

How to make a whitewashed stacked plot

Important!! If your data is NOT an array (a single data file of a series of spectra), you must combine your separate 1D data files into a single arrayed data file. Please see the end of this help sheet to see how to combine multiple data sets to a single ‘arrayed’ data set. Note that all commands are in **bold**.

The easiest way to make a whitewashed stacked plot is to start with an arrayed data set (such as a kinetics experiment using pad (**pre**acquisition **delay**)). As usual I believe the best way to describe this is with an example. Suppose we have a just performed a 50 step kinetics run.

- Load the data set in the typical fashion.
- Type **wft dssh**. This will perform a **weighted Fourier transform** on the data set and **display** all 50 spectra **stacked horizontally**.
- Suppose the peak of interest is most intense in the last spectrum acquired. Type **ds(50) vsadj dssh**. This will display spectrum #50, automatically adjust the vertical scale for all the array spectra with respect to #50, and finally display all of the spectra stacked horizontally.

Creating a whitewashed plot:

To create or display the whitewash plot, you can simply type **dsww** (**d**isplay **s**pectra **w**hitewashed). Since this is probably not exactly what you had in mind, you will probably need to adjust several parameters to ‘tune’ the final product to your liking.

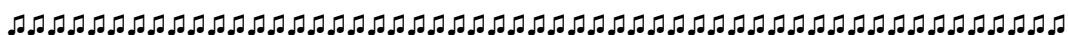
- **sc** *start of chart* from the right edge of the paper. Acceptable values are from 0mm to 250mm. To shift right-to-left, $sc=0$. To shift left-to-right, $sc = (\# \text{ of spectra} - 1)(ho)$.
- **wc** *width of chart*. This value is in mm and may range from 5 to wc_{max} (250). A rule of thumb would set wc equal $250 - (\# \text{ of spectra} - 1)(ho)$. If you would like the bottom spectrum to cover 25% of the full width type **wc=wcmax*0.25**.
- **ho** *horizontal offset* in mm. This is the horizontal offset for *each* of the spectra from the initial (or 1st displayed spectrum). Negative values will shift from left-to-right, positive values will shift from right-to-left.
- **vo** *vertical offset* in mm. Sets the vertical offset of each spectrum in a stacked display.
- **vs** *vertical scale*. In normalized mode (nm), vs is the height of the largest peak in the spectrum. In absolute intensity mode (ai), vs is the multiplier used to produce the desired vertical scale.

To display the arrayed spectra in ‘whitewash mode’ starting from the lefthand side of the page type **sc=0 wc=wcmax*.25 ho=-2.5 vo=2.5 dsww(1,50,1)**. This will set the horizontal scale to 25% of the full range, starting on the lefthand side of the page and display all 50 spectrum with horizontal and vertical offsets of 2.5mm. If you wish to only view every 4th spectrum, type **dsww(1,50,4)**.

As with most things, you’ll need to ‘play’ with **sc**, **wc**, **ho**, **vo**, and **vs** to suit your individual taste. Once you’ve decide on a suitable display it’s time to plot the data! Remember, only that data visible in the display window will make it to the plot.

Plotting the data:

- Type **pscale plww page** to plot the whitewashed stacked plot with the horizontal scale. You may also want to plot text, parameters, etc. , so you’ll need to experiment.
- More information for using the **dsww** and/or **plww** commands type **man(‘plww’)** in the Input Window.
- These plots can be ‘dumped’ to postscript files in the usual fashion.



To combine multiple data sets to a single data set follow these steps...

- Type **jexp5** (join experiment #5). Recall a 1D data file and type **wft**. This step verifies that your exp5 area is OK.
- Type **jexp1** (join experiment #1). Recall the first 1D data file for the stacked plot (this will be the spectrum on the bottom of the stack). You don't have to process the data.
- Type **clradd** to clear the add/subtract buffer (in exp5) and type **add** to add the first spectrum to the add/subtract buffer.
- Recall the next data file (**rt ('/pathname/mydatafile.fid')**) for the stacked plot. Type **add('new')** to add this spectrum to the add/subtract buffer. The 'new' argument adds a new FID element to be created in the add/subtract experiment (exp #5). The 'added' file may also be scaled (let's say by .75) by typing **add(0.75, 'new')**.
- Repeat the previous step for the remaining spectra.
- Type **jexp5**.
- Select a parameter to act as a "dummy array variable". A good choice is usually some variable such as d2. Type **d2=1,2,3,4**. In this case we will be making an array with a dimension of 4.

Open a UNIX shell (right mouse click on the background and select *Programs > Terminal...*). You will need to edit the process parameter file. Type **vi vnmrsys/exp5/procpar**. Edit this file so that the following text:

```
arraydim 7 1 32768 1 1 2 1 5 1 64  
1 1  
0
```

is changed to:

```
arraydim 7 1 32768 1 1 2 1 5 1 64  
1 4  
0
```

The vi commands that accomplish this are:

- Type **/arraydim** This will search the file 'procpar' for the string 'arraydim'. Use the arrow keys to place the cursor over the *second 1* in the *second line* in the above example).
- Type **rX<escape>**. r will 'replace' the current character with X, where "X" is the number of spectra and <escape> is the escape key. Note that in the above example the new array dimension is 4.
- Type **:wq** (colon w q) to exit the vi editor
- Type **exit** to close the UNIX window.

In the Input Window:

- Type **gain=1 calcdim arraydim?** The value of arraydim should now equal the number of spectra of your array.
- Type **groupcopy('current','processed','acquisition')**. This will copy all of the 'new' parameters to a new tree.
- Type **svf** and give the 'new' data set a name. This will save your new array data as a new data file.