and (c), fix $PHI1$ at >800 , then $PHI1=PHI2$. The choices a,b and a,c give the EDAE2 (symmetric) and EDAE1 (asymmetric) models, respectively. The b choice alone leads to the EDAE3 (asymmetric when $U1$ not equal to $U2$).

There is one more choice, particularly appropriate for the EDAE1 model, which is invoked by entering TDAE (=TAU) as a negative number (NFREE = 2 or 0: free or fixed). It will actually be used as a positive number, but this choice multiplies $TAU*OMEGA$ by $\exp(-U1)$. Then the actual fitting value of TAU will usually be closer to unity and may be less correlated with other free parameters, allowing it to be better determined. When only a five- parameter EDAE is needed in a DE circuit position, use the DE with NELEM = 12 or 13 (see Section 4).

Alternatively, when DCH is not zero, $P(4) - P(9)$ are used for the six parameters of the general Gaussian distribution of activation energies, GDAE. Then these parameters are defined as RGDAE, TGDAE, XLGDAE, XUGDAE, XIGDAE, and THGDAE. The lower integration limit is $-XLGDAE$ and the upper one is $XUGDAE$. If $XLGDAE$ is fixed at >800 , then $XLGDAE=XUGDAE$ (taken free to vary), yielding the symmetric GDAE2 model when $THGDAE$ is fixed at zero. For 5-parameter GDAE models, use the DE with NELEM=12 (Sect. 4). For more information on these models, see J.R. Macdonald, J. Appl. Phys. **61**(1987)700. When $P(4) = 0$, no DAE is present.

An appreciably more general exponential DAE, which includes both the EDAE and the GDAE, the GEDAE, is included in the O Circuit.

Circuit E



Parameter Assignments

P(1) = RDE1
 P(2) = TDE1
 P(3) = UDE1
 P(4) = PDE1
 P(5) = NDE1

P(11) = RDE3
 P(12) = TDE3
 P(13) = UDE3
 P(14) = PDE3
 P(15) = NDE3

P(21) = RDE5
 P(22) = TDE5
 P(23) = UDE5
 P(24) = PDE5
 P(25) = NDE5

P(6) = RDE2
 P(7) = TDE2
 P(8) = UDE2
 P(9) = PDE2
 P(10) = NDE2

P(16) = RDE4
 P(17) = TDE4
 P(18) = UDE4
 P(19) = PDE4
 P(20) = NDE4

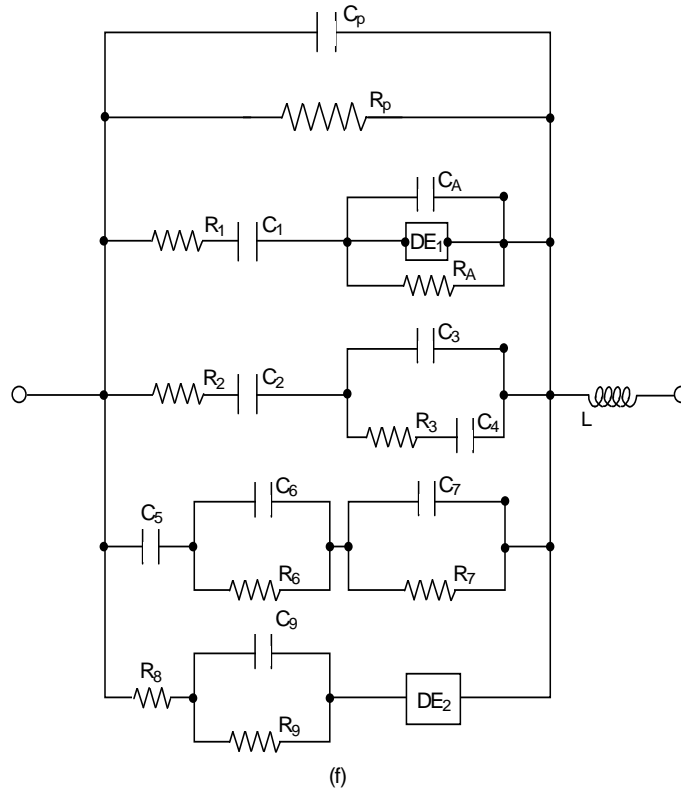
P(26) = R1
 P(27) = R2
 P(28) = R3
 P(29) = CP
 P(30) = L

POSSIBLY FREE, WEIGHTING PARAMETERS:

U = P(31)

Xi = P(32)

Circuit F



Parameter Assignments

$R_1 = P(1)$	$RDE2 = P(11)$	$R_6 = P(21)$
$C_1 = P(2)$	$TDE2 = P(12)$	$C_6 = P(22)$
$R_2 = P(3)$	$UDE2 = P(13)$	$R_7 = P(23)$
$C_2 = P(4)$	$PDE2 = P(14)$	$C_7 = P(24)$
$CA = P(5)$	$NDE2 = P(15)$	$R_8 = P(25)$
$RDE1 = P(6)$	$RA = P(16)$	$R_9 = P(26)$
$TDE1 = P(7)$	$R_3 = P(17)$	$C_9 = P(27)$
$UDE1 = P(8)$	$C_3 = P(18)$	$RP = P(28)$
$PDE1 = P(9)$	$C_4 = P(19)$	$CP = P(29)$
$NDE1 = P(10)$	$C_5 = P(20)$	$L = P(30)$

POSSIBLY FREE, WEIGHTING PARAMETERS:

$U = P(31)$	$\xi = P(32)$
-------------	---------------

Circuit G

This 'circuit' is different from the others. It has been reserved for ordinary nonlinear least squares fitting, with X as the frequency data and Y as the dependent data. Thus X need not be a frequency but could be, for example, a temperature variable used in generalized Arrhenius fitting (see, however, the more general possibilities available in the NELEM=33 DE choice). In order to accommodate many possible fitting functions, the one provided at present involves up to 30 possible parameters. In the following, P1 means P(1), etc. By setting any of P1,P6,P9,P13,P14,P17, P19, and/or P22 fixed to zero, various simpler fitting functions are produced. Use input choices (line 2) of ZZRR only. If DATYYP=R or I, the fit uses as the Y data that in the real or imaginary input data columns, respectively.

The present full function is (using $x^y \equiv x^y$):

$$Y = \text{(If P5 is non zero):} \quad P1*(X**P2)*EXP[(P3/P4)*(P4-X)/(X-P5)]$$

$$\text{(or if P5 is zero):} \quad P1*EXP(-X/P4)$$

$$+ (P6/P7)*(P7-X)/(X-P8)$$

$$+ P9*(X**P10)$$

$$+ P13*(P11 + P12*X)$$

$$+ \text{(when P30 = 0) } P14/[1.DO+P15*(X**P16)]$$

$$+ P17*EXP(-X/P18)$$

$$+ P19*EXP(-(X/P20)**P21)$$

$$+ P22*(X**(P23 - 1))*P23*(P24**(-P23))*EXP(-(X/P24)**P23)$$

$$+ \text{GDAE transient response (parameters P25-P30) when P15=0 and P14 and P30 are non zero.}$$

If $P30 < 0$, P4, P18, and P20 are replaced by their inverses. Thus, e.g., $P1*Exp(-x/P4)$ becomes $P1*Exp(-P4*x)$.

As an example of the use of this equation, suppose one wanted to find the least squares values of the parameters A,B, and C in $TAU = A*(T*T)*EXP[B/(T-C)]$. One would set $P1=A$, $T=X$, $P2=2$, $P3=B$, $P4=10**30$ (fixed), and $P5=C$. It would usually be appropriate to use proportional weighting. This sort of nonlinear fitting can reduce the bias which would be introduced if one fitted the logarithm of the equation with unity weighting (the usual approach).

Note that the present function can involve up to three exponentials in series. Also the P19 term represents Williams-Watts transient response at the polarization level, and the P22 one represents such response at the current level. The parameter assignments for the GDAE transient response are as follows:

$Xi = P(25)$	$XLG = P(28)$
$GAMMA = P(26)$	$XUG = P(29)$
$DELTA = P(27)$	$TAU = P(30)$

P(14) is the scale factor; XLG AND XUG are the lower and upper limits of integration; DELTA is related to the standard deviation of the GDAE; and Tau is a time constant of the response. See J. Appl. Phys. **61**(1987)700.

****New additions:**

IF P(15) is not zero and P(30) = 0

$$\begin{aligned} \text{IF P(40) = 0} \\ Y = P(14)/[1 + (X/P(15))^{**P(16)}] \end{aligned}$$

IF P(40) > 0

$$Y = P(14)/[1 + (X/P(15))^{**P(16)}]$$

IF P(40) = -1

$$Y = P(14)/[1 + (X/P(15))^{**P(16)}]^{**P(17)}$$

IF P(40) = -2

$$Y = P(14)/[1 + (X/P(18)) + (X/P(15))^{**P(16)}]^{**P(17)}$$

ENDIF

Polynomial and other fit functions

IF P(40) = -4 **See fitting example, **GPOLY3**, in FITTESTS, Circuit G.

$$Y = P(41) + X*[P(42) + X*\{P(43) + X*P(44)\}]$$

IF P(40) = -8

$$\begin{aligned} XI = 1/X \\ Y = P(41) + XI*[P(42) + XI*\{P(43) + XI*P(44)\}] \end{aligned}$$

IF P(40) = -16

$$Y = P(41)*\text{SIN}\{P(42)\}*X^{**P(43)} + Y$$

IF P(40) = -32

$$Y = P(41)*\text{ARCTAN}\{P(42)\}*X^{**P(43)} + Y$$

IF P(40) = -64

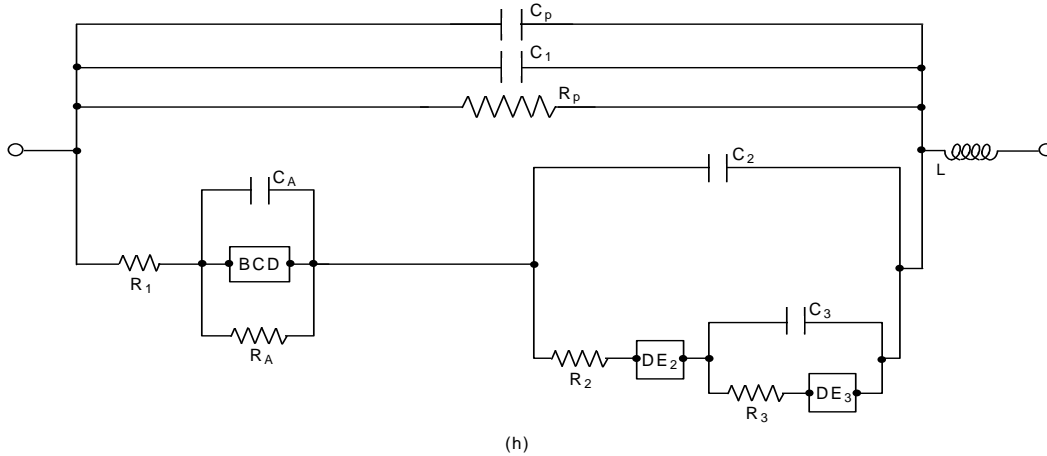
$$Y = P(41)*\text{TANH}\{P(42)\}*X^{**P(43)} + Y$$

IF P(40) = -128

$$Y = P(46)*[X^{**P(47)}]*\text{EXP}\[-\{(P(48)*X)^{**P(49)}\}]$$

ENDIF

Circuit H



Parameter Assignments

P(1) = RDE2	P(11) = *	P(21) = R2
P(2) = TDE2	P(12) = M,* null, or L	P(22) = C2
P(3) = UDE2	P(13) = PIZ (PI SUB Z),*	P(23) = R3
P(4) = PDE2	P(14) = PIM,*	P(24) = C3
P(5) = NDE2	P(15) = *	P(25) = ICH,*
P(6) = RDE3	P(16) = Xi (see below),*	P(26) = *
P(7) = TDE3	P(17) = RA*	P(27) = *
P(8) = UDE3	P(18) = CA*	P(28) = RP
P(9) = PDE3	P(19) = R1,*	P(29) = CP,*
P(10) = NDE3	P(20) = C1,*	P(30) = *

* SEE TABLE 1, p. 5-19

POSSIBLY FREE, WEIGHTING PARAMETERS:

U = P(31)

Xi = P(32)

For BCD (**PNP/PNPA**) elements, models, and responses, see the following discussion. For interface PNP response only, set P30 negative and MAVFEV=0:see its discussion below and get full PNP data values in LINOTRL.DAT (input) and interface ones also in (output) for plotting.

****FURTHER**, when the value of P25 is fixed at 5 and P40 fixed at 4, the DE2 element is moved so that it is then in parallel with the BCD one (a situation similar to that of the right-hand part of Circuit O). The DE2 and its parameters may then be used with dielectric models such as the dielectric Davidson-Cole one (DCD), e.g. by setting P10=6 and P8=1. See JRM paper #256 and later ones. This allows fitting with both conductive-system and dielectric system models in parallel. ********It is highly recommended that the P25=5 model choice be used in preference to other PNPA ones, since it yields mobility estimates for both positive and negative charges and also allows separate reaction and adsorption parameters to be estimated for the individual electrodes.

The above H circuit is somewhat like the D circuit, but the DAE element there is here replaced by a **BCD** (Blocking-Conductive) distributed element involving PNP or PNPA continuum-diffusion response models. The BCD element can yield several different PNP impedance responses, depending on the value of the parameter ICH (P(25)). These possibilities are summarized by the table at the end of the present description of Circuit H. It shows and names the special parameters involved in each choice. Currently, functional P(25) choices are 0, 1, 1', and 5. Others in the table will be activated later. See especially JRM papers: M252 through M256.

First, if ICH=0, BCD is null and P(11)-P(16) should all be set to zero. Second, when ICH=1 and P(11)>0, the first parameterization of exact completely-blocking response is invoked (see below). When ICH=1 and P(11)<0, the 1' second type of parameterization of such response is selected. Finally, when ICH=5, **partially-blocking/specific adsorption** response is modeled by BCD. Here, reactions and/or specific adsorption may occur at the identical electrodes for either or both the positive and negative charge carriers.

See J. Chem. Phys. **68**(1978)1614 [M124] (Sections B and F) and Electrochim. Acta **37**(1992)1007 [M194] for a discussion of completely or partially-blocking response and fitting with the present approach. See also J. Electrochem. Soc. **132**(1985)2939, and J. Electrochem. Soc. **135**(1988)2274 [M183] for descriptions of the physical system described and the appropriate equations involved here. In brief, the BCD DCE involves the exact small-signal solution for a completely or partly blocking system with charges of both signs having arbitrary mobilities and valence numbers and arbitrary (intrinsic) generation-recombination (GR) conditions. It also includes the possibility of anomalous diffusion (PNPA) when $\nu < 1$. See the M252 and M254 references above for more details of the definitions of the physical parameters involved in the BCD models. A long MS dealing with PNP and PNPA response and fitting, and including the possibility of GR effects, is JRM paper M252, while M254 involves partial blocking..

Note that when BCH=ICH=P(25)>0, both P(19) and P(20) (usually R_1 and C_1) must be nonzero (free or fixed) (CP may be used or not, but never allow both C_1 and CP to be free at the same time). For the first blocking parameterization (see the table), it is necessary that P(11) = N_0 be > 0. Note also that the Xi [ξ : parameter P(16)] of this choice is an entirely different quantity from the Xi [P(32)] used in weighting. The dielectric relaxation time, $TAUD \equiv \tau_D$, is $R_1 * C_1$ and the scale of this DCE is set by the magnitude of R_1 . The quantities R_1 and C_1 are usually named R_∞ and C_g in the above publications, and $R_\infty \equiv 1/G_\infty$, where G_∞ defines the high-frequency plateau of the real part of the admittance and C_g is the geometrical pure-dielectric capacitance. For specific data, these parameters would be named ρ_∞ and $\epsilon_V \epsilon_\infty$ or ϵ_∞ .

If full dissociation is wanted, for ICH=1 one may set both Λ_N and Xi to large fixed values such as 10^{20} . For uni-univalent situations, PIZ must be fixed at unity and should also be held fixed at unity for partial dissociation conditions. $\Pi_m \equiv PIM$ is the mobility ratio (that of negatively charged entities divided by that of positively charged ones), and M is the number of Debye lengths in a half-cell. PIZ should always be fixed, even for full dissociation conditions; if it is unknown, try various plausible fixed values to find the best fit.

The second blocking parameterization, invoked for ICH=1' and the input value of P11= N_0 set negative (although used in positive form), involves additional GR microscopic parameters. Here

N_0 is the original concentration (in cm^{-3}) of intrinsic entities that may dissociate and yield a concentration of $n_0 \leq N_0$ mobile negative charges and p_0 positive ones, mobile in the **two-mobile** case where $1 \leq \Pi_m < 10^{40}$, and immobile for $\Pi_m \geq 10^{40}$ (**one-mobile** situation). The GR quantity KDL is the log of $\text{KGR} = k_g / k_r$ (in dimensionless form) where these quantities are generation and recombination parameters. A summary of some of the relevant parameters is included below. Some of the different parameter assignments are as follows; ones not associated with an input parameter are calculated in the program as appropriate. See Table 1 for other choices not listed below

P(11) = N_0 , concentration (in cm^{-3}) of intrinsic dissociable entities (may be inputted as negative)

P(12) = M or LL=L, electrode separation length in cm. See the table

P(13) = $\Pi_z \equiv z_n / z_p$, valence number ratio.

P(14) = Π_m , mobility ratio (non-zero)

P(15) = k_g , generation rate constant (for ICH=2); Λ_N for ICH=1; $\text{KGR} \equiv k_g$ for ICH=1', 5

P(16) = k_r , recombination rate constant (for ICH=2); $\text{XI} \equiv \xi$ for ICH=1, 1', or 5

P(17) = $\mu_n \equiv \text{mu}_n$, mobility of negative charges (for ICH=2); or ICH=1': R_A , or CJ ρ_2

P(18) = $\mu_p \equiv \text{mu}_p$, mobility of positive charges (for ICH=2); or C_A , or $\rho_{2\infty}$ for ICH=5

P(19) = LL, electrode separation length in cm ((for ICH=2,4); μ_n for ICH=1', or R_1

P(20) = eps, bulk dielectric constant (for ICH=2); or C_1

P(26), P(27): ICH=5: ρ_{10} or ρ_{20} , respectively; Chang-Jaffe dimensionless reaction rates

P(30) Fractional exponent for PNPA. Set it fixed at 1 for PNP model.

The absolute temperature, T, must be entered (under ATEMP) in line 3 of the LEVM input file

The following specifications apply for ICH=4 (not currently activated). The positive bulk equilibrium concentration is not an input but is calculated from $p_0 = (z_n/z_p) * n_0$. If k_g is taken as $< 10^{36}$ (free or fixed), dissociation is taken incomplete and z_n is set equal to z_p (common value z_e). At least one of the two z's must be taken fixed. On the other hand, if k_g is taken as $> 10^{36}$ (fixed), one has a complete dissociation situation and n_0 , z_n , and z_p may be taken free or fixed as appropriate. Then k_r should be fixed at unity since it will play no role in the fit.

For the ICH=5 partially-blocking situation, the following parameters are used in addition to those appearing in the first blocking parameterization: Xi1a and Xi2a, and RHO1inf and RHO2inf (all associated with adsorption - set to zero if no adsorption) and RHO10 and RHO20, the normalized reaction rate constants for positive or negative charges (used both with and without adsorption).

*** For ICH=1, 1' or 5, a **valuable summary** of all relevant quantities is available **after** a PNP or PNPA fit is complete. Then set MAXFEV = 0 in the input LEVM file and run it. Invoke "Launch LEVMVIEW" and select the output file AUXPNTL for reading or printing. It first contains the values of the input parameters, followed by 12 or 13 lines of parameter names and values, and then by the data. I find it useful for situations with a lot of data to read this file on WORD and then select only the part down to the first few data lines for printing. The quantities not explicitly defined above and in the papers dealing with PNP response are identified below. There are 13 lines for the ICH=1' one-mobile, fully blocking case with $\Pi_m \geq 10^{40}$. See below.

In the usual GR case where k_g is set to unity, one finds that $\xi / k_{gr} = 8.8585 \times 10^{-12}$. For ICH=1, but not for 1' or 5, some of the summarized parameter values below are meaningless. For composite fitting models, carry out the summary for only the PNP/PNPA part of the full model to get its relevant values. Thus, set only the PNP/PNPA parameters to be present in the model.

SUMMARY QUANTITIES in actual calculations and the AUXPNTL output file (see above)

EEPS	EKG	TAUD	R1	TAUD is the Debye relaxation time, $R1 \cdot C1$
ε_∞	k_g	τ_D		
BLAM	EKR	TAUGR	C1	TAUGR is the GR time constant
Λ_N	k_r			
DM	ADL2	ELL	ANO	ADL is the Debye length; $L = ELL$ is the elect.separation
1 or 2		L	N_0	DM=2 for $\Pi_m \leq 10^{38}$; it is 1 otherwise, not appropriate.
ALAM	PIM	AK2	ENO	AK2 is reaction rate: $2\rho_2 D_{II} / L$ (p. 1630, M124)
Λ	Π_m		n_0	
EMUN	EMUP	DN	DP	n and p mobilities and diffusion coefficients (cgs units)
AK1	CELCAP	XI	EDS	CELCAP $\equiv \varepsilon_v (AREA / L)$, ξ , EDS: ε for DebD series model using P(23) and P(24) of H circuit
REZ	REE	CM2	MRAT	REZ and REE are the values of R' and ε' at the lowest frequency of the data. $CM2 = r \equiv M_2 \text{ctnh}(M_2)$ and $MRAT = (REE/EEPS)$
PIM	ADL1	RCM1	TEMP	ADL1 and RCM1 are appropriate values for M1

PNP/PNPA P(11) < 0 CHOICES: (GR=generation/recombination; BC=boundary conditions; CJ=Chang-Jaffe: see M124) See next page:

ICH=1': $\Pi_m < 10^{40}$ 1,2-mobile, GR, GBC; or $\Pi_m \geq 10^{40}$ 1-mobile, GR, CJBC

ICH=5: $\Pi_m < 10^{40}$ 1,2-mobile, GR, general CJ reaction/specific adsorption BCs

Table 1 - Different parameterizations for the Circuit H BCD model selected by values of P(25).

P(25) ICH	P(11)	P(12)	P(13)	P(14)	P(15)	P(16)	P(17)	P(18)	P(19)	P(20)	P(26)	P(27)	P(29)	P(30)
0	0	0	0	0	0	0	R _A	C _A	R ₁	C ₁	0	0	C _p	--
1	N ₀	M	Π _Z	Π _m	Λ _N	ξ	R _A	C _A	R ₁	C ₁	ε _∞	LL	C _p	ψ
1'	- N ₀	LL	Π _Z	Π _m	KGR	ξ	**R _A /ρ ₂	C _A	μ _n	C ₁	---	---	C _p	ψ
2	N ₀	---	z _n	z _p	k _g	k _r	μ _n	μ _p	LL	ε _∞	0	0	C _p	ψ
3	n ₀	M	Π _Z	Π _m	Λ	ξ	R _A	C _A	R ₁	C ₁	ε _∞	LL	C _p	N ₀
4	--	n ₀	z _n	z _p	k _g	k _r	μ _n	μ _p	LL	ε _∞	0	0	C _p	N ₀
5	- N ₀	LL	Π _Z *	Π _m	KGR	ξ	ρ _{1∞}	ρ _{2∞}	R ₁	C ₁	ρ ₁₀	ρ ₂₀	ξ _{2a}	ψ

* If P(13) < 0, then ξ_{1a} = -P(13) and Π_Z=1, else if P(13) > 0, P(13) = Π_Z. ** For 1', GPNP for Π_m < 10⁴⁰ if R_A > 0, or CJPNP for Π_m ≥ 10⁴⁰ with ρ₂ > 0

See FITTESTS: CKT.H for some PNP/PNPA typical LEVM input example files illustrating many response possibilities

Circuit I

This circuit is used to fit (real) transient data with a double exponential distribution of activation energies (EDA_E). The function fitted involves the incomplete gamma function. Details of the full response function appear in J.R. Macdonald, J. Appl. Phys. **61**(1963)538, **34**(1987)700, and **62**(1987)R53-R55. Note that G-circuit choices allow NLSQ fitting of Williams-Watts stretched exponential transient response (at either the polarization (charge) or current level). Here the parameter assignments are (some alternative definitions are given in parenthesis - see the above references. Further, G= the parameter gamma, and SE denotes a script E):

P(1) = response scale factor	$\tau = \tau_a \exp(GxSE)$
P(2) = τ_1 (τ_c)	$\tau_1 = \tau_a \exp(GxSE_1)$
P(3) = ϕ_1 ($-\rho_1$)	G = alpha + beta
P(4) = ϕ_2 ($-\rho_2$)	$\tau_{inf} = \tau_a \exp(GxSE_{inf})$
P(5) = τ_1/τ_{inf}	$\tau_o = \tau_a \exp(GxSE_o)$
P(6) = τ_1/τ_o	

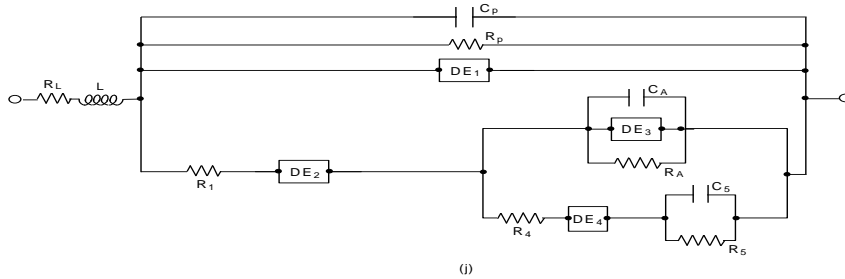
POSSIBLY FREE, WEIGHTING PARAMETERS:

$$U = P(31)$$

$$Xi = P(32)$$

Other parameters are not used (set them to zero). The transient response here is calculated using the exact theoretical formula; in contrast, that for the GEDA_E, which includes the EDA_E, is calculated by numerical integration of the applicable DRT.

Circuit J



Parameter Assignments

- | | | |
|--------------|--------------|--------------|
| R1 = P(1) | RDE2 = P(11) | RDE4 = P(21) |
| RA = P(2) | TDE2 = P(12) | TDE4 = P(22) |
| CA = P(3) | UDE2 = P(13) | UDE4 = P(23) |
| R4 = P(4) | PDE2 = P(14) | PDE4 = P(24) |
| RL = P(5) | NDE2 = P(15) | NDE4 = P(25) |
| RDE1 = P(6) | RDE3 = P(16) | R5 = P(26) |
| TDE1 = P(7) | TDE3 = P(17) | C5 = P(27) |
| UDE1 = P(8) | UDE3 = P(18) | RP = P(28) |
| PDE1 = P(9) | PDE3 = P(19) | CP = P(29) |
| NDE1 = P(10) | NDE3 = P(20) | L = P(30) |

POSSIBLY FREE, WEIGHTING PARAMETERS:

- | | |
|-----------|------------|
| U = P(31) | Xi = P(32) |
|-----------|------------|

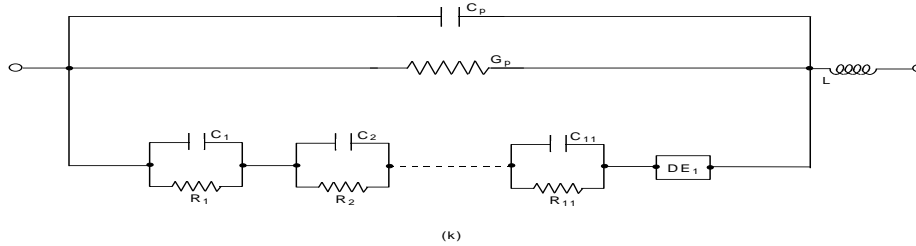
Although it is unlikely that one would often need to use a fitting model with as many as 11 free parameters unless a DRT were being estimated (Circuit K or R), I have included an example of such a fit in the JTST file. It involves data and fit results taken from the EQUIVCRT CNLS program of B. Boukamp. Note that this J fitting circuit involves a DE specialized to a resistor and inductance in parallel (NELEM = 17) and one specialized to a pure capacitance (NELEM = 2 with PHI fixed at unity).

The parameter values in the JTST input file are not very close to their final values and LEVM requires about 30 iterations to reach convergence. Run it and see. Note that parameter P(4) is poorly determined by the JTST fit. A subsequent run without it would be statistically as significant.

A test of what happens with a bad input choice may be made by changing the value of P(1) in JTST from 393 to 3930, fixing P(4), and setting P(34) to -1. On running this input, one will get a singular matrix error, even though the final P(1) estimate will be near its proper value. With P(34) negative, gradient values appear during iteration, and they may be compared to the values obtained with a more appropriate choice for P(1). A rerun with the original JTST values except using the new estimate of P(1) should converge rapidly.

You may have noticed that JTST uses modulus weighting (IRCH = 3), as also employed by Boukamp. These data involve an inductive (or negative capacitance-resistance) loop, and in the low-frequency region $-\text{Im}(Z)$ becomes very small. During fitting, corresponding model values may even pass through zero and go negative. Under these conditions pure PWT (or FPWT) (IRCH = 2 or -2) may be inappropriate since it makes the very small imaginary values possibly play too important a role in the fitting - their weights are too large. To avoid this difficulty in situations of this kind, one can either use modulus weighting or, probably preferably, use IRCH = 2 or -2 with a non-zero U input. It is appropriate to use a fixed value of U = P(31) set as small as possible to still allow convergence. For the JTST data, an initial value of 1 or less might be reasonable. This ensures that when the imaginary-part proportional weighting uncertainty becomes very small, it is dominated by an independent additive-error component, just as one might expect to be the case in many actual measurement situations.

Circuit K



Parameter Assignments

R1 = P(1)	R6 = P(11)	R11 = P(21)
C1 = P(2)	C6 = P(12)	C11 = P(22)
R2 = P(3)	R7 = P(13)	RDE1 = P(23)
C2 = P(4)	C7 = P(14)	TDE1 = P(24)
R3 = P(5)	R8 = P(15)	UDE1 = P(25)
C3 = P(6)	C8 = P(16)	PDE = P(26)
R4 = P(7)	R9 = P(17)	NDE1 = P(27)
C4 = P(8)	C9 = P(18)	GP = P(28)
R5 = P(9)	R10 = P(19)	CP = P(29)
C5 = P(10)	C10 = P(20)	L = P(30)

POSSIBLY FREE, WEIGHTING PARAMETERS:

U = P(31)	Xi = P(32)
-----------	------------

- P(36) = 0: CP as shown above
- P(36) = < 0: CP in parallel with DRT elements only
- P(36) = >0: CP in parallel with DE1 only

See discussion of several other fixed parameters such as P(38) in the R-Circuit section below. Note that the dotted line connecting the elements in the bottom line of the circuit above is scarcely visible but prints OK. Also, the R circuit is often preferable to the present K circuit.

When $P(30) < 0$, $|P(30)|$ here becomes a series resistance, not an inductance. Before using this circuit, read the discussions of the P(34), P(35), P(38), and P(40) choices in the R-Circuit discussion. See below for a redefinition of the R and C parameters of a pair (R1,C1, R2,C2, etc.) depending on the sign of P(40). For the K-Circuit, the $|P(40)| = 5$ choice, transient response inversion, is not available. Furthermore, for both the K and R circuits, the choice $P(40) = -4$ works, but is physically unreasonable. Note that the K Circuit requires that N (line-3 of input file) be equal to 40 or more.

This circuit, and that of Circuit R, are optimized for fitting data with what Orazem and co-workers have termed "measurement models" (see Refs. 5 (pp. 96,181), 18, 39, and 49 in Section 1). Incidentally, although the name "measurement models," not a very appropriate choice and is the only thing intrinsically new about their work, this designation will sometimes be used herein. They suggested fitting of data with a distribution of relaxation times (the Voigt model involved in the present K-circuit, a part of LEVM V.6 of January 1993 and before) and then fitting with a specific circuit, such as possible versions of Circuits A-D. Discussion of some erroneous claims about the provenance of the "measurement model" approach appear in the item, "The Orazem Controversy" included in the section *Miscellaneous Writings* (of JRM) in his home page, <http://jrossmacdonald.com>.

Through the use of Circuits K or R, one can also check the validity of the Kronig-Kramers relations without the need for extrapolation and without the need for a specific fitting model choice (see Refs. 5 (p. 181), 31-36, 38, 39). This approach has been fairly widely used. The K and R circuits may be used either with all τ_i variables free or fixed, as in JRM papers (see p.1-25) #'s 205 of 1994 and 206 of 1995. See especially Ref. 38 of Bernard Boukamp for the fixed situation. The R-circuit has also been used to solve an important response problem when the KK relations are unhelpful (JRM #229).

Even more valuable for some applications is the use of these circuits to determine estimates of a **discrete distribution of relaxation times (DRT), (or activation energies)** which is associated with the data (see Refs. 31-36 and 40 and the discussion of P(40) choices in the R-circuit discussion) or which approximates a **continuous** distribution. The K Circuit is not restricted to the discrete function approach (DFA) of estimating relaxation times; see the R Circuit for more discussion of the continuous-function approach (CFA) and for the estimation of more relaxation times than is possible with the K Circuit.

Several test examples which illustrate various features of DFA and CFA fitting appear in the KTST.XYZ files. Note particularly that KTST.M2S and KTST.4S obtain dielectric and conductive-system DRT estimates, respectively, for the same experimental data set. These calculations obtain six or seven relaxation times, but it should be noted that 11 is the maximum number possible with the K-Circuit. LEVM itself is limited to a maximum of 42 free parameters, and the use of the R-Circuit instead of the K one allows one to extract up to 19 relaxation times from accurate data. This is usually more than enough to well define the actual distribution of relaxation times for experimental data, but this capability is also valuable for exact simulation studies. It is desirable that the number of frequencies in the data appreciably exceed the number of free parameters in the fit. I have used LEVM.EXE running under MS-DOS with up to 19 relaxation times (38 free DRT parameters) so far without reaching any limitations, and this, or even a larger number, can also be estimated if one revises and recompiles the R-Circuit.

For ordinary experimental data, it will usually be sufficient to extract no more than seven to eleven relaxation times. One should stop trying to obtain more when an added pair is not well defined by the data. To avoid including effects in the calculated distribution of relaxation times of bulk capacitance and DC conductance, fits of actual data should allow CP and GP to be free when

appropriate. Further, DE1 can be used, if necessary, to account for some electrode effects.

When $P(40)=0$ or is ≥ 0 , the K circuit is as above and one obtains a Voigt-circuit representation of the distribution of conductive-system relaxation times: do fit at Z or M level. On the other hand, when $P(40)$ is < 0 , the circuit is modified to Maxwell type: **then** R1,C1 through R11,C11 are **re-arranged** so that each RC of a pair is in series and the effects of each such pair are in parallel (See the R-circuit choices). This part of the circuit is then the dual of that shown in the diagram and fitting should be carried out at the Y or epsilon level. This choice has been included in order to allow one to **calculate a discrete or continuous distribution of relaxation times defined at the dielectric, rather than the conductive system, level.**

There are times when an isolated, discrete relaxation time turns out to be present but is not thought to be part of the DRT of interest. For $P(21) < 0$, $|P(21)|$ and $P(22)$ are defined, respectively, as a capacitance and a resistance in series, all in parallel with the rest of the circuit. They thus yield a single relaxation time of dielectric type, and this response may be used separately from that leading to the DRT part of the response for $P(40)$ either less or greater than zero. Although $P(21)$ must be entered as a negative parameter, its absolute value is actually used. See one of the KTST examples.

The following is a short description of what goes on in determining a DRT by the present approach. See Ref. 36 for much more details for the dielectric-system situation. Let U represent either an impedance or a complex dielectric constant. Then for $|P(40)| = 0$ or 1 and each omega value, it can be written in terms of the normalized relaxation strength, g_j , and the actual input-pair parameters, s_j and t_j , as

$$U = U_{inf} + \Delta U \sum [g_j / (1 + i\omega s_j t_j)],$$

where $\Delta U * g_j = s_j$; the sum is over the relaxation times: $\tau_j = s_j t_j$, and $j_{max} = 11$. Here $\Delta U = U_0 - U_{inf}$, and the sum of the g_j 's is unity. For a conductive DRT, $s_j = R_j$, $t_j = C_j$, and, in this case, $U_{inf} = R_{inf}$. This quantity can be included in the fitting by identifying it with $|P(30)|$ when $P(30)$ is taken negative so it is a resistance instead of an inductance, or, more simply, it can be identified with DE_1 when this is set to be a resistance. Finally, C_P will not always be negligible for actual data and must then also be included in the fitting circuit. See test files for examples.

For a dielectric system, the s_j 's may be (real) dielectric-constant elements (see below), and the t_j 's are τ_j / s_j . In addition, at the Y level, one may write $Y = G_0 + i\omega C_v U$, where C_v is CELCAP and $C_v U_{inf} = C_g = C_P$. Here G_0 is the low-frequency-limiting (DC) value of the real part of the conductance, here represented by G_P . Note that the sign of the input quantity ATEMP may be used to determine whether the s_j 's are dielectric constants or capacitances (see more details on the next page), but one must be careful to use the proper choice of the input quantity CELCAP.

When $|P(40)| = 2$, the above equation is replaced by:

$$U = U_{inf} + \Delta U \sum [g_j / (1 + i\omega \tau_j)],$$

and when $P(40) = 4$ (CSD1 fitting), by

$$U = U_{inf} + \Delta U \sum [\tau_j g_j / (1 + i\omega \tau_j)] / \sum \tau_j g_j$$

In both cases, the tau product has been replaced by τ_j itself, which is then the second one of the input parameters of each DRT pair. This allows the estimated standard deviations of the τ_j 's to be directly obtained. The $P(40) = 4$ choice should only be used to estimate g_j values which are

appropriate for a conductive system (See Refs. 25, 35, 40, 43). Only for a dielectric system are the g_j 's of the first and second equations above, the appropriate DRT normalized strengths.

When the K circuit is used to determine a set of distribution strengths and relaxation times, the output file **OUTPARM** (and corresponding screen output) is produced if P(40) is not zero. This file contains columns containing discrete values of: IORIG - the original number of the first of each parameter pair for easy identification with PNTOUTL output; the corresponding τ =tau in ascending order; its relative standard deviation (RSD) (for $|P(38)| > 1$) or zero otherwise; g - the normalized relaxation strength; the RSD of g; and x_s and x_t . These last two quantities are discussed below. In addition, additional output, as discussed later, appears in OUTPARM and on the screen.

Let us summarize the use of the K-Circuit for various forms of input data. First, use the appropriate value of CELCAP if the data are of impedance or M, E, or Y form. But if the data are available in the form of complex resistivity or conductivity, use $CELCAP = \epsilon_v$, the permittivity of vacuum, 8.8542×10^{-14} F/cm (Gaussian units). If the data are associated with a conducting system, one should use P(40) = 0 for measurement-model fitting or P(40) = 1 or 2 to obtain one form of the DRT parameters. But the most appropriate CFA form is obtained if P(40) = 4 (See the discussion of CSD0 and CSD1 response in the Preface). For P(40) = 0 or 1, the input relaxation-time parameters of a pair are defined as R and C; then x_s and x_t are the fit estimates of these R and C inputs. For P(40) = 2, 3, or 4, the two input parameters of a pair are R, or R/Tau for the 4 choice, and Tau; and x_s and x_t are the corresponding R and C, where $RC = \text{Tau}$.

The situation is somewhat more complicated for dielectric-system DRT estimation because whether ATEMP is > 0 or < 0 affects the analysis. Consider first the choice ATEMP > 0 . Then an input pair is defined as C and R and x_s and x_t are also C and R for P(40) = -1. But for P(40) = -2, the inputs are C and Tau, and again $x_s = C$ and $x_t = R$.

When ATEMP < 0 , all C parameters (including C_p) are defined instead as dielectric constants: $\epsilon = C/CELCAP$. Then for P(40) = -1, an input pair is defined as ϵ and R, and x_s and x_t are ϵ and $\epsilon_v \rho$, where ρ is a resistivity and $\epsilon \epsilon_v \rho = \text{Tau}$. But for P(40) = -2, the inputs are ϵ and Tau and x_s and x_t are the same as above. Thus we see that in all cases, x_s times x_t equals Tau. All of the preceding discussion deals with discrete distributions. For continuous ones, the sums which approximate discussion needed integrals are weighted (see Ref. 36) and so differ slightly from the ones given above.

One to three lines of output appear at the bottom of the OUTPARM file, and the screen output, when P(40) is non-zero. The estimates in these lines become more exact the larger the value of N. The first line, which is always present, contains the following three or four estimates: the DRT sum Σg_j , which should be unity for discrete distributions but not for continuous or mixed ones; $\Delta U = U_0 - U_{inf}$ the low frequency limiting value of the real part of either Z or rho (conductive system) or C or epsilon (diel. system); the high frequency limiting value of G (or sigma) for a dielectric system DRT or of C or epsilon (conductive system) associated only with a discrete distribution (equal to $A=1/[\Sigma(x_t)^{-1}]$). For $|P(40)| = 4$, the quantity actually used for normalization also appears at the end of the line.

The remaining two lines only appear for continuous distributions with the fixed fit parameters $|P(38)| > 1$ and $|P(40)|$ not zero. The first line shows estimates of the moments $\langle \tau^m \rangle$, for $m = -1, 1,$ and 2 . The final output quantity on this line is only of interest when $|P(40)| = 5$, transient response (R-circuit only). It is the estimated value of the transient response at a time set by the input value of $P(36)$. For other values of $|P(40)|$, it is set to unity. The final line presents estimates of the normalized moments $\langle x^m \rangle$, for $m = -1, 1,$ and 2 . Here, $x \equiv \tau/\tau_0$, where τ_0 is a characteristic relaxation time of the distribution, always positive. When it is at least approximately known, its value should be input as the value of the fixed input parameter ATEMP. When any other value of ATEMP is used, ignore the results in this last line. The last value shown on this line is that of $|ATEMP|$. When ATEMP is set to $-\tau_0$, τ_0 is actually used. A value of τ_0 may often be obtained by fitting the original data to a DRT response model, such as the KWW0 one selected by NELEM = 10 or 36. Alternatively, when the form of the actual distribution is known, the DRT estimates in the OUTDRPT file (see below) may be used as LEVM input data for fitting to the actual distribution, for example, by using the NELEM = 37 choice for the KWW0 model, and thus obtaining a direct estimate of τ_0 . See, for example, the CSD1 test files KTSTKWW1.11 and RTSTKWW1.19.

The present estimates of the CSD0 $m = -1$ moments (for $|P(40)| = 2$) should be very large when no cutoff is present, but those for $m = 1$ and 2 are found to be quite accurate for large N values. See Refs. 36, 37, 40, and 43 for further discussions of DRT moments and especially Ref. 47 for discussion of CSD0 and CSD1 DRT differences.

When $|P(40)| > 1$, the output files **OUTDRT** and **OUTDRPT** are produced. The columns of OUTDRT are: τ_j , $\log_{10}(\tau_j)$, g_j , $\log_{10}(|g_j|)$, and the RSD's of g_j and τ_j , respectively. The OUTDRPT file is in the format required for direct plotting using the CTD plot program. Its first two columns are τ_j and g_j , and CTD plotting should involve only the real-part choice. Finally, when $|P(40)| > 1$ and **P(31)** is taken fixed and < 0 , the output file **DRTLSD.DAT** is produced. This file allows one to plot estimated standard deviation rectangles around each g_j plotted point. When $P(31) = -1$, the rectangle points (tau, g-value) are untransformed; for -2 , \log_{10} values appear for both quantities; and for -3 , only a $\log_{10}(\text{tau})$ transformation is made. When the input $P(31)$ is < 0 , its actual value for subsequent calculations is set to zero, the most common choice.

Summary

As usual in LEVM, I provide more choices than most people need! I recommend starting with one of the following: (A) "Measurement Model" fitting - $P(40) = 0$ or -1 ; (B) Dielectric DRT fitting - $P(40) = -2$ or possibly -1 ; (C) Conductive-system DRT estimation - $P(40) = 2$ or 4 .

Finally, to avoid getting caught in a local minimum, start fitting of the present type using PWT with $P(1)$ and $P(2)$ free (and possibly C_P and/or G_P free). After convergence, add $P(3)$ and $P(4)$ free and then fit. Keep adding free parameters in pairs and fitting after each addition until FQF and/or SD (e.g., S_F , the standard deviation of the relative fit residuals) stop decreasing appreciably and the relative standard deviations of some of the pair parameters possibly become as large as 0.3 or more. **See the R-circuit discussion for another much simpler fitting procedure alternative.**

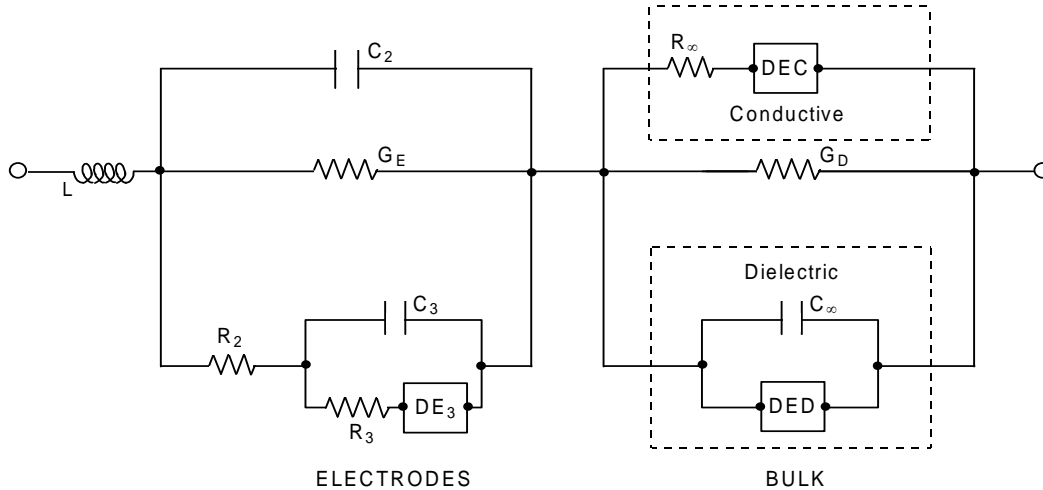
Also note that it is sometimes easier and quicker for continuous distributions to first find a discrete DRT using $P(38) = 0$, then change $P(38)$ to 2 for example, run the fit, and thus obtain a proper continuous distribution fit. When the underlying data involve a continuous distribution, note that any discrete DRT derived from it is only an approximation to the actual continuous one and is only of computational value. In actuality, a set of discrete DRT points derived from either discrete or continuous DRT data, should be considered only as made up of unconnected isolated points and it

then is not sensible to define its moments.

There may be times, particularly with CFA fitting ($|P(38)| > 0$ - see the R-Circuit), when a DRT fit ends with error messages. Do not be disturbed; press ENTER and continue; then run the file OUTIN produced by the first fit (use, e.g., IRE = -10 or -11 in the input). Continue the cycle until there is no change in the final output, most or all of which may still be significant (although g_j RSD deviation estimates which are zero or very large generally indicate that the associated relaxation times are meaningless).

In general, for any LEVM fitting, it is a good idea to ensure that OUTIN is always produced and that it then be run after apparent convergence to verify that the fit cannot be further improved. This procedure is particularly important to follow for measurement-model fitting and for DFA and CFA DRT estimation.

Circuit O



(o)

Parameter Assignments

RINF = P(1) (see p. 5-32*)	GE = P(11)	RDE3 = P(21)
CINF = P(2) (see p. 5-32*)	GD = P(12)	TDE3 = P(22)
GAMC1 = P(3)	GAMD1 = P(13)	UDE3 = P(23)
GAMC2 = P(4)	GAMD2 = P(14)	PDE3 = P(24)
RD,RDE = P(5)	CD,CDE = P(15)	NDE3 = P(25)
TAUC,TDEC = P(6)	TAUD,TDED = P(16)	R2 = P(26)
UC1,UDEC = P(7)	UD1,UDED = P(17)	C2 = P(27)
UC2, -- = P(8)	UD2, -- = P(18)	R3 = P(28)
PHIC1,PDEC = P(9)	PHID1,PDED = P(19)	C3 = P(29)
PHIC2,NDEC = P(10)	PHID2,NDED = P(20)	R, L, or C= P(30)*
NCH1 = P(35)	NCH2 = P(40)	

POSSIBLY FREE, WEIGHTING PARAMETERS:

U = P(31) Xi = P(32)

*P(37) > 0: P(30 = series R

*P(37) = -4: P(30 = series L

*P(37) = -16: P(30 = series C

Otherwise, no series element appears in the L position above.

This is a very important multi-purpose circuit particularly appropriate for fitting bulk dispersion of the conductive-system type (CSD) at the conductive level (symbol DEC in the figure), or bulk dispersion of the dielectric-system type (DSD) at the dielectric level (DED), or for simultaneous treatment of both effects when both are present in the data. In addition, it allows parameters P(11), and P(21) through P(30), to be used for electrode effects. Proper use of the O Circuit requires that N (line-3 of input file) be ≥ 40 . As well as being valuable for fitting of ordinary solid or liquid electrolyte data, the O Circuit is particularly useful for fitting dielectric and high-resistivity data associated with mobile ions, vacancies, and/or electrons and holes in disordered or single-crystal materials.

General, inputs and outputs

For either CSD or DSD situations, the O Circuit provides four distinct fitting choices. The first three are a new generalized exponential distribution of relaxation times model (GED), the Cole-Davidson (CD) model, and the Havriliak-Negami (HN) model. Their frequency or transient responses are calculated by direct integration over their distributions, and such integration involves a finite (i.e., cutoff) range, thus ensuring physical realizability when it would otherwise be lacking. The fourth model (DE) is a multi-model because it allows any one of the distributed circuit elements (DCE) of Section 4 to be used as a dispersive element in the O Circuit. Thus, for example, the CD and HN responses, with or without cutoffs, are available here. For economy, let us designate the first three dispersion models with cutoff collectively by DWC when they need not be separately distinguished.

Parameters P(3) through P(10) are used for CSD response (identification letter "C") and P(13) through P(20) for DSD response (identification letter "D"). When two parameter choices appear above together, such as UC1,UDEC, the first designates a DWC fitting parameter, and the second a parameter associated with any one of the distributed-element responses listed in Section 4. For example, RD is the DWC value of (R0 - RINF) and RDE is the DE value of the same quantity: the dispersion scale factor for conductive-system response. For dielectric dispersion, CD is the DWC value of (C0 - CINF) and CDE is the DE value of the same quantity. Similarly, TAUD and TDED are the dielectric-level-equation relaxation times for dielectric-level DWC and DE response, respectively.

Note that when MODE=8, the GED model fits transient response data rather than the usual frequency response data. For this choice of MODE, it is necessary to pick DATTYP=R, a real-part fit only.

Note that although P(30) may be a negative quantity in the input file, it is actually positive in the fitting circuit. Note that the parallel conductances (or conductivities for specific data), GE and GD, are treated as conductances, not resistances, in the fitting.

The two crucial O-circuit input choice parameters, NCH1 (P(35)) and NCH2 (P(40)), are used to define which aspects of the circuit are actually used. NCH1 is used to select CSD fitting possibilities for the DEC block of the circuit and NCH2 selects DSD models for the DED block. Let NCH represent either of these choice parameters. Then the available choices are:

NCH

0	---	No response: skip. P(1) and P(2) are still operative
1	GED	Generalized exponential response (GED): three types
2	CD	Cole-Davidson response (CD)
3	HN	Havriliak-Negami response (HN), (not including CD)
4	DE	Any relevant Section-4 distributed-element response
5,6	DE	Like NCH=4 choice but includes coupling-model Debye response at high frequencies; set P(41) to minimum tau. Then P(42) and P(43) are Debye strength and tau parameters. Generally, use cutoff model (see Ref. 46) instead of coupling model

It is very important that NCH1 and NCH2 values be chosen correctly and consistent with the non-zero input parameters. Otherwise, a run error will probably occur.

O-Circuit Admittance

The total O-circuit admittance may be expressed as follows. First, note that C_g is the present CINF; DEC is $(R_0 - RINF)I_C$; DED is $(C_0 - CINF)I_D$, and I_C and I_D are normalized, distributed circuit dispersion elements, of either GED or DE type. See Appendix 1 of the Preface. Define the admittance of the left part of the circuit as Y_{el} ; that of the CSD response as $Y_C = 1/Z_C = 1/[RINF + (R_0 - RINF)I_C]$ and that of the DSD response as $Y_D = i\omega[CINF + (C_0 - CINF)I_D]$, which is readily expressed at the dielectric level. Next, let $Y_S = Y_C + Y_D + G_D$, the admittance of the right part of the circuit. Finally,

$$Y(\omega) = Y_S(\omega) / [1 + \{Y_S(\omega) / Y_{el}(\omega)\}].$$

The quantities CINF, CD, and CDE (P(15)) listed above are capacitive dielectric-level input parameters (whose units are Farads) provided the quantity ATEMP in the input file is ≥ 0 and < 1000 . But if ATEMP is set to < 0 , these quantities are defined as dielectric constants (involving epsilon sub infinity and epsilon sub zero). Thus, one can use either type of scaling in dielectric level fitting. But since actual calculations require impedances, it is necessary to be able to convert dielectric-constant parameters to capacitances in the latter case. Therefore, if ATEMP < 0 , it is imperative that a proper value of CELCAP be provided in the input file. See the various OTST example test files. In general, a proper input value of CELCAP should be provided for any fitting of real data.

There are times when it is desirable to have a conductance GD in parallel with the CINF branch of the circuit; here it is represented by P(12) and might model a leakage conductance. It then can be subtracted, when desired, by setting MAXFEV =2, NFREE =3 for P(12), and running LEVM. Subtraction of such quantities as GD and CINF must be carried out when the LEVM immittance level is Y. Similarly, subtraction of such a quantity as RINF must be carried out at the Z level.

O-Circuit: two CSD response models in series (bulk and electrode)

See the discussion on p. 4-7. This is important when one wants to use both CSD1 and CSD0 response elements in a composite fitting model. When Mode < 0 , ordinarily both elements will be of CSD1 character. If the second element (here DE3) is a HN, it may be forced to be of CSD0 character by setting |TEMP| > 900 . For the KWW (NELEM = 10), the second element may be made to be KWW0 by setting P(23) < -900 , fixed. Alternatively, for any DCE invoked with NCH1 = 4, such as the KWW with cutoff (NELEM = 36), one needs only to set TEMP < -900 . Example files are OTSCOMP1, OTSCOMP2, and OTSCOMP3.

Further input-output information

As usual, when $MODE = 0$ or 1 , ordinary CSD (=CSD0) or DSD response is calculated. Alternatively, when $MODE < 0$, CSD = CSD1 response is calculated. The sign of $MODE$ has no effect on DSD-type GED response. When $|MODE| = 8$, transient, rather than frequency, response is calculated and is available for fitting transient data.

For CSD response: use P(5) through P(10); DSD response: P(15) through P(20). In addition, the γ_k quantities P(3) and P(4) (CSD), and P(13) and P(14) (DSD), are used for the GED and HN choices discussed below. The range of γ is 0 to ∞ but 0.5 to about 4 is that of main interest.

*Here inputs P(1) and P(2) are infinite-frequency R and Cg values. If the ATEMP input value is < 0 , then P2 and P15 are at the epsilon level; otherwise they are capacitances, as shown on the parameter-listing page.

The quantities RN and AIN, which appear in the final dielectric-related output when that is present, are distribution normalization-related values. They are equal when only CSD response is calculated, but when both CSD and DSD effects are present, AIN is that for the CSD response and RN is that for the DSD part, which is calculated last of the two. Zero values indicate a quantity not calculated.

RN and AIN are not actual normalization values but involve approximate or exact calculated normalization values. Actual, direct normalization values are calculated by quadrature from the distribution itself and then themselves normalized with the values calculated directly from normalization formulas. When the latter are exact, the values of RN and AIN are unity.

For CSD0 with $NCH = 1$ and $\gamma_1 = 1$, the calculated RN and AIN values take exact account of the finite size of U1, so actual values of RN and AIN are then essentially unity. But for γ_1 unequal to 1, these values may be greater or less than unity but approach it as $|U1|$ increases. Thus in this case, the difference from unity is a measure of the effect of a finite cutoff value, allowing one to see how large a value of $|U1|$ is necessary to yield a good approximation to no-cutoff response.

For HN/CD cases, the calculated normalization is exact only in the no-cutoff limit. This is often well approximated by a value of U1 of 30 or more. Thus for $NCH = 2$ or 3 , and U1 less than infinity, RN will exceed unity, and the degree to which no cutoff response is well approximated is measured by the excess over unity. A value of 1.01 therefore indicates a pretty good approximation. Generally for HN/CD, we expect U1 to fall in the range $5 < U < 25$, but a limited data range often precludes the possibility of finding its most appropriate value by taking it free to vary in the fitting.

For $NCH = 2$ or 3 , the sign of U1 does not affect the calculation result, but a negative value turns on extra screen output, showing the progression of the calculation by means of a data-point count. These calculations may become very slow for $U1 > 15$, for a large number of data points, and for $IGACC > 3$. Luckily, $IGACC = 2$ or 3 yields adequate accuracy for HN and CD fitting.

But slow approach to final convergence, even with $U1 < 15$ and $IGACC = 2$, suggests that it is usually most appropriate to first use $NCH = 4$ with $NELEM = 6$ or 7 , obtain a converged fit, and then use it as the input to a $NCH = 2$ or 3 run.

The generalized exponential distribution (GED)

The new GED dispersion response function (NCH = 1) is somewhat similar to fractional-exponential KWW response, but instead of involving a fractional exponential in the time domain, it here appears in the distribution of activation energies (or logarithmic relaxation times) domain. It is actually a modification and generalization of the Weibull statistical distribution function. This distribution is modified here by introducing cutoffs so that its domain is not $-\infty \leq y \leq \infty$ but $-U_1 \leq y \leq U_2$, where the U s are here taken positive. Here $y = \ln(x)$, where $x \equiv \tau/\tau_0$. The distribution must be cut off for some values of its parameters in order that it lead to physically realizable response.

Second, it is generalized both by allowing the ϕ parameters to fall in the range $-\infty \leq \phi \leq \infty$ rather than be only positive and also by allowing the possible use of different distribution parameters on the left, low-tau (subscript 1) and right (high-tau) (subscript 2) sides of the distribution. The slope-related ϕ parameters are the inverses of corresponding dispersion (width) parameters. The present distribution includes both an ordinary exponential ($\gamma = 1$) (see the NELEM = 12 and 13 choices of Section 4 and DAE choice of Circuit D) or a Gaussian distribution ($\gamma = 2$) (see the NELEM=14 choice of Section 4). For positive U values, we may write the GED distribution as

$$F(y) = \begin{cases} Nf_1(y) & -U_1 \leq y \leq 0 \\ Nf_2(y) & 0 \leq y \leq U_2 \end{cases}$$

where

$$f_k \equiv \exp[-\text{sign}(\phi_k) |\phi_k y|^{\gamma_k} / \gamma_k]$$

and

$$N^{-1} \equiv \sum_{k=1}^2 \Gamma(1/\gamma_k) \gamma_k^{(1-\gamma_k)/\gamma_k} \{(1 - e^{-\phi_k U_k}) / \phi_k\},$$

where Γ is the ordinary gamma function. For exact normalization, it should be replaced by an incomplete gamma function, but this is not done here for simplicity. For the DWC, values of P(10) and/or P(20) have no effect except in the GED case c (see below). They may have any valid values for other DE models.

- There are three different GED fit models set by the choices:

- $U_1 < 0, U_2 = 0$: use γ_1 only. Asymmetric. If $\gamma_1 = 1$: EDAE = EDAE1 case
- $U_1 > 0, U_2 = 0$: use γ_1 only. Symmetric. If $\gamma_1 = 1$: EDAE2 case, and if $\gamma_1 = 2$: Gaussian DRT case. Note: $0 < \gamma_1 < \infty$
- $U_1 < 0, U_2 > 0$: use γ_1 and γ_2 . General, with possibility of different U 's, ϕ 's and γ 's.

Note: ϕ_1, γ_1 and U_1 affect the high frequency region of the response, and ϕ_2, γ_2 , and U_2 the low-frequency region. Thus, $-\phi_1$ is the log-log slope of a hf region (especially when $|\phi_1|$ is 0.5 or less) of, say, $-\text{IM}(Z)$ vs. Ω , and ϕ_2 is the slope of a corresponding lf region. The calculation of frequency or transient response from a DRT, such as that above, is discussed in Ref. 40. The [MODE] = 0, 1, 2, and 8 choices are valid here. **But for a single ϕ , $\phi_0 = 1 + \phi_1$.

The CD and HN models, and CSD1 response possibilities

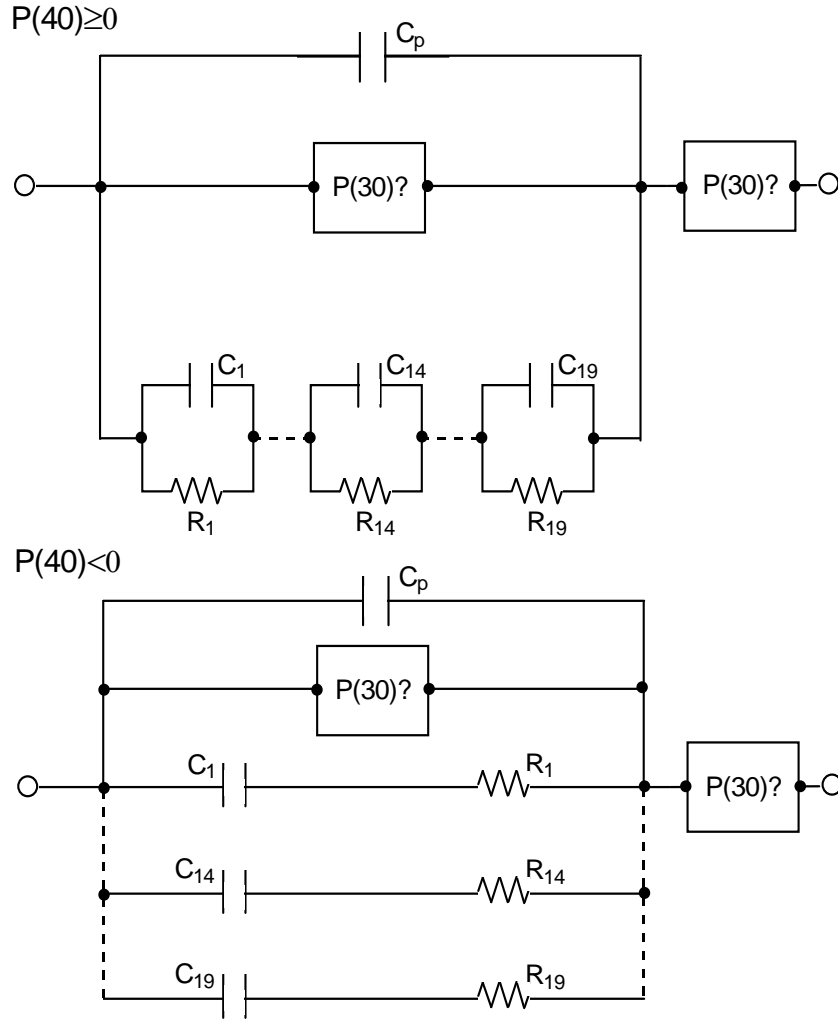
For the CD model, which provides DSD and CSD0 response (MODE = 0 or 1) and CSD1a response, one uses, for example, the parameters RD, TAUC, UC1 and PHIC1 with GAMC1 = 1, fixed. Because a simple expression for the first moment of the CD distribution is available, calculation of CSD1a response is straightforward. Here the choice MODE = -1 (or -2) yields such response using the CSD0 distribution and the subtraction method described in the Preface, Part H. In addition, for **MODE = -6**, such response is calculated directly from the related CSD1 distribution. The latter method yields more accuracy in the low-frequency region. These calculations are valid whether RCINF is zero or not.

The choices MODE = 0 or 1 yield DSD and CSD0 HN response. Because no simple expression for the first moment of the general HN distribution is available, MODE = -2 CSD1b response is also available, both here and for the NELEM = 6, 7 DE choices of Section 4. It will be useful to refer to the comparison of parameter assignments for HN fitting (not including the CD case) presented in the Section-4 NELEM = 7 discussion. As well as the parameters mentioned above for CD fitting (with RD here redefined as a dielectric constant), one must also specify a value of the exponent parameter $\gamma = \gamma_1$, here GAMC1 = P(3).

CSD1b HN response is calculated by the difference method when MODE = -1. But the actual RD resistance or resistivity can only be easily calculated from CSD1b fit results when RINF = 0. Even if the first moment were known, and it can be calculated from the known distribution, it would still be necessary, when RINF was not zero, to solve a quadratic equation involving R0 and (R0 - RINF) at each step in the fitting iteration. This is why the NELEM = 7 DE CSD1 calculation, which only involves the distribution for the final extra output, with U = 15 (in order to allow quick convergence), is restricted to CSD1b response and applies only for the RINF = 0 situation. Incidentally, such extra output (see Section 1, Part 4b) appears for any of the CSD NCH1 = 1, 2, or 3 choices.

Since in the present O-circuit, CSD1a response can be calculated directly using the CSD1 HN distribution (with MODE = -6), one avoids the complexity of the CSD1b approach, and HN fit results are valid whether RINF is zero or not. For the NCH1 = 2 and 3 choices, the necessity of integration over the distributions can sometimes make a fitting run extend for a long time. In extreme cases, it will be appropriate to allow the fit to run all night.

Circuit R



POSSIBLY FREE, WEIGHTING PARAMETERS:

$U = P(31)$

$\Xi = P(32)$

Parameter Assignments

P(40):	X	-X	Y	5	
P(1) =	R1	C1	D1	R1	P(29) = CP
P(2) =	C1	R1	τ_1	τ_1	
.				.	
.				.	
P(27) =	R14	C14	D14	R14	FOR P(37):
P(28) =	C14	R14	τ_{14}	τ_{14}	= 0: P(30) = parallel G
P(41) =	R15	C15	D15	R15	> 0: P(30) = series R
P(42) =	C15	R15	τ_{15}	τ_{15}	-4: P(30) = series L
.				.	-16: P(30) = series C
.				.	
P(49) =	R19	C19	D19	R19	
P(50) =	C19	R19	τ_{19}	τ_{19}	

POSSIBLY FREE, WEIGHTING PARAMETERS:

$$U = P(31)$$

$$X_i = P(32)$$

In the above, X is 0 (+X column only) or 1. The quantity Y may be + or - with values of 2, 3, or 4, but the choice -4 is inappropriate. The D's in the above are either R's ($P(40) > 0$) or C's ($P(40) < 0$). You should read over the discussion of the K Circuit before using this circuit since they are very similar and produce the same output files. See the K-circuit write-up for a discussion of these files and final screen output. In the figures above, P(30) is shown in two possible positions. Depending upon the fixed value of P(37) used, one or the other of these positions for P(30) is selected, as well as the character of P(30) itself. The quantity MODE is not used for K- or R-circuit calculations.

It is important to note that both the K circuit and the R circuit are appropriate for estimating either discrete or continuous distributions (or their combination), including Kronig-Kramers testing, and measurement-model analysis, while the R circuit includes, in addition, the possibility of carrying out inversion of transient data. For continuous distributions, two estimation approaches are available: that using CNLS and that using Phillips-Tikhonov regularization. The latter approach is described later in this section. The R-circuit requires that $40 \leq N \leq 50$, where N is the total number of parameters.

**Some LEVM/LEVMW input files for dielectric data are included in the R-circuit part of FITTESTS. See files beginning: RTSTDIEL.... Other files are for conductive-system data.

CNLS Estimation

General

It will be seen from the above circuit that up to 19 RC pairs may be used in measurement model fitting, conductive-system DRT estimation, or dielectric system DRT estimation, while only a maximum of 11 pairs is possible for the K circuit. Because the parameters from P(33) to P(40) are always fixed, the additional possible free parameters beyond P(27) and P(28) are here those from P(41) to P(50). To use this circuit with more than 14 pairs (or, equivalently, relaxation times) one must add needed lines to the parameter section of the input file and set N in line 3 of the input file equal to the maximum total number of parameters (<51). The executable LEVM file LEVM.EXE allows this.

Besides allowing more DRT parameters to be estimated than is possible with the K Circuit, the present circuit model also allows continuous function approximation (CFA) DRT estimation. This uses a fitting equation similar to the second or third one in the K-circuit discussion, but the τ_j 's need not be free parameters (See Ref. 36). They may be fixed at equal intervals on a logarithmic scale whose range is simply related to that of the data themselves. When they are fixed, their relative standard deviations are all set to zero in the output file OUTPARM. A value of P(39) may be used to either expand or contract the Tau range. CFA with the τ_j 's free is usually superior; see Part 3.

For many situations, it seems likely that DFA DRT estimation is likely to be superior to fixed-tau CFA fitting. Note especially the test file RTST.M8P, which attempts to fit a clearly discrete DRT with less-appropriate CFA estimation. See the discussion on apparent fitting errors at the end of the K-Circuit section before you run this. Several other RTST files have been included in the distribution disk to provide examples of R-Circuit fitting/inversion. In addition, all KTST files can also be used as tests of the R-Circuit procedure (but not necessarily *vice versa*). Note particularly the RTST files: RTSTT.9, RTSTWC0.9, RTSTWC1.9, and RTSTWD.19. These all involve exact KWW beta=0.5 data. The first demonstrates transient-data inversion, the next two CSD0 and CSD1 frequency-response inversion, and the last DSD inversion of the original CSD frequency-response data.

**When using a DRT obtained by inversion of frequency-response data and appearing in OUTIN, one can calculate the associated transient response by first using ONEMKE to prepare a file using the top part of OUTIN and desired time values in its bottom part, calculated, e.g., with MKFREQ. Then set P(40) = 5, MAXFEV = 0, IRE = -10, and MODE = -16 in OUTIN and then running the file. The result will involve the unnormalized transient response in the real-data column of the new OUTIN and in LINOTR, etc. To obtain the normalized transient response, $\phi(t)$, set MAXFEV = 0 and MODE = -16 again and fit once more. Finally, with MAXFEV = 0, iterate again to obtain $\phi(t)$ values in the data column and in the other output files.

The K-circuit and the R-circuit can both be used for fitting data to what Orazem and co-workers have termed "measurement models" (see Refs. 5, 18 and 39 in Section 1 and especially the discussion on p. 5-23 of this manual). One fits RC pairs to the data (e.g., R1,C1 ... up to R19,C19 in the R-circuit) and thus does not need to specify a more specific fitting circuit. This approach is thus useful for preliminary fitting and for testing whether the data are associated with a time-invariant system or not (a type of Kronig-Kramers test (see Refs. 35 and 38). These applications can be very useful, but these two circuits also provide another very important fitting capability: the determination of the distribution(s) of relaxation times (DRT), or activation energies (DAE) present in the measured material which led to its specific frequency response.

A main difference between the two circuits is that the R-circuit can be used for up to 19 pairs (or DRTs) and the K-circuit only allows a maximum of 11 pairs. Comparison of the two circuits shows that more other-circuit-elements with which to fit electrode-response behavior are available in the K-circuit than in the R one.

The 11 or 19 pairs can actually be of two basic types. For conductive-system response, the RC pairs consist of an R and a C in parallel, each in series with the next such pair (as shown in the figure above). Fitting is then most appropriately carried out at the impedance or complex-resistivity level.

But, if the data are thought to be associated with a primarily dielectric system, then the pairs may be taken as made up of a C and R (=1/G) in series, in parallel with the other such pairs (dielectric DRT fitting at the admittance or complex-conductivity level). See Refs. 34 and 40. The appropriate connections are shown in the bottom R-circuit diagram. If the data are expressed as complex-dielectric-constant values and fitting is carried out at the complex dielectric-constant level, then each series C and R element of a pair of the figure may be interpreted, respectively, as the dimensionless real or imaginary part of a complex dielectric constant associated with basic single-Debye dielectric response. See Ref. 36. Furthermore, depending on the input data, an R-parameter may be a resistivity one, rho, etc.

LEVM allows some further generality in fitting of the above types. Instead of determining the most appropriate R and C values of all pairs by CNLS fitting, one can instead determine the R (or C) value of a pair and its associated elemental time constant) or relaxation time: tau = RC. Then tau is taken as a basic fitting parameter. It is important to note that when all such taus are taken fixed, rather than free to vary, the response function (fitting model for the pair parts of the circuit) is linear in its R or C free variables, leading to very simple and quick fitting (see Refs. 35,38).

Transient response data fitting and inversion is invoked when P(40)=5. In addition, P(38) must be set to a desired value. Transient response requires the real-part fitting choice and involves R and tau values, as in the table above. Let TR(t) denote the unnormalized transient response model.

Then

$$TR(t) = \sum R_j \exp(-t / \tau_j),$$

where the sum is over the (R, tau) pairs. The final normalized DRT response replaces R_j in the above by g_j = R_j / ∑ R_j, ensuring that the normalized value of TR is unity at t = 0.

Because of the close similarity between the K and R circuits, the details of the fitting possibilities and input parameter choices needed to select a particular type of fitting will be described in connection with the R-circuit but they apply to the K one as well. In order to provide some general background, the rest of the present section involves a discussion of the equations involved in RC and DRT fitting.

The most pedestrian and safest way to determine best-fit pair parameters is take $IRCH = 2$ and to start with guesses for the values of the two parameters of a single pair [e.g., $P(1)$ and $P(2)$], take both parameters free, carry out a CNLS fit which leads to an OUTIN resultant file, enter guesses for $P(3)$ and $P(4)$ in OUTIN, let them be free, and fit again using OUTIN. This process is continued until a sufficient, or limiting number, number of fitted pair values has been found. After fitting with three or four pairs, the standard deviation of the fit, SF, should begin to decrease with each additional pair added by a factor of between 1.5 and 2. When this factor approaches unity reasonably closely, which will eventually happen because of the presence of noise in the data, it is clear that additional pairs are unable to improve the fit significantly and then the process should be terminated. When a synthetic, noise-free data set is used, however, the process only terminates because of the limited number of pairs available in the present circuits.

Linear Estimation/Fitting

The present version of LEVM incorporates several much simpler, and usually quite satisfactory, ways to obtain various kinds of multiple-pair least-squares fits without the laborious, sequential fitting procedure described above. One starts with an ordinary linear least squares fit, obtained by picking a set of taus, and holding them fixed in the fitting. No initial guesses for the R (or C) parameters are needed for such linear fitting, but one either needs to specify the number of pairs desired or use an input file which already has non-zero values (of any kind) present for the pair parameters which are to be estimated.

In the linear-fit procedure, the taus may be automatically selected in either of two ways. The simplest sets $\tau_{min} = 1/\omega_{max}$ and $\tau_{max} = 1/\omega_{min}$, and spaces the taus between these values evenly on a logarithmic scale. Alternatively, a smallest value of tau may be selected, along with the number of desired pairs. Again the taus will be evenly spaced on a logarithmic scale. A choice is available to allow the R parameters to be either positive or negative, as required by the fit, or constrained to be positive. The latter choice does not yield a linear fit (and so requires many iterations before full convergence) if the actual linear fit is one which requires one or more R's to be negative. Thus, it is wise to begin the fit with no sign constraints.

A linear fit of the above kind will usually be sufficient if one is only interested in testing whether the data are associated with a time-invariant system or not, but the quality of the fit is constrained by the fixed values of the taus. When they are set free to vary, as the next step in the overall fitting process, an appreciably better fit is generally obtained, and the parameter estimates (e.g., R and tau or R and C) are less uncertain and then represent the result of a nonlinear least squares fit. This step is necessary if one wants to obtain meaningful estimates of the parameters of a distribution of relaxation times or activation energies. Note that when R and tau values are determined from a fit, LEVM allows one to automatically change the results to the corresponding R and C parameters - which will appear in a new OUTIN file. Details of the various possible fitting procedures are presented below.

Further Parameter Choices

Particular fitting possibilities are illustrated by the various K- and R-circuit test files included with LEVM. The input file parameters P(40), P(34), and P(38) are of especial importance in defining the kind of fit desired, and fixed parameter choices for them and others are discussed in detail below. Asterisks are used to identify the most usual choices.

Functions controlled by P(40): + or - values

When $P(40) \geq 0$, the DRT fitting circuit is that shown for the K-circuit or is the top structure shown for the R-circuit. These circuits are appropriate for conductive-system analysis, either at the impedance or complex modulus levels (see Refs. 34,40). Alternatively, when $P(40) < 0$, one deals with a dielectric system DRT and the fitting structure is that of the bottom part of the R-circuit figure (see Ref. 36). Estimation can be carried out either at the dielectric level or the admittance or complex conductivity level.

Values of |P(40)|:

- 0: The DRT fitting parameters are R and C.
- 1: The parameters are R and C, or C and R in the dielectric system case when $P(40) = -1$. A DRT output file is created for all $|P(40)| > 0$.
- 2*: The parameters are R and tau (or C and tau). Tau sorting occurs at each iterative step. The DSD or CSD0 g_j DRT values define not the normalized G_D or G_0 DRTs but these quantities multiplied by $x \equiv \tau/\tau_0$, the F quantity in Refs. 40 and 47.
- 3: Like 2 choice, but no tau sorting. Appropriate when taus are fixed in ascending or descending order of magnitude.
- 4: Use for CSD1 situations. The choice -4 is nugatory and should not be used. Fitting uses $R*\tau$ in place of R. Therefore, the odd DRT fitting parameters are no longer R's but are essentially R/τ ; the g_j 's then define the G_1 DRT (see p. 0-19 and Refs. 40, 43, and 47). Iterative tau sorting enabled.
- 5: Analysis of transient response data. Independent variable is time, not frequency. Transient response data in column 2; do real, not complex, fit, with $|P(38)| \geq 1$. The last output value on the screen and in OUTPARM will be the estimated response at a time set by the P(36) input value. The $|P(40)| = 5$ choice is not operative for the K circuit.

Functions of P(38):

- 0: Discrete-distribution fitting. DV or DF fitting
- 1: Use with fixed-tau, continuous-distribution fitting. CF fitting. This involves fixed weighting.
- 2*: Use with variable or fixed tau, continuous (CV) or continuous and discrete distribution fitting (CDV). Uses variable quadrature weighting.
- 3: Like 2, but uses a different numerical integration routine (Cubint without error-estimate correction)
- 4: Like #2, but uses a different numerical integration routine (Cubint with error-estimate correction)

**For CF fitting, with or without regularization, preferably use $P(38) = 2, 3, \text{ or } 4$ with $P(35)$ set and all taus fixed, rather than $P(38) = 1$ which involves less accurate integration in this situation. The choice $P(38) = 2$ ensures proper DRT normalization.

Functions of P(34):

- 1*: Gradient output on screen during iteration
- 0: No gradient output, but normal operation otherwise
- >=1: All the P(34) choices with P(34)>0 set up or transform different input-file header situations but do not immediately do a fit. After invoking a "fit" with P(34)>0, one obtains (provided IRE <0) a new OUTIN file and can then either change any of the input parameters in it before fitting (e.g., check values of P(38) and P(40)) or immediately do a fit run using RNLO. If convergence is not rapid (as it should be for a linear situation), it may be that a linear fit requires some of the parameters to be negative. Use the P(34) = 2 choice to produce the appropriate input. When P(34) = 1, 2 or 5, P(40) is automatically set to 3 in the OUTIN file, except if the initial P(40)=4, then it remains at 4, the CSD1 DRT situation.
- 1*: Use P(34)=1 for setting up linear-least-squares fitting. Starting from an input file containing data and number-of-data-points information, this choice creates a new OUTIN input file with P(38) = 1 (i.e, fixed tau values; see above discussion of simple starting possibilities). The other DRT parameters are constrained to be positive and are initially set to unity. If P(36) = 0, the taus are set by the range of the omega data. Otherwise, P(36) is set to the smallest desired tau value and P(39) to the largest value. Set P(35) equal to the desired number of DRT pairs if necessary. When the file is ready, enter: RNL filename to create but not immediately run the file (in case you wish to make further changes in it).
- 2*: Like 1, but DRT strength parameters are not constrained to be positive only.
- 3: Uses values of R (or C) and tau present in an input file and sets all DRT parameters, including the taus, free to vary but constrained to be positive. Use this choice to convert a converged linear least squares fit to a nonlinear one.
- 4: Like 3, but DRT strength parameters may be positive or negative.
- 5: Use to convert nonlinear fit results to a linear-fit OUTIN file, as in choice 1, but does not change the initial values of the strength parameters.
- 6: Converts an input file which contains R (or C) and tau DRT input parameters to one containing the associated R and C (or C and R) parameters. P(40) is set to Sign(P(40)) (+1 or - 1). If the initial file involves P(40)=4, it will be reset as above, but the value of 4 is necessary in the input file to allow identification of the type of odd parameters (CSD1, or CSD0 or DSD type).
- 7: Converts an input file which contains R (or C) and C (or R) DRT input parameters to one containing the associated R and tau (or C and tau) parameters. P(40) is set to 2*Sign(P(40)) (+2 or - 2).

P(35) values:

P(35) can be used to set the desired number of DRT parameter pairs. Its use is unnecessary in sequential DRT fitting starting from one or two initial pairs and increasing by one pair for each subsequent fit. It is needed, however, in fits where one or both values of a pair are held constant, and its value is automatically set by some of the P(34) choices.

Regularization Estimation

This DRT-estimation approach uses a Phillips-Tikhonov regularization parameter, lambda, here P(33). For regularization, one must use a set of fixed taus, with equal "tau" or log(tau) intervals (e.g., use P(34)= 1). See the preceding section, Functions of P(38), for discussion of the value of P(38) to use. The allowed values of P(39) are 0, 1, 2 or 3. They set the order of the regularization. For example, these values, respectively, apply a type of regularization smoothing which makes the strength parameters tend toward zero, a constant, linear behavior, or quadratic behavior.

A common choice for regularization order is 1, but values of 2 or 3, not usually available as they are here, are often more appropriate. The most appropriate value of lambda is not obvious: if it is too small, regularization produces no effect, while if it is too big the detailed structure of the response can be smoothed out. Try increasing the value of lambda from a very small value until the standard deviation of the fit, SF, begins to increase significantly. A somewhat larger value of lambda may then provide the most appropriate tradeoff between best fitting of the original data and a useful amount of smoothing. A possible initial choice for lambda is the inverse of the maximum value of the odd fitting parameters. Regularization is of most use for noisy data and is not needed for exact simulation data.

An example of the effect of regularization is provided by the two test files, RTST303.NOR, which does not involve any regularization, and RTST303.REG, which does. The experimental data here is that for a material at T=303K with electrode and Cinf effects removed after fitting of the original data with a KWW CSD1 model (P(40)=4).

For the RTST303.REG fit, lambda= 10^{-11} and the order is taken as 2. If these two files are run, their DRT strength parameters may be most readily compared by using CTD to examine their corresponding OUTDRPT files. One would use CTD with the input file name OUTDRPT and invoke the real vs. freq. plot choice (actually thus plotting the DRT strength parameter vs log(tau)). When one makes such a comparison, one sees that the DRT curve is quite irregular for the best-fit RTST303.NOR fit, but much smoother and probably more appropriate for the RTST303.REG fit.

Circuits S and T

These circuit are not currently implemented. They exist only as dummies that the user can adapt for his/her own purposes and then recompile using the LEVM FORTRAN source-code (see, for example, the Microsoft FORTRAN Power Station V.1 compile and link files: CMLPV.BAT and LNK.BAT included in the LEVMFOR subdirectory). For LEVMW, I use the Compaq Visual Fortran Standard Edition V6.6 compiler.

CTD Program

This program, appropriate for **LEVM MS-DOS fits**, was written by Sudath Amarasinghe, and is copyrighted by him 1993-1996. CTD is a graphing utility program capable of generating 2D and 3D plots of immittance data. It can read the AUXPNTL file generated by the LEVM program as well as immittance data which have been stored in a file using the format described in the following (Main Menu/Read Impedance Data) section. The maximum number of data points supported by CTD is 1001. EGA or VGA graphics capability is required to run this program. Both EGAVGA.BGI and CTD.EXE files should be in the same directory to run CTD.

Some of the options are disabled when only Real or Imaginary data are used in the fitting instead of both Real and Imaginary data. If one tries to invoke these disabled options, a warning ("Insufficient Data") is displayed. Note that the word "impedance" as used herein, can often be taken to mean "immittance."

The program can be exited at any menu by pressing **Alt-x**. The **Esc** key can be used to exit the current menu and go back to the previous menu.

Main Menu

Read Impedance Data

This option will read the impedance data file in the following format.

Line #1 :	File description (up to 80 characters)		
Line #2 :	Data mode (Two characters) . Ex: ZR, ZD, YR		
Line #3 and onwards :	Frequency	Real	Imaginary
	or	Frequency	Magnitude Phase

See the IMP.DAT file as an example. If the data mode in line 2 is invalid or missing, ZR mode is assumed. **Frequency should be in Hz** (not in angular frequency or Log[frequency]).

Read AUXPNTL file

This choice reads the AUXPNTL file or any other file with the same data format as in AUXPNTL. When this option is chosen, the program asks the file name. Hit the <return> key to read the AUXPNTL file. To read a different file in AUXPNTL data format, type the name of the file and hit the <return> key.

Exit

Exit the program.

AUXPNTL MENU

Mode ZR (or ZD, YR etc.)

This option allows the user to convert data between 16 different modes. Refer to other sections of the LEVM manual for a complete description of the data modes.

Frequency Hertz (or Omega)

Change the units of frequency Hz to ω and *vice versa*.

Cell Cap. 1.00E+00

Allows the user to change the capacitance of the cell. This option is activated when the data are in (Z and Y) or (M and E) formats depending on the immittance form of the AUXPNTL data file.

This restriction has been imposed to prevent changing the Cell Cap. value during the conversions between (Z, Y) and (M, E) formats.

Do 2D Plots

Graph data in a 2D plot. User can choose different types of plots.

Do 3D Plots

Graph data in a 3D plot. In ZR mode $x \Rightarrow Z_{\text{real}}$; $y \Rightarrow -Z_{\text{imag}}$; $z \Rightarrow \text{Log}(\text{Freq.})$.

Overlay On/Off

When overlay is "ON" both experimental and estimated results are graphed on the same plot. When it is "OFF" only one of them is displayed.

Log Impedance Plot On/Off

When Log Impedance is "ON", Log values of absolute real and absolute imaginary data are used to plot the data. They can be converted back to original data by turning the LogPlot OFF. One should be careful in doing this because the information about the signs (+/-) of the data may be lost.

Log frequency On/Off

When Log frequency is "ON", Log values of the frequency are used instead of the input frequency values. This is the default. But there are some cases, where one wants to plot versus frequency rather than $\log(\text{frequency})$. This option makes this possible.

Save Format ASCII/C (or T or S)

Set the delimiter of the ASCII output file. C, T and S correspond to Comma, Tab and Space respectively. On hitting the <return> key, one can choose different options.

Save Data

Save data in the ASCII format selected from the *Save Format* option. The file-saving utility is included so that the data can be imported into programs like Lotus, Excel and QuattroPro. These data file formats cannot be read back by the CTD program.

Options in a 2D Data Display

Measured data are plotted in **green** color with *, and estimated data are plotted in **purple** using the + symbol.

Title

The program labels the X, Y and Z axes according to the data mode. The user can rename these by using this option.

Rescale

Rescale the graph.

Type

User can choose different symbols for the scatter plot or choose line plot.

Set Labels

Define the number of tic labels for each axis. This is useful when the aspect ratio of the plot is 1 (Sometimes one of the axes will be too short to print several tic labels, and as a result, they will write on top of each other).

Aspect

Two choices are available. *Full scale* uses the whole screen as the plotting area. *Unit Aspect Ratio* graphs data so that units/pixel is same for both X and Y axes. The latter is useful to see semicircle correctly in the impedance plane plot.

Hard Copy

Plot data on a HP Plotter or screen dump on a HP Laser Jet/Deskjet Printer. Screen Dumping uses 75 dots/inch resolution to decrease the printing time. It assumes that the printer is connected to LPT1. The graphs can be plotted on a HP Plotter connected to the COM1. Select the *HP Plotter* option if present, escape back to the preceding menu, and select *Redraw* option to get a hard copy.

Redraw

After changing any parameters select this option to redraw the graph.

Options in a 3D Data Display

Rotate

Rotate around X, Y and Z axes. The initial rotation is 45 degrees around X and Y axes.

Cont. Rot.

Continuous rotation of the 3D plot around X, Y or Z axis by a given number of degrees at a time. (This looks nice in a fast computer)

Project

Data projection on XY, XZ and YZ planes. Data can be projected on the 3D plot or as a 2D plot.

Descenders

Draw descenders from the 3D plot to each plane.

Type

Same as in 2D plots.

Titles

Same as in 2D plots.

Hardcopy

Same as in 2D plots.

Redraw

Same as in 2D plots.

The 2D and 3D **plot-select** menu allows the user to plot a great many different types of graphs. A new 2D option has been added to allow the plotting of the absolute value of the tangent of the data (only in rectangular form) (OTN) or its inverse (ITN). Thus, the OTN curve involves $|\text{imaginary part}/\text{real part}|$. Log(immittance) should be set to "off" when using this option. For the four immittance levels, there are only two basic responses because the identical OTN curves for E and M are the same as the ITN curves for Z and Y, and the ITN curves for E and M are the same as the OTN ones for Z and Y. Some uses of ITN at the epsilon level are discussed in the paper by Niklasson, J. Appl. Phys. 66 (1989) 4350.

If the Overlay is ON, Redraw will graph both experimental and fitted data. CTD can read the AUXPNTL file produced by LEVM when the number of iterations is forced to zero. In this case experimental and estimated are same and the relative error percentages are forced to zero. This is also valid in the case where the user chooses the "Read Impedance Data" option.

The following format has been used to label X and Y axes tic marks:

$m \Rightarrow 10^{**}(-3)$ (Example: $3.2m = 3.2 \times 10^{-3}$)
 $u \Rightarrow 10^{**}(-6)$ $n \Rightarrow 10^{**}(-9)$ $p \Rightarrow 10^{**}(-12)$ $f \Rightarrow 10^{**}(-15)$ $a \Rightarrow 10^{**}(-18)$
 $k \Rightarrow 10^{**}(+3)$ $M \Rightarrow 10^{**}(+6)$ $G \Rightarrow 10^{**}(+9)$ $T \Rightarrow 10^{**}(+12)$ $P \Rightarrow 10^{**}(+15)$
 $E \Rightarrow 10^{**}(+18)$

When the numbers are out of these ranges, they are labeled in exponential format.

INLEV program

The INLEV program can be used to prepare input data files for LEVM. Two types of data file formats can be read by INLEV. The first format is useful when preparing a new data file. The format is as follows.

Line #1 : Comment line (up to 80 characters)
 Line #2 : Number of data points.
 Line #3 and onwards : Frequency Real (or Magnitude) Imaginary (or Phase)

(See file INLEV.DAT as an example.)

The second format is as same as the input data file format for the LEVM program. This is useful when editing a data file which has been prepared before.

I hope the INLEV and CTD programs will be useful tools in your data fitting process. These two programs were written while I was in the University of Minnesota. If you have any questions about the programs, I can now be reached at the address below.

Dr Sudath Amarasinghe
 Ceramic Fuel Cells Ltd.
 Monash University Gippsland Campus
 Gate3, Switchback Road
 Churchill, Victoria 3842
 Australia

The MKLIO Input/Output Programs

The MKLIO **MS-DOS** programs were written by Evgenij Barsoukov <e-barsoukov2@ti.com> and are used in **LEVM** with his kind permission. The executable routines (MKLIO.EXE and EXTRLVM.EXE) may also be copied and used elsewhere as needed but are COPYRIGHT 1995 - 1998 by Evgenij Barsoukov. The uses of these routines are first summarized below and then their descriptions are provided along with typical examples. These programs should, most conveniently, be in the same subdirectory as that containing LEVM.EXE. If you start MKLIO or EXTRLVM without command line parameters, you will obtain a help window. Note that the present routines not only allow very simple construction of LEVM input files, but they also allow one to make runs automatically with many different data files and to produce a simulated set of LEVM input files.

Summary

MKLIO

This program combines either the top part of a LEVM input file (header) or a complete input file (whose data will be eliminated and replaced) with an arbitrary data file (in any format and even with reversed columns) to produce a new, complete input file, all with proper spacing and ready to be used. One does NOT need to specify on line 3 of the header the value of M, the number of data points present in the data file.

Not only can MKLIO be used to prepare a LEVM input file *ab initio* (see details below), but it has other uses as well. It is often necessary or desirable to start a new fit using the parameters from the previous one. The simple sum of the old header and a new data file does not solve the problem, because:

- the exact number of the data points must be changed in the header, and
- information about the data file name must be in the header to allow further automatic extraction of parameters from output files (see EXTRLVM).

MKLIO is a command-line-driven program which solves both of these problems. It is also suitable for using in batch files for processing multiple immittance data files. Further, it allows one to drop a number of points at the beginning and end of the data file, and it writes backup of the used header input configuration file to prevent it from being overwritten during LEVM fitting.

EXTRLVM

When one deals with a series of immittance spectra associated with variation of an external quantity, it is often important to see how some of the fit parameters change with the serial number of the immittance data file, where the number corresponds to potential, concentration, thickness, etc. values of the investigated material. LEVM handles each immittance output file individually, so such dependencies can only be summarized by hand when EXTRLVM is not used. However it is very useful to be able to see the parameter dependencies after each new fit of a series of fits, in order to see, for instance, if the parameter dependencies remain physically justifiable as the external variable is incremented.

EXTRLVM writes all needed parameters of a fit to a parameter file which consists of a row of data values preceded by a number corresponding to the number of the data file used in the fit. A single application of EXTRLVM (as in the example batch file: ONEFIT.BAT data-z1, see below) creates a

parameter-variation file involving just the first line of the three shown below. Subsequent runs of ONEFIT with the data files data-z2 and data-z3 then add the second and third lines to the par1.dat data file.

```
1 3.366487E+02 7.503447E+03 7.030335E-01 9.228649E-01 2.754648E-07
2 4.544281E+02 7.503458E+03 7.030349E-01 9.228649E-01 2.754645E-07
3 6.134140E+02 7.503453E+03 7.030340E-01 9.228650E-01 2.754646E-07
```

Here the first number of a row is the serial number of the fitted file, which will be read by the program from the first information line of OUTIN. The remaining columns list the parameter values which were selected to be extracted from OUTIN. For most situations, it will be much easier to use the batch file MULTFIT which automates the above procedure. Then all available input files are run sequentially with ONEFIT and produce the output file par1.dat. Such a run with the above data files produces the output above.

Preparing LEVM Input Files

A complete input file includes a header part and, below that, a data section. The header part requires that a number of choices be made, and estimated values for free and fixed parameters be present. As discussed elsewhere in the present manual, such preparation can be carried out with a word processor, with the program INLEV, or by other methods. Here it will be assumed that such a header (or a complete input file) is available, and that one wants to combine this with a (new) set of data. For choice 1 below, the data need not be formatted, but for choices 2 and 3, as presently implemented, the data should be in usual form with columns of frequency, real-part, and imaginary-part in that order. Particularly for simulation work, it is important to be able to combine the data in a full input file with another header (or that part of another full input file), producing a new input file which can then be edited if necessary.

1. Simple file preparation:

Use MKLIO with one or more %n choices, as discussed in Section C. The output file will be INFL.

2. More flexible file preparation:

```
ONEMKE %1 %2 %3
```

Here, the batch file ONEMKE invokes MKLIO. The input choice %1 is the initial header or full LEVM input file; %2 is the name of the data file; and %3 specifies the desired name of the new full output file.

3. Move data in a full input file to another full input file with a different header:

The data part of a full, formatted LEVM input file includes a first column which serially numbers the data values. This numbering needs to be removed for use with MKLIO. This process is carried out using the batch file CONLFL.BAT (construct a LEVM file), one which invokes MKLIO with the special input choice -filename. It first makes the new unnumbered data file, filename.dat, and then combines it with a specified header or full input file (%1), and finally puts the result in (%3). The needed command is:

```
CONLFL %1 %2 %3,
```

where %1 is the name of the header (or complete file) which is to be combined with the unnumbered data; %2 is the name of the full LEVM input file containing the data of interest; and %3 is the desired name of the full output file, the usual quantity of interest. But, when CONLFL is run, the resulting new data file, %2.dat, may be used as an input to DINPUT (using RNDIN) to change/transform the data if needed. Then the result can be used in Process 2 above to produce a new input file.

Details of the Programs

MKLIO

If the header, or full file used to provide a header, is named OUTIN, then the only **necessary** parameter in running MKLIO is the name of the data file to process, e.g., data-z1.dat. Then to make a new full input file for LEVM one would write the command: MKLIO data-z1.dat.

Other possible command-line parameters (order not significant): -r; -d; -g; -f; and -a. See also the important new commands listed at the end of this section.

If the source of fit parameters and first estimates is not named OUTIN, its name must be included on the command line. For example, if the source is named AZC40 and the data file is data-z1.dat, then one would write MKLIO data-z1.dat -razc40. No -r command is needed for the default name, OUTIN.

To drop N1 frequencies at the beginning of the data file and N2 at the end, use, e.g., MKLIO data-z1.dat -dN1dN2. The default numbers are d0d0.

If one has available (Grapher) data files in the format (Z',Z'',f) rather than (f,Z',-Z''), they may be used by adding -g to the command line (default: disabled). For example, use MKLIO gdata-z1.dat -g. The new input file will have the proper LEVM ordering.

The -f command-line input is used in conjunction with a full LEVM input file, as discussed in Section B.3 above.

Alternative base name for backup: <-aname>; default: input data file. If this command-line parameter is not used, the backup file for OUTIN will be named as follows:

If data-z9.dat is the data file, the name for the backup is data-i9.inp. If the command: -abase is given, for the input file data-z9, the backup file will be named basei9.inp. For example: MKLIO data-z9.dat -abase)

New Options

Option -w
All frequencies in a data file will be converted to angular when written to INFL.

Option -pNfilename
The N first initial parameters from the LEVM-format header file can be overwritten with parameters from a user-supplied file with the option: -pNfilename. Parameters for each spectrum in this file should be placed in one line, free numeric format, with spaces as separators.

The number of lines corresponds to the number of spectra; the first position in the line is used for the data file number.

An example of such file is attached (par_star.prn.prn):

```
2 5.848e-006 6.389e-016 0.00274 0.4628 0.00363 -12.17
5 5.848e-006 6.389e-016 0.00274 0.4628 0.00363 -12.17
10 5.848e-006 6.389e-016 0.00274 0.4628 0.00363 -12.17
^file number
```

The command MKLIO -p6par_star.prn data-z5.dat will first extract all fitting options and init. parameters from the default header file OUTIN, then replace the first 6 initial parameters with ones from par_star.prn. The line is chosen automatically so the first number 5 corresponds to the number at the end of data file name (data-z5). If file data-z2 will be used, the first line will be chosen.

This -pNname option is useful for creating initial parameters for LEVM in a different mathematics package. Also parameter files created by EXTRLVM during a previous fit can be used for a new fit or simulation.

Option -oN_N1filename
This is the same as -pN option, N1 will be used for line selection instead of the data file number. So if the command MKLIO data-z5 -o6_10par_star.prn is given, the first 6 parameters in INFL will be replaced with 6 parameters from the third line in pfilename (starting with 10).

Option -eNfilename

This allows one to replace one of initial parameters (the Nth) in a header file with one from a numbered parameter file. This file (example: numpar.prn) has a format like:

```
1 5.848e-006
2 3.181e-006
3 3.803e-006
4 3.761e-006
5 3.762e-006
6 3.762e-006
```

The command `MKLIO -e3numpar.prn data-z5.dat` will use the file `data z5.dat` and the default header file `OUTIN` to create an input file `infl`, where the third input parameter will be replaced by `3.762e-066`. `MKLIO` automatically chooses the right parameter from the numbered parameter file (in this case 5th because the file name ends with 5). This option is necessary if you want to analyze a series of spectra and one of the initial parameters is known from a separate experiment and changes with spectrum number. Specially, if this parameter is fixed in the fit, the present option is the only way to automatize the fitting procedure. Only `maxparam 40` is supported by this option.

EXTRLVM

Since a single invocation of `EXTRLVM` produces only a single output line, this program is most useful when one wishes to fit several, or many, data sets with the same fitting model. One then automatically runs `EXTRLVM` many times. Included are two typical `LEVM`-fit batch files: `ONEFIT.BAT` and `MULTIFIT.BAT` which can be altered to suit the needs of the user.

In addition to automatically backing up `OUTIN`, `EXTRLVM` also backs up the `LEVM` output files `LINOTR` and `PNTOUTL`. The backup of `LINOTR` is `basemN.prn` and that of `PNTOUTL` is `baseeN.fou`, where `base` is the name of the data file used (without the number). These backups prevent overwriting of the files during multiple file processing.

The following input commands are **necessary** in running `EXTRLVM`:

- Name of the parameter file, e.g., `set1_p.dat`
 - Serial numbers of `LEVM` parameters to be extracted from `OUTIN`: e.g., `-s1,2,3,4,7,12`.
- For example: `EXTRLVM set1_p.dat -s1,2,3,4,7,12`

Optional command parameters:

In the case where the parameter estimates to be extracted are at the beginning of the parameter line in `OUTIN`, command `-nN` can be used, where `N` is the serial number of the last parameter to be extracted. Thus, the command `-n4 -s7,12`, is analogous to `-s1,2,3,4,7,12`.

For example, use: `EXTRLVM set1_p.dat -n4`.

- Source for fit parameters and parameter estimates: `-filename` (default OUTIN). For example
: `EXTRLVM set1_p.dat -no -razc40`

- Alternative base name for backup; `-aname` (default: input data file). If this command line parameter is not used, the backup file for OUTIN will be named as follows:

If `data-z9.dat` is the data file, the name for the backup is `data-o9.inp`.

If the command `-abase` is given for the input file `data-z9`, the backup file will be named `baseo9.inp`. For example: `EXTRLVM set1_p.dat -s1,2,3,4,7,12, -abase`.

Set alternative maximum number of parameters in the input header file. Default value is 40.
`-pN`. For example: `EXTRLVM set1_p.dat -n4 -p50 -razc50`.

When the command `-g` is given, a file with statistics of each estimated parameter value will be generated (the parameter file should include more than 2 parameter lines). For example, `EXTRLVM set1_p.dat -n4 -g`). The resulting file includes 3 lines:

1. Mean values, \bar{x}
2. Average rms deviations, where the deviations are defined as $(x - \bar{x})$, d
3. Absolute average deviations, d/\bar{x} .

MKLIO and EXTRLVM Batch Files

The included batch files, ONEFIT.BAT, MULTFIT.BAT, and MSIM.BAT are particularly appropriate for multiple processing of a series of impedance spectra.

In order to become familiar with these batch files, it is desirable to run them using the included sample data files, `data-z(1-7)`. The data files and batch files need to be in the same directory as MKLIO, EXTRLVM, and LEVM.

EXAMPLE 1 - a single fit using MKLIO and EXTRLVM

The illustrative batch file ONEFIT.BAT is

```
MKLIO %1
LEVM
EXTRLVM par1.dat -s1,6,7,9,29
```

For example, the `%1` input above could be `data-z1.dat`. In general, you will need to alter ONEFIT.BAT so that it selects your own desired parameter choices.

EXAMPLE 2 - multiple fits using MKLIO and EXTRLVM

The illustrative batch file MULTFIT.BAT, which allows one to fit automatically the series of impedance spectra data-z(1 to 7).dat, is

```
for %%f in (1 2 3 4 5 6 7) do call ONEFIT2.BAT data-z%%f.dat.
```

Here, the included file ONEFIT2.BAT is

```
MKLIO -aser1_i %1
LEVM
EXTRLVM par1.dat -s1,6,7,9,29 -aser1_o
```

On running MULTFIT.BAT, the 7-row file par1.dat, with the 1th,6th,7th,9th and 29th parameter estimates obtained from each individual fit, will be automatically produced.

```
Input INFL files (only headers with parameters) will be stored as ser1_i(1-7).inp;
output files will be stored as ser1_o(1-7).inp;
The last 2 files are only produced, when IRE<9. (See the LEVM manual):
LINOTR files with simulated and original impedance values will be stored as
data-m(1-7).dat;
PNTOUTL files with information about the quality of the fit will be stored as data-e(1-7).dat;
```

EXAMPLE 3 - multiple simulation using MKLIO and EXTRLVM

If source files with fit parameters are already available (for example as backups from previous fitting), a series of impedance spectra can be simulated using the following batch files:

```
MSIM.BAT:
```

```
for %%f in (1 2 3 4 5 6 7) do call LSIM.BAT ser2_o%%f.inp %%f
```

where the file LSIM.BAT is:

```
MKLIO -aser3_i data-z1.dat -r%1
It is necessary that the parameter MAXFEV be set to 0 in %1 (the source files).
LEVM
copy linout data2-z%2.dat
```

Here, the name of the source file is supplied by MSIM.BAT to LSIM.BAT as the first command line argument, and the number of the file as the second. In the above, running MSIM.BAT will produce the files data2-z(1-7).dat.

Copyrights and History

This LEVM Manual and the program software are COPYRIGHTED by James Ross Macdonald, except as noted below. Permission is hereby granted that the program and manual or any parts thereof may be freely reproduced and used for individual scientific purposes but none of this material may be used by others as either a separate program or as part of such a program sold commercially, except by permission of the copyright owner. First, as noted therein, the manual material and programs of Sections 6 and 7 are individually copyrighted by their authors. The following algorithms of the LEVM program are COPYRIGHT 1986 by Cambridge University Press and are used herein by permission: GAMMLN, GCF, GSER, LUBKSB, LUDCMP, POLINT, QROMB, and TRAPZD. Finally, the subroutine LEV3 and some parts of the subroutine LEV2B are taken from the open literature and are thus not copyrighted. Note that **LEVM** is not an acronym but a name.

An excellent, but considerably less general CNLS program than LEVM is the EQUIVCRT program of Boukamp described in Ref. 11. In addition, the program ZView of Scribner Associates incorporates a considerable part of the much earlier LEVM V.6.0 program and operates in WINDOWS, but neither the manual nor the program includes all of V.6.0, reducing their utility considerably. Besides being based on an earlier, less powerful version of LEVM, one that was **improperly** used commercially without requesting permission, this program is far less general and more circumscribed than either LEVM.EXE or the present LEVMW program.

MEISP is a “Multiple EIS parameterization” software package, developed by Korea Kumho Petrochemical Co. research group led by Dr. E. Barsoukov in the early part of this century, that uses the LEVM engine for fitting but adds to it the ability to analyze multiple spectra together and to observe tendencies of parameter changes. It supports automatic pre-fit of initial guesses, which eliminates the need of manual guessing and allows forcing of time-constant order to different elements. Graphical presentation of the spectra, results and parameters is also included. In addition it includes a rich library of commonly used equivalent circuits and distributed elements (such as limited length diffusion), that can be included as part of the circuits. There is also a graphical circuit editor that allows one to create arbitrary circuits, as well as support of user-created models as DLLs. One may download a free, fully-functional trial version of MEISP from <http://impedance0.tripod.com/#3>.

The early history of the LEVM program begins with the paper by Macdonald and Garber, J. Electrochem. Soc. 124(1977)1022, the first description of a practical CNLS approach for IS data. Further work is described in Part B of the Reference list (end of Section 1). The first versions of LEVM were called LEVM-OLSON or LOMFP (1978-1988), but in 1988 the OLSON part of the fitting program was eliminated and a much superior nonlinear least-squares **iteration** procedure was incorporated (subroutine LV3 or LEV3, written by J. J. Moré). Some of the principal versions of LEVM have been: 2.21, 3/88; 3.01, 7/89; 3.05, 5/90; 4.0, 8/90; 5.0, 9/91; 6.0, 1/93; 6.1, 3/94, and 7.01, 2/97. There were no versions issued between 6.1 and 7.01. Version 7.1 was completed in the summer of 1998, and V.7.11, which incorporates some corrections and additions to V.7.1, was issued in October 1999. Version 8.0 was issued in 2003, and V.8.06, 8.07 in 2005. The present version 8.12, of 08/12, is only slightly expanded/corrected over the 2011 one.

In the years 1977-1986, the program was developed by my students, postdocs, and me. Since 1986 I have been entirely responsible for changes and improvements. Between 1975 and 1988,

work on the program was supported first by the NSF and then by the Army Research Office. I much appreciate this support and the excellent work of my associates on the various developmental stages of the program. In addition, I thank Sudath Amarasinghe and Evengij Barsoukov for allowing their valuable programs (Sections 6 and 7) to be included in LEVM. Most importantly, **Andriy Gorkovenko** of Kiev deserves great thanks for his expert work in making the **LEVMW** version of LEVM a reality.

Finally, if you have problems with using LEVM/LEVMW or if you find an error in the program, please get in touch with me and ask for help. My complete publication list appears in the website listed below, and all of the listed publications are now available for downloading in PDF form.

Dr. J. Ross Macdonald

**Home: 354 Carolina Meadows, Villa
Chapel Hill, NC 27517, USA
919/967-5005**

macd at email.unc.edu

WWW Website: <http://jrossmacdonald.com/>

**Office: Department of Physics and Astronomy
University of North Carolina
Chapel Hill, NC 27599-3255, USA
(use home address)**

Changes In LEVM

Summary of important changes/additions made to LEVM from V. 6.1 to V. 8.11

1. V. 6.1 to V. 7.01

First, many improvements have been made in the methods of analysis of bulk response, particularly important for high-resistivity disordered materials, glasses, and dielectrics. The distinction between conductive-system response (CSD), dielectric-system response (DSD), and a combination of the two is emphasized and incorporated in the CNLS analysis possibilities. In addition, two different types of CSD response are now distinguished, CSD0 and CSD1, and important new CSD1 fitting algorithms are provided which correct and replace earlier Moynihan modulus analysis methods.

In addition, for inversion of data to estimate distributions of activation energies (DAE) and relaxation times (DRT), linear inversion has been simplified, useful for Kronig-Kramers analysis and for "measurement model" fitting, and the Phillips-Tikhonov inversion method has been added as an alternative to the full CNLS inversion methods.

Input-Output

Many new data-transformation possibilities have been added to the DINPUT program. In addition, a new and powerful input-output program, MKLIO, which, among other things, simplifies preparation of LEVM input files, has been provided by Dr. Evgenij Barsoukov, and is incorporated as a part of the LEVM package. The present version of LEVM handles data with a maximum of 1001 frequencies (or times) and up to 42 free parameters. It requires extended memory. In addition, a smaller version of LEVM, which involves a maximum of 150 points and 28 free parameters and can operate with ordinary memory, is also included. Several useful new output files are now produced, especially LINOTLR.

A new program, MK4LF.EXE, uses input data or fitting data predictions at one of the four immittance levels and produces files of the corresponding data and fit at all four levels, particularly valuable for plotting and examination of the response at all of these levels. In addition, it provides an output file containing the log-log slopes of the real (with or without subtraction of its zero-frequency value) and imaginary parts of the data. The new file, SIZE.INC, makes it very easy to change the maximum number of frequencies and parameters default values when recompiling LEVM.

Now, output files with log or regular responses can be automatically made after a fit. Subtraction from the data of the effects of one or more parameters has been improved. The new version of LEVM includes several files and options which make it very easy to investigate the temperature dependencies of quantities obtained from CNLS fitting, including Arrhenius and Vogel-Fulcher NLS fitting models.

Several additional DCE's have been added to LEVM. Especially important and useful are the particular new DCE's discussed below. For these, valuable new quantities are calculated and are output at the end of CNLS fits for CSD situations. These quantities involve moments of the distribution involved in the fit and various useful high- and low-frequency limiting dielectric constant values. They make it easy to distinguish between CSD and DSD contributions to the total dielectric constant, an important distinction which has largely been ignored heretofore.

New Distributed Circuit Elements (DCE's)

Several new DCE algorithms have been added which can be used for either CSD or DSD fitting situations. Particularly important is one to fit frequency-response data to a highly accurate Kohlrausch-Williams-Watts stretched-exponential response model. Further, a new very general DCE: the generalized exponential DAE, GEDAE, has been added to the O-circuit. It allows one to fit data using sub-exponential DAE's, super-Gaussian DAE's, and all possibilities between an ordinary exponential DAE and a Gaussian DAE. The responses use adjustable cutoffs and allow both symmetrical and asymmetrical response. *The NELEM=9, Generalized Finite Warburg Diffusion DCE has been much extended to include a Poisson, Nernst, Planck (PNP) microscopic treatment based on my papers #'s 124 and 183.

In addition, starting from their exact DRT's, accurate empirical Havriliak-Negami, Cole-Cole, and Davidson-Cole response models may now be calculated with adjustable cutoffs and used for fitting, for the first time making these empirical response models physically realizable. Moreover, they can be used for CSD1 as well as ordinary CSD0 fitting, a possibility unavailable until now. The choice between CSD0 and CSD1 is dictated by the sign of the input quantity MODE. Further, MODE can be used to change from frequency response fitting to transient response fitting for these DCE's.

Miscellaneous

In addition, the fixed MODE input parameter is used to choose between fitting with many DCE's that can involve either the ordinary relaxation time τ_o (tau sub o) or an effective dielectric constant ϵ_τ (epsilon sub tau). The latter quantity is particularly valuable in showing any difference between the temperature dependencies of ΔR (or $\Delta\rho$) (delta R or delta rho) and τ_o (tau sub o).

When the quantity M in the LEVM input file is initially set negative, the program sets CELCAP to the permittivity of vacuum, appropriate for CSD data in specific form and so also for DSD dielectric-level data.

When the input quantity IOPT is set negative, its value determines the accuracy of iterative CNLS fitting and so the number of iterations and the number of significant figures in the fit estimates. Thus, for rather inaccurate data it is unnecessary to iterate to the limit of the machine accuracy of the computer.

Several useful changes and additions have been made to the 2- and 3-D plot program CTD, and two other such programs incorporated in earlier versions of LEVM have been eliminated. Two new fitting output quantities have been added: real and imaginary values of the average absolute relative residuals. Comparison of the values of these quantities with those of the corresponding standard deviations of the weighted residuals is valuable in identifying the presence of residual outliers. When LEVM is set to produce the file OUTIN as part of a fit, now the two standard deviations, S_F ($S_{sub F}$): SDWC (weighted residuals) and SDRC (relative residuals) are automatically included at the end of the third line of OUTIN. This is useful for archival purposes in identifying particular fits. Finally, the LEVM manual has been thoroughly revised and augmented.

2. V. 7.01 to V. 7.1

These changes/additions are less comprehensive than those above but are, nevertheless, important.

First, some KWW response models have been added that use the KWW DRT and thus allow arbitrary cutoff and both transient response and frequency response simulation and fitting, unlike the previously available NELEM=10 no-cutoff DCE choice. The first of these, NELEM=35, uses the exact DRT expression for BETA=1/3. The second, NELEM=36, uses the approximate, but quite accurate, KWW DRT algorithm of NELEM=37 to provide the above fitting choices for arbitrary values of BETA. The NELEM=37 choice separately allows fitting of DRT estimates, obtained either theoretically or from inversion, to be fitted to a KWW DRT with free cutoff and BETA parameters. The KWW model is very important for fitting both CSD and DSD data.

The length of the present LEVM source code now precludes useful compilation with Microsoft FORTRAN V. 5.1 because the resulting executable file, LEVM.EXE, requires too much space in lower memory. The file LEVM.EXS provided in LEVM V. 7.01 was produced by this approach and was smaller than that also provided previously as LEVM.EXL, compiled with Microsoft FORTRAN Powerstation V. 1. Here, only the executable compiled with the Power Station, designated LEVM.EXE, has been provided. It is compiled with the maximum possible number of free parameters and for up to 1001 independent variable (e.g., frequencies) values. It runs mostly in extended memory and so uses only a negligible amount of lower-memory space. It may be used directly in MS-DOS or from Windows 3x or Windows 9x using MS-DOS Mode. For those with LEVM 7.01, in the unlikely event that a smaller executable file is needed, one can use the V. 7.01 LEVM.EXS file mentioned above - at the cost of unavailability of the V. 7.1 changes and additions. Alternatively, one can reset the SIZE.INC file for smaller values and recompile LEVM V.7.1, possibly using the MS PowerStation-V.1 LEVM compile file CMLPV.BAT in the LEVMFOR subdirectory.

LEVM 7.1 makes it easier to discriminate between CSD and DSD frequency-response data, as discussed in Refs. 45 and 47 cited at the end of Section 1. In addition, it can be used to fit data with the Ngai coupling model, or with the usually more appropriate cutoff model, both discussed in Ref. 46.

A number of errors and misprints in the V. 7.01 manual have been corrected, as well as bringing it up to date for V. 7.1, and a few minor bugs in LEVM and in MKLIO have been eliminated.

Finally, a summary of some tips and fitting suggestions, otherwise buried throughout the Manual, have been collected and now appear at the end of the PREFACE.

3. V. 7.1 to V. 7.11

Some minor errors in the program and in the Manual have been corrected. A few additional features have been added and are described in the Manual. See especially the discussion on p. 5-30 and elsewhere about how to carry out, using Circuit-O, composite conductive-system fits, representing bulk and electrode effects, with CSD1 and CSD0 elements in series.

****4. V. 8.0 – 8.06, 8.07-8.11 (October, 2011)**

Several errors/bugs were corrected in these versions and further generality was added. In addition, some distributed circuit elements have been added and/or modified. See the present version of the manual and especially the discussion of the WINDOWS version in the Preface. Also see the revised discussion of ROE and A/L on p. 3-5. **New diffusion-related material and fitting models have been added on pp. 4-9, 4-10, and 4-26, as well as in Circuit H.