THE CIFIT PROGRAM

August, 2000

Alex D. Bain

INTRODUCTION

CIFIT is a program for fitting data from a selective inversion relaxation experiment on a system undergoing slow chemical exchange (1). In slow exchange, each different site is observable. They can be perturbed away from equilibrium, and observed as they return. This return to equilibrium is governed by a set of coupled differential equations. Given a table of observed intensities as a function of time, and a proposed mechanism, CIFIT will try to find the set of parameters (rates, relaxation times, initial and equilibrium intensities) that best fits the experimental data.

BACKGROUND

For a system with *n* sites, the general process of spin relaxation coupled with exchange is described as follows (2-4). If $M_i(t)$ is the magnetization of site *i* at time *t* and $M_i(¥)$ is the equilibrium magnetization at the same site, then the time dependence of the vector of magnetizations $M_i(t), i = 1..n$ is given by equation (1).

$$\frac{\partial}{\partial t}\begin{pmatrix} M_1(\infty)-M_1(t) \\ \vdots \\ M_n(\infty)-M_n(t) \end{pmatrix} = -\mathbf{A}\begin{pmatrix} M_1(\infty)-M_1(t) \\ \vdots \\ M_n(\infty)-M_n(t) \end{pmatrix}$$

where (1)

$$\mathbf{A} = \begin{pmatrix} R_1 & -k_{21} & -k_{31} & \cdots & -k_{n1} \\ -k_{12} & R_2 & -k_{32} & \cdots & -k_{n2} \\ -k_{13} & -k_{23} & R_3 & \cdots & -k_{n3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -k_{1n} & -k_{2n} & -k_{3n} & \cdots & R_n \end{pmatrix}$$

In this equation,

$$R_i = \frac{1}{T_{1i}} + \sum_{j \neq i} k_{ij} \tag{2}$$

$T_{1\mathrm{i}}$ is the spin-lattice relaxation time of site *i* and k_{ij} is the rate of exchange from site *i* to *j*. This formulation ignores any off-diagonal relaxation terms, which would give rise to nuclear Overhauser effects. The formal solution (5) to this coupled set of differential equations is given in (3).

$$\begin{pmatrix} M_1(\infty)-M_1(t) \\ \vdots \\ M_n(\infty)-M_n(t) \end{pmatrix} = \exp(-\mathbf{A}t)\begin{pmatrix} M_1(\infty)-M_1(0) \\ \vdots \\ M_n(\infty)-M_n(0) \end{pmatrix} \tag{3}$$

where **A** is the matrix in (1). If **U** is the matrix with the eigenvectors of **A** as its columns, and **L** is the diagonal matrix with the eigenvalues of **A** down the diagonal, equation (3) can be solved as in (4).

$$\begin{pmatrix} M_1(\infty)-M_1(t) \\ \vdots \\ M_n(\infty)-M_n(t) \end{pmatrix} = \mathbf{U}\ddot{\mathbf{E}}\mathrm{xp}\mathbf{U}(- \quad t) \quad {}^{-1}\begin{pmatrix} M_1(\infty)-M_1(0) \\ \vdots \\ M_n(\infty)-M_n(0) \end{pmatrix} \tag{4}$$

Even though there are many individual rates in (1), the exchange itself may be written as a rate, k, times a matrix which describes the mechanism. For instance, consider a two-site exchange. Let us define the rate, k, as the rate from site 2 to site 1, for notational convenience. If K, the equilibrium constant, is defined as in equation (5),

$$K = \frac{[site\ 2]}{[site\ 1]} \tag{5}$$

then the exchange can be written (ignoring relaxation) as in (6).

$$\frac{\partial}{\partial t}\begin{pmatrix} M_1(\infty)-M_1(t) \\ M_2(\infty)-M_2(t) \end{pmatrix} = -k\begin{pmatrix} K & -1 \\ -K & 1 \end{pmatrix}\begin{pmatrix} M_1(\infty)-M_1(t) \\ M_2(\infty)-M_2(t) \end{pmatrix} \tag{6}$$

The mechanism matrix containing K is independent of the rate, k, and so the single parameter k can be varied in the program to fit the data.

This constitutes a mathematical model of the coupled relaxation of a spin system. The role of the CIFIT program is to vary the parameters in this model until the sum of the squares of the differences between the modelled and observed data is minimized. Since the model is non-linear, this is a non-linear least-squares problem. We seek the global minimum in the sum of squares of the differences between modelled and observed, as a function of the model parameters. The number of parameters in the model is as follows. For each site, there is a value of the relaxation rate in the absence of exchange, a value for the equilibrium intensity and a value for the intensity at the start of the relaxation. Then for each exchange process there is a rate. For *nsts* sites and *nrks* processes, there are (*3*nsts+nrks*) parameters. The algorithm for finding the best set of parameters is that of Marquardt, which is an improvement on the standard non-linear least-squares method. In this algorithm, we need partial derivatives of the data points with respect to the parameters. These are taken from McClung and Muhandiram (6), except for the derivatives of the eigenvectors (7). For these we have used Wilkinson's formula (8), which is slightly different from McClungs's. Numerical Recipes (9) gives a good introduction to these methods and for more detail, the reader is referred to Draper and Smith (10), or Seber and Wild (11).

The Marquardt method is a combination of the method of steepest descents and the method of linearization. Steepest descents is better for poor guesses of the parameters, but linearization is better near the minimum. The Marquardt algorithm uses a parameter l to combine the methods. Initially a large l approximates steepest descents and then l is decreased as the fit gets better and the method resembles linearization. The parameters that control the initial value of l, and how much it is increased or decreased, are not entirely clear in the literature. They are constants in the program, and may be adjusted to suit particular types of data.

COMPILING THE PROGRAM

The program is written in fairly standard ANSI-style C, and it should compile under most C compilers. The main problem is function prototypes. Most compilers (Borland products, gcc, the SGI C compiler) support ANSI-style prototypes, but the standard compiler shipped with SunOS on SUNs does not. In the code, therefore, there is a preprocessor directive to allow either type of prototype. For ANSI-style prototypes, define __STDC__ (note two underscores on either side) to be 1. This may be already defined for your compiler - in Turbo C, this is done if the ANSI Keywords Only option is set. Note also that Turbo supports both types (even mixed!) of prototypes. For a PC based, 32-bit C compiler, we have used CodeWarrior from MetroWerks.

On a SGI Indigo running IRIX 5.3, the command for compiling the program is

```
cc cifit.c -D__STDC__ -o cifit -lm
```

where the -D option has set __STDC__.

On the SUN, or for old-style prototypes define __SUN__ (again, two underscores on either side) and compile the program as follows.

```
cc cifit.c -D__SUN__ -o cifit -lm
```

On an IBM RS/6000 Model 630 workstation running AIX version 2, __STDC__ is predefined for the xlc compiler. The command for compiling the program is as follows.

```
xlc cifit.c -o cifit -lm
```

The PC version of the program was compiled for Windows using the CodeWarrior C compiler (MetroWerks), and runs as a console application.

RUNNING THE PROGRAM

The program requires two input files and produces one output file. All of these are simple text files, which can be handled with the editor of your choice on your system. One input file is called the mechanism file, and contains initial values of the parameters, along with the definitions of the mechanism matrices. The second file, the data file, simply contains the observed magnetizations as a function of time. Both the mechanism and data files allow a fairly free format, including comments within the file. The output file contains all the results, for reference. There is also the option of putting out a plot file in a format that can be read into a spreadsheet (Excel, in this case) for inspection and plotting.

CIFIT supports a command-line argument for the filenames of these files. Typing

```
cifit test
```

will make the program look for a file named test.mch as the mechanism file and test.dat as the data file. The output will go into test.out. If no name follows cifit on the command line, then the user will be prompted individually for the names of the files. The command-line usage assumes the extensions .mch, .dat, and .out, but these are not required if the filenames are entered explicitly. In this case, the program will look for the exact filename entered.

Once the files have been opened, the program does a detailed balance calculation. This means that it calculates what the equilibrium values should be, based on the exchange mechanism entered (1). This is compared to the observed equilibrium values. This is merely for reference purposes, as a check on the mechanism, and the calculated values are not used in the rest of the program.

The program will then print out the initial guesses for the parameters, and ask for which ones need to be varied. Varying all the parameters at once is usually asking for trouble, so some strategy for finding the best set of parameters a few at a time is the best approach. The parameters are identified as to what they are (relaxation rates, infinity values, initial values or exchange rates) and are numbered. Note that the numbering starts at 0, not at 1 - this is C, not FORTRAN.

Once the variable parameters have been defined, the program will start trying to find better values, and will report on its progress. This will be printed on the computer console, as well as being put into the output file. Depending on the system, the calculation may or may not converge.

If it converges, the final values will be printed out, and the user will be asked whether a plot file is needed. This is simply a text file of observed and calculated data in a format that can be imported directly into a spreadsheet program.

STRUCTURE OF THE INPUT FILES

The input files are text files, created by any text editor, and are read in using the C function fscanf. This means that entries need to be separated by "whitespace" - spaces, tabs or carriage returns. Otherwise the format is free, and fairly robust. The exception is the title that occupies the first line of both the mechanism and the data file. Here, the line is read in (up to 80 characters) until a carriage return. Before each line of input, there is a procedure to strip off any whitespace (e.g. blank lines) and to ignore any line that starts with #. Such lines are encouraged for use as comments, for self-documentation of the input files. See below for examples.

The mechanism file has the following format. The places where # occurs indicate where comments are allowed.

Mechanism file
#
1st line: Title (<80 characters)

#

2nd line: Number of sites and number of rate processes (both integers). Because all the memory allocation is dynamic, there is no restriction on the number of sites. The maximum number of rate processes is defined by the constant MAXEX in the program. At the moment, this is set to 5, but this is arbitrary and can be changed.

#

3rd line: Initial guess at relaxation rates ($1/T_1$) for each site, in seconds^{-1}

#

4th line: Initial guess at M(∞) for each site.

#

5th line: Initial guess at M(0) for each site. The program actually works with M(0) - M(∞), but we input M(0), since that is easy to copy from the data file. The program does the subtraction.

For each mechanism:

#

6th line: Initial guess at the rate for this process

#

7th line: Number of non-zero, off-diagonal elements in the exchange matrix for this process.

for each non-zero element:

8th line: The row and column address of the element (integers) and its value (real). Even though the elements in equation [1] are negative, we put in positive values here, which are negated by the program. Note again that the addresses start at 0, not 1.

There are as many of these lines as needed for the system.

The data file has the following format - again comments are allowed where # is indicated.

#

1st line: Title (<80 characters, one line)

#

2nd line: Number of times at which the magnetization is sampled - i.e. the number of spectra acquired

#

next lines: The delay time and the value of each magnetization at that time.

POSSIBLE NUMERICAL PROBLEMS

Generally speaking, it is best that all the parameters be of the order of 1. This means that the units of intensity should give magnetizations that are not too big or small, and the time scale for the observed data should give rates that are within an order of magnitude or two of 1. The results should not depend on the scaling of the data, and this is a good test of a reliable fit. However, our experience is that magnetizations in the tens of thousands can give numerical problems.

It is also possible that mechanism matrices may give numerical problems. These may be close to degenerate and may cause the eigenvalue routine to find a singular matrix. In this case, a small perturbation of the exchange matrix may help. For instance, changing an element from 1 to 1.001 will have no effect on the data or on the conclusions (unless you can really measure intensities to better than 0.1%!) but may relieve the singularity of the matrix. Again, a test of a good fit is that it should be invariant with respect to small variations in the mechanism file.

USE OF THE PROGRAM

The return to equilibrium is determined by both the spin-lattice relaxation and by the exchange. It is our experience that these should be determined separately, in two different experiments. First, a non-selective T_1 experiment, then a selective experiment should be performed. The non-selective inversion-recovery is sensitive to the T_1 values, but relatively insensitive to the rate. However, the observed decays can not be used to extract T_1 directly, since the spin-lattice relaxation will be averaged by the exchange. The selective experiment is sensitive to all the parameters, but it is possible for the rates and the T_1's to interact strongly in a full fit using all the parameters.

Our strategy is to guess a value of the rate, fix it, and fit the non-selective data using all the other parameters. Then, fix the T_1's and fit the selective data for the rate. Put the rate back into the analysis of the non-selective data, to get improved T_1's, and continue this iteration. This method seems to give more reliable results than simply fitting all the parameters to one set of data.

Treat the error estimates in the output very carefully - they are probably overly optimistic. For a detailed analysis of the errors, see Seber and Wild (11) or our work (1).

EXAMPLES

The first example is the exchange in furfural, a two-site, unequal-populations case. The mechanism file is as follows. There is only a single process in this case, so we only need specify the two off-diagonal elements for the mechanism, in the ratio of 10:1, which is the ratio of the two equilibrium populations. Remember that the first site has address 0.

The mechanism file is furf.mch.

```
#Test data on furfural for cifit
furfural in cd2cl2 at 175 K, June 6/94

#two sites, one process
2 1

#1/T1 for each site
.177 .24

#M(infinity)
.0973 .9830

#and M(0) for each site
.0997 -.8768

#rate for the process
.36
```

```
#and two non-zero off-diagonal elements
#note that since there are unequal populations
#the off-diagonal elements are not equal, and
#must be the right way around
2
0 1 .1
1 0 1
```

and the data file, furf.dat, is

```
#Data file for cifit
furfural data from June 6, T=175K
#12 data points to be read in
12

# the data themselves, first the time, then
#the magnetizations of each site
.1 .0997 -.8768
30 .0973 .9830
5 .042 .4517
4 .0351 .3075
2 .0357 -.1361
10 .0744 .8114
1 .0562 -.4763
.5 .0755 -.6885
15 .0905 .9407
6 .0500 .5688
.7 .0676 -.6037
8 .0641 .7201
```

When you just type "cifit", the following dialogue occurs on the console.

```
cifit
cifit 2
Mechanism file: furf.mch
Data file: furf.dat
Output file: furf.out
Mechanism title: furfural in cd2cl2 at 175 K, June 6/94
Data title: furfural data from June 6, T=175K

Calculated values for z magns are:
 No. 0=  0.1000 No. 1=  1.0000
Normalized values for z magns are:
 No. 0=  0.0990 No. 1=  1.0000

Values for 1/T1 are:
 No. 0=  0.1770 No. 1=  0.2400
How many do you want to vary?:0
All values of 1/T1 fixed
Values for M(inf) are:
 No. 2=  0.0973 No. 3=  0.9830
How many do you want to vary?:2
Varying all  2 values for M(inf)
Values for M(0)-M(inf) are:
 No. 4=  0.0024 No. 5= -1.8598
How many do you want to vary?:2
Varying all  2 values for M(0)-M(inf)
Values for Rates are:
 No. 6=  0.3600
How many do you want to vary?:1
Varying all  1 values for Rates
Varying  5 parameters, with numbers  2  3  4  5  6
Starting the calculation
        2        3        4        5        6
   0.0973   0.9830   0.0024  -1.8598   0.3600
  -0.0034   0.0010   0.0117  -0.0579   0.0084

Iteration #  1      chi-squared:  0.0002843    alamda:  1.00e-04
        2        3        4        5        6
   0.0939   0.9840   0.0141  -1.9177   0.3684
   0.0000   0.0001   0.0001  -0.0002  -0.0005

Iteration #  2      chi-squared:  0.0002842    alamda:  1.00e-05
        2        3        4        5        6
   0.0939   0.9841   0.0141  -1.9179   0.3679
   0.0000   0.0000   0.0000   0.0000   0.0000

Final Values of Fitted Parameters and Uncertainties:
#  2 =     0.093943 +/-     0.001658
#  3 =     0.984056 +/-     0.001848
#  4 =     0.014138 +/-     0.003349
#  5 =    -1.917900 +/-     0.003840
#  6 =     0.367900 +/-     0.021948
Covariance matrix
  0.000003   0.000000  -0.000002  -0.000001   0.000016
  0.000000   0.000003  -0.000002  -0.000004  -0.000004
 -0.000002  -0.000002   0.000011  -0.000003   0.000022
 -0.000001  -0.000004  -0.000003   0.000015  -0.000044
  0.000016  -0.000004   0.000022  -0.000044   0.000482

Raw chi squared =      0.000284226, Scaled by M(inf) =      0.000290860
Percent Sqrt((scaled chisq)/(degr of freedom)) =     0.3913
Do you want to calculate a plot file? (y/n):y
Plot file: furf.qq
Enter start time, final time, increment:0 30 .5
```

We requested that everything be varied except for the relaxation rates. This gives the following output file, furf.out.

```
CIFIT 2.0
Mechanism title: furfural in cd2cl2 at 175 K, June 6/94
Data title: furfural data from June 6, T=175K
Number of sites = 2, 1 process
12 Time points read in
 0.100:  0.0997  -0.8768
30.000:  0.0973   0.9830
 5.000:  0.0420   0.4517
 4.000:  0.0351   0.3075
 2.000:  0.0357  -0.1361
10.000:  0.0744   0.8114
 1.000:  0.0562  -0.4763
 0.500:  0.0755  -0.6885
15.000:  0.0905   0.9407
 6.000:  0.0500   0.5688
 0.700:  0.0676  -0.6037
 8.000:  0.0641   0.7201

Initial values of parameters:
1/T1's
 No. 0=  0.1770 No. 1=  0.2400
M(inf)'s
 No. 2=  0.0973 No. 3=  0.9830
M(0)-M(inf)'s
 No. 4=  0.0024 No. 5= -1.8598
and M(0)'s for reference
 No. 4=  0.0997 No. 5= -0.8768
Rates
 No. 6=  0.3600
Chi squared value =   0.01054895

Iteration # 0
        2        3        4        5        6
   0.0973   0.9830   0.0024  -1.8598   0.3600
  -0.0034   0.0010   0.0117  -0.0579   0.0084
New chi^2 =   0.00028432

Iteration #  1    alamda:  1.00e-04    chi^2 =   0.00028432
        2        3        4        5        6
   0.0939   0.9840   0.0141  -1.9177   0.3684
   0.0000   0.0001   0.0001  -0.0002  -0.0005
New chi^2 =   0.00028423

Iteration #  2    alamda:  1.00e-05    chi^2 =   0.00028423
```

```
        2        3        4        5        6
   0.0939   0.9841   0.0141  -1.9179   0.3679
   0.0000   0.0000   0.0000   0.0000   0.0000
New chi^2 =   0.00028423

Final Values of Fitted Parameters and Uncertainties:
#  2 =     0.093943 +/-     0.001658
#  3 =     0.984056 +/-     0.001848
#  4 =     0.014138 +/-     0.003349
#  5 =    -1.917900 +/-     0.003840
#  6 =     0.367900 +/-     0.021948

Final values of all parameters:
1/T1's
 No. 0=  0.1770 No. 1=  0.2400
M(inf)'s
 No. 2=  0.0939 No. 3=  0.9841
M(0)-M(inf)'s
 No. 4=  0.0141 No. 5= -1.9179
and M(0)'s for reference
 No. 4=  0.1081 No. 5= -0.9338
Rates
 No. 6=  0.3679
Chi squared value =   0.00028423

Covariance matrix
  0.000003   0.000000  -0.000002  -0.000001   0.000016
  0.000000   0.000003  -0.000002  -0.000004  -0.000004
 -0.000002  -0.000002   0.000011  -0.000003   0.000022
 -0.000001  -0.000004  -0.000003   0.000015  -0.000044
  0.000016  -0.000004   0.000022  -0.000044   0.000482

Observed and Calculated Values
  0.10  Calcd:   0.1006  -0.8811
         Obsd:   0.0997  -0.8768
         Diff:   0.0009  -0.0043

 30.00  Calcd:   0.0938   0.9825
         Obsd:   0.0973   0.9830
         Diff:  -0.0035  -0.0005

  5.00  Calcd:   0.0437   0.4513
         Obsd:   0.0420   0.4517
         Diff:   0.0017  -0.0004

  4.00  Calcd:   0.0365   0.3043
         Obsd:   0.0351   0.3075
         Diff:   0.0014  -0.0032

  2.00  Calcd:   0.0355  -0.1391
         Obsd:   0.0357  -0.1361
         Diff:  -0.0002  -0.0030

 10.00  Calcd:   0.0754   0.8216
         Obsd:   0.0744   0.8114
         Diff:   0.0010   0.0102

  1.00  Calcd:   0.0552  -0.4757
         Obsd:   0.0562  -0.4763
         Diff:  -0.0010   0.0006

  0.50  Calcd:   0.0760  -0.6865
         Obsd:   0.0755  -0.6885
         Diff:   0.0005   0.0020

 15.00  Calcd:   0.0880   0.9336
         Obsd:   0.0905   0.9407
         Diff:  -0.0025  -0.0071

  6.00  Calcd:   0.0515   0.5653
         Obsd:   0.0500   0.5688
         Diff:   0.0015  -0.0035

  0.70  Calcd:   0.0665  -0.5982
         Obsd:   0.0676  -0.6037
         Diff:  -0.0011   0.0055

  8.00  Calcd:   0.0654   0.7238
         Obsd:   0.0641   0.7201
         Diff:   0.0013   0.0037

Raw chi squared =      0.000284226, Scaled by M(inf) =      0.000290860
Percent Sqrt((scaled chisq)/(degr of freedom)) =     0.3913
```

In this example, we asked for a plot file, furf.qq, which is a comma and quote delimited file for Excel. It is as follows.

```
"Observed and Calculated"
 0.10000,    0.1006,   -0.8811,    0.0997,   -0.8768,    0.0009,   -0.0043,
30.00000,    0.0938,    0.9825,    0.0973,    0.9830,   -0.0035,   -0.0005,
 5.00000,    0.0437,    0.4513,    0.0420,    0.4517,    0.0017,   -0.0004,
 4.00000,    0.0365,    0.3043,    0.0351,    0.3075,    0.0014,   -0.0032,
 2.00000,    0.0355,   -0.1391,    0.0357,   -0.1361,   -0.0002,   -0.0030,
10.00000,    0.0754,    0.8216,    0.0744,    0.8114,    0.0010,    0.0102,
 1.00000,    0.0552,   -0.4757,    0.0562,   -0.4763,   -0.0010,    0.0006,
 0.50000,    0.0760,   -0.6865,    0.0755,   -0.6885,    0.0005,    0.0020,
15.00000,    0.0880,    0.9336,    0.0905,    0.9407,   -0.0025,   -0.0071,
 6.00000,    0.0515,    0.5653,    0.0500,    0.5688,    0.0015,   -0.0035,
 0.70000,    0.0665,   -0.5982,    0.0676,   -0.6037,   -0.0011,    0.0055,
 8.00000,    0.0654,    0.7238,    0.0641,    0.7201,    0.0013,    0.0037,

"Calculated Smooth Curve"
 0.00000,    0.1081,  -0.9339,
 0.50000,    0.0760,  -0.6865,
 1.00000,    0.0552,  -0.4757,
 1.50000,    0.0425,  -0.2949,
 2.00000,    0.0355,  -0.1391,
 2.50000,    0.0325,  -0.0043,
 3.00000,    0.0322,   0.1128,
 3.50000,    0.0338,   0.2150,
 4.00000,    0.0365,   0.3043,
 4.50000,    0.0400,   0.3825,
 5.00000,    0.0437,   0.4513,
 5.50000,    0.0477,   0.5119,
 6.00000,    0.0515,   0.5653,
 6.50000,    0.0553,   0.6125,
 7.00000,    0.0589,   0.6542,
 7.50000,    0.0623,   0.6911,
 8.00000,    0.0654,   0.7238,
 8.50000,    0.0683,   0.7528,
 9.00000,    0.0709,   0.7785,
 9.50000,    0.0733,   0.8013,
10.00000,    0.0754,   0.8216,
10.50000,    0.0774,   0.8396,
11.00000,    0.0791,   0.8556,
11.50000,    0.0807,   0.8698,
12.00000,    0.0822,   0.8824,
12.50000,    0.0834,   0.8936,
13.00000,    0.0846,   0.9036,
13.50000,    0.0856,   0.9125,
14.00000,    0.0865,   0.9204,
14.50000,    0.0873,   0.9274,
15.00000,    0.0880,   0.9336,
15.50000,    0.0887,   0.9392,
16.00000,    0.0893,   0.9441,
16.50000,    0.0898,   0.9485,
17.00000,    0.0902,   0.9525,
17.50000,    0.0906,   0.9559,
18.00000,    0.0910,   0.9590,
18.50000,    0.0913,   0.9618,
19.00000,    0.0916,   0.9642,
19.50000,    0.0919,   0.9664,
20.00000,    0.0921,   0.9684,
20.50000,    0.0923,   0.9701,
21.00000,    0.0925,   0.9716,
21.50000,    0.0926,   0.9730,
22.00000,    0.0928,   0.9742,
22.50000,    0.0929,   0.9753,
23.00000,    0.0930,   0.9763,
23.50000,    0.0931,   0.9771,
24.00000,    0.0932,   0.9779,
24.50000,    0.0933,   0.9786,
25.00000,    0.0934,   0.9792,
25.50000,    0.0934,   0.9797,
26.00000,    0.0935,   0.9802,
26.50000,    0.0935,   0.9806,
27.00000,    0.0936,   0.9810,
27.50000,    0.0936,   0.9813,
28.00000,    0.0937,   0.9816,
28.50000,    0.0937,   0.9819,
29.00000,    0.0937,   0.9821,
29.50000,    0.0937,   0.9823,
```

The second example is the same data as in the sifit manual, from Ted McClung. All parameters, except the relaxation rates and one of the exchange rates, are varied. In this case, the command-line argument was used.

Mechanism file (mccl.mch)

```
mechanism from ted's sifit manual
# 4 sites, 2 processes
4 2
# 1/T1 for each site
1.71  1.53 2.07     2.2
#M(infinity)
1.064 1.105 0.8497  0.526
#M(0)
0.99  1.1   0.85    -0.35
#Rate for process 1
0.4
#6 non-zero off-diagonal elements
6
0 1 1.
0 2 1.
1 0 1.
2 0 1.
2 3 2.
3 2 1.
#Second process, for numerical stability
#very small rate
0.000001
#six elements, to make sure all sites are connected
6
0 1 1.
0 3 2.
1 0 1.
1 2 1.
2 1 1.
3 0 1.
```

Data file (mccl.dat).

```
ted's 4 site data from sifit manual
18
.0005 1.0   1.106 0.873 -0.346
.0007 0.998 1.109 0.861 -0.349
.001  0.995 1.098 0.841 -0.356
.003  0.993 1.106 0.852 -0.333
.005  0.993 1.109 0.867 -0.341
.007  0.988 1.092 0.833 -0.341
.01   0.979 1.109 0.841 -0.322
.02   0.973 1.087 0.804 -0.314
.05   0.986 1.095 0.801 -0.242
.08   1.01  1.107 0.805 -0.167
.1    1.00  1.093 0.792 -0.13
.2    1.06  1.09  0.759  0.0308
.4    1.029 1.096 0.758  0.25
.7    1.037 1.099 0.78   0.408
1.0   1.044 1.097 0.801  0.467
```

```
1.5   1.053 1.102 0.837  0.516
2.5   1.069 1.105 0.855  0.529
10    1.068 1.106 0.848  0.532

cifit mccl
cifit 940720.2
Mechanism title: mechanism from ted's sifit manual
Data title: ted's 4 site data from sifit manual

Calculated values for z magns are:
 No. 0=  2.0000 No. 1=  2.0000 No. 2=  2.0000 No. 3=  1.0000
Normalized values for z magns are:
 No. 0=  2.0228 No. 1=  2.1008 No. 2=  1.6154 No. 3=  1.0000

Values for 1/T1 are:
 No. 0=  1.7100 No. 1=  1.5300 No. 2=  2.0700 No. 3=  2.2000
How many do you want to vary?:0
All values of 1/T1 fixed
Values for M(inf) are:
 No. 4=  1.0640 No. 5=  1.1050 No. 6=  0.8497 No. 7=  0.5260
How many do you want to vary?:4
Varying all  4 values for M(inf)
Values for M(0)-M(inf) are:
 No. 8= -0.0740 No. 9= -0.0050 No.10=  0.0003 No.11= -0.8760
How many do you want to vary?:4
Varying all  4 values for M(0)-M(inf)
Values for Rates are:
 No.12=  0.4000 No.13=  0.0000
How many do you want to vary?:1
Enter the parameter number
12
Varying  9 parameters, with numbers  4  5  6  7  8  9 10 11 12
Starting the calculation
Number          4        5        6        7        8        9
Value        1.0640   1.1050   0.8497   0.5260  -0.0740  -0.0050
Increment    0.0036  -0.0008  -0.0003   0.0000  -0.0033   0.0020
Number         10       11       12
Value        0.0003  -0.8760   0.4000
Increment   -0.0010  -0.0052   0.0012

Iteration #  1     chi^2 =   0.00734283        alamda:  1.00e-04
Number          4        5        6        7        8        9
Value        1.0676   1.1042   0.8494   0.5260  -0.0773  -0.0030
Increment    0.0000   0.0000   0.0000   0.0000   0.0000   0.0000
Number         10       11       12
Value       -0.0007  -0.8812   0.4012
Increment    0.0000   0.0000   0.0002

Iteration #  2     chi^2 =   0.00734282        alamda:  1.00e-05
Number          4        5        6        7        8        9
Value        1.0677   1.1042   0.8494   0.5260  -0.0773  -0.0030
Increment    0.0000   0.0000   0.0000   0.0000   0.0000   0.0000
Number         10       11       12
Value       -0.0007  -0.8812   0.4014
Increment    0.0000   0.0000   0.0000

Final Values of Fitted Parameters and Uncertainties:
#  4 =     1.067653 +/-     0.004898
#  5 =     1.104161 +/-     0.005161
#  6 =     0.849445 +/-     0.005140
#  7 =     0.526029 +/-     0.005318
#  8 =    -0.077347 +/-     0.006429
#  9 =    -0.002999 +/-     0.006633
# 10 =    -0.000681 +/-     0.006378
# 11 =    -0.881189 +/-     0.006430
# 12 =     0.401378 +/-     0.031024
Normalized Covariance matrix
  0.000024   0.000002   0.000002   0.000000  -0.000027  -0.000002  -0.000002    0.000000
 0.000008
  0.000002   0.000027   0.000001  -0.000001  -0.000002  -0.000030  -0.000001    0.000000
 0.000009
  0.000002   0.000001   0.000026  -0.000003  -0.000002  -0.000001  -0.000027    0.000001
 0.000059
  0.000000  -0.000001  -0.000003   0.000028   0.000000   0.000000   0.000000   -0.000028
-0.000077
```

```
 -0.000027  -0.000002  -0.000002   0.000000   0.000041   0.000002   0.000002   0.000000
-0.000008
 -0.000002  -0.000030  -0.000001   0.000000   0.000002   0.000044   0.000000   0.000000
-0.000006
 -0.000002  -0.000001  -0.000027   0.000000   0.000002   0.000000   0.000041   0.000001
-0.000022
  0.000000   0.000000   0.000001  -0.000028   0.000000   0.000000   0.000001   0.000041
 0.000043
  0.000008   0.000009   0.000059  -0.000077  -0.000008  -0.000006  -0.000022   0.000043
 0.000962
Raw chi squared =      0.007342867, Scaled by M(inf) =      0.002187146
Percent Sqrt((scaled chisq)/(degr of freedom)) =     0.5892
Do you want to calculate a plot file? (y/n):n
```

And the output file (mccl.out) is

```
CIFIT 2.0
Mechanism title: mechanism from ted's sifit manual
Data title: ted's 4 site data from sifit manual
Number of sites = 4, 2 processes
18 Time points read in
 0.001:  1.0000   1.1060   0.8730  -0.3460
 0.001:  0.9980   1.1090   0.8610  -0.3490
 0.001:  0.9950   1.0980   0.8410  -0.3560
 0.003:  0.9930   1.1060   0.8520  -0.3330
 0.005:  0.9930   1.1090   0.8670  -0.3410
 0.007:  0.9880   1.0920   0.8330  -0.3410
 0.010:  0.9790   1.1090   0.8410  -0.3220
 0.020:  0.9730   1.0870   0.8040  -0.3140
 0.050:  0.9860   1.0950   0.8010  -0.2420
 0.080:  1.0100   1.1070   0.8050  -0.1670
 0.100:  1.0000   1.0930   0.7920  -0.1300
 0.200:  1.0600   1.0900   0.7590   0.0308
 0.400:  1.0290   1.0960   0.7580   0.2500
 0.700:  1.0370   1.0990   0.7800   0.4080
 1.000:  1.0440   1.0970   0.8010   0.4670
 1.500:  1.0530   1.1020   0.8370   0.5160
 2.500:  1.0690   1.1050   0.8550   0.5290
10.000:  1.0680   1.1060   0.8480   0.5320

Initial values of parameters:
1/T1's
 No. 0=  1.7100 No. 1=  1.5300 No. 2=  2.0700 No. 3=  2.2000

M(inf)'s
 No. 4=  1.0640 No. 5=  1.1050 No. 6=  0.8497 No. 7=  0.5260

M(0)-M(inf)'s
 No. 8= -0.0740 No. 9= -0.0050 No.10=  0.0003 No.11= -0.8760

and M(0)'s for reference
 No. 8=  0.9900 No. 9=  1.1000 No.10=  0.8500 No.11= -0.3500

Rates
 No.12=  0.4000 No.13=  0.0000
Chi squared value =   0.00771789

Iteration # 0
Number          4        5        6        7        8        9
Value        1.0640   1.1050   0.8497   0.5260  -0.0740  -0.0050
Increment    0.0036  -0.0008  -0.0003   0.0000  -0.0033   0.0020
Number         10       11       12
Value        0.0003  -0.8760   0.4000
Increment   -0.0010  -0.0052   0.0012
New chi^2 =   0.00734283

Iteration #  1    alamda:  1.00e-04    chi^2 =   0.00734283
Number          4        5        6        7        8        9
Value        1.0676   1.1042   0.8494   0.5260  -0.0773  -0.0030
Increment    0.0000   0.0000   0.0000   0.0000   0.0000   0.0000
Number         10       11       12
Value       -0.0007  -0.8812   0.4012
Increment    0.0000   0.0000   0.0002
New chi^2 =   0.00734282
```

```
Iteration #  2    alamda:  1.00e-05    chi^2 =   0.00734282
Number          4        5        6        7        8        9
Value        1.0677   1.1042   0.8494   0.5260  -0.0773  -0.0030
Increment    0.0000   0.0000   0.0000   0.0000   0.0000   0.0000
Number         10       11       12
Value       -0.0007  -0.8812   0.4014
Increment    0.0000   0.0000   0.0000
New chi^2 =   0.00734282

Final Values of Fitted Parameters and Uncertainties:
#  4 =     1.067653 +/-     0.004898
#  5 =     1.104161 +/-     0.005161
#  6 =     0.849445 +/-     0.005140
#  7 =     0.526029 +/-     0.005318
#  8 =    -0.077347 +/-     0.006429
#  9 =    -0.002999 +/-     0.006633
# 10 =    -0.000681 +/-     0.006378
# 11 =    -0.881189 +/-     0.006430
# 12 =     0.401378 +/-     0.031024

Final values of all parameters:
1/T1's
 No. 0=  1.7100 No. 1=  1.5300 No. 2=  2.0700 No. 3=  2.2000

M(inf)'s
 No. 4=  1.0677 No. 5=  1.1042 No. 6=  0.8494 No. 7=  0.5260

M(0)-M(inf)'s
 No. 8= -0.0773 No. 9= -0.0030 No.10= -0.0007 No.11= -0.8812

and M(0)'s for reference
 No. 8=  0.9903 No. 9=  1.1012 No.10=  0.8488 No.11= -0.3552

Rates
 No.12=  0.4014 No.13=  0.0000
Chi squared value =   0.00734282

Covariance matrix
  0.000024   0.000002   0.000002   0.000000  -0.000027  -0.000002  -0.000002    0.000000
 0.000008
  0.000002   0.000027   0.000001  -0.000001  -0.000002  -0.000030  -0.000001    0.000000
 0.000009
  0.000002   0.000001   0.000026  -0.000003  -0.000002  -0.000001  -0.000027    0.000001
 0.000059
  0.000000  -0.000001  -0.000003   0.000028   0.000000   0.000000   0.000000   -0.000028
-0.000077
 -0.000027  -0.000002  -0.000002   0.000000   0.000041   0.000002   0.000002    0.000000
-0.000008
 -0.000002  -0.000030  -0.000001   0.000000   0.000002   0.000044   0.000000    0.000000
-0.000006
 -0.000002  -0.000001  -0.000027   0.000000   0.000002   0.000000   0.000041    0.000001
-0.000022
  0.000000   0.000000   0.000001  -0.000028   0.000000   0.000000   0.000001    0.000041
 0.000043
  0.000008   0.000009   0.000059  -0.000077  -0.000008  -0.000006  -0.000022    0.000043
 0.000962

Observed and Calculated Values
  0.00  Calcd:   0.9904   1.1011   0.8484  -0.3538
         Obsd:   1.0000   1.1060   0.8730  -0.3460
         Diff:  -0.0096  -0.0049  -0.0246  -0.0078

  0.00  Calcd:   0.9904   1.1011   0.8482  -0.3533
         Obsd:   0.9980   1.1090   0.8610  -0.3490
         Diff:  -0.0076  -0.0079  -0.0128  -0.0043

  0.00  Calcd:   0.9905   1.1011   0.8480  -0.3525
         Obsd:   0.9950   1.0980   0.8410  -0.3560
         Diff:  -0.0045   0.0031   0.0070   0.0035

  0.00  Calcd:   0.9909   1.1011   0.8466  -0.3473
         Obsd:   0.9930   1.1060   0.8520  -0.3330
         Diff:  -0.0021  -0.0049  -0.0054  -0.0143
  0.01  Calcd:   0.9913   1.1010   0.8451  -0.3420
         Obsd:   0.9930   1.1090   0.8670  -0.3410
         Diff:  -0.0017  -0.0080  -0.0219  -0.0010

  0.01  Calcd:   0.9916   1.1010   0.8437  -0.3368
         Obsd:   0.9880   1.0920   0.8330  -0.3410
         Diff:   0.0036   0.0090   0.0107   0.0042

  0.01  Calcd:   0.9922   1.1009   0.8416  -0.3291
         Obsd:   0.9790   1.1090   0.8410  -0.3220
         Diff:   0.0132  -0.0081   0.0006  -0.0071

  0.02  Calcd:   0.9940   1.1007   0.8349  -0.3039
         Obsd:   0.9730   1.0870   0.8040  -0.3140
         Diff:   0.0210   0.0137   0.0309   0.0101

  0.05  Calcd:   0.9990   1.1000   0.8170  -0.2326
         Obsd:   0.9860   1.0950   0.8010  -0.2420
         Diff:   0.0130   0.0050   0.0160   0.0094

  0.08  Calcd:   1.0035   1.0995   0.8021  -0.1678
         Obsd:   1.0100   1.1070   0.8050  -0.1670
         Diff:  -0.0065  -0.0075  -0.0029  -0.0008

  0.10  Calcd:   1.0062   1.0992   0.7938  -0.1277
         Obsd:   1.0000   1.0930   0.7920  -0.1300
         Diff:   0.0062   0.0062   0.0018   0.0023

  0.20  Calcd:   1.0171   1.0980   0.7665   0.0394
         Obsd:   1.0600   1.0900   0.7590   0.0308
         Diff:  -0.0429   0.0080   0.0075   0.0086

  0.40  Calcd:   1.0309   1.0972   0.7564   0.2535
         Obsd:   1.0290   1.0960   0.7580   0.2500
         Diff:   0.0019   0.0012  -0.0016   0.0035

  0.70  Calcd:   1.0429   1.0975   0.7799   0.4089
         Obsd:   1.0370   1.0990   0.7800   0.4080
         Diff:   0.0059  -0.0015  -0.0001   0.0009

  1.00  Calcd:   1.0509   1.0986   0.8063   0.4742
         Obsd:   1.0440   1.0970   0.8010   0.4670
         Diff:   0.0069   0.0016   0.0053   0.0072

  1.50  Calcd:   1.0595   1.1006   0.8327   0.5119
         Obsd:   1.0530   1.1020   0.8370   0.5160
         Diff:   0.0065  -0.0014  -0.0043  -0.0041

  2.50  Calcd:   1.0661   1.1031   0.8473   0.5248
         Obsd:   1.0690   1.1050   0.8550   0.5290
         Diff:  -0.0029  -0.0019  -0.0077  -0.0042

 10.00  Calcd:   1.0677   1.1042   0.8494   0.5260
         Obsd:   1.0680   1.1060   0.8480   0.5320
         Diff:  -0.0003  -0.0018   0.0014  -0.0060

Raw chi squared =      0.007342824, Scaled by M(inf) =      0.002187111
Percent Sqrt((scaled chisq)/(degr of freedom)) =     0.5892
```

IF THERE ARE PROBLEMS

If you find bugs, or you have some useful suggestions, please contact

Alex D. Bain
Dept. of Chemistry
McMaster University
1280 Main St. W.
Hamilton, Ontario, Canada
L8S 4M1

Phone: (905) 525 9140 ext 24524
fax: (905) 522 2509
email: bain@mcmaster.ca
home page: http://www.chemistry.mcaster.ca/faculty/bain

REFERENCES

1. A.D.Bain and J.A.Cramer, *J.Magn.Reson.* **118 A**, 21 (1996).

2. H.M.McConnell, *J.Chem.Phys.* **28**, 430 (1958).

3. A.D.Bain and J.A.Cramer, *J.Magn.Reson.* **103 A**, 217 (1993).

4. A.D.Bain and J.A.Cramer, *J.Phys.Chem*. **97**, 2884 (1993).

5. C.Moler and C.F.van Loan, *SIAM Review* **20**, 801 (1978).

6. D.R.Muhandiram and R.E.D.McClung, *J.Magn.Reson.* **71**, 187 (1987).

7. A.D.Bain and D.A.Fletcher, *Mol.Phys.* **95**, 1091 (1998).

8. J.H.Wilkinson, *The Algebraic Eigenvalue Problem,* Clarendon Press, Oxford, (1965).

9. W.H.Press, B.P.Flannery, S.A.Teukolsky and W.T.Vetterling, *Numerical Recipes in C. The Art of Scientific Computing,* Cambridge University Press, Cambridge, (1988).

10. N.R.Draper, H.Smith, *Applied Regression Analysis,* John Wiley and Sons Inc., Toronto, (1981).

11. G.A.F.Seber and C.J.Wild, *Nonlinear Regression,* Wiley, New York, (1989).