

Quantum simulation program instructions

Introduction

Model:

This program follows the evolution of a system of coupled spins, using a quantum mechanical density matrix method. Each spin included has a location in space based on the local coordinate system, a spin and a nuclear gyromagnetic ratio. Spin 1 is usually taken to be the muon and starts with 100% polarisation, while all the others are unpolarised.

Options are: to detect the polarisation of any spin, to give any other spin the initial 100% polarisation, or to set the initial polarisations based on Boltzmann statistics and their individual Zeeman splittings. The program can therefore simulate a number of cases in NMR, NQR and ESR in addition to muon spin resonance/relaxation.

Spatial or waveform information, which would for example enable hyperfine constants to be calculated, is not modelled, it is assumed that such parameters are known, calculated by other means, or will be fitted from experimental data.

Coordinate system

This is fixed relative to the molecule/crystal lattice. Applied fields and muon polarisation can then be at any angle.

Units, unless otherwise defined, are:

- Time: microseconds
- Field: Gauss
- Angle: degrees
- Position: Angstroms
- Frequency (RF, hyperfine splitting, etc): MHz (but frequencies of components in output files are Mrad/s, ie. $*2\pi$)
- Transition rate: μs^{-1}
- Gyromagnetic ratio: MHz/T
- Temperature: K (also Activation energies)
- Polarisation: from -1 to 1, not %.

Internal interactions:

- Dipole-dipole interaction between spins, based on their relative locations in space and dipole moments. As special cases there is no coupling if two atoms are at the same location, or between the electron and others (the dipolar part of the hyperfine coupling can be used to give an equivalent effect).
- Hyperfine interaction between one spin labelled "electron" and each of the others, divided into parts:
 - Isotropic
 - Dipolar, along the indicated easy axis of the "other" spin

- Non-axial, about two perpendicular axes.

The electron is usually given $S=1/2$ and the other spins are usually assumed to be the muon and nuclei, though with care an exchange coupling J between atomic moments can be modelled using this term.

- Quadrupolar interaction for any spin with $I > 1/2$, with principal axis as specified, ν_Q and η . (η along the same perpendicular axes as for HFC). With care this can be used to model crystal field splitting for atomic moments too.

External influences:

- Zeeman interaction of each spin with the external static magnetic field. Additionally each spin can have its own random, static field included, to allow for next-nearest nuclei. The electron can have its g-tensor specified (6 components).
- RF magnetic fields, also as Zeeman interaction, with arbitrary frequency, amplitude and phase, and either linear or circular polarisation. "Red-Green" measurements are modelled by repeating the calculation without and with RF applied. The frequency can be automatically set to each of the splittings between eigenvalues, to track the resonances, measuring their amplitudes and sensitivity to B1 fields.
- Spin flips of any spin (by some external interaction), at a specified rate.
- Site changes: The muon can be considered to occupy one of several possible "sites". The coordinates, gyromagnetic ratios, hyperfine constants, quadrupole moments and easy axis direction can be specified for each site. Transition rates between each pair of sites, and the initial population of any site, can be specified. For ionisation and capture the electron can be given zero hyperfine constant in the ionised site, and can optionally have its spin flipped on return to represent capture of a new electron from the conduction band.

Currently RF cannot be used in conjunction with spin flips or site changes.

Averaging:

Monte Carlo averaging can cover all possible angles for an external field (powder sample) and all phases for the RF field. Transverse field axes for TF and RF measurements are chosen relative to the random "longitudinal" axis.

Alternatives are a uniform distribution of field directions, giving a noise-free plot at the expense of some artefacts if using a low number of averages, using specified field and muon axes for a single crystal simulation, or a special-case zero field powder result.

Results:

The method gives the resulting muon polarisation as a numerical sum of sine/cosine terms. For the spin flip or site change option these may be exponentially damped with individual relaxation rates. The individual coefficients can be saved direct to a file or used to build up a frequency spectrum.

Usually the polarisation is evaluated numerically as a function of time, then used to plot out:

- average asymmetry over a specified time interval
- integral counting asymmetry (specify muon lifetime and start time)
- fit a selection of functions to it and plot the fit parameters such as relaxation rate or frequency shift
- RMS amplitude or second moment of a TF spectrum

Some of the plotted values use the eigenvalues directly, instead of the calculated $P(t)$.

Sweeps:

The above simulation can be performed as a function of:

- field magnitude
- field angle
- RF frequency
- Spin flip rate
- Site change rate
- quadrupole splitting

or as a 2D plot varying two of these parameters in some combinations.

Fitting:

The simulation can be used to model experimental data, either in the form of one or more $P(t)$ curves or as fitted asymmetry/frequency/relaxation rate versus field. Most input parameters can be varied during the fit. Spin flip and site change rates can be fitted directly, or calculated from a temperature specified for each data point using variable activation energy and attempt rate.

The asymmetry and background can also be varied, and an extra "normal" muonium component can be added.

Plots:

During calculation each $P(t)$ curve is plotted, overlaid on each other. If a relaxation function is fitted, this is overlaid on the plot. On completion of a fit the experimental data and fit curve are shown. At completion the result plots e.g. asymmetry versus field for each simulation are plotted. This plot can be printed when the program exits. All curves from the plots can be saved to files for plotting or further uses.

Output:

The program can output a log of all parameters used in the simulation or fit.

All files are plain text format with extension ".dat" by default. Existing files will never be overwritten.

Installation

The program as compiled here runs on Windows XP and may be expected to run on other Windows operating systems or emulators.

The following files are required:

quantum_win4.exe – the main program, which may be in any suitable directory.
DFORRT.DLL (the FORTRAN run time library), PGOLEDriver.DLL (the PGPLOT run time library), PGPLOT.LIB, PGOLEDRIVER.LIB, GRFONT.DAT and RGB.TXT (the PGPLOT data files). These should all be in the same directory as the main program.

(As an alternative, the DLLs can be in the Windows system directory and the PGPLOT files can be in C:\PGPLOT.)

musrfunctions.dll – this is optional, only required if linking to Wimda. Put in the Wimda directory where Wimda will be able to load it.

Usage:

Start the program (double click quantum_win4.exe). It will open:

- Console window for text output (check here for diagnostic or error messages if something seems wrong, also shows progress of simulation or fit)
- PGPLOT window for graphs (title PGOLEDriver)
- Main dialog-box type window for parameter entry.

Note the graph within the PGPLOT window is not maximised, you should probably do this before proceeding further. Resizing the plot once drawn just stretches the bitmap image, it is only replotted at the new resolution if the program redraws it, e.g. on starting a new simulation.

Main window:

The screenshot shows the 'Enter Quantum Parameters' dialog box with several sections highlighted and labeled:

- Size of problem:** Points to the 'Num of Spins' (3) and 'Num of charge states or sites' (1) fields.
- Select charge state to edit:** Points to the 'State 1' dropdown menu.
- Model:** Points to the table of parameters for each charge state.
- Experimental measurement:** Points to the 'Calculation type' section.
- Monte-Carlo settings:** Points to the 'Averaging' section.
- Select fitting mode:** Points to the 'Field' and 'Activated transitions?' options.
- RF:** Points to the 'RF parameters' section.
- Sweep:** Points to the 'Sweep' section.
- Control buttons:** Points to the 'OK', 'Save Setup', 'Reset Scale', 'Clear Plot', and 'Cancel' buttons.

The table in the 'Model' section is as follows:

State 1	position	spin	easy axis	Electron?	hyperfine splitting	quadrupole	Spin flip rate	Random field
X	Y	Z	I	MHz/T	x	y	z	
0	0	0	1/2	135.5	1	0	0	4460
0	0	0	1/2	-2802	1	0	0	0
0	0	0	1	16	1	0	0	20

On the top line: enter the number of spins (1-6) and number of charge states (1-6).

Next block: enter the model.

The window shows the coordinates etc. for one of the charge states at a time. Use the up/down controls at the top left to switch between charge states, if more than one is selected. The button "Copy to Others" copies the values for this charge state to all the other states, for example if there are only small differences between them such as orientation.

- Coordinates in Å (molecule/crystal reference frame) used to calculate dipolar coupling. These can all be left at zero if only hyperfine interactions are relevant.
- Spin I and gyromagnetic ratio (MHz/T) for each particle.
- Easy axis for each particle, used for hyperfine and quadrupole interactions.
- Select which spin if any is the Electron (enables hyperfine coupling).
- Hyperfine splittings A, D, non-axial in MHz. The second and third axes required for the non-axial term are not entered directly. One axis is perpendicular to both the easy axis and y-axis (or easy axis and x, if too close to y), and the 3rd mutually perpendicular axis calculated from these. The 3rd is the one with positive contribution from E.

Note on definitions: with the easy axis set along z (2nd axis along x, 3rd along y), entering hyperfine constants of A, D and E gives values along the principal axes $A_x=A-D/2-E/2$, $A_y=A-D/2+E/2$, $A_z=A+D$

- Quadrupole splitting (ν_Q and η) (again note axes for η)
- Spin flip rate (μs^{-1}) if using the "hopping" method, and the polarisation to be applied to "flipped" spins
- Random field. The magnitude of the number gives the width of the field distribution, sign of entry 1 indicates +=Gaussian or -=Lorentzian distribution for all spins. Negative values for subsequent spins mean keep the same value for that spin. For 2nd and subsequent charge states, the value is the multiplier from the first charge state – e.g. -1 means the random field changes sign between sites.
- g-tensor for the Electron (six components xx, yy, zz, xy, xz, yz) which default to an isotropic $g=2$ state. The gyromagnetic ratio for the electron should always be entered as the "free electron" value, ie. 28025 MHz/T.

Next block: "experimental measurement"

Maximum time in microseconds to evaluate and plot $P(t)$, and the number of time bins (default 1000, up to 10000 is allowed). Choose according to frequencies expected in data, note the plot is "filtered" to prevent high frequencies in the analytical result appearing as spurious effects when sampled at low resolution. Using too many time bins can slow down the calculation.

Calculation method to use (plain, RF or hopping). The RF mode has two variants. One is intended for high RF frequencies much greater than the plotted $P(t)$, and ideally a whole number of periods per plotted time bin, although it can evaluate "RRF" results. The other is for low frequencies where the effects within the RF cycle are of interest or the induced precession signal must be plotted out directly. Both can be slow if the RF frequency is much lower than the frequencies in the Hamiltonian (hyperfine constants or electron Zeeman splitting for example).

"Integration": type and parameters, eg:

- average $P(t)$ over a time window (from, to)
- integral counting (muon lifetime, initial "dead" time)
- RMS asymmetry over window (from,to)
- various fits to Gaussian or Lorentzian functions, giving relaxation rates, TF frequency shift or asymmetry. Parameters are the muon lifetime (for weighting) and number of Mevents for simulated data (0 for infinitely good data, otherwise simulated "statistical noise" is added before fitting). Note these modes cannot be used for fitting to data values as the fit subroutine is not re-entrant.
- similar results using dominant terms from the analytic solution for $P(t)$ (which can be used for data fitting).
- frequency shift or asymmetry of the dominant term for $P(t)$ in a given frequency window, also second moment and apparent electron polarisation from asymmetry of a pair of sidebands.

Frequencies are plotted relative to the muon's Larmor frequency, also frequency windows can be moved if B is to be swept.

Some measurements are only implemented for certain calculation types.

"Monte Carlo" type and parameters:

Possible orientation averaging modes are:

- Monte Carlo averaging, number of steps
- Uniformly distributed averaging, number of steps. For both of these, either LF (muon parallel to B) or TF (muon perp to B) may be chosen. For transverse orientation the phase angle between the initial muon polarisation and detector (as rotated about B) can be set. (Phase zero is in the direction of the initial polarisation and positive phase values increase in the direction of precession of a diamagnetic muon, in positive fields.) Uniform averaging with 1 orientation can be used where all the interactions in the system are isotropic, e.g. to plot repolarisation or $\Delta m=0$ level crossing lines. Uniform averaging has equal spaced points in $\cos(\theta)$ so when simulating a system with axial symmetry, this is best placed along the z-axis.
- Single Crystal specified axes for B_0 , B_1 , muon initial polarisation and detector – specify directions relative to the crystal coordinates as "direction cosines" x,y,z in the three boxes. The values will be normalised by the program so "1 1 1" is valid and will reappear as "0.577 0.577 0.577" next time.
- Powder average (zero field only)
- List of orientations taken from a file. This contains three sets of direction cosines for B_0 , B_1 , and polarisation (the detector is set parallel to the initial polarisation) and the hyperfine constants A and D for spin 1. A "file open" box to select the file will appear before simulation starts. For fitting, the file must already be loaded into memory, by running the same model in simulation mode.

Polarisation types are:

- "100% muon" – along the specified or randomly chosen axis, with all the other spins initially unpolarised
- "With muon": The system starts in thermal equilibrium, so polarisation may be determined by Zeeman splitting and the B_0 axis. This is only useful for RF or hopping calculations.
- "Before muon": the system with the muon removed (but including other hyperfine and quadrupole splittings) is in thermal equilibrium, then the 100% polarised muon is added.

For the last two, you need to enter a temperature. A negative temperature for "Before muon" equally populates all states within kT of the ground state, while a positive value uses the usual Boltzmann statistics. Zero temperature populates the ground state only. For the "hopping" calculation type, charge state/site no. 1 is used to calculate the initial polarisation for all sites, while in the other modes each possible site is polarised independently.

This mode can be used to model quenching of electronic moments, by adding another spin to represent L, giving it a large hyperfine coupling to the electron to represent spin-orbit coupling, and a large quadrupole splitting to represent crystal fields. However it is faster to use only the electron spin and change its g-values.

RF parameters:

- Frequency in MHz.
- Phase of RF relative to start time (flag to select Monte-Carlo random phase)
- RF interaction type (as magnetic field linear or circularly polarised, change of A, change of D). Note change of D is an extra component aligned along the z-axis, not the easy axis specified or "B₁" axis and only applies to spin 1. B₁-fields are applied to every spin.)
- RF interaction strength (G or MHz)
- Flag for Red-Green subtraction (runs calculations twice, one with no RF, and subtracts P(t).)
- RRF measurement at applied frequency (n=1) or a harmonic of it, and the phase relative to the applied RF signal. n=0 and ph=90 for standard measurements.

Sweep:

Possible sweep modes are:

1st coordinate	2nd coordinate
<i>(Sweeps including angle – use single crystal mode)</i>	
Field angle theta	-
Field x-component	Field z-component
Field angle theta	Field angle phi
Field magnitude	Field angle theta
RF Frequency	Field angle theta
<i>(Sweeps of one variable only)</i>	
Field	-
RF Frequency	-
Spin flip rate	-
Temperature	-
Quadrupole splitting(s)	-
Hyperfine constant A for spin 3	-
Site change rate (relative)	-
Site change rate	-
RF B1 field	-
<i>(2-dimensional sweeps)</i>	
Field	RF Frequency
Field	Site change rate (relative)
Field	Site change rate
Field	Flip rate
Field	Temperature
Site change 1→2	Site change 2→1

(Not all sweep modes are appropriate for, or work in all calculation methods)

Theta is the angle in the z-x plane. All orientation sweeps require "single crystal" orientation to be selected. With the "TF" measurement types and sweeping theta only,

only the field B_0 is varied, the muon polarisation and B_1 are unaffected. The sweep theta/phi mode rotates everything.
Range to sweep for each axis: enter "from", "to" and either "step" or "number of points" (the other one is calculated).
Flag for log scale - if set, "step" is the ratio between successive points on the sweep.

Fitting, etc:

Enter the static magnetic field (if not used as a sweep variable).
Selecting "Activated Transitions" enables transition rates to depend on the specified temperature, mostly useful for fits.
Tick "Fitting" to enable fitting mode, fit parameters and data are specified later.

Buttons

"OK" to proceed with simulation or fitting. (If any combination of options is not valid, you get a pop-up box to say so and the main window re-opens.)
"Save Setup" to save the last used parameters to a file (note, ignores most changes just made in the window)
"Reset Scale" pops up two boxes for new min/max values (x then y) in which the defaults are calculated from the last plot - then replots the results based on this scale.
"Clear plot" removes all but the last plot from the graph, but does not re-scale it.
"Cancel" exits the program, first offers to "print" plots in window: this actually generates a Postscript file "pgplot.ps".

After "OK":

If "Activated" mode is selected and the number of sites is >1 then three windows appear where the power of T, activation energy and attempt rate can be entered for each transition. Tick the boxes in the first window to select which transition rates will be set in this way, and to request automatic calculation of the initial population based on the rates at that temperature.

Actual transition rates

Change from this...

	1	2	3	4	5	6
1	0.4722222 <input type="checkbox"/> Flip	<input type="checkbox"/> Flip	<input checked="" type="checkbox"/> Flip	<input type="checkbox"/> Flip	<input type="checkbox"/> Flip	<input type="checkbox"/> Flip
2	<input type="checkbox"/> Flip	0.1666666 <input type="checkbox"/> Flip	<input type="checkbox"/> Flip	<input type="checkbox"/> Flip	<input type="checkbox"/> Flip	<input type="checkbox"/> Flip
3	<input type="checkbox"/> Flip	<input type="checkbox"/> Flip	0.3611111 <input type="checkbox"/> Flip	<input type="checkbox"/> Flip	<input type="checkbox"/> Flip	<input type="checkbox"/> Flip
4	<input type="checkbox"/> Flip	<input type="checkbox"/> Flip	<input type="checkbox"/> Flip	<input type="checkbox"/> Flip	<input type="checkbox"/> Flip	<input type="checkbox"/> Flip
5	<input type="checkbox"/> Flip	<input type="checkbox"/> Flip	<input type="checkbox"/> Flip	<input type="checkbox"/> Flip	<input type="checkbox"/> Flip	<input type="checkbox"/> Flip
6	<input type="checkbox"/> Flip	<input type="checkbox"/> Flip	<input type="checkbox"/> Flip	<input type="checkbox"/> Flip	<input type="checkbox"/> Flip	<input type="checkbox"/> Flip

Diagonal entries = initial probabilities. Tick boxes to flip Electron spin on state change

OK Calc Equilibrium Populations Cancel

Next, if the number of sites is >1 the Transition Rate window appears. Enter the transition rates from each site to any other in the off-diagonal elements, and the initial populations in the diagonal elements. Rates which will be set by activation energy (as above) cannot be set here. The button "Calc Equilibrium" fills in the initial populations with the equilibrium populations resulting from the specified rates.

Next to each rate is a tick box to select if the electron spin is flipped (prob. 1/2, ie. depolarised) as part of that transition.

For calculation modes other than Ionise, only the populations are specified - the program will run a single site simulation for each site and then generate the weighted average of the final measured value – this is correct for integral or average asymmetry but may not be meaningful if fitting frequency or RMS amplitude for example. For fitting raw (time-domain) data the weighted average of $P(t)$ is used.

Next depends if Fitting or simulating.

Simulating:

In RF mode there is a pop up box for "Line Hunt" mode. If "OK" selected it behaves like Red-Green:

- calculate with RF off (frequency as specified in main window: should be very large for correct operation, at least twice the maximum expected)
- find all the transitions within the given band: $f=(E_i-E_j)/h$
- if "Double Quantum" (2) chosen, also selects frequencies within the band where $2f = (E_i-E_j)/h$
- then runs a simulation for each frequency, with RF on.

Results are saved to a file as for 2d sweeps.

Dialog box asking "Print analytical results" - if "OK" is selected the analytical formula for each $P(t)$ will be printed in the console window, together with a representation of the Hamiltonian, etc. It waits for RETURN to be pressed (in the console window) after printing each Hamiltonian. Enter "s" then RETURN to save this printed Hamiltonian to a file (a standard save box pops up). The file format is a sparse representation with one line for each non zero entry giving i and j indices, and real and complex parts. The analytical formula combines terms with the same frequency, and omits those with zero amplitude.

The Hamiltonians, when printed, have 1 character per element in the matrix:

"R": real and +ve

"r": real and -ve

"I": imaginary, +ve

"i": imaginary, -ve

"c": complex, any sign.

Space or dot: zero. Dotted background pattern aids in tracing rows or columns. The order of rows is such that the state of the first listed spin changes most rapidly.

Save box for Breit-Rabi plot: eigenvalues (as columns) versus sweep parameter (column 1). If a file is given, it will ask for a second file to save some information about the eigenvectors at each end of the sweep (for each level, as in the order in the file, gives the projection of each spin along the field axis, and their correlations). Not available with Hopping where energy levels are not well defined, and not offered if averaging over orientations. Energy levels in RF mode are modulo the applied frequency (the intercept on the $f=0$ axis, extrapolated from a small frequency range can be used to estimate the true value, if not too close to a resonance where energy is not a well defined quantity). The program attempts to match up the "same" level down each column by comparing the eigenvectors but sometimes this fails, especially near avoided level crossings. With a sufficiently fine step size the levels avoid each other and with a coarse step they may cross.

Save box for analytical results (not offered if too many Monte Carlo averages in use). The resulting file has columns for the first sweep variable, amplitude as $\cos()$ and $\sin()$ parts, frequency, and relaxation rate if appropriate. Each simulated orientation will give one line per pair of eigenvalues plus an extra "zero frequency" term – there will be the same number of results each time though they will not always be in the same order, especially if levels are degenerate or cross. (This differs from "Print analytical results".)

Save box for binned frequency spectrum (useful if averaging over orientations). This is a table: x-axis (top) = frequency with 500 points (fixed), y-axis (left) = sweep parameter, z=amplitude in that bin. Pop-up box allows entry of minimum and maximum frequencies of interest.

Save box for fit parameters (if one of the fitting-to- $P(t)$ measurements in use)

Dialog box for "Save P(t)" - if selected, will give a Save box for each P(t) plot as it is generated. "Cancel" those not wanted.

Save box for 2D results (if a 2D sweep type used). Must give it a filename. Values saved are the two sweep variables and the measurement (e.g. average P(t))

Load box for any additional interactions to add (expressed as an arbitrary matrix). Click "Cancel" if not required. The file format is the same as that saved after printing a Hamiltonian. It is assumed to be Hermitian so only one of each off-diagonal pair need be given (if the second is also given it overwrites both), and all diagonal elements must be real. Elements not specified are set to zero. In the RF modes extra columns in the file can specify a matrix to add to the RF interaction: real and imaginary parts for the sin() and cos() terms at the RF frequency, and then optionally for the second harmonic as well. The latter are needed for modelling magic angle spinning etc.

Then the program starts simulation. Progress box appears. Plots of P(t) appear (overlaid) in the plot window. With 2D plots, changes plot colour for each new value of the second variable. If a relaxation function is fitted to each curve, the fit curve is overlaid on the plot to check the quality of the fit.

Click "cancel" on the progress box to stop the calculation early - but it will only stop at the end of an individual sweep point calculation.

Once complete: "Cancel" in the progress box changes to "OK". Click it.

Then little dialog box "Calculations complete". Click "OK".

Then the final "measured vs. sweep" plot appears in the window, overlaid on previous ones (those for which Keep Curve was selected, below). For 2D sweeps only the last "row" of the sweep is plotted here.

Save dialog box for this curve.

"Keep curve" box. Select "OK" to keep in subsequent plots for comparison.

Then returns to main window.

Fitting:

There are 2 normal modes for fitting:

- fit to raw P(t) for one or more runs at different fields (and/or temperatures, implying spin flip rate or site change rate set via activation energy)
- fit to values of integral asymmetry, relaxation rate, frequency shift etc for several runs at different fields etc

The mode is selected by the type of control file opened (see later).

Box to enter initial values for asymmetry and background (and B^{*2} alpha shift term). Mainly for P(t) mode or integral or average asymmetry. Otherwise use asymmetry=1.0 and background=0.0. Put alpha shift term to zero unless required.

Box to enter the asymmetry and hyperfine constant of an extra normal, isotropic muonium fraction acting as another "background" (in addition to any in the simulation). Set to 0 if not used.

Main variable selection and initial value box:

Parameter type Indexes Initial value Constraints

Parameter type	Index 1	Index 2	Value	Min	Constraints Max	Accuracy
0 - Baseline asymmetry	1	1	0	0	0.5	0.01
1 - Relaxing asymmetry	1	1	1	0	1	0.01
a - Hyperfine const (spin.chg)	1	1	22.218414	0	0	0
a - Hyperfine const (spin.chg)	3	1	16.811242	0	0	0
d - Hyp. Anisotropy (spin.chg)	3	1	5E-002	0	0	0
h - log(Hop rate) (spin.chg)	3	1	12.03	0	0	0
(none)				0	0	0
(none)				0	0	0
(none)				0	0	0
(none)				0	0	0

One row for each variable parameter (up to 10 allowed):

left column: drop down boxes to select fit variable types:

next 2 columns: index variables i,j where required. Otherwise set 1,1

next column: starting value.

last 3 columns: minimum, maximum and accuracy for constraints.

Some options use a "reduced" value eg. log(conversion rate) or sign(log(hyperfine const * scale), hyperfine const) to reduce the range of values seen by the fit algorithm. This processed value is used here, the actual value is reported later.

Changing the selection fills in the initial value box from the value in the main simulation settings.

Constraints: set the accuracy to zero to disable the constraint. Otherwise the parameter is required to be within the bounds, or at least not beyond them by more than approximately the accuracy. (Effectively increases chi squared as the parameter goes beyond the bounds). Uses "reduced" values as in initial value column. The initial value must be within the bounds $\pm 1 * \text{accuracy}$, otherwise the upper or lower bound is adjusted to include the initial value.

Variable types:

- "(none)" – for unused slots in the table (such a blank entry between other active entries will be removed and the others moved up to close the gap, once fitting starts)

- "0 - Baseline asymmetry", "1 - Relaxing asymmetry" – as entered in the previous box, mainly for P(t) or integral asymmetry (repolarisation) fits. Multiplies result of simulation by asym, then adds baseline.
- "2 - Field zero offset", "3 - Field scale factor" – allow magnetic field to vary a bit from the x-values in the fit file.
- "4 - Muonium fraction", "5 - Muonium log hfc", "6 - B**2 shift of alpha" – as in dialog box.
- "a - Hyperfine const (spin, chg)" – signed log value.
- "A - lin Hyp const (spin, chg)" – direct value. For both these options j=0 means all charge states set, j=-1 means states 2-N only
- "d - Hyp. Anisotropy (spin, chg)" – as a multiple of A for this spin/charge
- "D - lin Hyp. Aniso (spin, chg)" – direct
- "n - Hyp. Non-axial (spin, chg)" – as a multiple of A
- "N - lin Hyp. Non-ax (spin, chg)" – direct. For all these, j=0: all charge states. j=-1: charge states 2..N. If i=electron: all other spins (multiplying by individual A values where relevant)
- "q - Quadrupole (spin, chg)" – splitting MHz, linear
- "h - log(Hop rate) (spin, chg)" – spin flip rate. i=0: all spins, j=0: all charge states for that spin. i=j=0 is not allowed.
- "i - pop or log(ion rate) (from, to)" – if i=j, sets (linear) initial populations. One population must remain non zero and not be used explicitly as a fit parameter and will be adjusted to give total=1.0. If i and j are different, sets log conversion rate. i=0 or j=0 allowed as above.
- "e - log(ion)(from, to)+eqm pop" – if used, initial populations are in equilibrium based on specified transition rates. For both these, i=0 or j=0 means all rates from/to that state that are already non zero. i=j=0 means every rate that is non zero.
- "t - Activated ion power of T", "u - Activated ion log(energy/K)", "v - Activated ion log nu0" – generates conversion rates from Arrhenius type formula $v = v_0 T^n \exp(-E_a/kT)$ where T is specified for each run in the fit data file. Allows i or j=0 to mean every transition.
- "b - RF B1" – linear, may mean A1 (MHz) etc.
- "r - Random field (spin, chg)" – in "signed log" format.
- "x - Coord (spin, chg)", "y - Coord (spin, chg)", "z - Coord (spin, chg)" - allow spins to move around and vary dipolar coupling. i=0 or j=0: all spins or charge states. Sign of each coordinate preserved so allowing symmetrical +x, -x movement.
- "p - Detector phase angle (deg)" – Detector phase, for transverse measurements.

Note that the variables are applied in the order given therefore e(0,0) followed by e(1,2) allows all rates to vary together, except the rate 1 to 2 which is independent. The equilibrium calculation of initial populations, or assigning the remaining population to a state not labelled as varying, is always done last.

Click OK when finished.

Next: "Open" box for file to fit. The file must have columns in of one of the formats below, with no header:

- "sel B T y err"
 - sel – a letter giving the averaging type, as in the main window drop down box (must be compatible with the start/end/lifetime values in main window but need not be the same for every data point)
 - B - field for simulation
 - T - temperature, for activated rates
 - y, err - observed values
- "sel B y err"
 - as above with no explicit temperature dependence
- "sel B T file"
 - file = name of data file for each run. This must contain time, asymmetry and error columns. Will skip over non-numeric header lines in this data file so output from CONVERT_ASCII or Wimda plot window "Save Data" is acceptable. These run files must be located in the same directory as the fit file (no path may be specified).

The maximum number of data points that can be fitted: 200 fitted values or raw files, 20000 time bins fitted in total if fitting raw data. It may be beneficial to bunch the raw data or reduce the time range fitted.

If raw data fitting selected, a dialog box pops up giving the option to truncate the time range used in the fit (normally all points in the range covered by the simulation will be included). As the time values are interpolated from the evaluated curve, fitting to long time spectra containing high frequencies will require that the number of bins is set to a high value (in the main window). It may be an advantage to adjust the number of bins so the bin widths of simulation and experiment match.

The fit now starts.

A Progress box displays the number of iterations and χ^2 of the latest iteration. (No estimate of finishing time is possible). Click "cancel" to stop the fitting after the next calculation and return the best fit parameters so far (with no error bars). While fitting, the raw P(t) curves are plotted (but with no correction for Asymmetry or Background).

After fitting the fit parameters are printed to the console window, and there's a "Save" box to allow them to be saved. The main simulation parameters are left set to these values so immediately running with "fitting" unselected will generate the best fit curves.

If fitting to average asymmetry etc, the data points (with error bars) and fit values are plotted as a function of field.

Special "Wimda" DLL-fit mode

This uses a DLL (usually musrfunctions.dll) to allow another program to request simulation data.

The DLL can be located anywhere and have any name - this is for the convenience of the other program, eg. Wimda. A suitable DLL is provided with this program.

You can start Wimda and Quantum in any order, but do not open Wimda's "Analyse" window yet. (If you are using a different program which is statically linked against the DLL, it should not be started at this stage)

Proceed through Fitting as above, select the fit parameters to use. Note: options for Asymmetry, background, B² correction, extra muonium, field zero offset and Arrhenius activation are not available. Select "Field scale factor" to allow Wimda to set or vary the field. Any values not chosen as fit parameters keep the values set in the main window. A maximum of 4 should be used with Wimda in "fit P(t)" mode – the "Analyse" window.

When the "Open" box appears, either select the .dll file (if in an unusual location) or click "Cancel" to use the default "C:\Program Files\Wimda\musrfunctions.dll".

A "progress" box appears which shows the number of evaluations performed.

Now open Wimda's "Analyse" window and select the fit function "Quantum" which should be in the list. Its parameters are named appropriately as selected above. You can use the same function for more than one component, with independent parameters, or accompany it with one of the standard functions.

The function can be used either as "Oscillating" or "Relaxing" though it will usually model both effects. When used as "Oscillating", any Detector Phase parameter is taken out and used specially: fit in "All Groups" mode and there is a separate Phase for each group.

When Wimda has finished fitting and you have closed its "Analyse" window, click "Cancel" on the progress window to regain control. Clicking "Cancel" early will make the DLL return $P(t)=0$ to Wimda which will confuse it if a fit was in progress.

Details of the Wimda DLL

The DLL exports two sets of functions. For Quantum, it provides:

subroutine Qlinkinit(name,descr,parnames(*),pardesc(*), npars, ntimebins, dt) – initialise the set of fit parameters to be used.

integer function Qgetcalc(pars(*),timebins(*),callbackfn) – does the "work". pars(*) is an array to be used for parameters, and timebins(*) for the returned P(t) in steps of dt as specified above. Repeatedly calls the subroutine "callbackfn" (as passed in by Quantum) with no parameters, to request a calculation. Eventually returns the number of calculations performed.

subroutine Qunlink – called when finished, un-registers the function.

For Wimda, the functions are:

subroutine GetOscFnList(n,fnam)

subroutine GetRelFnList(n,fnam) – these return the list of user-defined functions which will have 1 entry with a name as given in Qlinkinit().

character function OscFnDetails(n)

character function RelFnDetails(n) – these return the function description.

subroutine GetOscFn(n,np,pn,siz,fu)

subroutine GetRelFn(n,np,pn,siz,fu) – these return the number of parameters for a function, the parameter names, and a pointer to a function (fu) to actually evaluate it.

The P(t) function is defined as

double precision fn(t,p1,p2,p3,p4)

where t is the time and p1..p4 are parameter values. For the oscillating functions p4 is always the phase.

The DLL stores evaluated curves for P(t) for recently used parameter values and interpolates these curves to get a value to return to Wimda. The time range set in the main window must be not less than the time range over which Wimda will try to fit data. Evaluating points beyond this will return the value at the end of the curve, similarly evaluating for $t < 0$ will return the first value, usually close to 1. Fitting to long time spectra with high frequencies requires a high number of time bins to be selected.