

Dynamix

Ben Tatman

October 21st 2020

Dynamix is a program developed to fit dynamics models (see *Models*) to solid state NMR Relaxation data. This guide will go into what features are available, how to use them, how to format data, and also go into any caveats or issues with the model.

Features

As of writing, Dynamix can fit the following models;

- Simple Model Free (SMF) Fits two parameters, S^2 and τ , with spectral density given as;

$$J(\omega) = \frac{(1 - S^2)\tau}{1 + (\omega\tau)^2}$$

- Simple Model Free with Temperature Dependence (SMFT) Fits three parameters, S^2 , τ_0 and Ea , where the correlation time is temperature dependent;

$$\tau(T) = \tau_0 \exp\left(\frac{Ea}{RT}\right)$$

This is then fit as in SMF.

- Extended Model Free (EMF) Fits three parameters, S_{slow}^2 , τ_{slow} , τ_{fast} . The fast order parameter, S_{fast}^2 , is calculated as $S_{\text{dipolar}}^2/S_{\text{slow}}^2$. The spectral density is given as;

$$J(\omega) = \frac{(1 - S_f^2)\tau_f}{1 + (\omega\tau_f)^2} + \frac{S_f^2(1 - S_s^2)\tau_s}{1 + (\omega\tau_s)^2}$$

- Extended Model Free with Temperature Dependence (EMFT) Five parameter model, fitting S_{slow}^2 , $\tau_{0,\text{slow}}$, $\tau_{0,\text{fast}}$, Ea_{slow} , Ea_{fast} . S_{fast}^2 is calculated as in EMF, and the correlation times are temperature dependent as in SMFT. The spectral density is as in EMF.
- Extended Model Free without Dipolar Approximation (DEMF)^[1] Four parameter model, S_{slow}^2 , S_{fast}^2 , τ_{slow} , τ_{fast} . Spectral density as in EMF.
- Extended Model Free without Dipolar Approximation, with Temperature Dependence (DEMFT) Six parameter model, fitting S_{slow}^2 , S_{fast}^2 , $\tau_{0,\text{slow}}$, $\tau_{0,\text{fast}}$, Ea_{slow} , Ea_{fast} . Spectral density as in EMF.
- Gaussian Axial Fluctuations (GAF)¹ Eight parameter model: τ_{slow} , τ_{fast} , $\sigma_{\text{slow}}^\alpha$, $\sigma_{\text{slow}}^\beta$, $\sigma_{\text{slow}}^\gamma$, $\sigma_{\text{fast}}^\alpha$, $\sigma_{\text{fast}}^\beta$, $\sigma_{\text{fast}}^\gamma$. These axial fluctuations are then used to derive order parameters, which are then fit to the EMF spectral density function.

¹Note I'm not entirely convinced I have the sense of the rotations in `gen_build.py` correct.

$$S_{\mu\nu}^2 = \frac{4\pi}{5} \sum_{l,k,k',m,m'=-2}^2 = (-i)^{k-k'} \exp \left(-\frac{\sigma_\alpha^2(k^2 + k'^2)}{2} - \sigma_\beta^2 l^2 - \frac{\sigma_\gamma^2(m^2 + m'^2)}{2} \right) \times \\ d_{kl}^{(2)}\left(\frac{\pi}{2}\right) d_{k'l}^{(2)}\left(\frac{\pi}{2}\right) d_{mk}^{(2)}\left(\frac{\pi}{2}\right) d_{m'k'}^{(2)}\left(\frac{\pi}{2}\right) Y_{2m}(e_\mu^{pp}) Y_{2m'}^*(e_\nu^{pp})$$

Where parameters are defined as in Lienin 1998. The relaxation rates take into account dipolar contributions between N-H, N..H(rest), C-N, CA-N, C-H, C..H(rest), C-N, C-CA, as well as the anisotropic chemical shifts of nitrogen and carbon.

- Gaussian Axial Fluctuations with Temperature Dependence (GAFT) Ten parameter model. All of those in GAF, plus fast and slow activation energies. Temperature dependent time constants calculated as in SMFT, EMFT and DEMFT, angles used to calculate S^2 , then fit to EMF spectral density.
- Model Free with slow Gaussian Axial Fluctuations (EGAF) Six parameter model. Implements MF order parameter on fast time scale and GAF order parameter for slow motion.
- Model Free with slow Gaussian Axial Fluctuations with Temperature Dependence (EGAFT)

These models can be fit to ^{15}N and ^{13}C R_1 and $R_{1\rho}$ values, as well as C-C, C-H, N-H and C-N dipolar order parameters.

Passing the 'OR_VARY = 1' parameter alongside a GAF model (EGAF, EGAFT, GAF, GAFT) will transform it into a variable orientation model, in which the orientation of the axes fit to the peptide plane are allowed to vary according to three rotations, α , β , γ . This rotation is implemented as a rotation of the second order spherical harmonics in the GAF order parameter term via Wigner D matrices. In effect,

$$Y_l^{m'}(r') = \sum_{m=-l}^l l[D_{m'm}^{(l)}(R)]^* Y_l^m(r)$$

The initial orientation of the X, Y, Z axis has Z aligned along CA-CA, with the CA-N positive relative to CA-C. X is roughly parallel to the C-O bond (C->O positive) and Y is perpendicular to X and Z such that the standard X, Y, Z convention is retained (eg $Y = Z \times X$). These variable models are generally termed **VGAF**, **VGAFT**, **VEGAF**, **VEGAFT**.

For each of these models, Dynamix will perform a user specified number of optimizations with random starting points using the Nelder-Mead simplex method to find an optimum. Each optimum is output into a **residue_N.dat** file. Once complete, it will perform back calculations for each relaxation data point, outputting these into **backcalc_N.dat** files. If one of the GAF modes is used, it will calculate effective S_{NH}^2 order parameters and output these into **gaf.dat**.

If error mode is enabled, it will perform a further set of optimizations where the starting point is set to the optimized parameters. Back calculated relaxation rates are then varied within experimental error, and optimization performed. The new optimum points for each repeat in the error calculations are then used to determine standard deviations for the optimized values².

Compilation

To compile the program from source on Karplus, navigate to the directory above **src/** and run

```
gcc src/main.c -lm -pthread -o dynamix -O3
```

This will pull in all other required C files in the **src/** directory, load the math(s) and pthread libraries, using optimization level 3, and output the program into **dynamix**. Also contained in the **src/** directory is documentation (build using **doxygen** **Doxyfile** if not up to date) and test programs to verify the model against the MATLAB scripts.

²Note that the output will have the minimum optimized points and two standard deviations for the errors; it will not output the mean of the error calculations (unless you explicitly change the code to do so - at the moment this is line 408 of main.c, in which you should change **m.residues[1].parameters[i]** to **m.residues[1].errors_mean[i]**.

Data Formats

There are four data formats used to provide data to Dynamix. In all of these, beginning a line with ‘%’ marks it as a comment.

System File

The system file contains information about the model being fit, and how to fit it. It also tells Dynamix where to find all other data. Generally I use the file extension `.dx` for these, though it is not necessary.

The first portion of the system file consists of key value pairs laid out as

KEY = VALUE

The keys are all upper case, and there must be spaces on either side of the equals.

| Key | Value |
|-----------------|---|
| MODEL | Model being fit - see Features |
| S2DIP | File containing dipolar order parameters as defined below |
| CSISON | File containing isotropic chemical shifts for ^{15}N |
| CSISOC | File containing isotropic chemical shifts for ^{13}C |
| N_RESIDUES | Number of residues - must be the same as the number of lines in each input file (or bad things may happen) |
| OUTPUT | Directory (eg <code>output/</code>) to place output files into |
| N_ITER | Number of iterations for optimization |
| N_ERROR_ITER | Number of iterations to perform for error calculation |
| IGNORE | Residue to ignore; each residue to ignore should have its own line |
| OR_NH | N-H orientations |
| OR_NC | N-C orientations |
| OR_NCA | N-C α orientations |
| OR_NCSAxx/yy/zz | Nitrogen chemical shift anisotropy orientations |
| OR_CCAp | $^{13}\text{C}'$ - $^{13}\text{C}_{i-1}^{\alpha}$ orientation |
| OR_CCAc | $^{13}\text{C}'$ - $^{13}\text{C}_i^{\alpha}$ orientation |
| OR_CN | C-N orientation |
| OR_CNH | C-amide proton orientation |
| OR_CCSAxx/yy/zz | Carbon chemical shift anisotropy orientations |
| NTHREADS | Number of threads to run; generally, set to the number of processors you want to run it on |
| OR_VARY | Whether or not the orientation of the GAF axes relative to the peptide plane should be allowed to vary. |

For example

```
MODEL = EMF
S2DIP = system/s2_dipolar.csv
CSISON = system/csisoN.csv
CSISOC = system/csisoC.csv
N_RESIDUES = 56
OUTPUT = output/
N_ITER = 200
IGNORE = 42
N_ERROR_ITER = 200
% Orientations taken from Lienin 1998
OR_NH = system/or_NH.csv
OR_NC = system/or_NC.csv
OR_NCA = system/or_NCA.csv
OR_NCSAxx = system/or_NCSAxx.csv
```

```

OR_NCSAyy = system/or_NCSAyy.csv
OR_NCSAzz = system/or_NCSAzz.csv
OR_CCAp = system/or_CCAp.csv
OR_CCAc = system/or_CCAc.csv
OR_CN = system/or_CN.csv
OR_CNH = system/or_CNH.csv
OR_CCSAxx = system/or_CCSAxx.csv
OR_CCSAyy = system/or_CCSAyy.csv
OR_CCSAzz = system/or_CCSAzz.csv
NTHREADS = 4

```

This key value header should be followed by **#RELAXATION**, and then each file containing Relaxation Data should be listed below.

```

#RELAXATION
system/15N_R1_600_50_0_271.csv
system/15N_R1_600_50_0_282.csv
...

```

Relaxation Data

Each piece of relaxation data (eg a ^{15}N R_1 measurement made at 300 K in 600 MHz at 50 kHz) should be placed into an individual file. This file should begin with a header.

Note that there must be a space on either side of the equals sign

```

FIELD = {field in MHz}
WR     = {spinning frequency in Hz}
W1     = {spin lock frequency in Hz}
TEMP   = {temperature in kelvin}
TYPE   = {15NR1, 15NR1p, 13CR1, 13CR1p}

```

This defines the ‘global’ parameters for this relaxation data. Then, a line **#DATA** denotes the start of the actual relaxation data, and should be followed by the relaxation data.

```

#DATA
1 {relaxation rate in s-1} {error in s-1 (2 standard deviations)}
2 {relaxation rate in s-1} {error in s-1 (2 standard deviations)}
...

```

The program requires a line for each residue; if you have no relaxation data for one residue, denote the relaxation rate and error with -1 instead of leaving it blank. This will tell Dynamix that the data does not exist, as opposed to it just being omitted by mistake. The program may run without this, however it will not necessarily inform you that there is insufficient data for fitting which may lead to incorrect fitting and data response.

Orientation Data Formats

For the GAF models the orientation of each interaction vector related to the α , β and γ motional axis is required. In the form of Lienin 1998, this is taken as θ and ϕ angles. For each orientation vector there should be a file containing this data as;

```

1 {theta} {phi}
2 {theta} {phi}
...

```

This data is only needed for GAF fits, but all of it is needed. The orientations required are described below.

Residue Data Formats

Data such as isotropic chemical shifts and dipolar order parameters should be kept in files laid out as;

```

1 {value} {error}
2 {value} {error}
...

```

Running The Model

Once the file is setup, the model may be run as;

```
./dynamix {path to .dx file}
```

This will output the various threads being spawned as the program operates. Passing the `-e` option;

```
./dynamix {path to .dx file} -e
```

Will enable error calculation.

Visualising Results

Back Calculations

Using `gnuplot` the output of Dynamix can be quickly visualised. To view residue specific back calculated data, simply run (after entering the output directory)

```
> gnuplot
...
gnuplot> plot 'backcalc_N.dat' u 2:3 w points pt 7, x lw 3
```

This will produce a graph showing the calculated values (x) against experimental values (y), with a line of $y = x$. In order to view all back calculated points for all residues, run;

```
> cat backcalc_* > backcalc.dat
> gnuplot
...
gnuplot> plot 'backcalc.dat' u 2:3 w points pt 7, x lw 3
```

If the 'VERBOSE' key is enabled in `datatypes.c`, the back calculated files will also contain experimental information. Though beyond the scope of this brief introduction, these can be plot onto the graphs too to show where variation in relaxation rate arises (for example, I find that 700 and 850 MHz data tends to much better fit the $y = x$ line than 600 MHz data. I don't know if this is some systematic experimental difference, or just the model validity breaking down).

Visualising Parameters

In the `utils/` directory there are `gnuplot` scripts to plot data. These are set up assuming that the output directories are named as the model type (eg, an EMF model is in `emf/`). At the moment there are scripts to plot each model, with and without errors, except for the GAF models for which no scripts exist to plot errors as of yet. For non error plots, the colour of each point is related to the chisq value - darker points represent better fits.

| Script Name | Description |
|--------------|----------------------------|
| pp_smf.m | Plots SMF |
| pp_smf_e.m | Plots SMF with errorbars |
| pp_emf.m | Plots EMF |
| pp_emf_e.m | Plots EMF with errorbars |
| pp_smft.m | Plots SMFT |
| pp_smft_e.m | Plots SMFT with errorbars |
| pp_emft.m | Plots EMFT |
| pp_emft_e.m | Plots EMFT with errorbars |
| pp_demf.m | Plots DEMF |
| pp_demf_e.m | Plots DEMF with errorbars |
| pp_demft.m | Plots DEMFT |
| pp_demft_e.m | Plots DEMFT with errorbars |
| pp_egaf.m | Plots EGAF |
| pp_egaf_e.m | Plots EGAF with errorbars |
| pp_egaft.m | Plots EGAF |
| pp_egaft_e.m | Plots EGAF with errorbars |
| pp_gaf.m | Plots GAF |

| Script Name | Description |
|-------------|---------------------------|
| pp_gaf_e.m | Plots GAF with errorbars |
| pp_gaft.m | Plots GAFT |
| pp_gaft_e.m | Plots GAFT with errorbars |

These may be run as;

```
> ls
emf/   utils/
> gnuplot utils/pp_emf.m
> ls
emf/   utils/   emf.eps
```

The output is as a .eps file. This may be viewed using eg Okular (`okular file.eps`) or converted into a PDF using `epstopdf file.eps`.

There are two Python scripts, `aicbic.py` and `gen_bild.py`. `aicbic.py` is run as `python3 aicbic.py FOLDER MODEL OUTPUT` eg `python3 aicbic.py vgaf VGAF vgaf.csv`, and will calculate Akaike, Bayesian, and adjusted Akaike Information Criteria for each model. The output file has four columns; **residue**, **AIC**, **BIC**, **AICc**.

`gen_bild.py` generates ChimeraX BILD files for given models. A PDB representing the protein of interest should be present in the folder, and the path to this should be defined on line 9. Then `gen_bild.py` may be run as `python3 gen_bild.py INPUT_FILE OUTPUT_FILE OR_VARY TYPE`, where `INPUT_FILE` is the `final.dat` file generated by Dynamix, `OUTPUT_FILE` is your desired BILD file, `OR_VARY` is 1 if the model has variable orientation or 0 if not³. `TYPE` should be `fast` or `slow` depending on the GAF timescale of motions.

Output File Formats

Note: All taus and tauf values are output in nanoseconds (10^{-9} seconds), not seconds. This is done to improve precision.

residues_N.dat

These residue files are generated during the optimization process. Each line refers to a separate optimization process. The first column is the residue number, the second is the minimum value of the chisq function. Plotting the second column generates a nice plot showing how well the optimization went.

The remaining columns depend on which model is in use. In particular;

- SMF: tau (ns), S2
- SMFT: tau (ns), S2, Ea
- EMF: taus (ns), S2s, tauf (ns)
- EMFT: taus (ns), S2s, tauf (ns), Eas, Eaf
- DEMF: taus (ns), S2s, tauf (ns), S2f
- DEMFT: taus (ns), S2s, tauf (ns), S2f, Eas, Eaf
- EGAF: taus (ns), tauf (ns), sAs, sBs, sGs, S2f
- EGAF: taus (ns), tauf (ns), sAs, sBs, sGs, S2f, Eas, Eaf
- GAF: taus (ns), tauf (ns), sAs, sBs, sGs, sAf, sBf, sGf
- GAFT: taus (ns), tauf (ns), sAs, sBs, sGs, sAf, sBf, sGf, Eas, Eaf

This may be useful for plotting to see how varied the individual responses are, eg how responsive the model is to one parameter. If `OR_VARY` is enabled, the final three columns will be the α , β and γ angles.

final.dat

This contains the final fit parameters. The first column is the residue number, the second is the S2 dipolar value, then the minimum chisq value, followed by the model specific columns as above. The final column is a count of how many repeats went into error calculation (or 0 if this was not done).

errors.dat

³Note I'm not entirely convinced I have the sense of the rotations in `gen_bild.py` correct.

This file is only output if error mode is enabled. It is the same as `final.dat`, only after each parameter there is the error (2 standard deviations). eg;

```
{residue number}, {S2dipolar}, {chisq minimum}, {tau}, {tau error}, {S2}, {S2 error}
```

gaf.dat

This file is only produced for GAF models. This is laid out as;

```
{residue number}, {tau s}, {S2 slow}, {tau f}, {S2 fast}
```

Where S_{slow}^2 and S_{fast}^2 are the order parameters for the N-H dipolar interaction.

backcalc_N.dat

These files contain back calculations. They are set out;

```
{relaxation number} {calculated R} {measured R} {error in measured R}
```

Plotting {calculated R} against {measured R} can give a useful indication of goodness of fit. To gain more insight into how different experimental parameters affect the fit, enable the `VERBOSE` flag in `datatypes.c` (eg set it to 1). This will then output a further few columns;

```
{field in MHz} {spinning frequency Hz} {spin lock frequency Hz} {temperature K}
```