2003 NMR User Training Course

National Program for Genomic Medicine High-Field NMR Core Facility,
The Genomic Research Center, Academia Sinica

09/29-09/30, 2003

09/30, 2003 Course Handout

Software Training: NMRPipe & Sparky

by

Wen-Jin Wu

NMRPipe

- **Read the original paper first**

Delaglio, F., Grzesiek, S., Vuister, G.W., Zhu, G., Pfeifer, J. and A. Bax (1995) NMRPipe: A multidimensional spectral processing system based on UNIX pipes. J. Biol. NMR 6, 277-293.

- **Usage**

Processing of 1D-4D NMR data.

- **Availability&Installation**

Free for Academia purpose. NMRPipe website: http://spin.niddk.nih.gov/bax/software/NMRPipe/

- **Platform**

Unix, Linux, NT-NMRPipe (with fee, http://www.resdesigns.com/)

- **Command Help**

nmrPipe –fn command –help,
for example: "nmrPipe –fn SP –help" to find out more about the command SP.

Or, simply use "man SP", "man bruk2pipe" etc.

NMRDraw

- **Come with NMRPipe. For viewing NMRPipe processed data. Useful for getting phasing parameters or solvent subtraction scheme for NMRPipe processing script.**

- **Some Links for using NMRPipe/NMRDraw:**

http://lysine.pharm.utah.edu/nmrlab/nmrpipe.html

NMRPipe mailing list:
http://groups.yahoo.com/group/nmrpipe/

Linux NMR mailing list
http://www.nmrfam.wisc.edu/Listarchives/linuxnmr-archive/threads.html

Note on NMRPipe/Sparky:
http://www.pharmacy.umaryland.edu/PSC/NMR/proc_anal/Pipe2Sparky.html

http://www.cm.utexas.edu/hoffman/nmrpipe_notes.pdf

Varian data:
http://www.pharmacy.umaryland.edu/PSC/NMR/proc_anal/

- **Environment setup for remote display**

1. **Open your x-window program (i.e. Exceed for windows, xhost + for Linux/UNIX, disable firewall)**
2. **Remote login to the work station**
   1. Linux/Unix: *ssh <remote hostname>*
   2. Windows:use Putty or other ssh program
3. **Setting DISPLAY Variable to for correct color display**

> *setenv DISPLAY <your hostname>：0.0*(csh or tcsh shell)
(Example: *setenv DISPLAYarouca：0.0*
or
> *export DISPLAY=<your hostname>：0.0* (sh or bash shell)

Using NMRPipe/NMRDraw

2D Processing: HSQC

1. **[winston@GOAT 140]$ setenv DISPLAY wolf:0.0**

snp (source NMRPipe, "snp" is just an alias for "source /usr/nmr/NMRPipe/nmrpipe.cshrc" used in IBMS, check with your system administrator).

2. **open a xwindow program such as Exceed, and have the display set up.**
3. **Change to the directory containing the ser file:**

   cd /goat/data/winston/nmr_data/av600/tepcomp_fhsqc/1

(Data acquisition parameters for this file: 64 complex point (R+I=128), DQD in ^{1}H, STATES-TPPI acquisition mode in ^{15}N).

4. **type in "bruker" to bring up the conversion utility window (Figure 1) . (type in "varian" for Varian data).**
5. **Click on the "Read Parameters" button to update acquisition parameters (figure 1)**

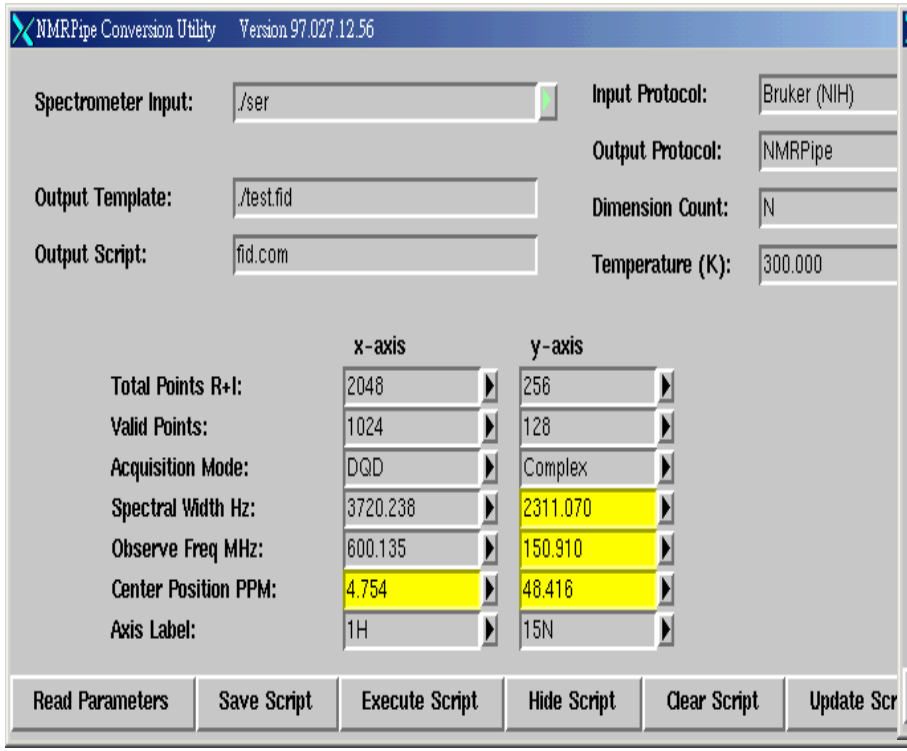


Figure 1

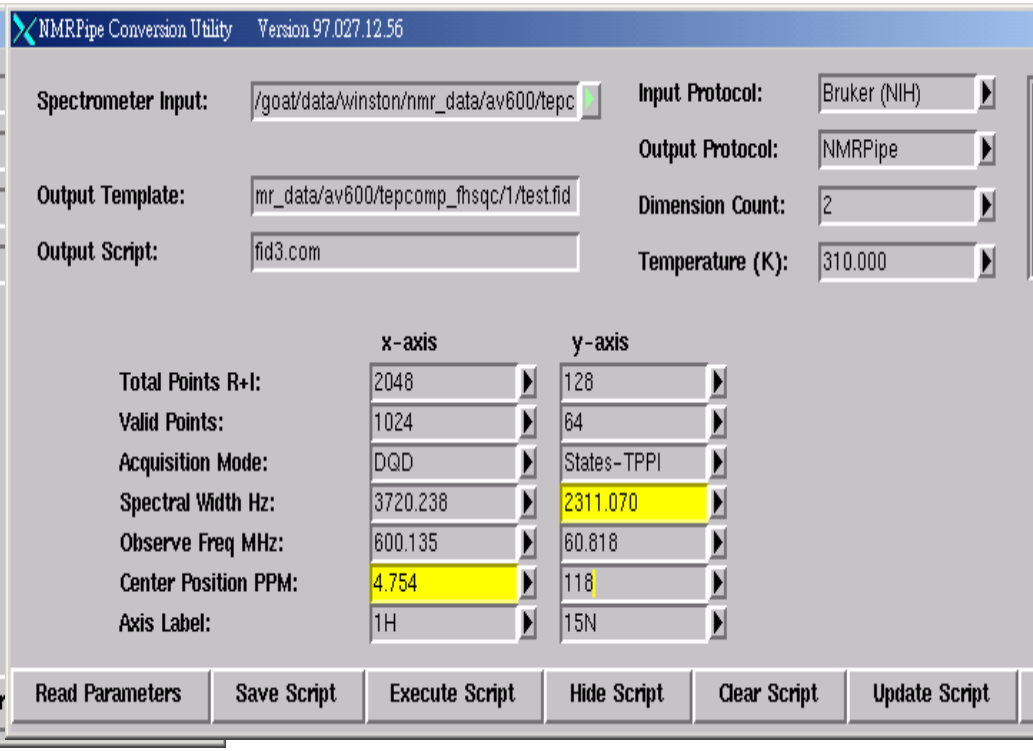
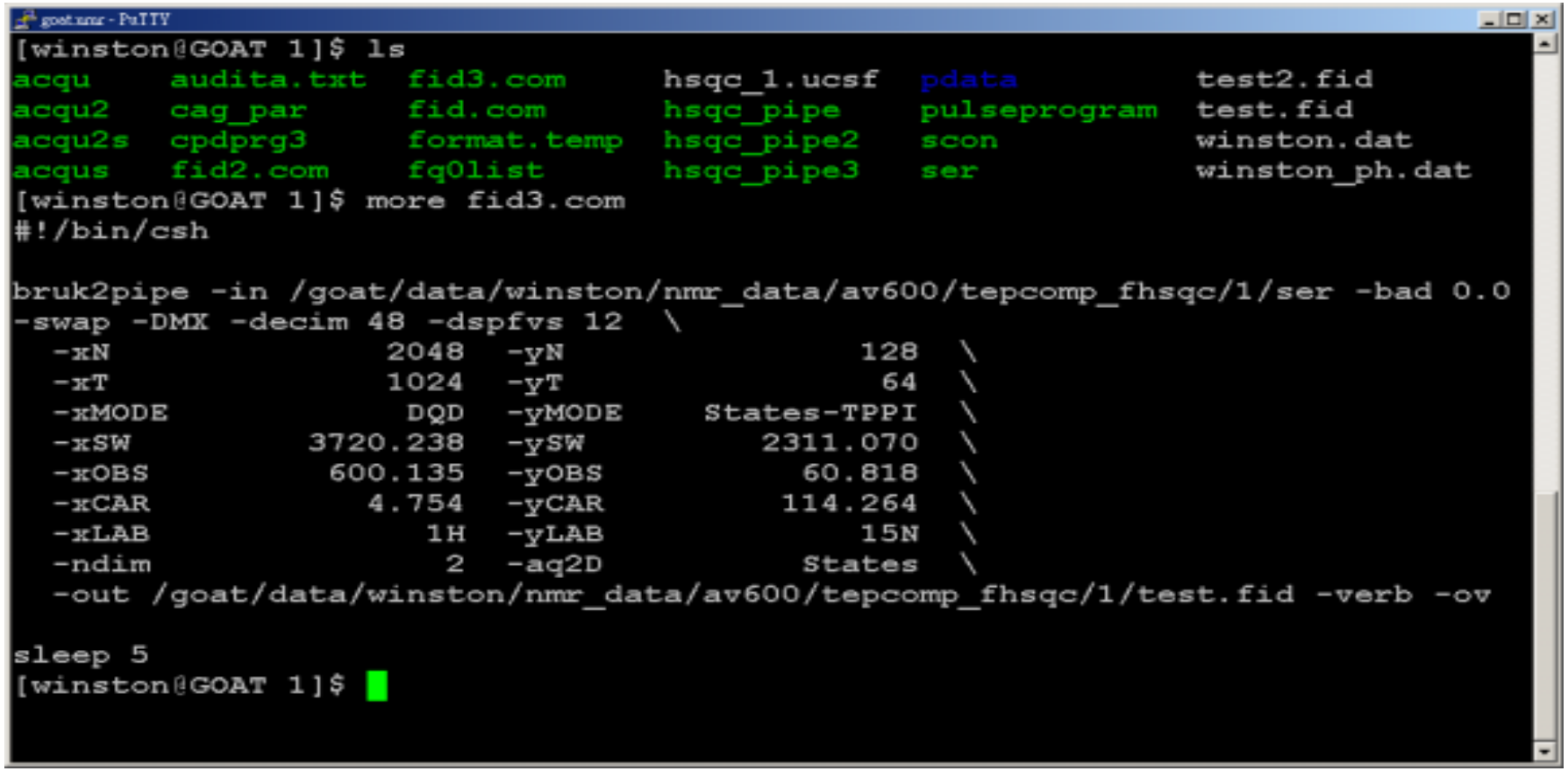


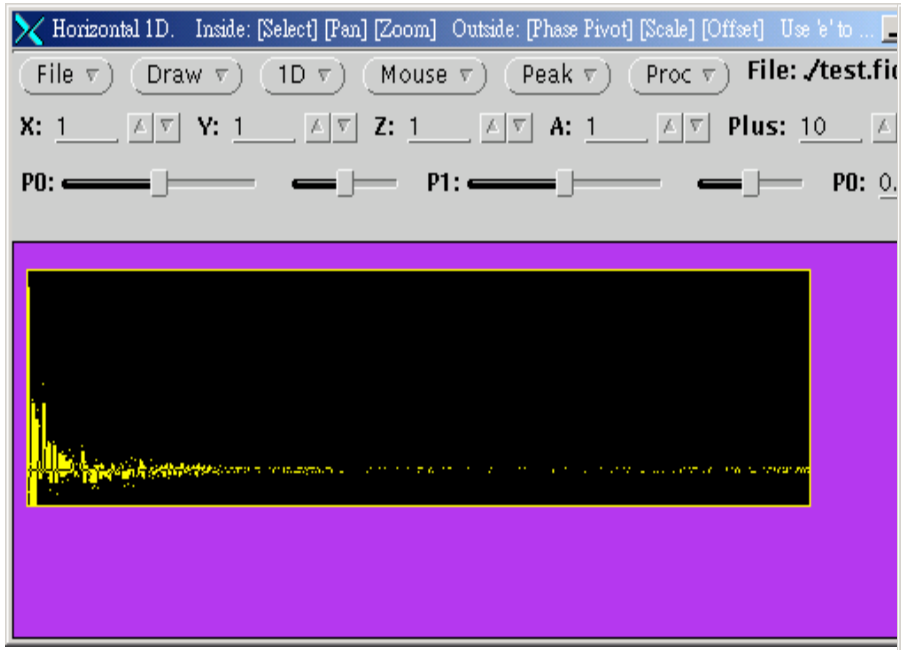
Figure 2

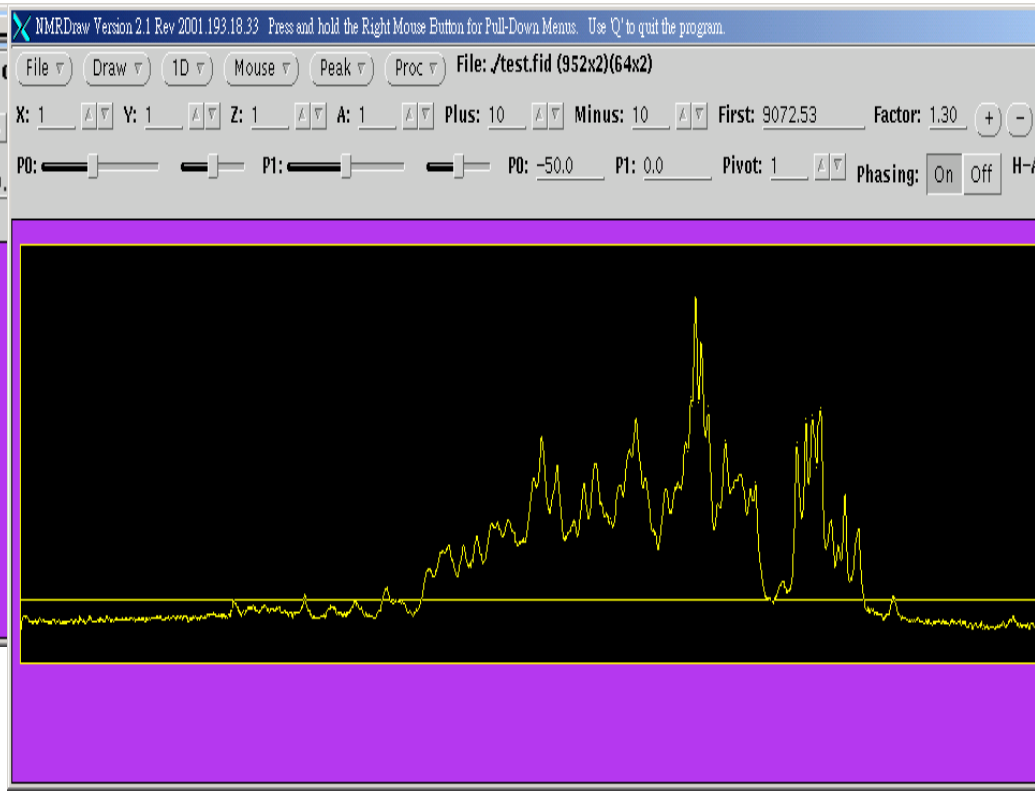
6. **Change the parameters that are incorrect. In this case, you should obtain something like Figure 2.**
7. **“Save Script” then “Execute Script” to do the conversion. “Quite” when is done.**



(Use "man bruk2pipe" for details on the "bruk2pipe" conversion script). (Use "man var2pipe" for Varian data).

8. **Open nmrDraw (type this in), and select the file "test.fid".**
9. **Under the *Mouse* button, select *1D Horizontal*, and choose X:1, Y:1 (the 1st fid).**
10. **Under *Proc*, choose Auto-Process 1D.**
11. **Have the *Phasing "on"*. Phase the spectrum and note down the values of ph0 and ph1.**





12. **Edit the hsqc_pipe macro and also input the above phasing parameters (ph0=p0, ph1=p1) in the proton dimension.**
13. **Make the macro executable, and place it under the directory containing the converted file "test.fid": /goat/data/winston/nmr_data/av600/tepcomp_fhsqc/1.**
14. **Execute the hsqc_pipe script by typing the name directly.**

Processing Status

```
[winston@GOAT 1]$ hsqc_pipe2
PS       128 of 128
PS      2048 of 2048
```

The "hsqc_pipe2" script:

```
#!/bin/csh
nmrPipe -in ./test.fid                          \     #Read in the file
| nmrPipe -fn SP -off 0.5 -end 1 -pow 1 -c 1    \     #SP: Sine bell window function
| nmrPipe -fn ZF –auto                          \     #ZF: zero filling
| nmrPipe -fn FT –auto                          \     # FT: Fourier Transformation
| nmrPipe -fn PS -p0 -50 -p1 0.0 -di –verb      \     #phase, delete imaginary
#| nmrPipe -fn EXT -left –sw                    \     #Extract region
| nmrPipe -fn TP                                \     #Transpose axis (xy to yx)
| nmrPipe -fn SP -off 0.5 -end 1 -pow 1 -c 1    \     #process on the 2nd dimension
| nmrPipe -fn ZF –auto                          \
| nmrPipe -fn FT –auto                          \
| nmrPipe -fn PS -p0 0.0 -p1 0.0 -di -verb      \
|nmrPipe -out ./winston_ph.dat –ov                    # Write out processed data
```

(use "man command Name", or "nmrPipe –fn Command Name –help" to find out more about each command !)

```
nmrPipe -fn SP [-off offset] [-end end] [-pow pow]
| nmrPipe -fn SP -off 0.5 -end 1 -pow 1 -c 1  \
```

- Use “nmrPipe -fn SP –help” for help on SP.

```
SP: Adjustable Sine Window. [SINE]
 -off   offset [0.0]  Sine Start*PI.  (Q1)        ;use 0.5 for a Cosine starts at 1 . Sine(0.5π)=1
 -end   end    [1.0]  Sine End*PI.    (Q2)        ;use 1 for a Cosine ends at 0.  Sine(π)=0
 -pow   exp    [1.0]  Sine Exponent.  (Q3)        ;1 for Sine, 2 for Sine square.
 -c     fScale [1]    Point 1 Scale.
```

- NMRPipe Processing Functions

A list of the nmrPipe functions used in the data processing:

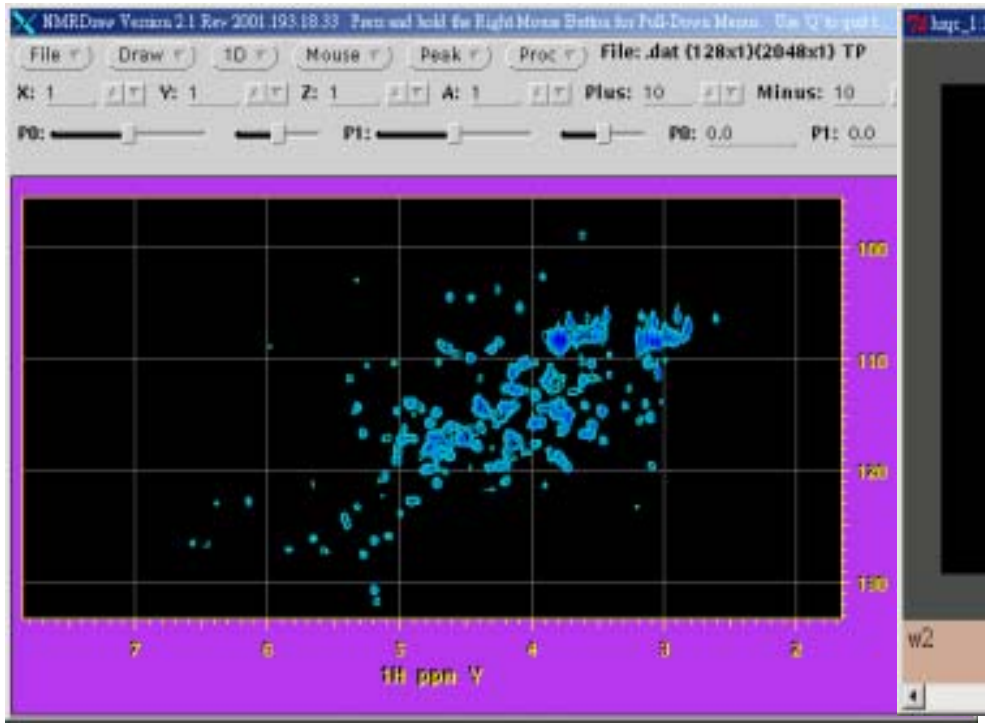
EXT extracts a region from the current dimension with specified limits
FT applies a real or complex forward or inverse Fourier transform
HT applies a Hilbert transform to reconstruct imaginary data
LP linear prediction
POLY When used in frequency-domain, applies polynomial baseline correction . When used in the time-domain, applies solvent correction
PS applies the zero- and first-order phase corrections
REV reverse data order in given dimension
SOL applies solvent correction

SP applies sine-bell apodization
TP/YTP exchanges data vectors from x and y axis of the data stream
ZF applies zero-filling
ZTP exchanges data vectors from x and z axis of the data stream

Generic Arguments

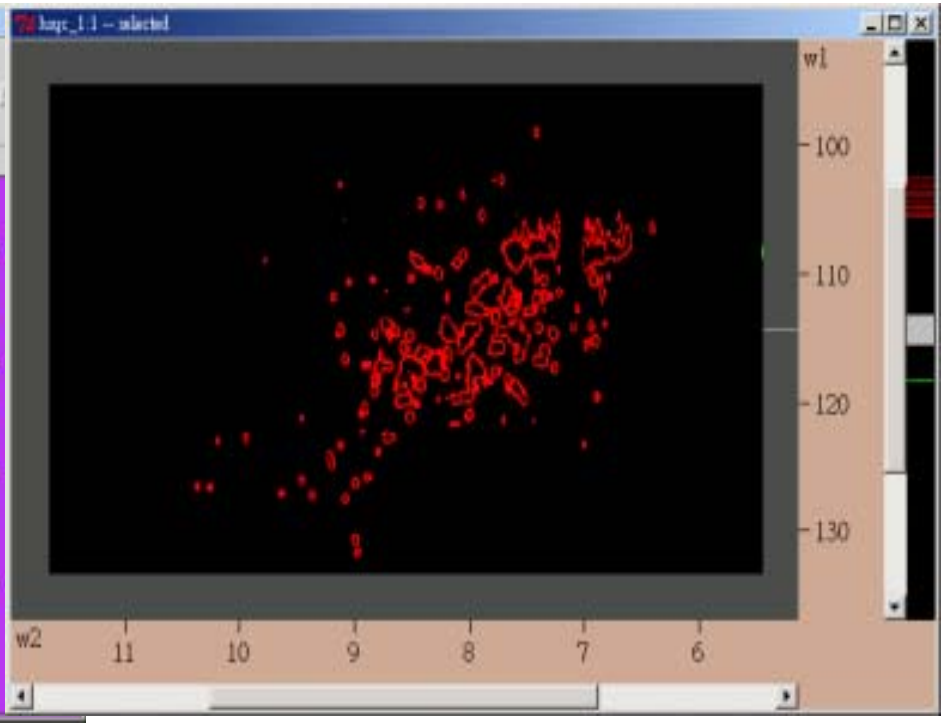
The following arguments are used by more than one function in the shell scripts above.

-di deletes imaginary data after the given processing function is performed
-hdr extracts parameters recorded during previous processing from spectral header
-in specifies the input file or file template
-inPlace specifies replacement of the input data by the output result (use with care)
-inn activates the inverse mode of a given function
-out specifies the output file or file template
-ov permits overwriting of any pre-existing files.
-sw updates the spectral width and other ppm calibration information
-verb permits processing in verbose mode, with status messages



NMRDRAW Display

(Since "frequency-jump" was used in this experiment, the 1H-dimension needs to be calibrated.)



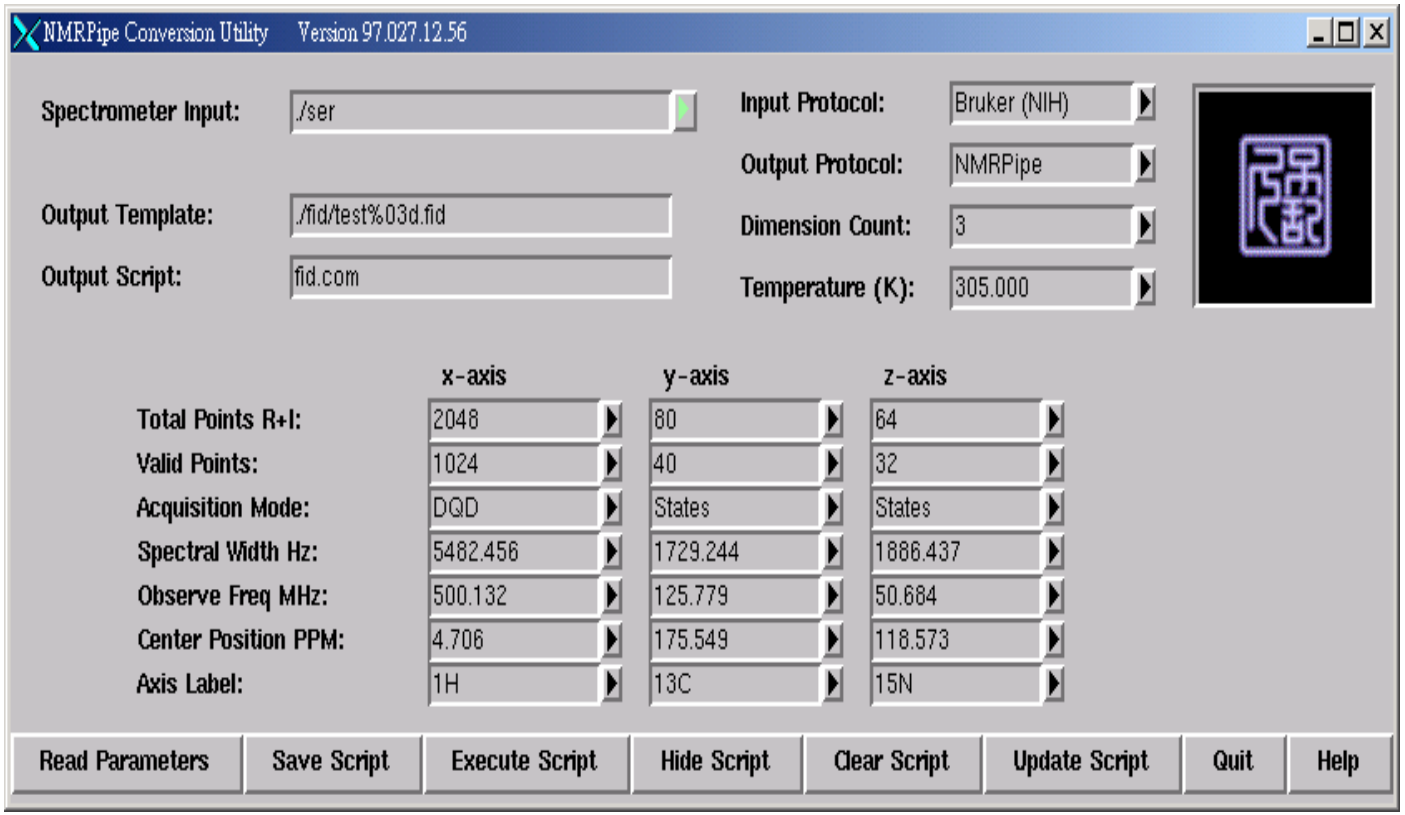
Sparky Display

[Use NMRDraw to inspect the spectrum]

- **Check phasing of column and row vector by using "1D Horizontal, or 1D vertical" under "Mouse".**
- **Convert to Sparky format (2D): pipe2ucsf winston_ph.dat hsqc_1.ucsf**
- **Open the spectrum in Sparky:**

3D Data Processing using NMRPipe/NMRDraw

- **change to the directory containing the ser file:**
  /goat/data/winston/nmr_data/nmrfam/hnco_omtky3/1
- **[winston@GOAT 1]$ bruker**



The ft_xyz.com nmrpipe macro:

****

```
#!/bin/csh
#
# 3D States-Mode HN-Detected Processing.

xyz2pipe -in fid/test%03d.fid -x  -verb                 \
| nmrPipe  -fn SP -off 0.5 -end 0.95 -pow 1 -c 1.0      \
| nmrPipe  -fn ZF –auto                                 \
| nmrPipe  -fn FT –verb                                 \
| nmrPipe  -fn PS -p0 0.0 -p1 0.0 –di                   \
| nmrPipe  -fn EXT -x1 10.0ppm -xn 7.0ppm –sw           \
| nmrPipe  -fn TP                                       \
| nmrPipe  -fn SP -off 0.4 -end 0.95 -pow 1 -c 0.5      \
| nmrPipe  -fn ZF –auto                                 \
| nmrPipe  -fn FT                                       \
| nmrPipe  -fn PS -p0 0.0  -p1 0.0 –di                  \
#| nmrPipe  -fn CS -rs 1.7ppm –sw                       \
| nmrPipe -fn POLY -auto -ord 0                         \
| nmrPipe -fn TP                                       \
| nmrPipe -fn POLY –auto                                \
| nmrPipe -fn TP                                       \
| pipe2xyz -out ft/B%03d.ft2 -y -ov

xyz2pipe -in ft/B%03d.ft2 -z –verb                     \
```

```
| nmrPipe  -fn SP -off 0.5 -end 0.95 -pow 1 -c 0.5          \
| nmrPipe  -fn ZF –auto                                      \
| nmrPipe  -fn FT –verb                                      \
| nmrPipe  -fn PS -p0 0.0 -p1 0.0 –di                        \
#| nmrPipe  -fn CS -ls 0.2ppm –sw                            \
| nmrPipe  -fn POLY -ord 0 –auto                             \
| pipe2xyz -out ft/C%03d.ft2 -z –ov
```

****

(The “ft_xyz.com” should be executable.)

Processing status message

```
[winston@GOAT 1]$ ft_xyz.com

XYZ2Pipe Partition: Plane 1 to 64 of 64
FT       5104 of 5120
FT       5120 of 5120
XYZ2Pipe Partition: Plane 1 to 128 of 128
FT      71760 of 71808
FT      71808 of 71808
```

Convert the Processed Data to Sparky-Readable Format

3D Data: For a processed hnco spectrum (3D data) saved as 2D planes with names hnco001.ft, hnco002.ft, ... the command to produce a single 3D NMRPipe file looks like:

```
% xyz2pipe -in hnco%03d.ft -x > hnco_3D.ft
```

you will then use the pipe2ucsf utility to convert the "single file" 3D NMRPipe data to Sparky format file:

```
% pipe2ucsf hnco_3D.ft hnco_3D.ucsf
```

Sparky

- **Utility:**

View and analyze multidimensional NMR data.

- **Availability & Installation:**

Free @ http://www.cgl.ucsf.edu/home/sparky/

- **Platform:**

Unix, Linux, MS Windows on PCs

- **Useful Links:**

(1) Sparky's website
http://www.cgl.ucsf.edu/home/sparky/
(2) Sparky Practical:
http://dolphin.chem.uu.nl/~henry/cursus/sparky.html

Using Sparky

Reading Files:
(Sparky's manual: http://www.cgl.ucsf.edu/home/sparky/manual/files.html)

- **Data processed with Bruker XWINNMR:**

To convert Bruker processed data 1/pdata/1/2rr to UCSF format:

% bruk2ucsf 1/pdata/1/2rr noe150.ucsf

● Data processed with NMRPipe:

(1). 2D Data: Given NMRPipe data noe150.pipe (2D data) you convert it to UCSF format with:

% pipe2ucsf noe150.pipe noe150.ucsf

(2). 3D Data: For a processed hnco spectrum (3D data) saved as 2D planes (with names hnco001.ft, hnco002.ft, …..hnco0032.ft), first "pack" all files into a single 3D file:
% xyz2pipe -in hnco%03d.ft -x > hnco_3D.ft
, then use the pipe2ucsf utility to convert the "single" 3D NMRPipe file to Sparky-formatted file:
% pipe2ucsf hnco_3D.ft hnco_3D.ucsf

Sparky Software Demonstration

lbd_3d_13_1 (CBCANH)
lbd_3d_12_1 (CBCA(CO)NH)

Contour level adjustment, changing plan, integrate, vector display, strip plot etc.