

CONTENTS

1. Specification	
1.1 Overview	2
1.2 History	2
1.3 File formats	3
1.4 Hardware requirements	3
2. Software and Installation	
2.1 Software	4
2.2 Installation	4
3. Experimental File Format	
3.1 Structure of input files	6
3.2 Converting electronic files from EPR spectra to *.csv format	6
4. Getting Started with PIP4Win	
4.1 Reading in a filenm.csv file	8
4.2 Simulating a simple spectrum (g-anisotropy)	8
4.3 Simulating a spectrum with g-anisotropy and hyperfine coupling	12
4.4 Simulating a spectrum comprising two species ($S = \frac{1}{2}$ and $S = 1$)	13
5. Useful Tips	
5.1 Zooming in	16
5.2 'Graining'	16
5.3 Linewidth effects	16
5.4 Parameter saving	16
5.5 More complex spectra	17
6. Feedback	18
Appendix	19

1. Specification

1.1. Overview

This program is essentially an interface for the Fortran program PIP written by Mark Nilges at the University of Illinois EPR Centre and it is important that any citations should include a reference to that program (as well as this one if you insist). PIP itself is substantially more versatile and powerful than this interface currently permits, e.g. PIP accommodates non-coincidence of the g-tensor and A-tensor matrices, whilst this interface assumes that g and A tensors are coincident. If you find that you have something more exotic than the current interface permits, then I have included some notes in the Appendix on the fixed file format required for PIP; you can write your own PIP files and run the program directly without the interface.

1.2 History

The developments in PIP4Win have primarily been driven by my own needs to simulate anisotropic EPR spectra. Thus version 1.0 of PIP4Win was designed to provide an interface for PIP for simulating spectra of $S = \frac{1}{2}$ radicals and was written in Qbasic in the early 1990's and actually ran in DOS mode (for those who can remember DOS!). It then received a major overhaul around 2008 when I transferred it to Visual Basic (v.1.1) and ran in a format similar to the one implemented now with the exception that (i) it would still only deal with $S = \frac{1}{2}$ and (ii) it had a delightful option to import spectra as bitmap files reflecting the fact that the ER200D spectrometer I used only produced analog output. Simulations could then be run 'over the top' of the imported image! More recently we began to study triplet systems ($S=1$) and I have updated (2011) the current version of PIP4Win to:

- Incorporate the capability of PIP to simulate EPR spectra for species with $S > \frac{1}{2}$ with zero field splitting parameters, $|D|$ and $|E|$;
- Simulate more than one component simultaneously so as to accommodate two species or low abundance nuclei ('satellites');
- Show the orientation dependence of g and the largest of the hyperfine coupling constants for one component to assist simulation and interpretation;
- Read EPR spectra in comma delimited (*.csv) format. (I have removed the option to read scanned bitmap images as redundant – though if there is demand it would not be difficult to reinstate this option.)
- Include a basic (linear) baseline correction and difference function for improving 'the fit'.

This is very much a beta-test version for $S > \frac{1}{2}$ ions and I would welcome comments about whether (a) it works and please report any specific bugs or errors you may encounter; and (b) if you have demands for additional functionality which are not currently included.

1.3 File Formats

The file format for both the observed and simulated spectra are written in comma delimited format (filenm.csv) to facilitate portability to spreadsheets such as Excel. This immediately raises an issue with regard to how to import 'real spectra' which are not typically in *.csv format. A description of how to transform real spectra into *.csv files is described later. However the advantage of this format is that the file sizes are small and readily imported into other software programs so that you can manipulate the way your spectra and simulations are displayed for publication, thesis, presentations etc. I shall, however, aim to include some basic capability in future versions to generate graphic files directly so that they can be inserted directly into other documents.

1.4 Hardware Requirements

All development has been undertaken on a PC running Windows XP 2002, service pack 3 with an Intel Core2 CPU T7100 @ 1.8GHz and 2 GB of RAM. The software appears to also run on Windows 7 without any issues (although with an unexpected change in colour of one of the graphics windows!).

The spectral simulation is undertaken by computing a series of 'single crystal' spectra at different orientations on a sphere defined by θ (rotation from z to the xy plane) and ϕ (rotation in the xy plane). Thus a nominal 20 steps in θ and 20 steps in ϕ requires the calculation of 400 (20 x 20) spectra. In many cases, especially with narrower linewidths or large g -anisotropy, more steps are required to provide a reasonable sampling of the powder spectrum to avoid 'graining'. Thus an increase from (20 x 20) to (40 x 40) spectra leads to a four-fold increase in computation time and calculation of 1600 spectra takes a couple of seconds, so some patience is needed to simulate anisotropic spectra.

Jeremy Rawson
jmrawson@uwindsor.ca

2. Software and Installation

2.1. The Software

The software comprises the following files:

a) Documentation:

Pip4Win.pdf (this file; with information on installation and running the software)

b) Installation file:

pip_setup.inf (essential start up information for the program)

c) The graphical user interface software:

PIP4Win.exe (this is the interface application)
pip4win.ico (poor excuse for an icon – sorry!)
pip4windows.res

d) The original Fortran 77 code:

PIP.exe (courtesy of M. Nilges, University of Illinois)

e) Files and programs to pass data to and remotely run the original Fortran code:

Interfacing to the Fortran code has proved particularly problematic, and functions which appear to work well in development stages did not always function well when fully compiled. The current approach is not particularly elegant but appears functional (the GUI shells to a second process which initiates the batch file which controls PIP!) Future versions will aim to streamline this process, though are not likely to lead to particularly noticeable improvements in time.

trial.dat	(data input file for pip)
trial.out	(output file from pip)
pip.bat	(batch file to run pip)
trial	(file to control the batch process)
runner	(routine to start the batch file)

The additional files (triala.dat, triala.out, pipa.bat, triala and runnera) are essentially identical to those above but control the simulation of the second spectrum when two spectra are considered.

2.2 Installation

Copy the PIP4Win folder onto your computer hard-drive where you wish to run the software from and paste all the files into it. This can be on your desktop or you can squirrel it away elsewhere on your harddrive and put a shortcut on your desktop or Start Menu.

Use notepad to edit pip_setup.inf; change the second line of that file to correspond to the location of the folder you have just created which contains the PIP4Win software.

Close pip_setup.inf with the changes made. It is essential that you copy pip_setup.inf to the root directory (C:\).

It is essential that you place pip_setup.inf in the root directory C:\ as PIP4Win is hard-coded to look for pip_setup.inf in C:\ in order to identify the location of the PIP4Win files on *your* computer. This does however provide you with the flexibility to store the software wherever you want – you can even put it on a second drive (D:\ or E:\ if desired).

3. Experimental File format

3.1 Structure of Input files

The current version of this software only reads comma delimited file formats (*.csv). The first two lines comprise the operating microwave frequency (in MHz) and a descriptor of the remaining file format, i.e. Field, Intensity. The first few lines of a typical file are presented below.

```
9.75692
Field, Intensity
3300.0, 1.2
3300.8, 1.6
3301.6, 0.7
.....
```

The software will accomodate up to 10,000 data points, though the simulation program is limited to 2048 points and averaging of data is undertaken when there are more than 2048 data points. This may potentially lead to some loss of resolution, though it is rare for anisotropic spectra to have sufficiently narrow line widths that this becomes a significant problem.

Several example files are included in the PIP4Win folder cunningly entitled “Examples”.

3.2 Converting electronic files from EPR spectra to *.csv format

Bruker *.spc and Jeol files can be converted to *.csv format via the freeware Winsim which can be downloaded from the following site:

<http://www.niehs.nih.gov/research/resources/software/tools/index.cfm>

Utilise the old version (Winsim.zip) rather than the new version (EPR-WinSim.zip); whilst the new version is much nicer for simulating isotropic spectra it does not have the capability to copy the raw data.

- a) Start Winsim
- b) Open your existing EPR spectrum from File\Open\Spectrum etc.
You're likely to want a Bruker OS/9 (*.spc) rather than *.lmb format.
- c) Once your spectrum is displayed on the screen, choose Edit/Copy Data from the pull-down menu.
- d) Now open Excel (or other spreadsheet s/ware) and paste your copied data into the worksheet
- e) Save your worksheet as filenm.csv (in *.csv format).
- f) Edit filenm.csv and add two lines to the top of the file:
The first line should be the microwave frequency for your experiment e.g. 9.45067. If you have a Bruker file then the filenm.par should contain a record of this provided your microwave counter was on (look for the line beginning MF).
The second line should contain the words "field, intensity"

If you have new EMX file formats which aren't read properly in the old version of winsim you can read them in the new version of Winsim, use 'Export as...' to save in the old *.lmb format which can then be read into the old version of Winsim!

If I can work out how to decode the file format for Bruker *.spc files then I'll include this in future versions of PIP4Win.

Whilst you might find this a bit tedious, it doesn't usually take very long in relation to the time spent trying to simulate your anisotropic spectra!

You should now be ready to use PIP4Win.

4. Getting Started in PIP4Win

The current version of PIP4Win has a daunting array of boxes, buttons and windows. If I can think of a good way to tidy it up, I'll incorporate it into a future version. A diagram outlining the main regions of the simulation 'form' is shown on the next page. The easiest way to find your way around is probably to try a simulation or two.

4.1 Reading in a filenm.csv file.

Click on the button which says "Load csv data" in the Spectra box on the left hand side. A window pops up which allows you to find and import your spectrum. Once you've selected it the program will automatically (i) display the spectrum in the window (original spectra are represented in blue); (ii) update the microwave frequency, centre field and sweep width parameters as well as the number of data points recorded, all of which are necessary for subsequent spectral simulation.

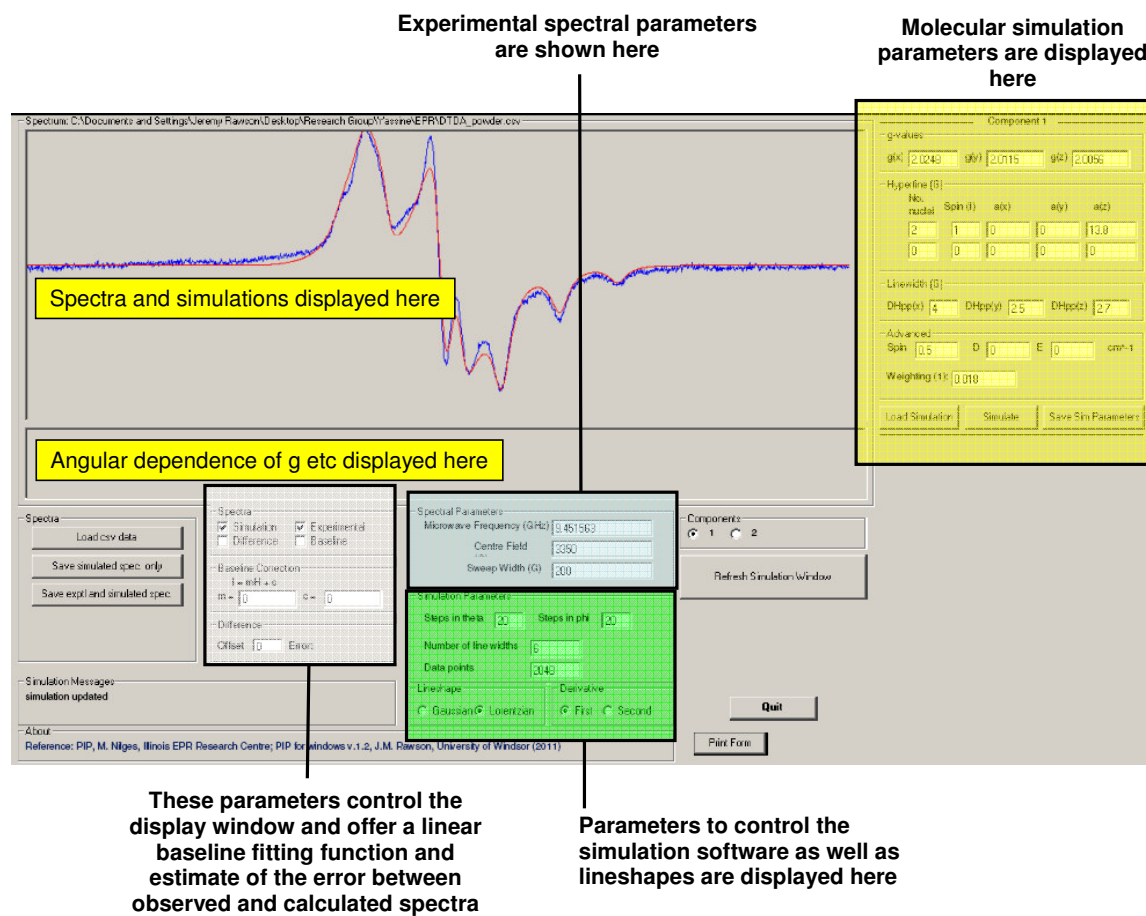
Note: should your spectrum have more than 4096 data, then the spectrum will be reprocessed so that the number of data employed matches with the number which the simulation software will handle.

4.2 Simulating a simple spectrum (g-anisotropy)

We'll start with a simple example of a copper (II) complex, $[p\text{-ClC}_6\text{H}_4\text{NH}_3]_2[\text{CuCl}_4]$. This compound adopts a layer-like structure in the solid state with chloride bridges between Cu^{II} ions. Because of the close proximity of other $S = 1/2$ ions in the lattice, there is a lot of dipolar broadening which inhibits the observation of any hyperfine coupling to Cu (both ^{63}Cu and ^{65}Cu have $I = 3/2$). As a consequence we just need to interpret the anisotropy in the g-tensor.

Open the following file, powder20a.csv.

At first glance the spectrum looks approximately axial $g_x = g_y \neq g_z$



Enter some initial numbers for the g-values in the box in the top right hand corner. [Values in the region 1.80 to 2.40 would be appropriate starting points]. Keep two of the numbers the same for now (for an axial type spectrum) and press the **Simulate** button.

Under the main spectrum the simulation will immediately show you the angular dependence of g for the values you provided. This angular dependence is shown from g_z to g_x and from g_z to g_y .

The full simulation takes a bit longer to run. Depending on how many steps you have in q and f, this is typically 2 – 10 seconds. The program does not automatically update the spectrum and you will need to hit the **Refresh Simulation Window** button to see if the calculation has yet been completed:

- (i) If no spectrum appears then Ensure that the Spectra simulation box is checked and then **press the ‘Refresh Simulation Window’ button.**
- (ii) If an ‘unexpected’ spectrum appears which does not appear to make sense in relation to the input parameters (usually when you first start and there is an old simulation resident), then it may be that the spectrum is still being simulated. Wait a couple of seconds and press **Refresh Simulation Window** again and see if something more appropriate now appears!

Hopefully by now you have a simulated axial spectrum. Simulated spectra are shown in red. Now continue to adjust the g-values to more appropriate values, by iteratively changing the g-values pressing **Simulate** and **Refresh**.

Now your peaks are in roughly the right place you might also wish to change the weighting which just scales the intensity of the spectrum to better match what is observed. Adjust **Weighting(1)** to provide a better intensity match. The weighting is not part of the simulation program so you **don’t** need to **Simulate** the spectrum again; just adjust the weighting and **Refresh**.

At this stage you’ll notice that your linewidths are not quite right. You should try adjusting (i) Lorentzian vs Gaussian lineshape and (ii) linewidth. Note that the Lorentzian lineshape is more ‘pointy’ and the linewidth is defined as half-width at half height whereas the Gaussian function is more rounded and its width is defined as a ‘peak-to-peak’ function. When changing from Gaussian to Lorentzian you’ll need to change the linewidths. As these change the shape of the simulation you have to **Simulate** and **Refresh** each time to update the effect of these parameters.

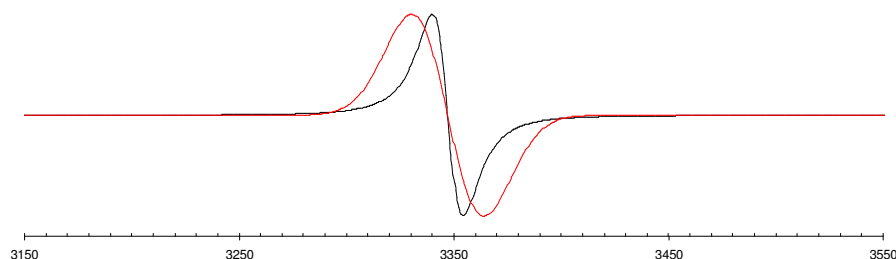


Figure. 2 Gaussian (red) and Lorentzian (black) lineshapes, both using a 12G linewidth.

The Gaussian and Lorentzian linewidths are approximately related by a factor of 2 with Lorentzian linewidths being \sim half as wide as a Gaussian for the same value of ΔH . Most spectra should be Lorentzian though you may find Gaussian behaviour in some circumstances, especially where there is unresolved coupling.

By now you should have a reasonable fit, by eye. However you'll get to the stage where its difficult to see whether there is a marked improvement. In many cases you may be happy with the closeness of the fit as it stands. However, if you want to try to improve your fit further, check the difference box and **Refresh** your spectrum. You can now see the difference between observed and calculated spectra (shown as a dotted black spectrum). In addition the quality of your fit is reflected numerically in a difference parameter. Generally a pretty good fit is reflected in a difference value less than 0.01.

You can add a small baseline correction and make further adjustments to your values to minimise the difference term. Include a small deviation of the g-tensors from axial anisotropy (i.e. $g_x \neq g_y \neq g_z$) might also improve matters at this latter stage.

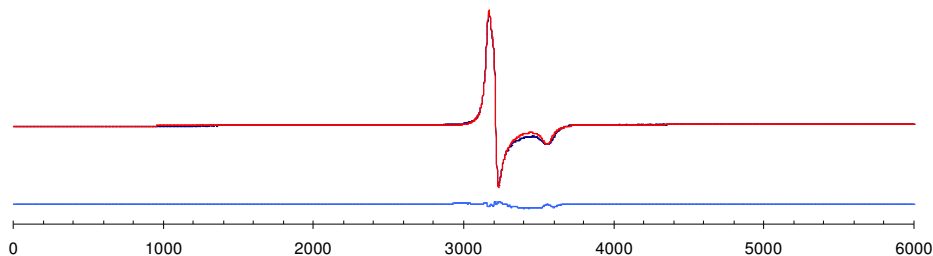


Figure 3. Simulation of the EPR spectrum powder20a.csv (difference is shown in blue); $g_x = 2.212$, $g_y = 2.178$, $g_z = 1.968$; Lorentzian linewidths $\Delta H_{(x)} = 23$ G; $\Delta H_{(y)} = 18$ G; $\Delta H_{(z)} = 44$ G; weighting 2.32; baseline correction ($m = -0.0068$, $c = -18$); Difference = 0.0048.

Once you're happy with your spectrum you'll probably want to save it somewhere:

- You can save the simulation parameters by clicking on the **Save Sim Parameters** button. A window will open up and you can store the data as `filenm.fit` or `filenm.sim` (there is no difference between these formats).
- If you want to save the actual spectrum and simulation together as a `filenm.csv` file, click on **Save Exptl and Simulated Spec** and again save your file through a normal window. This can be readily imported into Excel to personalise your format style. Note that the Save Exptl and Simulated Spec does not record the parameters used for the simulation so make a careful note of them and/or the final `filenm.fit` file.

4.3 Simulating a spectrum with g-anisotropy and hyperfine coupling

A frozen solution of the thiazyl radical $m\text{-MeC}_6\text{H}_4\text{CNSSN}^\bullet$ provided a good frozen solution spectrum in THF at 77 K. The dilute nature of the frozen solution means that the dipolar broadening is low and narrow line-widths are obtained ($\Delta H_{\text{dip}} \propto 1/r^6$).

We expect that the average of the g-tensor principle components approximate closely to the isotropic EPR spectral parameters. For the dithiadiazolyl radical, $g_{\text{iso}} \sim 2.01$ and $a_{\text{N}} \sim 5$ G so we should use these values to assist with the simulation process. An initial examination of the spectrum suggests rhombic character with some hyperfine coupling which is most clearly seen in the high field region. We expect coupling to two equivalent N atoms and the high field pattern appears to approximate to a 1:2:3:2:1 quintet of which the first 3 peaks are quite well resolved (marked * in Figure 4), permitting a good first estimate of the g-component and hyperfine ($g_x \sim 2.006$, $a_x \sim 14$ G).

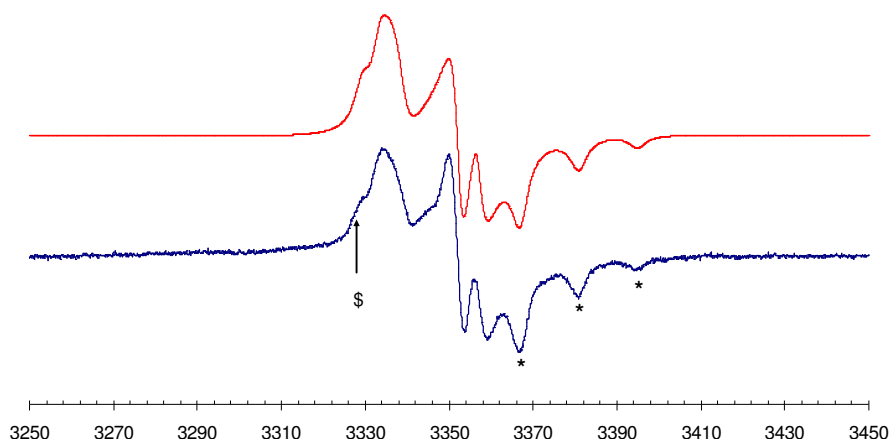


Figure 4 Simulation of the EPR spectrum of a frozen solution of $m\text{-MeC}_6\text{H}_4\text{CNSSN}^\bullet$ in THF. $g_x = 2.0057$, $g_y = 2.0115$, $g_z = 2.0245$, $a_x = 14.0$, $a_y = 1.0$, $a_z = 1.3$ G, $\Delta H_{\text{pp}}(x) = 2.5$, $\Delta H_{\text{pp}}(y) = 1.8$, $\Delta H_{\text{pp}}(z) = 2.3$ G with Lorentzian lineshape.

The approximate positions of the two remaining two g-values were estimated to fit with the residual peak positions. These were assisted by the orientation dependence of the g-factors (shown below the spectrum in PIP4Win). These clearly reveal the presence of an ‘undershoot resonance’ – marked \$ in Figure 4 – which arises through the angular dependence of g and hyperfine tensors. **This low field feature should not be ascribed to the g-value or hyperfine coupling to the low-field g-value!**

Iterative fine tuning of the hyperfine coupling constants, g-values and linewidths was undertaken to improve the difference parameter. Notably the best set of parameters has several small hyperfine couplings which are not resolved but lead to a better fit than no hyperfine coupling and slightly larger linewidths. The error in these parameters is considerable, though at least partially justified by the fact that $a_{iso} \sim 5.1\text{G}$ which cannot be met from the average of the anisotropic parameters if a_y and a_z are constrained to 0; $(14 + 0 + 0)/3 = 4.7\text{ G}$.

4.4 Simulating a spectrum comprising two species ($S = 1/2$ and $S = 1$)

In the solid state dithiadiazolyls typically dimerise to afford an EPR inactive singlet ground state configuration but may offer a thermally accessible triplet state ($S = 1$). This is not dissimilar to the behaviour of $\text{Cu}_2(\text{OAc})_4 \cdot 2\text{H}_2\text{O}$ which can be considered as a archetypal triplet dimer and was studied extensively by EPR spectroscopy by Bleaney and Bowers. The spectra are further complicated by the presence of low quantities of undimerised $S = 1/2$ lattice defects. At low temperatures only the defect $S = 1/2$ sites are observed, but at elevated temperatures thermal population of the excited state gives rise to a more complex spectrum comprising both $S = 1/2$ and $S = 1$ species.

The central resonance revealed sufficient line broadening so as to mask the hyperfine interaction and was initially modelled with equivalent g-values and a common Lorentzian linewidth. Some small modification of these values to rhombic symmetry was then employed to produce a reasonable fit to the central line shape ($\Delta H_{pp} = 12\text{G}$, Lorentzian).

The remaining features to low and high field of this resonance are associated with zero-field splitting of the $S = 1$ dimer. In order to simulate this second spectrum the number of components is increased from 1 to 2 (above the refresh simulation button). Activating a second component provides a second set of spectral features which work in an identical fashion to the first set.

In the current system the g-values for the triplet state appear very close to those observed for the monomer which is not unreasonable. For organic radicals the magnitude of the zero field splitting (ZFS) parameters $|D|$ and $|E|$ are small in relation to many transition metals (where $|D|$ can be up to 10's of cm^{-1}). Estimates of $|D|$ and $|E|$ can be made from the separation of the peaks (Fig. 5) and initially refined ‘by eye’ [The separation in Gauss should be converted to MHz ($\nu = g\beta H/h$) and then to cm^{-1}].

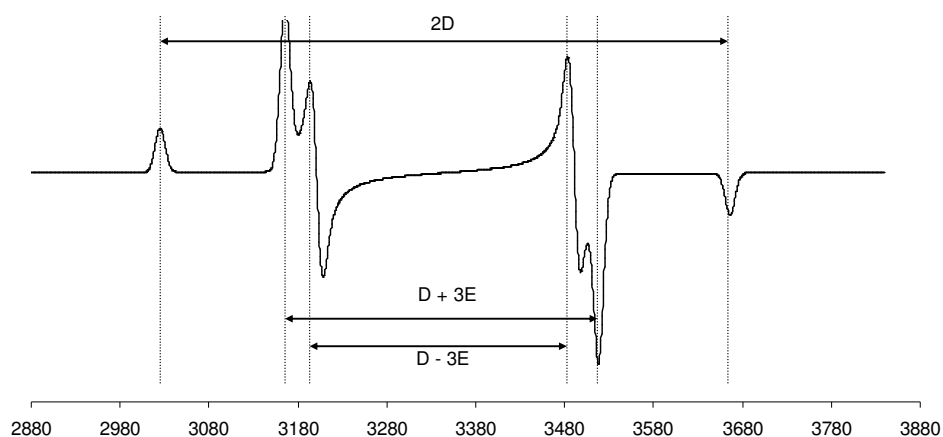


Figure 5. Form of an EPR spectrum for an $S = 1$ ion with rhombic zero field splitting ($g_x = g_y = g_z = 2.003$; $|D| = 0.03 \text{ cm}^{-1}$, $|E| = 0.001 \text{ cm}^{-1}$).

When fitting a two component spectrum the **Refresh** button displays the two separate components (a deconvoluted form) with the first component in red and the second in green. The **Combine Spectra** provides the weighted sum of the two components with simulated spectrum in red and experimental, as usual in blue.

Note: When using the difference map, the result varies depending on whether you hit **Refresh** or **Combine Spectra**. In refresh mode it looks at the difference between the first component only and the experimental data. In Combine mode, it looks at the difference between experimental and both components of the simulation. The difference between the two can be used to see the improvement in the fit of the spectrum. For the dithiadiazolyl radical described here, the difference = 0.0274 for the monomer only and reduces to 0.0084 when including the triplet component (Fig. 6).

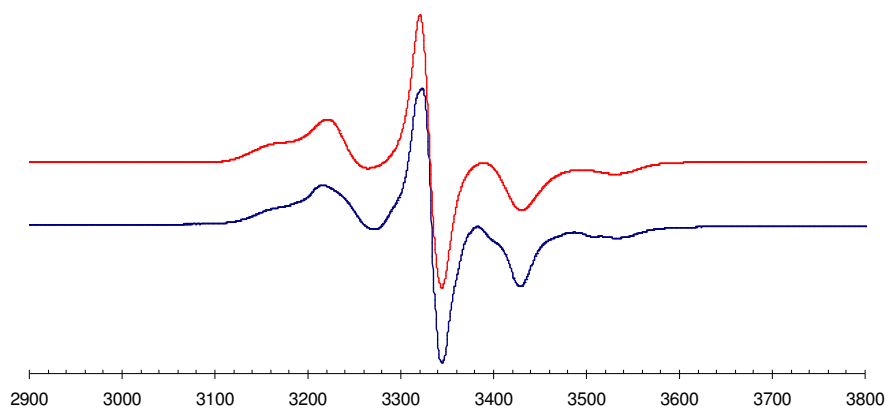


Figure 6. X-band EPR spectrum of 2,5-Cl₂C₆H₃CNSSN at 220 K; $S = \frac{1}{2}$ component $g_x = 2.018$, $g_y = 2.012$, $g_z = 2.003$, $\Delta H_{pp} = 12$ G (Lorentzian) for all components; $S = 1$ component $g_x = 2.018$, $g_y = 2.014$, $g_z = 2.003$, $\Delta H_{pp(x)} = 32$, $\Delta H_{pp(y)} = 10$, $\Delta H_{pp(z)} = 16$ G (Gaussian), $\mathbf{D} = 0.0174$, $\mathbf{E} = 0.0001$ cm⁻¹. Relative proportions ($S = \frac{1}{2}$) : ($S = 1$) = 0.70:0.22.

When writing the simulation and experimental file in this mode, the program deposits the combination of both spectra.

5. Useful Tips

5.1 ‘Zooming in’

If you need to ‘zoom in’ on part of your spectrum you can do so by changing the centre field and or sweep width of the spectrum, followed by **Refresh** (or **Combine**). However the program also uses these numbers to run the simulation so they should be switched back prior to running further simulations. I intend to improve this aspect of the program in the next release.

5.2 ‘Graining’

If your simulated spectra start to appear a bit jagged its usually because the simulation program is not sampling enough of the orientation space in θ and/or ϕ . Usually increasing θ and/or ϕ resolves this issue (Fig. 7) at the expense of slightly longer simulation times.

5.3 Linewidth effects

‘Number of linewidths’. This is normally set to 6 and corresponds to how many line widths from the centre of the resonance each spectrum is calculated at. Increasing the number of linewidths increases simulation time but as the simulation starts to provide a good fit then increasing the number of linewidths (residual intensity at the extrema of the spectral manifold) typically improves the fit a little.

5.4 Parameter Saving

I have tried so far as possible to iron out a lot of the bugs and the software does not ‘crash’ too often. However I’d recommend you save your fitting parameters from time to time in case of errors, particularly once your fit starts to be reasonable – a common one is to inadvertently put a non-numeric character in the simulation parameter boxes which is obviously not understood by the program and usually catastrophic!

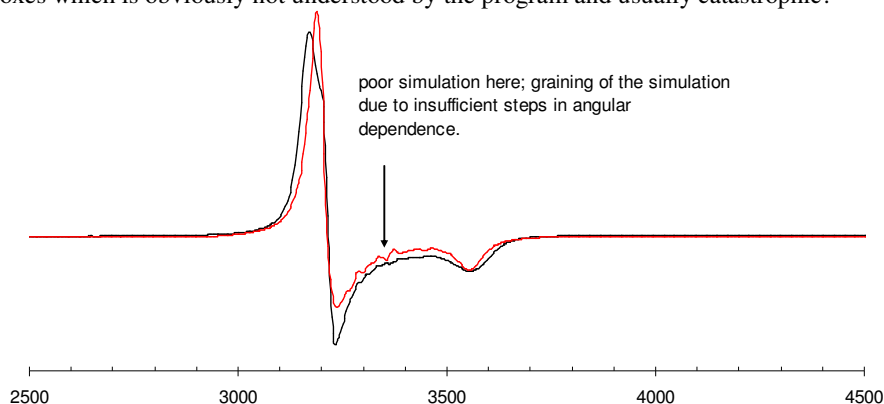


Figure 7. Axial simulation of the *m*-MeC₆H₄CNSSN[•] radical EPR spectrum using low values of θ and ϕ produces ‘mild’ graining. The resultant graining should be compared with Fig. 4. More severe graining is apparent with strongly anisotropic spectra and small values of θ , ϕ .

5.5 More complex spectra

There are certain features of PIP which are suppressed in the current version of the software. In particular this current version of PIP4Win assumes that g and A tensors are coincident. Whilst this might be the case under certain high symmetries, it is not generally so. In many cases the variation from perfectly coincident is small and the resultant errors in the simulation are therefore small. If the mis-orientation is large then you may prefer to simulate spectra using PIP directly. Instructions are included in the Appendix.

6. Feedback

Hopefully the software is relatively robust and helps you simulate at least some of your anisotropic EPR spectra satisfactorily. However I would welcome comments and feedback. Whilst no promises are made I shall try and fix reported bugs and incorporate suggestions where I can into future versions.

APPENDIX : PIP

The following information is adapted from the information originally supplied with PIP by M. Nilges.

This program generates a powder spectrum for spin $\frac{1}{2}$ or greater with one metal nucleus (calculated to full order: matrix diagonalization with up to a fourth order perturbation to convert from frequency domain to field domain) and two super-hyperfine spins for which there can be more than one equivalent nucleus. All hyperfine matrices can be rotated in any direction. The three Euler angles are read in the order α, β, γ . The Euler angles conform to the convention of Rose. Spectra are integrated using a two dimensional 4-point gauss-point integration.

Acknowledgements should be made to Mark Nilges and the Illinois EPR Research Center.

Input File format

Typical data file for a rhombic system with nuclear spin $\frac{1}{2}$, e.g. Cu^{2+} :

```

1.5      0.0      0.0      0.5      1      1
100.0    0.0      6.0
9.1047
2.02000  2.01000  2.10860
  04      01      035      6
060.00  060.00  420.00  0.0  0.0  0.0
000.00  000.00  000.00  0.0  0.0  0.0
  0.00   0.00   0.00   0.0  0.0  0.0
  0.00  000.00      0.0  0.0  0.0  0.0
  0.00   0.00      0.0  0.0  0.0  0.0
  08.0   08.0   10.0   G    1
1.4804   0.00      0    1    1    1
03100.0  0500.0   1024
```

The corresponding input parameters are:

SPINA	SPINB	SPINC	SPIND	NEB	NEC	FORMAT(4F8.1,2I8)
FACTR	FACT	CUTOFF				FORMAT(3F8.0)
NU						FORMAT(F8.0)
G(1)	G(2)	G(3)				FORMAT(3F8.0)
NTR	NTS	NTH	NPH			FORMAT(4I8)
A(1)	A(2)	A(3)	ANGSA(1)	ANGSA(2)	ANGSA(3)	FORMAT(6F8.0)
B(1)	B(2)	B(3)	ANGSB(1)	ANGSB(2)	ANGSB(3)	FORMAT(6F8.0)
C(1)	C(2)	C(3)	ANGSC(1)	ANGSC(2)	ANGSC(3)	FORMAT(6F8.0)
D	E		ANGSD(1)	ANGSD(2)	ANGSD(3)	FORMAT(2F8.0,8X,3F8.0)
QD	QE		ANGSQ(1)	ANGSQ(2)	ANGSQ(3)	FORMAT(2F8.0,8X,3F8.0)
W(1)	W(2)	W(3)	LS	IDERIV		FORMAT(F8.0,7X,A1,I8)
GN	HDIA	QR	NOP	NERR	NO	FORMAT(3F8.0,3I8)
HCR	HTOT	NTOT				FORMAT(2F8.0,I8)

And are defined overleaf:

SPINA	Nuclear spin for nuclei A, calculated to full order
SPINB	Nuclear spin for nuclei B, calculated to first order
SPINC	Nuclear spin for nuclei C, calculated to first order
SPIND	Value of the electron spin, S .
NEB	Number of equivalent nuclei for spin B
NEC	Number of equivalent nuclei for spin C
FACTR	Vertical scaling; full scale = 100.0.
FACT	Not used; set to zero.
CUTOFF	Number of linewidths from the center the lineshape is calculated
NU	Microwave frequency in GHz.
G(1)	$g(x)$
G(2)	$g(y)$
G(3)	$g(z)$
NTR	Number of transitions - nuclear spin; in order $\Delta M_I = 0, \Delta M_I = 1, \Delta M_I = 2$ etc. This ordering is valid only when the hyperfine coupling is larger than the nuclear Zeeman or nuclear quadrupole.
NTS	Number of transitions - electron spin; in order (Kramers): $\Delta M_S = -\frac{1}{2}$ to $\frac{1}{2}$; $M_S = -3/2$ TO $-1/2$, $M_S = 1/2$ TO $3/2$, $M_S = -5/2$ TO $-3/2$, etc.; (Non-Kramers): $M_S = -1$ TO 0 , $M_S = 0$ TO 1 , $M_S = -2$ TO -1 , $M_S = 1$ TO 2 , ETC.; This ordering is valid only when the electronic Zeeman is larger than the zero field splitting.
NTH	Number of integration units along the θ coordinate, each unit is four unequally weighted points.
NPH	Number of integration units along the ϕ coordinate. If NTH and NPH are too small extra peaks, known as "graining" will occur.
A(1)	First principal value of the hyperfine tensor for SPINA
A(2)	Second principal value of the hyperfine tensor SPINA
A(3)	Third principal value of the hyperfine tensor for SPINA
ANGSA(1)	Euler angle α relating the principal axes of the hyperfine tensor to g tensor
ANGSA(2)	Euler angle β
ANGSA(3)	Euler angle γ
B(1)	First principal value of the hyperfine tensor for SPINB
B(2)	Second principal value of the hyperfine tensor for SPINB
B(3)	Third principal value of the hyperfine tensor for SPINB
ANGSB(1)	Euler angle α relating the principal axes of the hyperfine tensor to g tensor
ANGSB(2)	Euler angle β
ANGSB(3)	Euler angle γ
C(1)	First principal value of the hyperfine tensor for SPINC
C(2)	Second principal value of the hyperfine tensor for SPINC
C(3)	Third principal value of the hyperfine tensor for SPINC
ANGSC(1)	Euler angle α relating the principal axes of the hyperfine tensor to g tensor
ANGSC(2)	Euler angle β
ANGSC(3)	Euler angle γ
D	Axial zero field splitting
E	Rhombic zero field splitting
ANGSD(1)	Euler angle α relating the principal axes of the zero field tensor to the g tensor
ANGSD(2)	Euler angle β
ANGSD(3)	Euler angle γ
QD	Axial nuclear quadrupole coupling constant
QE	Rhombic nuclear quadrupole coupling constant
ANGSQ(1)	Euler angle α relating the principal axes of the nuclear quadrupole tensor to the g tensor
ANGSQ(2)	Euler angle β
ANGSQ(3)	Euler angle γ
W(1)	Linewidth along $g(x)$
W(2)	Linewidth along $g(y)$
W(3)	Linewidth along $g(z)$

LS	Lineshape; L = Lorentzian with half-width-at-half-height in MHz (energy domain) G = Gaussian with the peak-to-peak linewidth in Gauss (field domain).
IDERIV	0 for absorption, 1 for first derivative, 2 for second derivative
GN	Nuclear g value
HDIA	Field at which the Hamiltonian is diagonalized; If set equal to zero, the field is auto-set to the centre of the spectrum If > zero, then the field is set to optimal field to minimize divergence of field-frequency calculation Note: if the zero-field or hyperfine splitting is of the same order of magnitude as the electronic Zeeman energy, more than one value of HDIA may be needed to properly simulate the entire spectrum and the entire spectrum need be simulated in sections.
QR	not used, set to 0
NOP	Perturbation order for field/frequency conversion: 1 to 4
NERR	Trapping level for divergence: 1 - 4; 4 - worst; only valid if NOP = 4
NO	B1 Polarization: 1=perpendicular 2=parallel
HCR	Field center in Gauss.
HTOT	Field sweep in Gauss.
NTOT	Number of points in spectrum.

Running PIP in a DOS Window.

Use Notepad or similar to edit an existing input file bearing in mind that it is a fixed format file.

Click on PIP.EXE to start the program. It will ask you for the name of the input file. If it is not in the same directory as the PIP.exe file, you will need to give it the full filename location e.g. C:\Windows\Desktop\EPRData\mydata.dat or wherever you have stored your data

After completing its simulation calculations it will ask for an output filename to write the data to.

Details of the Output file format are shown below.

Format of output data file

First line:		
NTOT	Number of points, FACT - scaling factor , PL - 0	FORMAT(I8,F8.4,F8.2)
Second line:		
HLW	Starting field (Gauss), HLH - ending field (Gauss), NU - frequency	FORMAT(2F8.1,F8.4)
For the next (NTOT/6) lines:		
Spectral data points		FORMAT(10I6)
Last 3+NSI(2+NSP) lines:		
Simulation parameters		FORMAT - Hollerith