Basic signal processing of experimental 4-pulse DEER data (Regularization)

Author: Luis Fabregas

Reviewed by: (Pending)

A very basic example of dipolar data processing using Tikhonov regularization for a simple evaluation of experimental data.

Load the data

Let's start by extracting the primary data from the spectrometer file using `deerload`.

```
[traw,Vraw] = deerload('data/deer_broad_50K_4us');
```

The function returns the experimental X-axis as `traw` and the Y-axis as `Vraw`.

At this point it is important to check the units of the time-axis vecor `traw`. While some commercial spectrometers return the time-axis data in nanoseconds, DeerAnalysis employs microseconds as physical unit.

```
traw = traw/1000; %ns -> us
```

Now we can also define the distance-axis vector `r` of our distance distribution. A simple approach is to call the `time2dist` function which will create a distance-axis adapted to the time-axis.

```
r = time2dist(traw);
```

Pre-Processing

Before the data can be fitted it (usually) needs to undergo a set of pre-processing steps:

1. The dipolar signal is usually complex, the first step if to perform a phase correction which will minimize the imaginary component / maximize the real component.
2. In common commercial spectrometers, the time-axes are measured in absolute values. This step aims to optimally determine and correct for the zero-time of the time axis.
3. The intensity of the signal is fiven in some arbitrary units (usually some kind voltage). All functions in DeerAnalysis require the dipolar signal to be scaled such that `V(0)=1`. This last set requires a fit of the scale require to correct the Y-axis of the dipolar signal.

```
%Optimization & Correction of phase
V = correctphase(Vraw);
%Optimization & Correctionof zero-time
t = correctzerotime(V,traw);
%Optimization & Correction of Y-axis scale
V = correctscale(V,t);
```

Prepare the dipolar kernel

Since experimental dipolar signals contain a inter-molecular contribution, i.e. a background, this must be fitted and included into the dipolar kernel before the regularization.

First we proceed to fit the background function using some time-domain parametric model. In this example we will use a stretched exponential function (`td_strexp`). Using the `fitbackground` function we obtain the fitted background as well as the fitted modulation depth.

```
[B,lambda] = fitbackground(V,t,@td_strexp);
```

Now we can use these fitted variables to generate the dipolar kernel which describes our signal.

```
KB = dipolarkernel(t,r,lambda,B);
```

Tikhonov regularization

We now have all the elements required to fit our distance distribution via Tikhonov regularization. For basic data analysis with Tikhonov it is recommended to use the Akaike information criterion (AIC) for the α - selection.

```
Pfit = fitregmodel(V,KB,r,'tikhonov','aic');
```

With our fitted distance distribution `Pfit` we can forward-calculate the fit of the dipolar signal by simply computing

```
Vfit = KB*Pfit;
```