

2D double-quantum/single-quantum (SQ/DQ) correlation spectroscopy analysis in ssNake

2nd February 2018

1 Introduction

The following will explain how 2D double-quantum single-quantum NMR data can be processed in ssNake. The tutorial delivered with the ssNake program is considered as prior knowledge. If you have not yet studied this, please do so before continuing with this example.

In liquid state NMR, homonuclear correlations (e.g. ^1H - ^1H) are commonly recorded using a COSY experiment. This experiment uses the scalar couplings (J-couplings) between the different nuclei to transfer magnetization between them, and to gain information into the connectivity between the spins. Another way to do this, is to use dipolar couplings, and go for a NOESY experiment. In solid state systems, scalar couplings are often too weak for COSY experiments (as the relaxation times are much shorter), and other experiments need to be considered. A common 2D homonuclear solid state experiment is a double-quantum single-quantum experiment. In this experiment, the direct dimension is the single-quantum dimension, and the indirect dimension the double quantum.

If two spins A and B are connected (via a dipolar coupling), they will show crosspeaks in the DQSQ experiment. In the DQ dimension, A and B can have a peak at their double frequency (because of the double quantum character). If there is a sizable dipolar coupling between A and B, there will also be crosspeaks at the sum shift $\delta_A + \delta_B$. The intensity of these peaks compared to the main (diagonal) peaks gives information on the strength of the dipolar coupling.

2 Data

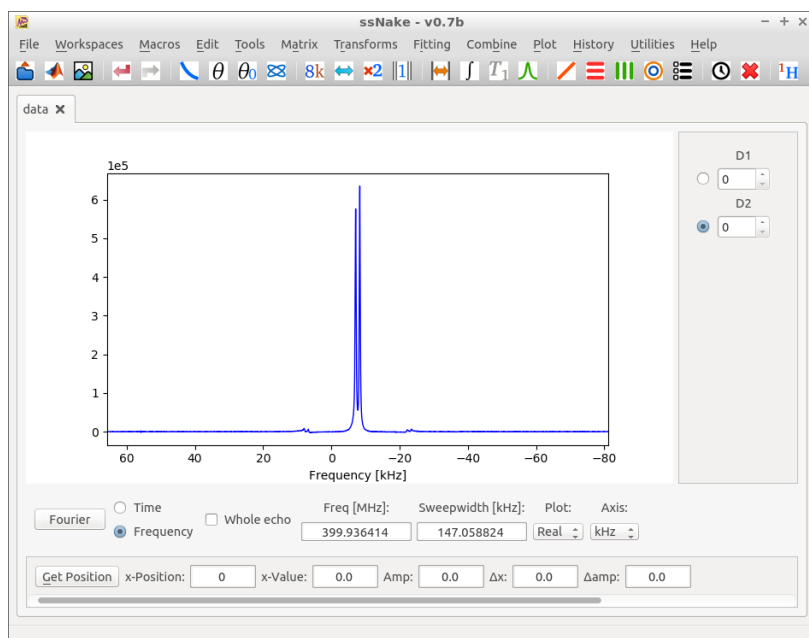
The data we will use in this tutorial is from a ^1H - ^1H DQSQ experiment on methylammonium lead iodide ($\text{CH}_3\text{NH}_3\text{PbI}_3$), a perovskite material used in a new generation of solar cells. The spectrum was recorded on a Varian 400 MHz machine, using a 3.2 mm rotor and 15 kHz MAS.

3 Processing

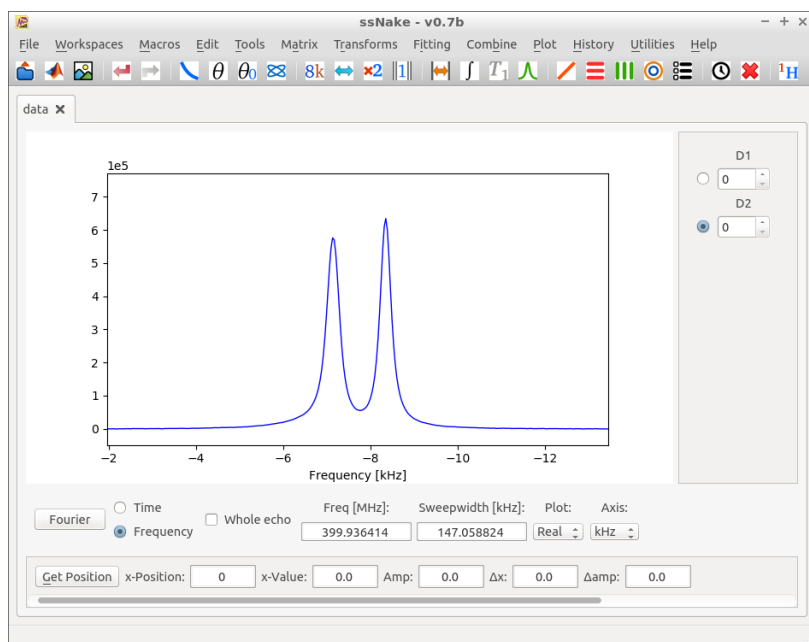
Processing a DQSQ experiment is mostly just regular 2D data processing. Only at the stage of referencing do things get a bit tricky.

- Open the data delivered with this tutorial in ssNake via Open → File
- Zero fill the data to 4096 points (Matrix → Sizing, and fill in 4096 at the size and leave the offset as is)
- Push the 'Fourier' button to go to the spectrum
- Phase the spectrum using only zero order phasing

This should give something like this:



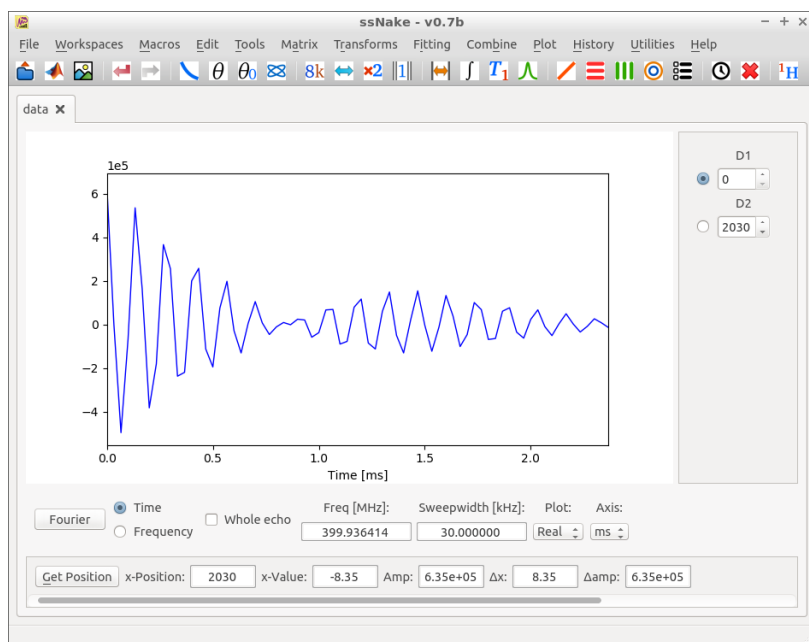
And zoomed on the main peaks:



This shows the first trace of our 2D data. We have now processed the direct dimension (D2), we now continue with the indirect dimension (D1).

- Get the position of the rightmost main peak (so that we can view it along D1). Use 'Get Position' from the bottom frame, and click on the peak. This give x-Position = 2030 (or something close to that).
- Fill in this value at the D2 box in the side frame and press 'Enter'

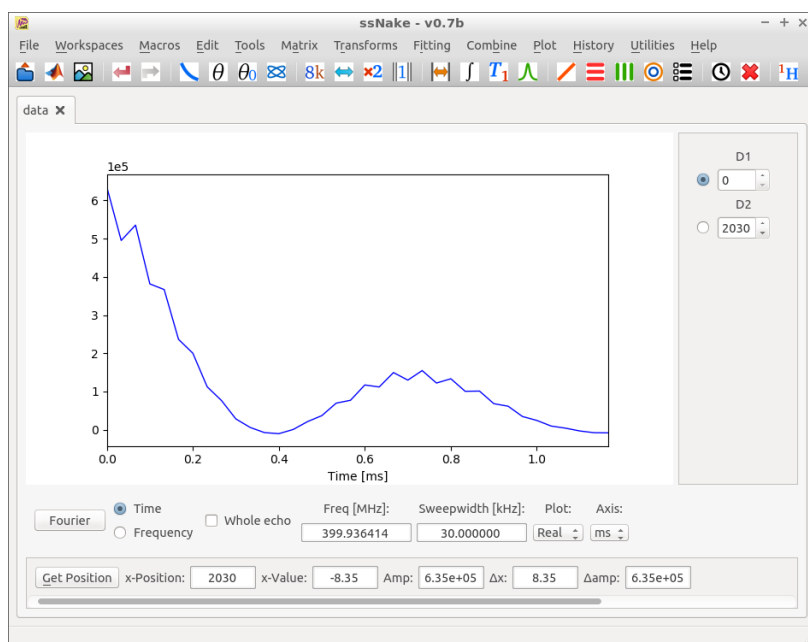
This should give:



This data was recorded with a hypercomplex scheme (states-TPPI), and requires a conversion:

- Use Tools → Hypercomplex → States-TPPI to convert it
- Use Tools → Complex conjugate to invert the sense of direction to the ssNake definition ¹

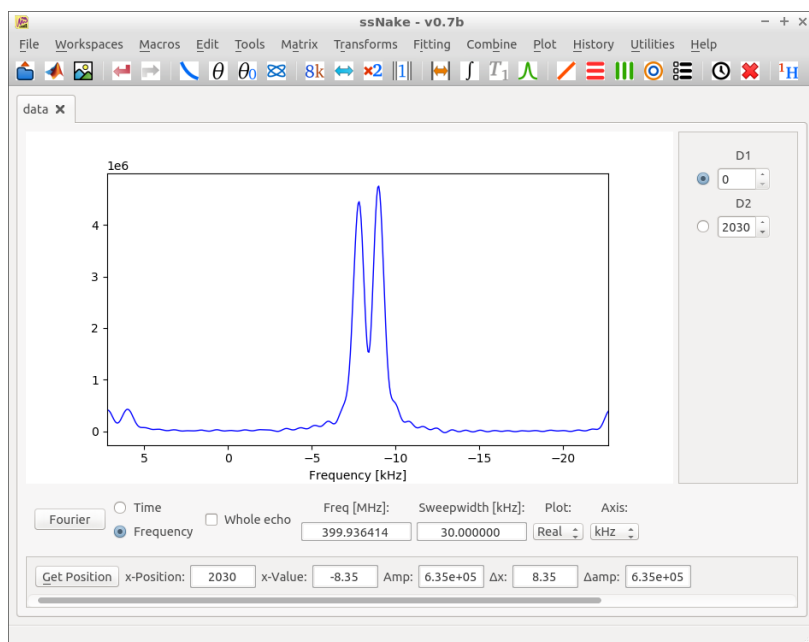
This should give:



- Zero fill to 512 points
- Fourier transform
- Phase (0 order only)

This gives:

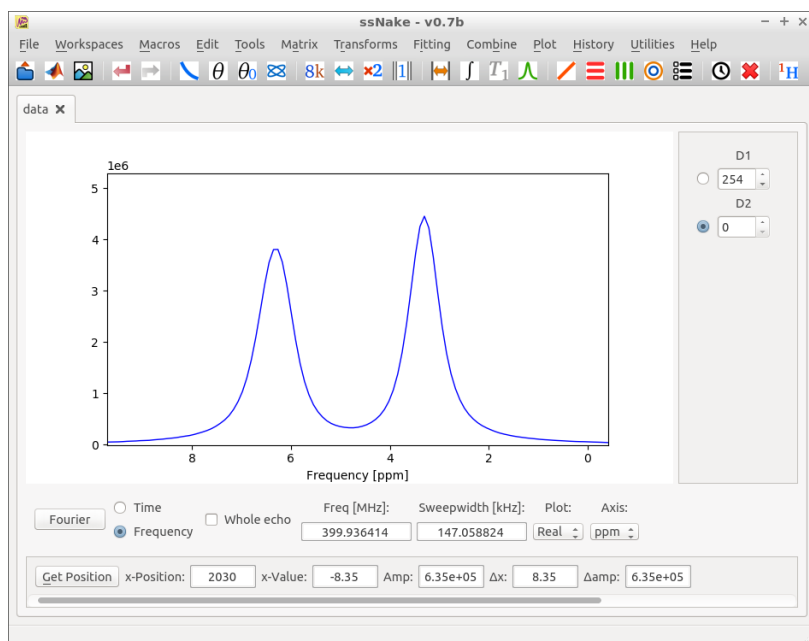
¹Most Varian sequences define the sense of the rotation in the indirect dimension in the opposite way as we do it. This is, however, dependent on the way the used pulse-sequence is written, so we cannot correct this for you automatically. Without this step, the spectrum along D1 will appear flipped.



We now need to set the correct chemical shift reference for both dimensions. The reference frequency (0 ppm) is 399.9344480 MHz (based on an external reference). We can immediately apply this to the direct dimension (D2):

- Go to D2 (use the radio button in the side frame)
- Use Plot → Reference → Set Reference, and fill in 399.9344480 for the frequency (and leave the other values unchanged).
- Set the current view to 'ppm' (in the bottom frame with 'Axis')

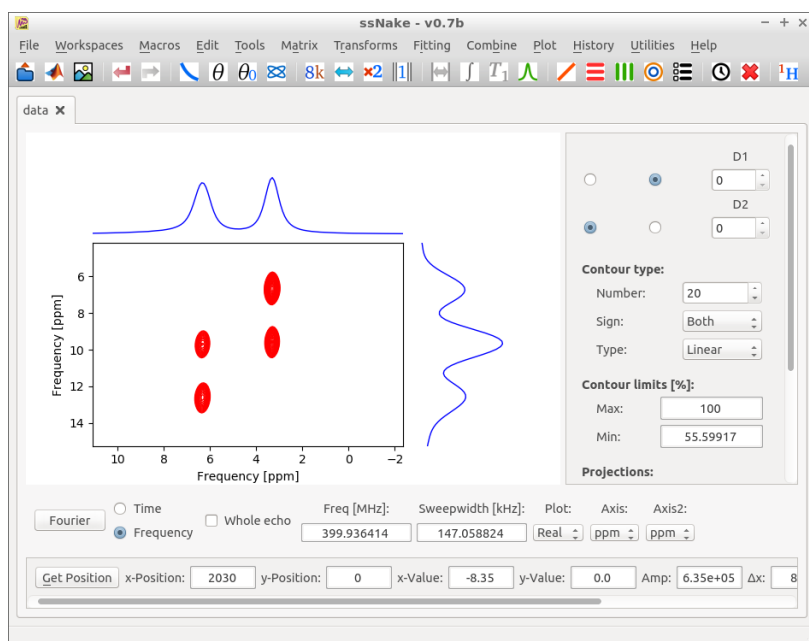
This gives:



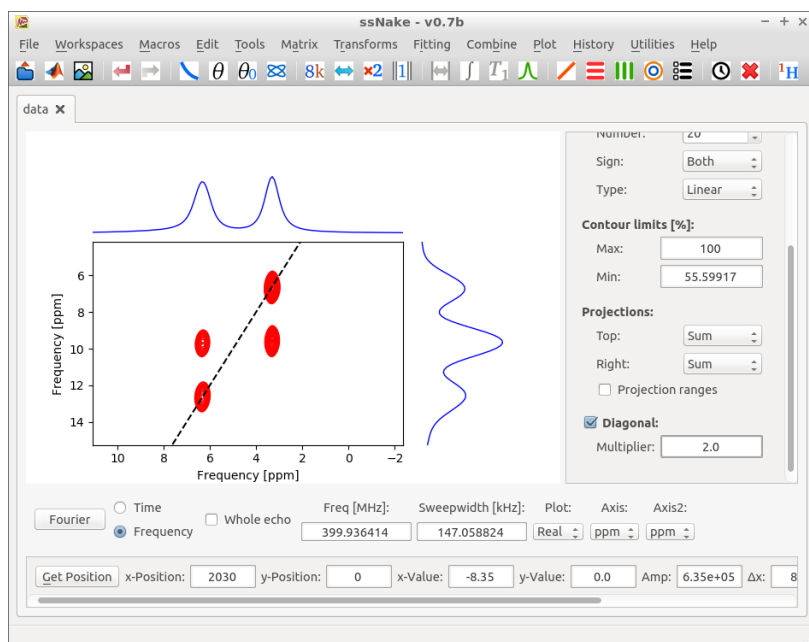
Now we continue with D1. The tricky part with referencing a double-quantum dimension is that our external reference is not correct anymore: all frequencies are doubled in the indirect dimension. This means that our zero ppm frequency (399.9344480 MHz) is different in D1: its difference with respect to the carrier frequency should be doubled. In our case, the carrier frequency (centre of the spectrum) is 399.936414 MHz (see the bottom frame). The difference with this and the reference frequency is -0.0019659 MHz. This value should be added to the reference frequency (this makes the new reference frequency to be twice as far away from the centre frequency as before). This makes the reference frequency in D1 equal to $399.9344480 - 0.0019659 = 399.9324820$ MHz.

- Go to D1 (use the radio button in the side frame)
- Use Plot → Reference → Set Reference, and fill in 399.9324820 MHz
- Set the current view to 'ppm' (in the bottom frame with 'Axis')
- Go back to D2 (radio button)

Now, we are finished with the processing, and it is time to view our 2D spectrum. Use Plot → Contour to make a contour plot. Zoomed in, this looks like (with the lower contour level at 55 %, see the side frame, contour levels can be zoomed with 'shift + scroll') :



This shows that the two peaks are connected. Both have an average peak in D1 at ≈ 10 ppm, as well as self peaks at twice there original chemical shift. These self peaks are located on the diagonal, which ssNake can also draw. In the side frame enable the 'Diagonal' and set the multiplier at 2 (we want the diagonal to go twice as fast in D1, due to the double-quantum character). This gives:



Which is the final spectrum. The pattern that is shown here is common for a DQSQ experiment, if the two chemical species are connected via a dipolar coupling. The final spectrum is also delivered with this tutorial (as a .mat file). Note that in the D2 dimension the width has been reduced to the relevant part via Matrix → Extract Part. This was done to reduced the saved data size.