NEMO, version 1.0

Short description: A Matlab toolbox to optimize CPMG B_1-relaxation dispersion experiment for determination of the correlation time of fast chemical exchange

Authors: Kateřina Vágnerová, Jan Lang
Department of Low Temperature Physics, Faculty of Mathematics and Physics, Charles University in Prague, V Holesovickach 2, Prague 8, 180 00, Czech Republic
Contact to authors:
katerina.vagnerova@seznam.cz
jan.lang@mff.cuni.cz

Toolbox content:

NEMO program
NEMO_fun function
NEMO_fun_fit function
COPYING.txt GNU GENERAL PUBLIC LICENSE Version 3, 29 June 2007

Installation

To be able to run the Nemo toolbox, you must have MATLAB installed with the Optimization toolbox (particularly, lsqnonlin function), version R2008b. For installation of Nemo, unpack the archive nemo.zip to a directory of your choice. Then, add this directory in the MATLAB path list (in the MATLAB menu File/Set Path. . .).

Description of NEMO functionalities:

NEMO is a tool to optimize the chemical exchange measurement setting based on qualified guess of the exchange paramaters.

Experiment definition (for one specific temperature)

Constant and variable setting:

```
I0;
R2num;          %(s-1) R2(only DD and CSA) - estimation
A;              %A=pA*pB*omega^2(Hz)=f^2*pi^2 if pA=pB=1/2 - estimation
tauex;          %(s) - estimation of exchange correlation time
pocet_bodu;     %how many points to measure
SNR;            %Signal-to-Noise Ratio (it is possible to read from your spectra by SINO
                    order)
pocet_MCsimul;  %number of Monte Carlo measurement simulation
minimalnider;   %starting Intensity derivation lower bound for "considerable" measurement
                (recomendation 0.7)
krok_der;       %starting iteration derivation step (recomendation 0.4)
```

Measurement simulation - definition

Constant and variable setting for measurement simulation and iteration.
A possibility to simulate uncertainty in frequency and tauex estimations
(the constants marked "...s")

```
AS
tauexS
```

Fitting rutine variables definition

Initial estimation [tauex*e-3, f/100, I0] (optimization starting points)

Another constants

DO NOT CHANGE

Algorithm:

1. Intensity plane and Intensity first derivation plane calculation

2. Intensity first derivation maximum finding (only in viewed plane)

3. Intensity plane 0,3 cut-off

The 0,3 cut-off was selected by simulation and statistic. DO NOT CHANGE.

Iteration:

4. Intensity first derivation plane cut-off

Cut-off is firstly defined by user ("mez" value), later iterated.

5. Measurement points (N, Techo) setting on the Intensity plane

Intensity and N,Techo measurement points plotting
Intensity derivation and N,Techo measurement points plotting

6. N,Techo points analyzing: measurement simulation and error evaluation

Measurement noise simulation and MC simulation.

```
[lam,err_fun,residual,exitflag]=lsqnonlin('myfun_fit',lam_init,[],[],options);
%[x,resnorm,residual,exitflag] = lsqnonlin(...) returns a value
% exitflag that describes the exit condition,
% residual = functional value in x, options are conditions of
% optimalization
```

7. Next iteration step

Expanding or lowering measurement district - derivation cut-off change.

8. Iteration analysis

9. Displaying measurement recommendation

recommendation: N techo(mikros)