

Note that most of the commands in this overview are geared towards Varians. Fundamental Concept of 2D: Resonances are phase-encoded during the t_1 period so that after Fourier transformation the f_1 dimension will reflect the 'evolution' of spins during t_1 delay in pulse sequence.

'Essential' ^1H -detected 2D Pulse Sequences:

COSY – **C**ORrelation **S**pectroscop**Y**, good for determining basic connectivity via J-couplings (through-bond).

TOCSY – **T**OTAL **C**ORrelation **S**pectroscop**Y**, same as **COSY**, but also able to generate cross peaks via intermediate spins (mix). Uses a spin lock that produces rf heating of the sample and hence requires many steady state scans (ss).

NOESY – **N**uclear **O**verhauser **E**ffect **S**pectroscop**Y**, allows one to see through-space effects, useful for investigating conformation and for determining proximity of adjacent spin systems. Not so useful for MWs in the 1 kDa range due to problems arising from the NMR correlation time.

ROESY – **R**otational **O**verhauser **E**ffect **S**pectroscop**Y**, same as **NOESY**, but works for all molecular weights. Has the disadvantage of producing more rf heating, hence it requires more steady state scans.

HMQC – **H**eteronuclear **M**ultiple **Q**uantum **C**orrelation, allows one to pair NH or CH resonances. Often uses X-nucleus decoupling and hence gives rise to rf heating, requires reasonably well calibrated pulses and many steady state scans.

HSQC – **H**eteronuclear **S**ingle **Q**uantum **C**orrelation, provides the same information as **HMQC**, but gives narrower resonances for ^1H - ^{13}C correlations. Also requires X-decoupling and hence a large number of steady state scans and is also more sensitive to pulse imperfections.

HMBC – **H**eteronuclear **M**ultiple **B**ond **C**orrelation, a variant of the **HMQC** pulse sequence that allows one to correlate X-nucleus shifts that are typically 2-4 bonds away from a proton.

Summary of Methodology:

- **Load 1D proton parameters** and be sure to choose your solvent correctly.
- **Regulate the temperature.** For example, type **temp=25 su** (be sure to run every experiment at 25 C or some other constant temperature, as stable temperature greatly improves the quality of 2D spectra, especially HMQC, HSQC, and HMBC). You may want to turn down the preconditioner temperature to 15 C and run with **temp=20**.
- **Tune and shim** with spinner off after allowing the sample to reach thermal equilibrium for 5-10 minutes.
- If you are going to collect an HMQC, HSQC, or HMBC, keep increasing the lock power and decreasing the lock gain until the equilibrium lock level begins to drop or become erratic, then back off the lock power until the lock is stable and maximized for a given value of lockgain.

- **Collect a good 1D** (set **tof** (transmitter offset) & **sw** (sweep width) and note what **nt** needs to be to give good signal-to-noise) (if you are going to do HMQC, HSQC, or HMBC), collect a one-scan spectrum of the X nucleus, perhaps even a reference compound, and set your **tof** & **sw** to span the chemical shift range of interest, record **tof** and **sw** from this experiment for later use as **dof** (decoupler offset) and **sw1** (spectral width in 1st indirectly detected dimension), also note the values of **rfl** (reference peak position in directly detected dimension) and **rfp** (reference peak frequency in directly detected dimension) to be used as **rfl1** (reference peak position in 1st indirectly detected dimension) and **rfp1** (reference peak frequency in 1st in directly detected dimension) later on).
- Set reference mark (put cursor on peak, type **dscale nl**, then **rl(7.27p)** for chloroform (or whatever shift you want).
- Save 1D, and transfer the data set (or at least the parameters) to another experiment and join that experiment (this will allow you to use the 1D you just collected for the ¹H projection(s). For example, if you are in experiment 1, then type **mf(1,2) jexp2 wft**.

Calibrating pulse widths:

- Join another VNMR experiment and calibrate pulsewidths (if necessary) (**mf(1,3) jexp3**) array **pw** from 4 to 40 in steps of 4 us, set **nt** to be the minimum needed for good s/n, set **d1** (first delay) to be 3-10 seconds, type **au**, phase the first slice by typing **wft(1)** and then phasing the spectrum; type **ai vp=90 wft dssh** and look for the inversion point (this is your 180), type **da** to display the array of pw's and find the value of pw that gives a fully-dispersive or zero-integral resonance and set pw90 to be half of the pw that gives you the peaks that are equally spread out above and below the baseline – i.e., dispersive instead of absorptive mode). After you do this, move the 1D parameters to another experiment.
- **Load 2D pulse** sequence by typing the name of the pulse sequence you want to run, e.g., TOCSY (typing the name of the pulse sequence in upper case loads the newer version of the pulse sequence, while typing it in lower case loads the old version without reading in any of the pulse calibration data from the probes file in /vnmr/probes).
 - Pulse calibration data is located in a file in /vnmr/probes/<probenam>/<probenam> where <probenam> is the value returned when you type probe? in the VNMR command window. To view these values, open a UNIX shell and type more /vnmr/probes/<probenam>/<probenam>.
- **Adjust the 2D parameters**, especially **nt** (number of transients), **ni** (number of increments or FIDs to collect – this determines the resolution in the F1 dimension), and **ss** (number of steady state scans executed prior to saving the data).

For all 2D pulse sequences these are important parameters:

- **d1** (relaxation delay)
- **gain** (receiver gain)
- **tof** (transmitter offset)
- **sw** (sweep width)
- **np** (number of point in FID)

- **fn** (Fourier number, i.e., size of final array in f2 dimension)
- **sw1** (sweep width in the f1 dimension, the 'second' dimension)
- **ni** (number of increments or the number of points collected in the f1 or 'second' dimension)
- **fn1** (Fourier number in the f1 dimension)
- **pw** (pulse width)
- **tpwr** (transmitter power)
- **nt** (number of transient or scans per FID)
- **phase** (how the data in the f1 or 'second' dimension will be collected)
- **hs** (homospoil flag)
- **hst** (homospoil time)
- **sspul** (steady state pulse flag)
- **trace** (which axis to display horizontally)
- **axis** (how to label the f1 and f2 axes)
- **d2** (evolution time – do not set! if you do, then you disable the software's ability to increment this variable which is required to collect a 2D)
- **PFGflg** (flag to indicate if you have gradients)

NOESY:

- **mix** (the mixing time during which spins undergo dipolar interactions, typically 0.03 to 0.3 seconds)

TOCSY:

- **p1** (hard 90 pulse at beginning of evolution, at end of evolution, and at end of mix)
- **p1lvl** (power level for p1 pulse)
- **mix** (mixing time for J-coupled or through-bond interactions, typically 0.03 to 0.3)
- **ss** (number of steady state scans to carry out to bring sample to thermal equilibrium in the presence of rf-heating from spin-lock during mix)

ROESY:

- **p1** (see TOCSY)
- **p1lvl** (see TOCSY)
- **mix** (mixing time for dipolar or through-space interactions)
- **rocomp** (resonance offset compensation flag)
- **ss** (see TOCSY)

HMQC/HSQC:

- **dn** (decoupler nucleus)
- **dof** (decoupler offset)
- **j1xh** (average one-bond H-X J-coupling in Hz, used to determine mixing time, use **j1xh=140** for dn='C13', **j1xh=90** for dn='N15')
- **mbond** (multiple bond correlation flag, set to 'n')
- **null** (enables BIRD nulling, is the time for recovery of resonances following inverting pulse, is set in seconds)
- **pwX** (pulse width for dn/X-nucleus)
- **pwXlvl** (power level for pwX)
- **dm** (decoupler mode, either 'nnn' for off or 'nny' for on)
- **dmm** (decoupler modulation mode, usually 'ccw')

- **dpwr** (decoupler power)
- **dmf** (decoupler modulation frequency)
- **ss** (number of steady state scans, set to 256-512 if **dm='nny'**, otherwise **ss=8**)
- Set **dof** equal to the value you wrote down for **tof** in the X-nucleus observation.
- Set **sw1** is the value you wrote down for **sw** in the X experiment.
- Set **rfl1** is the value of **rfl**, and **rfl1** is the value of **rfl**.
- If the T_1 's for your sample are long relative to the recovery time (**at + d1**, type **dps** to display the pulse sequence and view how long the spins have to relax between the read pulse before the FID is acquired and when the next pulse occurs), then you may want to set **sspul='y'** and **PFGflg='y'** (only if you have gradients!), this will randomize the net magnetization prior to the relaxation delay **d1**, thereby ensuring the net magnetization will be the same at the start of every scan (help minimize artifacts).

HMBC:

- **dn** (same as HMQC/HSQC)
 - **dof** (same as HMQC/HSQC)
 - **mbond** (multiple bond correlation flag, set to 'y')
 - **taumb** (mixing time for long-range correlations, usually 0.055 seconds).
 - 2D Sequences with Water Suppression (sometimes start with "tn", e.g., **ntocsy**)
 - **presat** and **sadly** (length of time to irradiate solvent peak)
 - **satfrq** (offset of solvent resonance, sometimes not used, in which case **tof** must be on the solvent resonance)
 - **satpwr** (power level to use when irradiating solvent peak)
 - See the end of the above paragraph with regard to setting **dof**, **sw1**, **rfl1**, and **rfl1**.
- Set the size of the 2D data matrix: **fn=2k fn1=2k** (4k can be used, but the processing takes significantly longer)
 - If you are running HMQC or HSQC and **dn='nny'**, then type **np=fn/2**, otherwise use **np=fn**
 - If you are running HMBC, be sure to set **dn='nnn'**
 - Make sure that **d1** is set such that **5*at** is less than **d1**. If you need to adjust the **d1:at** ratio and do not want to increase **d1** (doubling **d1** will make your experiment run for twice as long), you can decrease **at** by cutting down **np** to 1024 or even 768 or 512 (**at** should be less than 0.2 seconds or extreme sample heating may result).
 - If you are running HMQC, HSQC, or HMBC
 - set **dof** and **sw1** according to the 1D spectrum you already obtained, otherwise **sw1=sw**
 - Also make sure that you have the correct bandpass filter in place in the broadband channel going to the probe (X-nucleus decoupling will kill the wrong filter which costs about \$400). Using no filter will give you garbage for results.
 - If you want to use a squared, shifted sine bell for your apodization function in both dimensions, type **ssb ssb1** (these are MIT macros, otherwise you can use **wti** after the first slice has been collected). Gaussian functions are also very popular.
 - **Set the relaxation delay d1** to be 1-1.5 seconds.

- Collect one scan and let the autogain routine set the receiver gain: with **gain='n' phase=1 nt=1 ss=1 ni=1 au**
- Now set the gain back to manual mode, turn it down by 2 dB, turn off the digital signal processing, and collect another scan: **gain='y' gain=gain-2 dsp='n' au**
- When the acquisition is complete, type **ddff(1)** to display the data in the FID file, block 1. Look at the text window in the tcl/dg display. Verify that the numbers you see are in the + or - 10,000 range. If the numbers are low, increase the gain by 1dB, if the numbers are high, decrease the gain. The gain can never drop below 0.
- You can also set the receiver gain by typing **set2dgain** (an MIT macro, where set2dgain: **r1=nt r2=ni r3=ss wexp='set2Dgain2' nt=1 phase=1 ni=1 ss=1 gain='n' au** and set2Dgain2: **nt=r1 ni=r2 ss=r3 phase=1,2 wexp=" gain='y' if (gain > 3) then gain=gain-3 endif**)
- If you are running HMQC, null will load at 0.3 seconds. You may want to adjust this number, as this will allow you increase your receiver gain. You can array null with the rest of the parameters set as above. Type **au** to collect a bunch of FIDs with the different null values (**da** to display these values in the text window). Type **vp=90 ai wft dssh** and find the spectrum whose most intense peak is smallest. Type **da** and find the value of null that corresponds to that spectrum. Go back and find the highest gain setting you can use to give **ddff(1)** values that do not exceed + or - 10000.
- When the gain (and maybe null) is (are) set properly, restore the other parameters, for example: **dsp='i' phase=1,2 nt=16 ss=8 ni=32 au**
- If running TOCSY, ROESY, or HMQC/HSQC with **dn='nny'** set **ss=512 or 256** to allow the sample to reach a steady state temperature because of the rf heating that will occur from either the spin-lock or the decoupling.
- Set **nt** according to how many scans were needed to get a good proton 1D spectrum unless you are running an HMBC spectrum, in which case you should multiply nt by at least four.
- Set **ni**. Use an ni of at least 32, but an ni as large as 256 or 320 may also work. Type time will give an approximate estimate of how long the experiment will run.
- Type **au** to start the run. The acquisition status window should give you a good estimate of how long your experiment will run for.
- When the first FID has been collected (the acquisition status window will show FID: 2), transform the first FID by typing **wft(1)**.
- Phase the first slice of the 2D as you normally would phase a 1D spectrum.
- Calculate the first point multiplier by typing **cfpmult**. The value of **fpmult** determined by the **cfpmult** macro helps compensate for the fact that a zero delay cannot exist between pulses with differing phases when the first complex point of the 2D data set is collected. This zero delay corresponds to a zero t1 increment (**d2=0**).

- Set the integrals for all the peaks you see (type **cz** to clear the integral zeroes or resets).
- You can also play with the apodization in the f2 dimension with the **wti** (weight interactively) routine. Simply typing **ssb** (sets $sb=-1*at$, $sbs=sb$) will automatically set the apodization to a reasonable value, however. Typing **ssb1** will similarly set the apodization in the f1 dimension to a reasonable value.
- If the ADC overflow light starts flashing, you may need to abort the acquisition and go back and tweak parameters.

Do preliminary processing with the help of linear prediction.

When FID:33 shows in the acquisition status window, you can begin to look at the first 16 complex points of your 2D data set with the help of linear prediction.

- Type **dolp** (do linear prediction) to predict the remainder of the data set based on the first 32 FIDs (16 complex points). **dolp** makes use of the variable **celem** (completed fid elements) which keeps track of how many completed elements there are in the data set.
- **wft1da** (transform the data set halfway, i.e., only transform the f2 dimension, thereby producing what is termed an interferogram which shows how each FID's individual points vary as a function of the evolution time)
- **bc('f2')** (baseline correct the f2 dimension using the integral resets you set after you transformed the first slice with the **wft(1)** command)
- You will see an interferogram. If you want, you can manually apodize the t1 data points by placing the cursor on an intense peak and typing **wti**. Again, **ssb1** will automatically set a reasonable apodization value for you.
- Typing **wft2da** will transform the spectrum the rest of the way using the linear prediction values set by **lpr1** – to turn linear prediction off, set $proc1='ft'$ (instead of 'lp').
- Transforming a completed spectrum. Type **wft1da bc('f2') dolp wft2da**.

Manipulating the 2D spectrum:

- **dconi** will display to plot contours and make the cursor active (or interactive).
- **dpc** will display the plot contours and leave the cursor active.
- **vs2d** can be adjusted with the middle mouse button in **dconi** mode (but not in **dpc** mode) or **vs2d** can be set by typing in a new value, e.g., **vs2d=200 dconi**

To phase the 2D spectrum:

- Type **dconi**, click **[full]** and place the cursor on the top-right-most peak.
- Type **r1=[Index]** but do not hit enter. Index is the Index number at the top of the screen. Now position the cursor at the bottom-left-most peak of the spectrum and type **r2=[Index]**. You have just selected two rows with peaks.
- Type **ds(r1)** and click **[phase]**. Only adjust the right side phasing (you have just adjusted the zero order phasing).
- Type **ds(r2)**, click **[phase]**, click on the same spot on the right side of the spectrum BUT DO NOT ADJUST THE PHASE.

- Next, click on the left side of the spectrum and adjust the phase (this adjusts the first order phasing).
- Go back and display row r1 by typing **ds(r1)**. Tweak the right phase if needed.
- Type **ds(r2)** and follow the same row 2 phasing procedure as before (click right but make no adjustments, then click on the left and adjust the phase). Iterate back and forth until you are satisfied with the phasing.
- Type **dconi** to see what should now be a properly phased 2D spectrum. If there are purple and orange vertical streaks in your spectrum, you may have to go back and adjust the f2 phasing by typing **wft(1)** and readjusting the phase. If you do this, you then need to once again type **wft1da bc('f2') wft2da** to reprocess the data set as you did before.
- Another way to rephase is to set the parameter pmode from 'partial' to 'full' (**pmode='full'**), by then changing the trace parameter from 'f1' to 'f2', you can adjust the f2 phasing after the transform has taken place, but your transformed 2D data set will occupy much more memory (i.e., the experiment size will be larger when you type **explib**).

Setting the reference lines.

- Expanding a peak on the diagonal (if present, i.e., not HMQC, HSQC, HMBC), placing the cursors on top of the peak and typing **rl(7.27p) rl1(7.27p)** (use units of d instead of p (for rl1) if running a heteronuclear 2D, i.e., HMQC, HSQC, HMBC) will set the reference mark in both dimensions.
- **setref2D** should also work to set the reference marks, but small errors may occur.

Additional manipulations of the 2D data set.

- **bc('f1')** can be run to remove or reduce t1 ridges.
- **foldt** will symmetrize a data set (do not run on any data set where fn does not equal fn1 or where the f1 and f2 axes are not identical, e.g, HMQC).

Plotting the 2D data set.

- **ppc** will plot, just as **dpc** displays (these are custom macros, the manual version of **ppc** is **pcon(30,1.3) dpcon(30,1.3) dconi('restart')**)
- **th** will adjust how many of the lowest contours are left off when plotting or displaying plot contours. The middle mouse button can adjust **th** in **dconi** mode
- projections can be displayed using the **[proj]** button after typing **dconi**. Adjust the height of the projections with the middle mouse button. Click **[plot]** to send them to the plot buffer
- **wc** is the width of the chart (mm) in the horizontal direction, **wc2** is the width of the chart (mm) in the vertical direction.
- Type **page** to dump the contents of the plot buffer to the plotting device.

Instead of plotting with **pcon**, you can run the **plcosy** macro for homonuclear 2D plots, or the **plhxcor** macro for heteronuclear plots.

- **plcosy** is usually run with three arguments: **plcosy(30,1.5,1)**
 - the first is the number of contour lines to draw
 - the second is the spacing between contour lines (1.5 means that each successive contour line will denote 1.5 times the intensity of the previous line)

- and the last number will be number of the VNMR experiment where the processed 1D spectrum resides.
 - Note that the experiment with the processed 1D must have the exact same sweep width (sw) and transmitter offset (tof) in order to line up properly. That is why you collect the 1D and then use the movefid (mf) command to move the data including the parameters to another VNMR experiment where you then convert the parameter set to the 2D parameter set.
-
- Type **svf** to save your file.

Finishing up.

Be sure to turn off temperature regulation when you are done by typing **temp='n' su**. If you changed the temperature of the preconditioner, you will also need to turn the preconditioner temperature back to 20 C.