

Useful note:

saving a good shim set also saves you lots of time later. When you get shims you like, get into your shims directory and then type **svs('nameofyourshims')**. Alternately there is a button for the same operation. When you want to use those shims get into the shims directory and type **rts('nameofyourshims')**. Alternately, navigate the file system, get into the shims directory, select the desired shim set file and click the load shims button. NOTE that the shims in you experiment will not be actually overwritten by the retrieved ones unless you also type **load = 'y'**. Then **su**.

Checking probe linearity: the pw90 at some power tpwr1 should be twice the pw90 at a higher tpwr = tpwr1+6. For example if you get pw90 = 11.25 at tpwr = 60, you can be satisfied with pw90 = 23 at tpwr = 54, because 23 is $\approx 2 \times 11.25$. If you EVEN find that the probe is not linear, 1- don't use it and 2- report your observation. It is a good idea, when proposing to use high powers, to check probe linearity, as it often fails at the highest powers, and this is an indication that you should avoid those powers.

Solvent suppression

When the solvent is protonated, this represents a very concentrated sample, of up to 100 M. Such a situation arises for samples with exchangeable protons which must be suspended in protonated solvent (eg. proteins in water) but also when in-line NMR is being used to monitor HPLC effluent, considering the expense of using fully deuterated HPLC solvents.

Presaturation

This is one of the 'traditional methods'. It still has its place but should no longer be used as the default, because it also saturates both resonances near solvent and protons that exchange with solvent. It works because the small solvent molecules normally have a much longer T_1 than the sample, so that when the solvent has a single line that can be saturated, it will remain saturated for the time needed to collect an FID.

First, use the normal s2pul experiment with **gain = 0** and observe the solvent itself to calibrate tpwr and pw, and to measure the tof for the solvent resonance (place the cursor on the solvent and type **movetof**).

Type **presat** to load the basic presaturation experiment. Then correct the pw, sw, tpwr. Shim up to the best lineshape with the narrowest line base possible (this is crucial). Set satfrq equal to the tof you previously determined for the solvent, **d1=0**, **satdly=2**, **satmode='ynn'**, use at least **ss=2**, **nt=2**. Higher satpwr will give more complete saturation, but will begin obliterating and deforming lines near that of the solvent. Do not exceed 12. Array satfrq over a 20 Hz range in 1 Hz steps in 'av' mode (Since the above tof value represents the top of the resonance, not its center of mass, you may need to array tof over 10 Hz to find the value that gives the best presaturation.). Go back to 'ph' mode when you have finished. Shim, shim, shim and try to get gain up to 30 without overloading the receiver.

Counter-selective excitation

A composite excitation pulse can be used for frequency counter-selectivity: binomial pulse trains and the Sklenar-Bax sequence are two easy examples. The latter is easily understood as a long and thus very selective pulse that tips a particular (solvent) spin into the XY plane (eg. along the Y axis) and an immediately-following short (hard, non-selective) pulse that tips everything in the opposite direction by 90° with the effect that the magnetization along Y tips back up to Z (where it should not contribute to the spectrum) and the magnetization initially along Z is tipped down into the XY plane along the -Y axis, where it can be observed.

Sklenar-Bax

Retrieve my SKBax.par parameter set and update the tpwr, pw, tof. Calculate values for selpw and tpwrse, the duration and power of the selective (long, soft) pulse. You will have to optimize a tiny delay between the pulses that effectively adjusts the relative phases of the two pulses. rofl is defined in μ s.

A composite 180 degree pulse, will give a larger bandwidth of inversion.

Binomial

This is a combination of 'hard' pulses and delays that is counter-selective. Have a calibrated pw90 value present and then type '**binom**'. If you want on-resonance suppression set **offset**= a positive integer, the larger it is the wider the window of excitation. To suppress a line 1500 Hz off resonance set offset=-1500, maximum excitation will be at tof. The macro will calculate good values for the d2 delay accordingly (if you set **offset=0** you will have to calculate d2 yourself, $d2 = 0.5/\text{offset}$ of maximum excitation). Choose the type of composite pulse, type **seq=1331**. set **gain='n'** and **ga**. Collect a generous number of scans because you may have to work on the phase. This sequence does not give uniform excitation so choose offset to fully encompass your signals of interest. Signals very near the solvent will be weak and signals near offset Hz away from solvent will be weak again. Offset will be the frequency of maximal excitation.

Shaped pulses, sh2pul, pbox

Again, the starting point is a good 1D (emphasizing the importance of your ability to collect one). For greatest ease of later use, you should use a 1D that has the same solvent, sw and tof as you plan to use for the experiment utilizing your shaped pulse. The pbox software makes the generation and calibration of shaped pulses completely painless. However you will need a correctly referenced spectrum in hand (since you should not change tof later), and you will need a calibrated pw90 and related tpwr in the linear regime ('ref_pw90' appropriate for the 'ref_pwr') either set these in your experiment if they already exist, or just launch pbox and pbox will prompt for them and create them if they do not yet exist.. (Check to learn whether ref_pw90 and ref_tpwr are present before you initiate pbox, because pbox will not give you a chance to correct wrong values of these parameters).

Type **ds**, click on **pbox** in the second menu bar (Pandora's box). Choose the type of pulse you want, eg. 90° , set up the cursors to specify the spectral region to which the pulse should apply, choose the shape to be used (eg. eburp, = excitation band universal rotor pulse), **close**, **name** and give the computer a name for your pulse (an informative one for yourself), **close**. To see your pulse click **bloch**, **y close**. The computer will tell you what pulse width and power to use for it. WRITE THESE DOWN ! The files for the possible shapes are stored as wave files in vnmrSYS/wavelib.

Make sure that **ref_pw90** and **ref_pwr**, if they exist in your experiment, are correctly set.

If you haven't got a spectrum at hand to put cursors on, you can use the **options** button: To generate a 90° to dispose of water (at tof) click on **pbox**, 90° , **options**, **offset**, enter '0' and return, click on **bandwidth** and enter '600' and return. Click on **return**, **e-Burp1**, **close**, **name**, enter the name you choose (e.g. killwater) and return, **close**. Give the software **ref_pw90** and **ref_pwr** when prompted (if prompted). WRITE DOWN the pulse width and pulse power returned by the macro (7.5 ms and 27 dB).

wet and water sequences for solvent suppression

Type **wet1d**, set **tof** correctly, if necessary make an excitation pulse that simultaneously excites all the solvent lines, Use this for **wetshape** (ie enter **wetshape** = 'the_name_of_your_pulse'), use pbox's suggestions for **wetpwr**, **pwet**. Note that **c13wet** is by default set to 'y' and ¹³C is decoupled throughout. This is unnecessary unless you have a ¹³C labelled sample. Set **c13wet='n'**. **ssfilter** is also automatically set to 30, which will process out signals within 30 Hz of **tof**. Initially set this to **ssfilter='n'**. Set gain to at least 8 dB less. Set **gzlvlw** = 20000 (the default value in the parameter set is unnecessarily high).

- 1- Array **tof** over 20 Hz as above, and choose the value that minimizes the solvent signal.
 - 2- Array the power for the shaped pulse **wetpwr**
 - 3- Array the duration of the pulse **pwet** over $\pm 20\%$ in ≈ 7 steps.
- repeat the cycle if needed.

When you think you are doing pretty well, set **gain='n'** and let the system set its gain value. If your solvent is suppressed well, gain should come out to be above 30.