

RSFIT User's Guide v1.05

For use with RSFIT 4.00

Recent changes

version 4.00: There is a new constraint system. See CONSTRAINTS card below. The old CONSTRAINT and NONLINEAR cards still work, but you cannot mix and match the styles. Also, see Quick Start regarding using gnuplot to plot fits.

Quick start

RSFIT v3.4 and higher requires an input file. The syntax is

```
rsfit [-vef] [filename]
```

RSFIT must be run from a subdirectory called "input" or a directory called "output" with the same rank (that is, ../output). Both the input and output directories must exist. The input file in turn requires certain parameters for a fit to be performed. There are a substantial number of optional commands. At the minimum, one needs a file that contains roughly the following

```
#Sample input file
#Required parameters

#OUTFILE <filename> specifies the name of the file where the final fit parameters,
#includeing all the limits, constraints, names of files, starting and final parameters, etc.,
#are to be stored.
OUTFILE sample

#STDDIR <run_directory> gives the run directory for the k-space standard files
STDDIR jan00

#STDFILE <filename> <rdist> gives the name of a standard file to be used for fitting,
#and the pair distance corresponding to that standard
STDFILE u_o2.38.f8 2.3802

#KRANGE rkmin rkmax rkwidth ikwt gives the minimum and maximum wave vector
#to be used in the Fourier transform, the width of the window function used the
#transform, and the k-weight to be used, e.g.  $K^2$ :
KRANGE 2.5 13.5 0.3 2

RRANGE 1.2 3.5
FITDIR jan00
FITFILE 119b1
```

```
#Not required, but close!  
PEAK 0.05 0.0 3.0 -5 0.0 0.0  
#All fits must be signaled as “ready” with this:  
STARTFIT
```

To fit with more peaks, add STDFILE commands. The format of the file is free-format in the sense that the order of the input is not important, except that the STDFILE lines must be sequential. (The same is true of CONSTRAINT(S) and NONLINEAR commands. See the appropriate section.) To perform another fit, simply add the desired lines after the last STARTFIT command. If a given command is not given in the new fit, the values from the previous fit are used. There are some caveats to adding fits, however. The most important is that the number of STDFILES must be the same as in the first fit specified in the file. Also, at this time, you must not have CONSTRAINT(S) or NONLINEAR lines in the subsequent fits.

Standard output files to a “rsfit filename.x” (.x is a version number) command are

```
rsfit.dat  
rsfit_ks.dat  
../input/filename.x+1  
../output/filename.x  
../output/filename.x.cor      (if a correlation matrix is requested).
```

The ASCII r-space fit is given in rsfit.dat, and the equivalent k-space fit is in rsfit_ks.dat. Each also contains the data that was fitted to, and the k-space file has the filtered data as well as the “raw” data. Also, if a line “OUTFITFILE *filename*” is given in the input file, the fit will also be written to *filename*.

If a ‘-f’ option is given on the command line, the actual fit in binary format will be output into three files with names like

```
datafile_peak.zzz.x   This is the final pair distribution function used  
datafile_rspk.zzz.x  This is the actual fit  
datafile_diff.zzz.x  This gives the fit residuals
```

Plotting data can be performed with PLOT, or using a gnuplot script that lives in /home/exafs/bin/rsfit.gp, or the equivalent path on your system. For example

```
hahn$ gnuplot  
gnuplot>load “/home/exafs/bin/rsfit.gp”
```

... will produce a 2-panel plot showing the fit result in both k-space and r-space by reading the rsfit.dat and rsfit_ks.dat files. This method of looking at fit results is very convenient if one keeps a separate window open with gnuplot running, and then just “up arrow” to repeat the load command to refresh the plot after a new fit.

Finally, the format of the output file is not for the faint-of-heart. To translate this file into a more readable form, from the ../output directory type

`readxxo filename.x`
this will output a file called `read.zzz.x`. One may also type
`xxo2tex filename.x`
to generate a ReVTeX table of the results. The ReVTeX table makes some assumptions
(for instance, symmetric sigma errors), so it should be scrutinized before publication, etc.

Command summaries

Overview

Syntax is `rsfit [-OPTIONS] input_filename`

Command line options are `-h`, `-v`, `-s`, `-l`, `-f`, `-e`, `-c`, `-a`. Type `rsfit -h` for description.

Required input file commands are `STDDIR`, `STDFILE`, `KRANGE`, `RRANGE`, `OUTFILE`, `FITDIR`, `FITFILE`, and `STARTFIT`.

Starting values for pair parameters are set either with `PEAK` or with `SIGMAS`, `RSHS`, `AMPS`, `E0S`, `C3S`, and `C4S`. `DUMMIES` specifies starting values for dummy variables

Commands that constrain various parameters include `LIMITS`, `INCSIG`, `INCRSH`, `INCAMP`, `INCE0`, `INCC3`, `INCC4`, `INCDUMB`, `E0SAME`, `SIGSAME`, `RSHSAME`, `AMPRSAME`, `SIGHARDLIM`, `CONSTRAINT`, `NONLINEAR`, and `CONSTRAINTS`.

Commands to set how the error analysis is performed are `CORREL`, `ERRTYPE`, `VARSERR`, `VARRERR`, `VARAERR`, `VAREERR`, `VARTERR`, `VARFERR`, and `VARDERR`.

Commands for specifying I/O: `ALLPATHSOUT`

Other optional commands are `MAXITER`, `LAMBDA`, `NOMFP`, `DERIVWT`, `NODERIVWT`, `REMGGAUS`, `KDATA`, `RDATA`, `IWRIT`, and `HAMILTON`.

Comment lines start with either a `*` or a `#`. Blank lines are allowed.

The order of the commands is not important, except `STDFILE` lines must be sequential, as must `CONSTRAINT(S)` and `NONLINEAR` command lines. Also, if multiple fits are specified within an input file, the number of `STDFILE` lines must either be zero or the same as the last fit, and no `CONSTRAINT(S)` or `NONLINEAR` lines can be specified. Other kinds of constraints, such as `E0SAME` and `INCRSH` may be specified. The `HAMILTON` option is not defined for the first fit in the file (see the detailed summary for `HAMILTON`).

Fits begin with a `STARTFIT` command; that is, all information up to the `STARTFIT` command is used, and information below such a command is used for the next fit.

Detailed summaries

Command line options

- v verbose output. Includes a continuously updated line with the number of calls to the EXAFS function routine, an updated R(%) and C2 (=chi2 with data errors taken as unity). The last two numbers on this line are a continuous update of C3 and C4.
- s silent mode: absolutely nothing is printed to screen, except errors.
- f output fit files. This flag MUST be present to obtain fit files.
- e perform error analysis. This flag MUST be present to perform error analysis.
- c append to “chi2.dat” the chi2 value from the fit
- a append to “alpha.dat” the alpha value from a Hamilton test
- h use this for a complete list of options

Required commands

Parameters starting with ‘i’ are integers. All others are real or character strings.

KRANGE *kmin, kmax, kwin, ikwt*

Specifies range in k-space for the Fourier transform. *kmin* is the minimum and *kmax* is the maximum for the Gaussian window. *kwin* is a width for the Gaussian window. *ikwt* is the exponent in $k^{ikwt} * \chi(k)$ used for the transform, and thus for the fit.

RRANGE *rmin, rmax*

Specifies the range in r-space over which the statistical χ^2 will be calculated.

STDDIR *run_directory*

Specifies the *run_directory* where the standard files are stored. For instance, if the standard files are in a directory /home/exafs/exdata/jan00/ks (all standard files are stored in k-space), then *run_directory* is ‘jan00’.

STDFILE *filename, r_std*

Specifies the filename of an EXAFS standard, together with the effective pair distance *r_std* of that standard.

FITDIR *datafile_dir*

Specifies the directory where the data files are stored. For instance, if the data files are in a directory /home/exafs/exdata/jan00/ks (data files are assumed to be in k-space unless an RDATA command is given, then *datafile_dir* is ‘jan00/rs’).

FITFILE *datafile1,..., datafileN*

Specifies the datafile filename(s) to be fit. Indicating more than one filename causes RSFIT to average the files together and calculate the error *as a function of r*. This error is

then used in the calculation of the statistical- χ^2 , and in any subsequent error analysis. See also ERRTYPE.

STARTFIT

Command that signals RSFIT that all the input has been entered.

Parameter input

PEAK $\sigma, \Delta r, A, \Delta E0, C3, C4$

Specifies starting parameters for a given atom pair. The number of the pair is given by the number of PEAK lines in the input file. So, for instance, if there are 5 standards, there should be 5 PEAK lines, the first one corresponding to the first standard, the 2nd PEAK line to the 2nd standard, etc. The starting parameters are the distribution width, σ (Å); the shift in pair distance from the standard's value, Δr (Å); a multiplicative coefficient A, the shift in E0, $\Delta E0$ (eV); the third cumulant, C3 (Å); and the fourth cumulant C4 (Å).

A few important points to keep in mind:

- All of these values are relative to the corresponding standard file.
- If one uses a PEAK command, it is best not to use the SIGMAS, RSHS, AMPS, EOS, C3S, or C4S commands; though the DUMMIES command can be used with PEAK, if necessary.
- If PEAK lines are not repeated for a new fit, each subsequent fit will start from the final parameters of the previous fit. This is also true for the SIGMAS, RSHS, AMPS, EOS, C3S, C4S, and DUMMIES commands.

SIGMAS $\sigma_1, \sigma_2, \dots, \sigma_{TOT}$

Specify the starting distribution width σ 's (Å) for each standard pair 1-TOT.

RSHS $\Delta r_1, \Delta r_2, \dots, \Delta r_{TOT}$

Specify the starting r-shifts (Å) for each standard pair 1-TOT.

AMPS A_1, A_2, \dots, A_{TOT}

Specify the starting multiplicative coefficients A for each standard pair 1-TOT.

EOS $\Delta E0_1, \Delta E0_2, \dots, \Delta E0_{TOT}$

Specify the starting value of the E0 shifts (eV) for each standard pair 1-TOT.

C3S $C3_1, C3_2, \dots, C3_{TOT}$

Specify the starting value of the third cumulant C3's (Å) for each standard pair 1-TOT.

C4S $C4_1, C4_2, \dots, C4_{TOT}$

Specify the starting value of the fourth cumulant C4's (Å) for each standard pair 1-TOT.

DUMMIES D_1, D_2, \dots, D_{TOT}

Specify the starting value of each dummy variable, 1-TOT. Dummy variables should be used when additional parameters (not associated with a FEFF standard file, such as S_0^2) are needed in a fit. They can be used in constraints, just as other parameters.

A note about determining the number of dummy parameters: If the number of dummy parameters in the constraints differs from the number specified in a DUMMIES line, the number in the constraints is used. This behavior is useful if many dummy variables are necessary, but they are mostly dependent variable. Unexpected results can occur if the number of dummy variables is not consistent throughout the input file. Some of this behavior can be controlled or understood by looking at or simply using the new input file generated by RSFIT.

Constraining commands

LIMITS *sig_limit, rsh_limit, ampr_limit, e0_limit, c3_limit, c4_limit(, dummy_limit)*
Default is to not impose limits. These limits are on the amount of *change* a parameter can incur in the current fit, and are sometimes referred to as “soft limits.” The dummy limit is optional (for compatibility with old input files); if it is not specified, it defaults to 1000.0. NOTE: Limits on Amplitude and Dummy variables are fractional change. For example, limits of 0.5 means do not change the value by more than 50%. Hence, the dummy_limit default of 1000.0 means do not change by more than 1000.0%.

SIGHARDLIM, RSHHARDLIM, AMPHARDLIM, E0HARDLIM DUMHARDLIM
lower_limit <upper_limit>

These limits specify absolute limits on a class of parameters. For instance, one could limit sigmas to never be below 0.02 Angstroms (SIGHARDLIM 0.02), or dummy variables to only be in the range 0 to 1 (DUMHARDLIM 0 1). The latter case is common if a dummy variable is used to express the fraction of a given phase.

INCSIG *sig_inc1, sig_inc2... sig_incTOT*
The default value for all peaks is 0.01 Å.

INCRSH, INCAMP, INCC3, INCC4 and INCDUMB are similarly specified, with the same default values. INCE0 is similarly specified, but the default value is 1 eV.

EOSAME
Forces all E0 shifts for all standards to be the same.

SIGSAME
Forces all widths σ for all standards to be the same.

RSHSAME
Forces all pair distance shifts Δr for all standards to be the same.

AMPRSAME
Forces all amplitude ratios to be held fixed. For instance, if one uses a command
AMPS 1.0 2.0

for a 2-peak fit, AMPRSAME will keep the amplitude of the second peak to be twice that of the first peak. A possible result might be

AMPS 0.95 1.90

NONLINEAR *Pfix, P1, P2, C1, C2, C3*

(*deprecated, replaced by CONSTRAINTS*)

This line specifies a nonlinear constraint. The P's indicate a parameter in the form Sx, Rx, Ax, Ex, Tx, Fx, and Dx for the σ , Δr , A, $\Delta E0$, C3, or C4 parameter of the xth peak or the xth dummy variable, respectively. For example, 'A3' is the parameter for the amplitude of the 3rd peak, and 'F19' is the fourth cumulant for the 19th peak. The constraint is in the form:

$$Pfix=C1*P1*P2 + C2*P1/P2 + C3$$

All NONLINEAR lines must precede any CONSTRAINT lines, and must be sequential in the input file. Also, NONLINEAR lines must only be specified in the first fit in an input file.

CONSTRAINT *Pfix, P1, P2, C1, C2, C3*

(*deprecated, replaced by CONSTRAINTS*)

This line specifies a linear constraint. The P's are described above under NONLINEAR. The constraint is in the form:

$$Pfix=C1*P1 + C2*P2 + C3$$

All CONSTRAINT lines must follow any NONLINEAR lines, and must be sequential in the input file. Also, CONSTRAINT lines must only be specified in the first fit in an input file.

CONSTRAINTS

dep variable=formula1

dep variable=formula2

dep variable=etc.

The *formulae* specify general constraints. References to parameters must be in the form Sx, Rx, Ax, Ex, Tx, Fx, and Dx for the σ , Δr , A, $\Delta E0$, C3, or C4 parameter of the xth peak or the xth dummy variable, respectively. For example, 'A3' is the parameter for the amplitude of the 3rd peak, and 'F19' is the fourth cumulant for the 19th peak.

For example, to make a dummy variable act as an S02:

CONSTRAINTS

A1=D1*12

A2=D1*6

A3=D1*24

One may also use exponents and certain functions (i.e. sin,cos,tan,asin,log10,log,sqrt,exp). So, for instance, one can constrain changes in a bondlength to be along a right triangle:

$$R1=sqrt((R2+3)^2+(R3+4)^2)-5 \text{ #For a 3,4,5 triangle}$$

Once the CONSTRAINTS line has been specified, all the constraint formulae must immediately follow it, not including comment lines.

Error analysis

CORREL

Performs correlation matrix calculation.

ERRTYPE *ierrtype*, [*method*]; where *method*=<number>, “stern”, or “chi2”
Specifies whether errors are calculated in the positive direction (*ierrtype*=1), the negative direction (*ierrtype*=-1) or both (*ierrtype*=0). The default value is *ierrtype*=1. The method used for error calculations can either be to specify a constant error for the real and imaginary parts of the Fourier transform, or indicate the method to be used in calculating the statistical-chi2. The string “stern” indicates that RSFIT is to assume the fit model accurately describes the data, that is, that the statistical-2 is equal to the degrees of freedom in the fit, as given by the number of independent variables from Stern's rule minus the number of variables in the fit. The string “chi2” uses the standard deviation distribution for each point in *r* over multiple data files, as specified on the FITFILE line.

Default behavior:

- If no method is given, use “stern” if only one data file is given, “chi2” if multiple data files are given.
- If a method is given, use that method, if possible. The only exception is that if “chi2” is indicated, but only one input file is specified, theoretical error calculation will fall back to “stern”.

VARSEERR *i1, i2, ..., iTOT*

Specifies which σ parameters will have their error bars calculated, where *ix*=0 indicates to not determine the error on the σ parameter for the *ith* peak, and *ix*=1 indicates to determine the error. Default values set all *ix*=1. Note that errors are only calculated if the parameter in question is varied in the fit.

VARRERR, VARAERR, VAREERR, VARTERR, VARFERR, and VARDERR
commands specify same information for the other variable types.

HAMILTON *iham1, (iham2, iham3, ... ihamTOT)*

Perform Hamilton test on fit after removing the peaks listed (must be at least 1). NOTE: the HAMILTON flag cannot be used on the first fit, since the first fit is defined as the baseline fit.

The Hamilton test gives the significance level (as a percentage) of the peak being removed. The baseline fit should include the peaks to be tested. The Hamilton flag calculates how much worse the fit becomes when the peak is removed; hence, the significance of the peak. The idea is that if one continues to add more peaks (and hence more degrees of freedom), the fit can always be improved (i.e. a lower R-value); however, the improvement of the fit may not be *statistically significant*

To use this flag, set up two fits or more fits within an input file. The first fit will be the baseline and should not contain the HAMILTON flag. Subsequent fits can remove any number of peaks for testing. For instance:

<...other lines with fit set up>

```
STDFILE peak_1 2.0
STDFILE peak_2 3.0
STDFILE peak_3 4.0
PEAK 0.05 0.0 12.0 -10 0 0
PEAK 0.05 0.0 12.0 -10 0 0
PEAK 0.05 0.0 12.0 -10 0 0
STARTFIT
HAMILTON 2,3
STARTFIT
```

In this example, the 2nd and 3rd peaks will be “zero’d out” in the second fit. Statistics will be reported in the ../output file.

Tip: When using this flag, make sure you haven’t set the coefficient of a parameter from a peak you are testing to zero, since RSFIT will attempt to vary that peak even though it will then have no effect on the R(%). For instance, using the example from above, one might include a constraint:

```
R1=sqrt((R2+3)^2+(R3+4)^2)-5 #For a 3,4,5 triangle
```

which would work fine. But if for some reason you had a constraint like this:

```
R1=sqrt((R2+3)^2+0*(R3+4)^2)-5 #For a 3,4,5 triangle
```

RSFIT would attempt to vary R3 after its amplitude was set to zero (the HAMILTON 2,3 line), but since it wouldn’t affect R1 at all, the fit could never converge.

I/O Handling

ALLPATHSOUT

If a -f flag is given, output all fit files, including the individual paths. The default is to not include the individual paths.

Miscellaneous commands

MAXITER *imax_iterations*

Sets the maximum number of calls to the EXAFS function routine. The default value is *imax_iterations*=3000. This is useful for not allowing a fit to go on and on and on...

LAMBDA *mfp*

Indicates to *mfp* (Å) as the mean-free path for the amplitude correction when applying r-shifts. The default value is 8.0 Å.

NOMFP

Turns off the default mean-free path correction.

NODERIVWT

Indicates not to use derivative weighting when calculating χ^2 . This is the default.

DERIVWT

Turns on derivative weighting for calculating χ^2 .

REMGAS *width*

Indicates that negative values of fitted σ 's are possible (not default)! This is often the case when using experimentally derived standards. In this case, *width* should be set as the maximum expected difference between the data and the standard, and should be a positive number. For low temperature experimental standards, *width*=4.0 Å is generally sufficient.

KDATA

Indicates data file is in k-space. This is the default.

RDATA

Indicates the data is in r-space. In this case KRANGE is not required. THIS OPTION HAS NOT BEEN TESTED!

IWRIT

Turns on some extra verbosity...

Statistical Analysis in RSFIT

Error calculations in RSFIT do not use a covariance matrix, as in the standard Levinberg-Marquart minimization method typically used in other fitting programs. There are two drawbacks to using the standard covariance matrix. The first is that the Levinberg-Marquart method has a tendency to get trapped in a local minimum in the minimization function. The second drawback is that the curvature in the statistical- χ^2 as a function of a given parameter is used to calculate the standard deviation. This assumption can fail in complicated fitting landscapes, especially in the presence of local minima or complicated lineshapes, the latter of which is nearly ubiquitous to EXAFS fits.

For these reasons, RSFIT uses a stepping fit routine STEPIT for minimizing the best-fit parameter, and routine FIDO that searches the parameter space for the point at which the statistical- χ^2 has increased an amount corresponding to one standard deviation in the parameter. More specifically, the reduced- χ^2 in a statistically relevant fit is about equal to the degrees of freedom in that fit, that is, $\chi_r^2 = \chi^2/\nu \approx 1$. The degrees of freedom are given by $\nu = n_{\text{indep}} - n_p$, where n_{indep} is the number of independent data points in a given data set, and n_p is the number of parameters varied in the fitting model. One then determines the error on the best estimate for a given parameter p_{best} by fixing p at $p_{\text{best}} + \sigma_p$, at which point χ^2 will be increased from its minimum value by one degree of freedom, that is $\chi_{1\sigma}^2 = \nu + 1 = \chi_{\text{min}}^2 + 1$.

The difficulty in EXAFS analysis with determining errors has been that there is no easy way to determine the appropriate data errors σ_d in order to accurately calculate $\chi^2 = \sum (y_i - Y_i)^2 / \sigma_d^2$. The reason for this deficiency is that EXAFS data are not limited in their accuracy by the total number of photons collected (“The EXAFS technique is not count-rate limited” is the usual phraseology). That is not to say that one cannot collect an EXAFS spectrum that is count-rate limited, but rather that it is possible to collect a spectrum where the errors between the data and the fit, even in model systems such as a Cu foil, are larger than those given by counting statistics. Even if all other sources of experimental error could be brought under control, discrepancies still exist between the calculations of the scattering lineshapes, such as those provided by FEFF, and real data.

RSXAP has several options for estimating the correct statistical-2. Each involves using Stern's rule for determining the number of independent data points $n_{\text{indep}} = 2\Delta k \Delta r / \pi + 2$. This estimate is important for determining the true χ^2 , since in the sum $\sum (y_i - Y_i)^2$, the y_i are assumed to be independent data points. However, in an r-space spectrum with, say, 400 data points (200 in the real part and 200 in the imaginary part of the Fourier transform within the fitting range), one will achieve a much larger estimate of χ^2 if one sums over all 400 only partially independent data points! Therefore we first normalize χ^2 by N/n_{indep} , where N is the total number of data points in the fit range. This normalization is the same as used in Frenkel et al., PRB **49**, 11662 (1994).

RSFIT References

RSFIT uses a SIMPLEX-like algorithm for least-squares minimization called STEPIT, written by J. P. Chandler and distributed by the Quantum Chemistry Exchange Program.

STEPIT obtains a best-fit solution by searching the statistical-2 hyperspace, rather than using an analytical solution of the EXAFS equation. Therefore, techniques that involve the local curvature in the statistical-2 cannot be used. The proof of the method of freezing a given parameter and searching for the point at which $\chi^2 \sim 1$, such as used by Frenkel et al., PRB **49**, 11662 (1994), has been given in R. A. Arndt and M. H. MacGregor, "Nucleon-nucleon phase shift analyses by Chi-squared minimization," in Methods in Computational Physics: Advances in Research and Application (Academic Press, New York), v. 6, pp. 253-296. It is also in P. R. Bevington, Data Reduction and Error Analysis for the Physical Sciences (McGraw-Hill, New York) 1969, p.243.