VnmrJ Liquids NMR User Guide

Varian NMR Spectrometer Systems
with VnmrJ Software

Pub. No. 01-999250-00, Rev. A0604

VARIAN

VnmrJ Liquids NMR User Guide

Varian NMR Spectrometer Systems
with VnmrJ Software

Pub. No. 01-999250-00, Rev. A0604

VARIAN

VnmrJ Liquids NMR User Guide
Varian NMR Spectrometer Systems
with VnmrJ Software
Pub. No. 01-999250-00, Rev. A0604

Revision history:
A0802 – Beta test release
A1002 – Updated for VnmrJ 1.1B
C0503 – Updated for VnmrJ 1.1C

Applicability of manual:
Varian NMR spectrometer systems with VnmrJ 1.1C

Technical contributors: Bayard Fetler, Maj Hedehus, Dan Iverson, He Liu, Chris Price, Evan Williams
Technical writer: Everett Schreiber, Dan Steele
Technical Editor: Dan Steele

3120 Hansen Way, Palo Alto, California 94304
1-800-356-4437
http://www.varianinc.com

Overview of Contents

Table of Contents

List of Figures

List of Tables

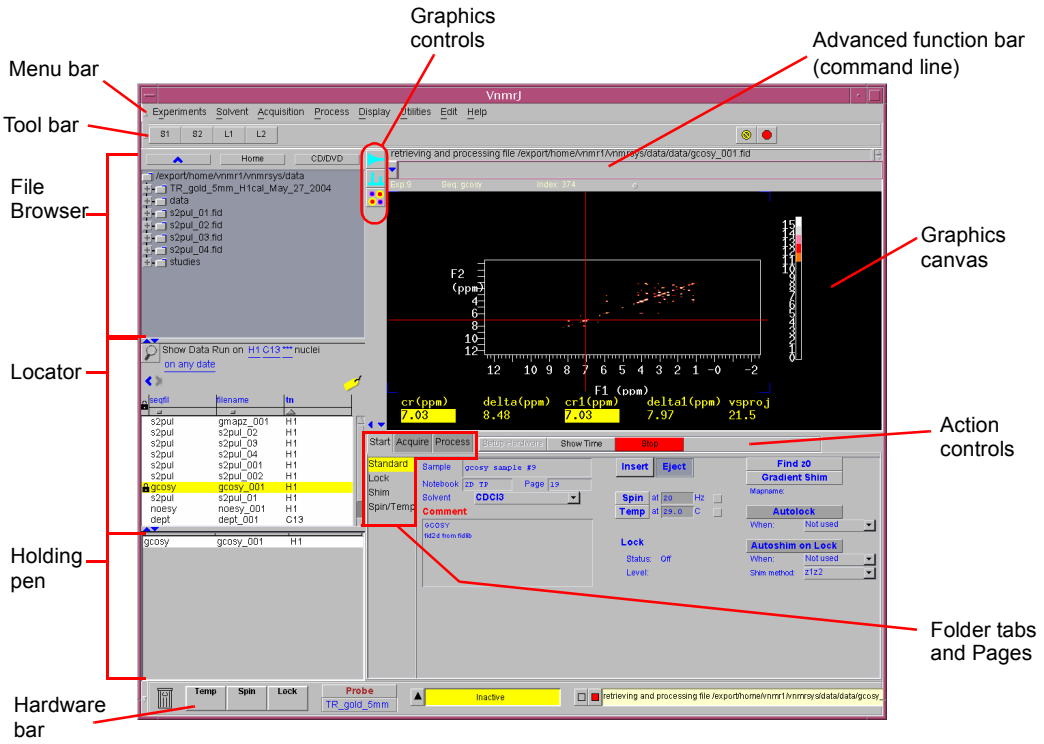
Chapter 1. Running NMR Liquids Experiments

Sections in this chapter:

This chapter is an introduction on how to use VnmrJ to run NMR liquids experiments using the VnmrJ experimental interface. The tasks involved in running an NMR liquids experiment generally follow the VnmrJ interface layout, moving from left to right over the interface from the Locator to the Start, Acquire, and Process tabs.

The VnmrJ interface for liquids experiments is described in Appendix A, “VnmrJ Liquids NMR Interface,” page 237.

VnmrJ Experimental Liquids Interface



1.1 NMR Experiment Tasks

The following table lists the tasks and where to get for information.

Task	*For more information*
Prepare for an experiment	Chapter 2, “Preparing for an Experiment,” page 21
Select an experiment	3.1, “Selecting an Experiment,” on page 29
Set up an experiment	Chapter 3, “Experiment Setup,” page 29
Acquire NMR data	Chapter 5, “Data Acquisition,” page 65
Process the data	Chapter 6, “Processing Data,” page 87
Display the data	Chapter 7, “Displaying FIDs and Spectra,” page 97
Print the data	Chapter 8, “Plotting and Printing,” page 115
Save the data	1.2, “Saving the NMR Data (optional),” on page 18

Prepare for an Experiment

Preparing for an experiment is described in Chapter 2, “Preparing for an Experiment,” page 21. Before beginning an experiment, the following must be done:

- Start VnmrJ
- Prepare the sample and position the sample tube in a turbine
- Load the probe file
- Install the probe, tune, and calibrate if necessary.

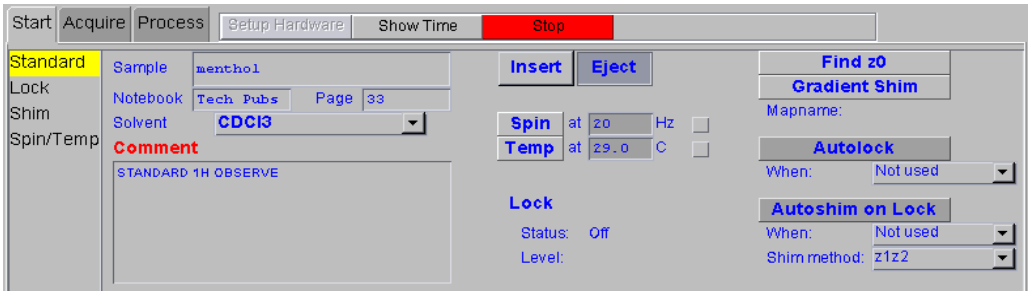
Select an Experiment

Select an experiment from the Experiments menu, or drag-and-drop a protocol from the Locator. Refer to 3.1, “Selecting an Experiment,” on page 29 for more details.

Set Up an Experiment

VnmrJ experimental setup and the functions available under the Start tab are described in Chapter 3, “Experiment Setup,” page 29.

Set up the experiment using the pages in the **Start** tab.



1. Select the **Standard** page.

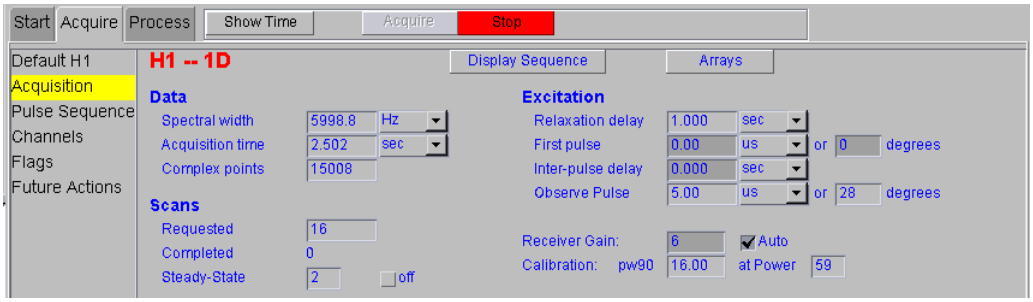
   Fill in the information for the sample, select a solvent, and enter comments. If you want to name your sample, enter a name for the in the **Sample** field. You can further define your sample by filling in the **Notebook** and **Page**.

2. Insert the sample.
3. Regulate spinning and temperature on the **Spin/Temp** page.
4. Find Z0 adjust the lock using the **Shim** and **Lock** pages.
5. Shim the system to adjust the field homogeneity using the controls provided on the **Shim** page.

Acquire a Spectrum

VnmrJ NMR data acquisition and the functions provided under the Acquire tab are described in Chapter 5, "Data Acquisition," page 65.

Set acquisition and acquire data using the pages in the **Acquire** tab.

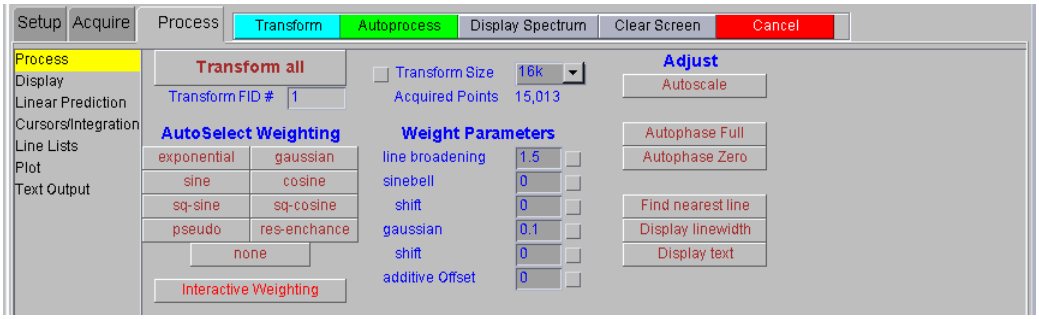


1. Set up experimental parameters and post acquisition actions.
2. Click the **Acquire** button to acquire NMR data.

Process the Data

VnmrJ NMR data processing and the functions provided under the Process tab are described in Chapter 6, "Processing Data," page 87.

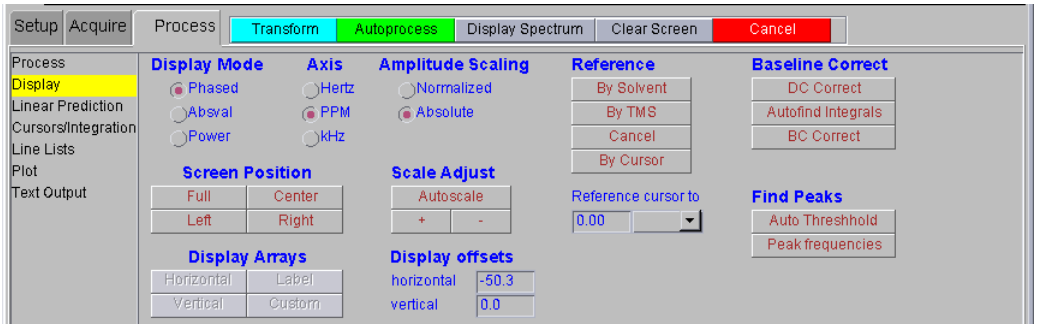
Process the NMR data using the pages in the **Process** tab.



Display the Data

VnmrJ data display is described in Chapter 7, "Displaying FIDs and Spectra," page 97.

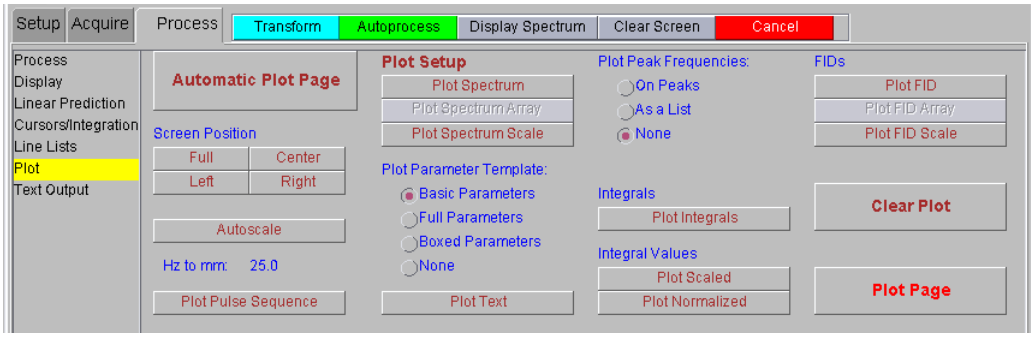
Use the **Display** page and the graphic control buttons to manipulate the display of the NMR data.



Print or Plot the Data

VnmrJ data display is described in Chapter 8, “Plotting and Printing,” page 115.

Use the **Plot** page to create a print or plot. The Plot Designer program is available from the menus, **Display -> Create a Plot Design**.



1.2 Saving the NMR Data (optional)

If you acquired the data but did not select the **Automatic FID save** feature in the **Future Actions** tab of the **Acquire** panel and you now want to save the data, you can save the data using either the menu method or the parameter panel.

Menu Method

To save data, click on **Utilities** in the menu bar, then **Save data**. You can control how and where data is saved by clicking on **Save data setup**.

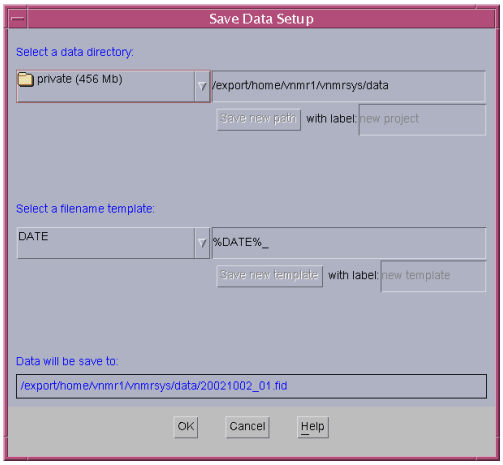


Figure 1. Save Data Setup Window

Parameter Panel Method

To save data, click on the **Acquire** tab, **Future Actions** tab, and **Save FID Now** button. If you checked the **Automatic FID Save** in the **Future Actions** panel before starting data acquisition, the data has already been saved when acquisition completes.

1.3 Stopping an Experiment

There are four ways to stop an experiment:

- Clicking on the **Stop** button  .
- Clicking on **Acquisition** in the main menu, then **Abort Acquisition**.
- Clicking on the **Stop** button in the **Acquire** panel.
- Enter **aa** on the command line

Chapter 2. Preparing for an Experiment

Sections in this chapter:

2.1 Starting VnmrJ

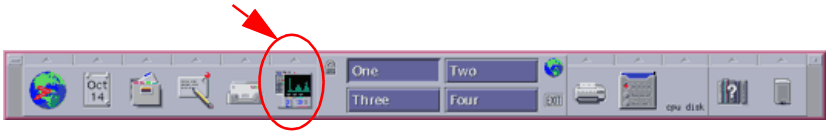
Before you can run VnmrJ, you must have a user name assigned by your system administrator. The standard software is installed so that `vnmr1` is always configured as a user, but your system administrator probably defined others as well. After logging into Solaris with your user name and password, you can start VnmrJ from a command line in a Terminal window or by clicking the VnmrJ icon in the CDE window.

To Start VnmrJ from a Terminal Window

1. Open a Terminal window:
   a. Right-click in an open area of the workspace.
   b. In the Workspace Menu, go to **Tools**->**Terminal**.
2. At the command line, enter `vnmrj`.
   `host:username> vnmrj`
   Remember to press the Return key after typing vnmrj.

To Start VnmrJ from the CDE Toolbar

Click on the VnmrJ icon in the CDE tool bar



2.2 Preparing the Sample

Reliable and fast accumulation of data from multiple samples depends greatly on the way samples are prepared and positioned in the turbines, and the autoshimming methods and lock power used. Variations in bulk magnetic susceptibility at air-to-glass, glass-to-solvent, and solvent-to-air contact points can contribute a dominant portion of the variation of field homogeneity from sample to sample, whether in an automation run or in manual operation. The time spent shimming, or even the need to shim is largely dependent on the care in controlling the effects of these contact points.

Solvent Selection

Samples can be run as neat liquids or in solutions. In most cases, you will probably be running compounds in solution. The solution should be chosen to be inert (does not react with the sample) and available in deuterated form. The instrument can be run unlocked, that is, without locking onto the deuterium of a deuterated solvent, but resolution is better with a deuterium lock, especially for lengthy accumulations. Probably the most commonly used solvents are deuterated acetone, chloroform, methylene chloride, and DMSO.

Sample Height

Experimentation and calculation show that the liquid column length must be at least three times the length of the observe coil to minimize end effects. This suggests a column length of close to 5 cm for a standard broadband or switchable probe, and about 4 cm for a $^{1}H/^{19}F$ probe. Solvent volumes of 0.6 ml in a 5-mm tube and 3.1 ml in a 10-mm tube are adequate for removing the end effects.

Reduction of sample volume to attain higher concentration usually fails because the increased signal is found around the base of the NMR resonance, not within the narrow portion of the signal. In fact, a well-shimmed 0.4 ml sample will be lower in sensitivity than the same solution diluted to 0.6 ml and also shimmed well. The questionable gain in sensitivity is further degraded by the longer time it will take to shim the system. Small variations of sample height that would be insignificant in a 0.6 to 0.8 ml sample can be dominant when the sample is only 0.4 ml in volume.

For best results and minimum shimming time, samples should be prepared to be the same height as much as possible. Above 0.7 ml there is little sensitivity to sample length as long as the bottom of the tube is positioned properly. You should make every sample up to the same height and obtain your shim values using samples of that height.

For Wilmad 528 or 535 tubes with no restricting plugs, typical samples with volumes listed in Table 1 should be placed at the depths shown in the table, where depth is the distance in mm from the bottom of the green spinner turbine to the bottom of the sample tube.

Table 1. Sample Tube Depths

Volume	*Length*	*Depth (Range)*
700 µL	50 mm	68 mm (65–69 mm)
600 µL	42 mm	65 mm (63–67 mm)
500 µL	36 mm	62 mm (60–64 mm)
400 µL	28 mm	59 mm (58–62 mm)

Sample Position Using the Depth Gauge

Set the sample position to a repeatable position. Use the sample depth Gauge provided, shown in Figure 2. If you have a sample changer, use location 0 on the sample tray.

1. Insert the turbine into the top of the sample depth guage.

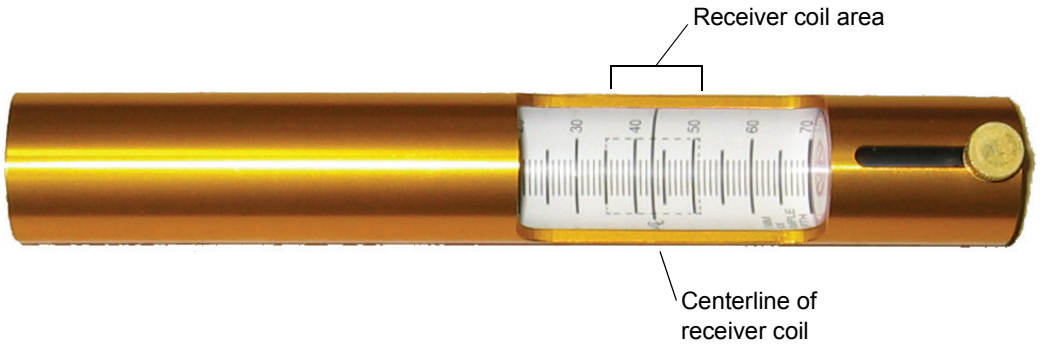


Figure 2. Sample Depth Gauge

2. Insert the NMR sample tube into the turbine. Gently push the sample tube down until it touches the moveable bottom of the sample depth Gauge.
3. Loosen the knob on the sample depth Gauge.
4. Raise the bottom of the Gauge, along with the sample tube and turbine, until the sample volume is centered on the centerline mark (CL, between 35 and 51 mm) in the back of the Gauge.
5. Tighten the knob.
6. Remove the sample tube and turbine from the depth Gauge.
7. Gently pull up on the sample tube in the turbine, replace the turbine/sample tube into the depth Gauge, and gently push down on the sample tube until it touches the repositioned bottom of the depth Gauge.

Sample Tubes

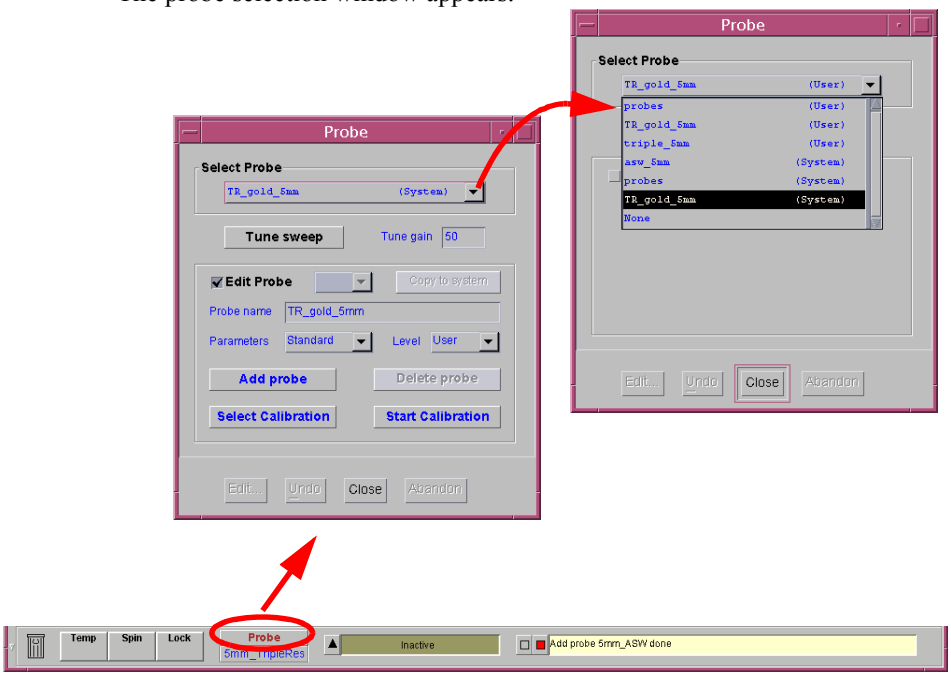
Finally, it is helpful to buy the best quality NMR sample tubes and to clean the outside of each tube with a solvent such as isopropyl alcohol, followed by a careful wiping with a wiper tissue before placing the tube in the probe.

2.3 Loading a Probe File

Probe files are typically created during system or probe installation. Procedures for creating probe files and probe calibration files are provided in the system *Acceptance Tests Procedures* manual.

1. Click the **Probe** button on the Hardware bar, at the bottom left corner of the VnmrJ interface.

The probe selection window appears.



2. Click the **Select Probe** drop-down menu and select the desired probe.
3. Click **Close** to dismiss the window.

2.4 Probe Tuning and Sample Changes

In general, if the probe is already tuned to the proper nucleus (as is almost always the case for proton and carbon observation), only a small amount is gained by tuning the probe to match your particular sample.

An exception to this rule occurs when switching from "normal" organic solvents to strongly ionic samples, such as a water solution with 1M buffer. If the probe is tuned for an organic solvent, such as $CDCl_3$, and a strongly ionic sample is then inserted, you may find a lengthening in the 90° pulse width by a factor of two or three.

For single-pulse experiments, this detuning of the probe will cause an apparent deterioration of signal-to-noise (since you will only be using a 30° pulse, for example, when you intended to use a 90° pulse) but in many cases this effect will be small.

2.5 Quarter-Wavelength Cable

When a large change is made in the frequency of the observe nucleus on broadband systems, such as switching from ^{13}C to ^{15}N, an additional change is made in the quarter-wavelength cable, a coiled cable located on the system as follows:

- Attached to the preamplifier housing for 500–900-MHz systems.
- Attached to the inside of the left magnet leg on the *MERCURYplus/-Vx*, or on the side of the Magnet Interface Box.
- Attached to the inner face of the magnet console interface unit as part of the observe circuitry on other systems.

The quarter-wavelength cable is *not* changed for each nucleus, but only for broad ranges of frequencies (for example, 40 to 80 MHz), usually covering a factor of two (an octave) in frequency. An incorrect cable does not typically affect signal-to-noise, but may have a profound effect on the 90° pulse length.

2.6 Tuning Probes on INOVA Systems

Typically, probes are tuned using the TUNE INTERFACE, shown in Figure 3.

- "TUNE INTERFACE," this page
- "Tuning a Probe," page 25

TUNE INTERFACE

The panel is located either on the magnet-console interface or on the dual-preamplifier assembly. The panel contains the following displays, readouts, and ports:

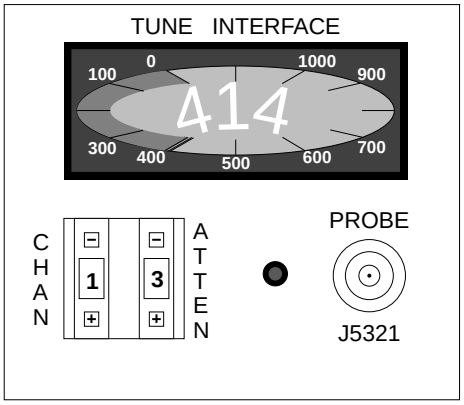


Figure 3. TUNE INTERFACE Panel

- At the top of the panel is the TUNE INTERFACE display, a rectangular liquid-crystal display that shows a numerical value two ways—as a digital readout and as an analog representation along the oval surrounding the digital readout.
- Below the display are two single-digit readouts labeled CHAN and ATTEN. The CHAN readout can be set to 0 for OFF or to the channel being tuned (1, 2, 3, etc.), and the ATTEN readout is the amount of attenuation (analogous to the TUNE LEVEL knob on older systems). The attenuation is selected in units of 10 dB. The maximum attenuation is 79 dB, which is selected by a setting of 8. Above and below each readout are buttons for setting the value of the readout.
- At the lower right of the panel is a red indicator light and a BNC probe port labeled PROBE J5321.

Tuning a Probe

Tuning a probe on a UNITY*INOVA* system using the TUNE INTERFACE panel takes the following steps:

1. Set up the spectrometer to observe the nucleus of interest.

   Often, the system is already set to the correct nucleus; if not, proceed as if you were setting up an experiment.

2. Change the rf cable attached to the probe channel you plan to tune. No filters should be in-line during the tuning procedures.

   Warm probes – Disconnect the cable from the PROBE J5311 port on the broadband preamplifier or the $^{1}H/^{19}F$ J5301 port on the preamplifier. Connect this cable to the PROBE J5321 port on the TUNE INTERFACE panel. If a TUNE OUTPUT J5323

BNC connector is present, disconnect the cable from the OUTPUT port (J5312 or J5302) and connect it to the TUNE OUTPUT J5323 port.

Cold Probes – refer to the *Cold Probe Installation* manual.

3. Two methods are available to set the tune frequency. Until you set up the tune frequencies with one of the methods (`su` or `tune`), the TUNE INTERFACE panel will not work after powering on or after resetting the acquisition console.
   - The first method is to enter `go` or `su`. Each time `go` or an `su` executes, the console receives a frequency for each channel defined for the experiment. This frequency also becomes the one used during tune. The table below shows the relationships between the channel selected and the associated parameters:

Channel 1	`tn`	`sfrq`	`tof`
Channel 2	`dn`	`dfrq`	`dof`
Channel 3	`dn2`	`dfrq2`	`dof2`
Channel 4	`dn3`	`dfrq3`	`dof3`

     For descriptions of these parameters, refer to the *Command and Parameter Reference*.
   - The second method is to enter `qtune` for a swept display. Refer to Appendix D, “Probe Tuning with qtune,” page 267 for details. The settings remain in effect until the next `go` or `su` command executes.
4. Press the **CHAN** buttons until the readout is the number of the rf channel you want to tune. Start with channel 1.

   This turns on the tuning function for the channel. The TUNE INTERFACE display should shows a number, and the red indicator light should not flash. (If the light flashes, check the connector to the cable for an improper connection.)
5. Press the **ATTEN** buttons until the readout is **6, 7, or 8**.
6. If necessary, insert the appropriate sticks into the probe. Refer to the probe installation manual as to which sticks are needed to tune to the desired nucleus.
7. Tune the probe. As the probe gets closer to being tuned, the number on the TUNE INTERFACE display will decrease.
8. Press the **ATTEN** button until the readout is **8**, to increase the tuning level sensitivity. Continue tuning until the number displayed on the TUNE INTERFACE display is as close to zero as possible.
9. Disconnect the tuning function by pressing the **CHAN** buttons until the readout is **0**. (During normal operation, CHAN must be set to 0 or acquisition is *not* allowed.)
10. Reconnect the rf cables to their original position.

    Disconnect the cable from PROBE J5321 port on the TUNE INTERFACE panel. Connect this cable to the PROBE J5311 port or the $^{1}H/^{19}F$ J5301 port, whichever was the original port. Then disconnect the cable to the TUNE OUTPUT J5323 port and connect it to the OUTPUT port (J5312 or J5302, as appropriate).

    At this time, the red indicator light should turn off.
11. Repeat the steps above for each channel on the system.

For further information about probe installation and tuning, refer to the probe installation manual that shipped with your probe.

2.7 Tuning Probes on *MERCURYplus/-Vx*

On *MERCURYplus/-Vx* systems, `btune` turns on the observe transmitter, directing about 0.5 watts of rf at frequency `sfrq` + `tof` to the probe. Before using `btune`, switch the cable on the magnet leg. `tuneoff` turns off the transmitter.

- "Observe Coil Tuning," this page
- "Decoupler Coil Tuning," page 27

CAUTION: **Only qualified service personnel should tune the lock channel. An incorrectly tuned lock channel can damage equipment and cause erratic results.**

Observe Coil Tuning

This example shows how to tune to ^{13}C. To tune to another nucleus, enter the name of that nucleus instead of `'C13'` in step 1.

1. Join an appropriate experiment and enter **`tn='C13' su`**.
2. Move the cable from the BB connector (J5302) to the TUNE connector, J5402 or for the Magnet Interface box from BB Probe, J6001, to J5402, Tune.
3. Move the cable from the BB connector, J5603 on the rear of the magnet leg to the TUNE connector, J5604, just above it or for the Magnet Interface box from J5603, Lo Bnd Tx, to J5604, Tune.
4. Enter **`btune`**.
5. Turn the meter switch to the **Tune Mode** position.
6. Adjust the **Tune knob** for a mid-range reading.
7. Turn the observe coil tuning rod until the meter reaches a minimum reading.
8. Turn the observe coil matching rod for a minimum meter reading. Adjust the **Tune knob** if needed.
9. Switch back and forth between the observe coil tuning rod and the observe coil matching rod until you achieve an absolute minimum meter reading. After a minimum is obtained, enter **`tuneoff`**.

   If the tuning is far off, it may be better to turn each rod past the minimum meter reading before turning the other rod.
10. Return the two cables to their original positions and turn the meter switch to **Spin Mode**. If a TUNE OUTPUT J5323 BNC connector is present, disconnect the cable to the TUNE OUTPUT J5323 port and connect it to the OUTPUT port (J5312 or J5302 as appropriate).

Decoupler Coil Tuning

CAUTION: **Before tuning the decoupler coil, check that air is flowing through the probe dewar and decoupler cooling line, cooling both the sample and decoupler coil. Excessive heat will damage the sample and the decoupler tuning capacitors. During VT operation, the probe dewar requires nitrogen for cooling. For maximum power, use at least 20 CFH or 9.5 LPM.**

1. Join an appropriate experiment and then enter **`tn= H1 su`** (for ^{1}H) or **`tn= F19 su`** (for ^{19}F).
2. Move the proton probe cable from the ^{1}H connector, J5102, on the magnet leg to the TUNE connector,J5402, or for the Magnet Interface box from J5103, Hi Bnd Preamp, to J5402, Tune.
3. Move the proton cable in the rear of the magnet leg to the coaxial tuning jack labeled TUNE, J5604, or for the Magnet Interface box from J5602, Hi Bnd Tx, to J5604, Tune.
4. Enter **`btune`**.
5. Turn the meter switch to the **Tune Mode** position.
6. Adjust the **Tune control knob** for a mid-range reading.
7. Turn the **decoupler coil tuning control** to obtain a minimum tuning meter reading. Adjust the **Tune knob** as needed. Once a minimum is obtained, enter **`tuneoff`**.
8. Return the two cables to their original positions and turn the meter switch to **Spin Mode**.

Chapter 3. Experiment Setup

Sections in this chapter:

These sections are in the same order as typically performed by most users.

3.1 Selecting an Experiment

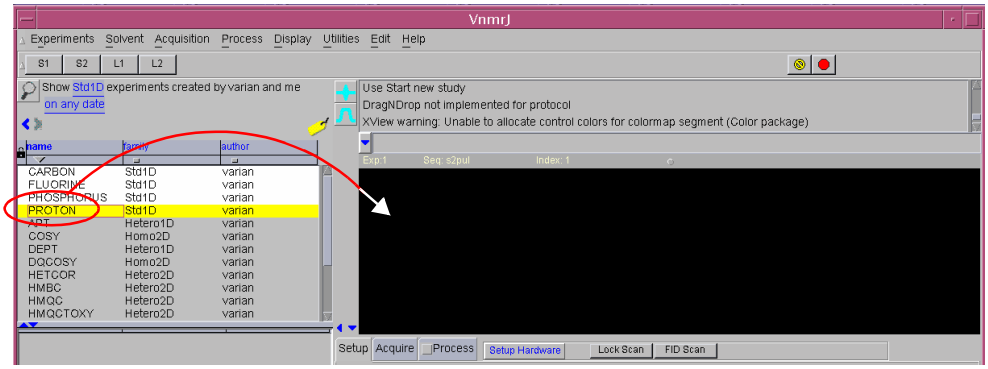
VnmrJ provides you with several ways to choose and load an experiment. This section describes two, from the menu bar or from the Locator. After an experiment is selected, VnmrJ reads and loads the standard parameters and then reads the probe file and loads the probe calibrations.

Menu

Click on **Experiments** in the menu bar, then click on a 1D experiment in the list. The list of experiments contains some submenus.

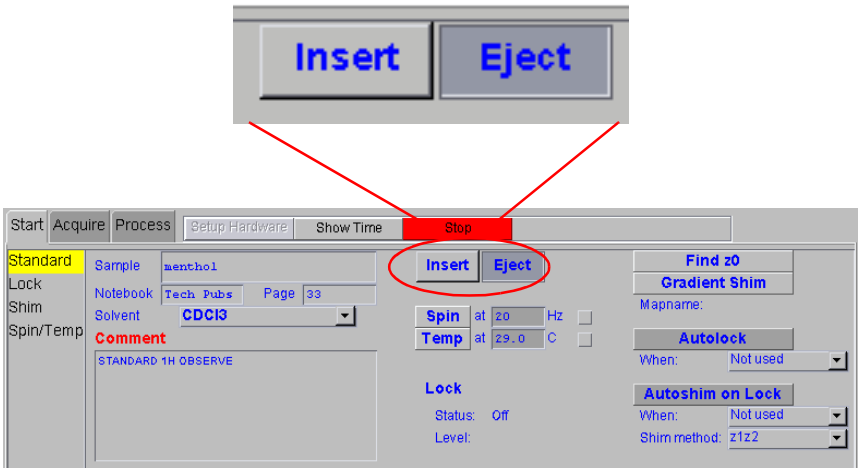
Locator

Click on an experiment in the Locator and drag it into the graphics canvas (or double-click on the experiment).



3.2 Ejecting and Inserting the Sample

The spectrometer is equipped with hardware and software to provide computer control of sample ejection, insertion, spinning, locking, and shimming. This section covers computer-controlled sample ejection and insertion.



Manual control of ejection and insertion is also provided on each of these systems to enable you to withdraw samples if necessary, but we strongly recommended that you use computer control for maximum reproducibility and safety.

To Eject a Sample

You should always eject first (even if no sample is in the magnet) to start airflow to carry the sample. The eject air is turned on and, under computer control, the sample, if present, rises back to the top of the upper barrel. You can now remove the sample and replace it with another sample.

Using the Start Tab

The Insert and Eject buttons are on the Start tab.

1. Click the Start tab to open the setup tab.
2. Click the **Eject** button.

Manual Ejection

The manual eject button is *used only in emergencies*.

- Press the black button on the top of the left leg of the magnet or inside the Magnet Interface Box.

To Insert a Sample

When inserting a sample, the sample tube gradually lowers down the upper barrel under computer control. After a five-second delay, the bearing air is turned off momentarily, allowing the turbine to seat properly.

The two-stage sample insertion operation is provided for safety reasons, particularly when working with the 5 mm upper barrel, which uses smaller turbines. Because the tube itself is used as the bearing surface in this barrel, the tube must drop down the barrel slowly enough to avoid breaking when contact is made with the conical guide. The second stage drop then permits the tube to slide into the bearing cylinder. Operation using the larger upper barrel, which can hold 5-, 10- and 16-mm tubes, is less susceptible to these problems because the turbine makes initial contact and alignment before the sample tube encounters any close tolerance.

Using the Start Tab

1. Perform a sample ejection (even if no sample is in the magnet) to start airflow to carry the sample.
2. Insert the sample by placing it in the top of the upper barrel.
1. Click the Start tab to open the setup tab.
2. Click **Insert**.

Manual Insert

Used only in emergencies.

1. Press and hold the black button on the top of the left leg of the magnet or inside the Magnet Interface Box.
2. Insert the sample by placing it in the top of the upper barrel to start airflow.
3. *Slowly* release the black button to slowly drop the sample. When the button is completely released, close off the top of the upper barrel with your hand for a second to properly seat the sample.

3.3 Spinning the Sample

You can adjust spin rate from the input window or the Acquisition window. Typical spin rates are 15 for 10-mm tubes and 20–26 for 5-mm tubes.

When a sample is inserted, the last entered spin rate is used to regulate sample spinning. The actual spin rate is indicated three ways:

- The Spin chart displays button on the hardware bar displays a history of the sample spin rate.
- In the Acquisition Status window, the actual rate is given as well as a spin regulation indication.
- In the remote status unit, the magnet leg, or the Magnet Interface Box on the *MERCURYplus/-Vx* the spin rate is shown by the spin light:

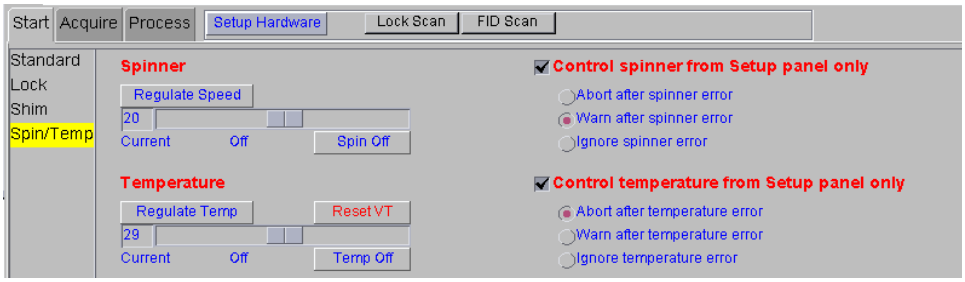
  If light is off, the sample is not spinning.

  If light is blinking, the sample is spinning but not at the last requested rate.

  If light is steady, the spin rate is being regulated at the last requested rate.

Using the Start Tab

The Spin/Temp page is under the Start tab.



1. Click the **Start** tab. Select the **Spin/Temp** page.

   The controls for changing spinning speed consist of an entry field, a slider bar, and a button for disabling or enabling spinning.

2. Adjust the spinning speed with either of these methods:
   - Enter a spin rate in the text entry field.
   - Drag the slide control. The value changes proportionally as the mouse moves.
   - Click in the slider bar to move the slider by one increment.

Using the Spinner Control Window

The `spinner` command opens the Spinner Control window for control of sample spinning, see Figure 4. From this window, the spinner can be started or stopped, and experiment control of spinning can be turned off. That way, if an experiment you just joined has the `spin` parameter set to a value other than the current spinning speed, and you forget to set `spin` to `'n'` and type `go`, the spin speed will not be changed.

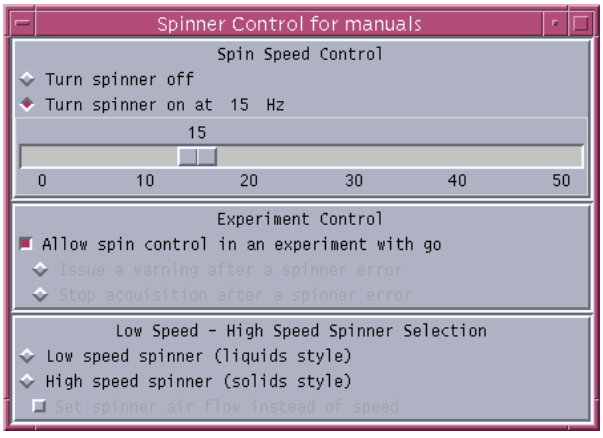


Figure 4. Spinner Control Window (`spinner` Program)

On the UNITY*INOVA* system, high-speed, solids-style sample spinning and low-speed, liquids-style sample spinning are both under computer control. The `spinner` program can be used to select these spinner types.

1. Enter the command `spinner` in the input window.

   The Spinner Control window appears.

2. Set the desired spinning speed by clicking the diamond next to Turn Spinner On and enter the speed value in Hz.

3. To disable experimental control of spinning, click the button next to Allow spin control in an experiment with go. A button that is indented (and red) is selected.

   When experimental control of spinning is disabled, you can choose how spinner errors are handled by the system—a warning is issued or acquisition is stopped.

4. Select a spinner mechanism type, low-speed (liquids-type spinning), high-speed (solids-type spinning), or automatic spinner type selection.

   If you select high-speed, you can choose to set the spinner air flow instead of the speed.

   If you select the automatic spin selection, you must also enter threshold values that tell the system when to switch to solids-type spinning.

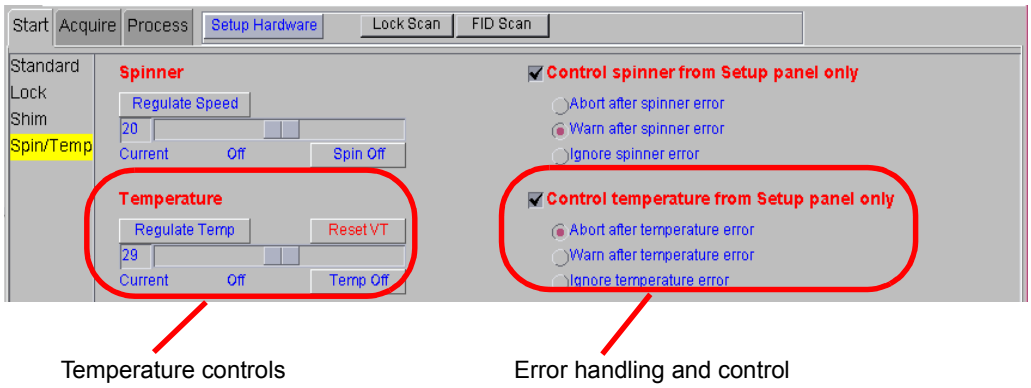
For the high speed spinner probes (e.g., MAS), the following safety measures have been implemented to prevent rotor and stator damage:

- The air flow selected from the spinner window is ramped to the new value at a safe rate.
- If the spinner speed drops to zero and the spin setting is nonzero, the air flow will be shut off. This measure prevents spinner runaway if the tachometer fails.
- If for some reason the spinning speed cannot be reached, the air flow will be shut off. This measure prevents the continued spinning of a crashed rotor with more air flow.

3.4 Sample Temperature

You can set the temperature for VT control in the **Standard** page and regulate the temperature by clicking on **Spin/Temp** page.

For more information on using the Variable Temperature module, refer to Chapter C, “Variable Temperature System,” page 261. The following steps describe a typical operation sequence:



1. Open the **Spin/Temp** page under the **Start** folder.

2. Set the desired temperature by entering a value or using the slider, and click **Regulate Temp**.
3. Set up the acquisition for the experiment as usual, using the Acquire folder.
4. Click the **Temp** button on the hardware bar to display the temperature display chart.
5. Start temperature control with the **Setup Hardware** button on the Start folder, or with the Acquire button on the Acquire folder. These commands act as follows:

Setup Hardware	the temperature control and acquisition hardware controls are set and the sample temperature changed to the desired temperature. The experiment is not started when the desired temperature is reached. After waiting the delay time (seconds), the Acquire button must be pushed must be issued before data acquisition begins.
Acquire	the same actions as Setup Hardware occur, except that after reaching the desired temperature, the system waits the delay time, then begins the pulse sequence and data acquisition. The delay time occurs every time the temperature is changed under program control.

After clicking Setup Hardware or Acquire, the selection of the VT gas routing occurs, and the VT controller begins to control the gas temperature in the probe at the requested temperature. The temperature readout will begin to change and the VT indicator light will begin flashing. At this time, if the requested temperature is below ambient, add coolant liquid to the coolant bucket.

After the temperature reaches the requested temperature (it may initially overshoot), the VT indicator light stays on steadily. A sample that could not be handled at ambient temperature can now be transferred into the probe. The VT readout is the temperature of the cooling/heating gas and may be different from the true sample temperature. The exact temperature of the sample is correctly determined by a calibration curve that must be constructed for each probe, and must include flow rate and equilibration time. Refer to the *VT Accessory Installation* manual for the NMR calibration method.

CAUTION: **Do not use aromatic, ketone (including acetone), and chlorinated solvents in the coolant bucket. Such coolant media attack the standard polystyrene bucket. Another type of container must be used (not supplied by Varian).**

CAUTION: **Operating the system with the coolant bucket filled with liquid nitrogen and with the temperature greater than the value of VT cutoff results in the condensation of liquid nitrogen inside the exchanger coil tube. If the exchanger coil is then warmed above –210°C or if nitrogen gas is passed through the coil (when temperature is less than VT cutoff), very cold liquid nitrogen is forced through the transfer line and into the probe. This will cause a sudden pressure surge in the transfer lines and probe as the liquid nitrogen boils, and it can blow the flexible connector apart. If the liquid nitrogen reaches the glass components of the probe and sample tube, the glass will probably break. Instrument damage can be avoided by following these precautions:**

Do not immerse the exchanger coil in liquid nitrogen when no nitrogen gas is flowing through the coil.

Do not stop the VT nitrogen gas flow while the exchanger is immersed in liquid nitrogen.

Arrayed VT experiments that have a temperature range from above VT cutoff to below VT cutoff should be set up starting at the lowest temperature and ending at the highest temperature. When the experiment passes the VT cutoff crossover, remove the liquid nitrogen coolant.

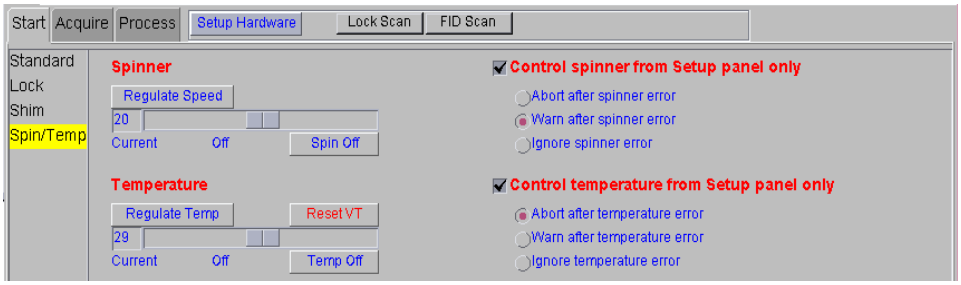
To avoid water in the exchanger when the low temperature experiment is complete, warm up the exchanger by removing it from the liquid nitrogen and maintain a flow of dry nitrogen until room temperature is reached.

WARNING: **Sealed samples containing volatile materials can rupture when heated, resulting in potential injury, exposure, and equipment damage. Before running sealed samples at elevated temperatures, heat the samples in an oven at a temperature higher than the highest temperature during the experiment. If the tube ruptures while in the probe, the glass components and insert coil will probably be destroyed.**

WARNING: **Sealed samples containing materials with boiling points at or below room temperature can rupture as the sample warms, causing potential injury, exposure, and equipment damage. Equilibrate the probe to a temperature below the sample boiling point before the sample is placed into the probe.**

3.5 Spin and Temperature Error Handling

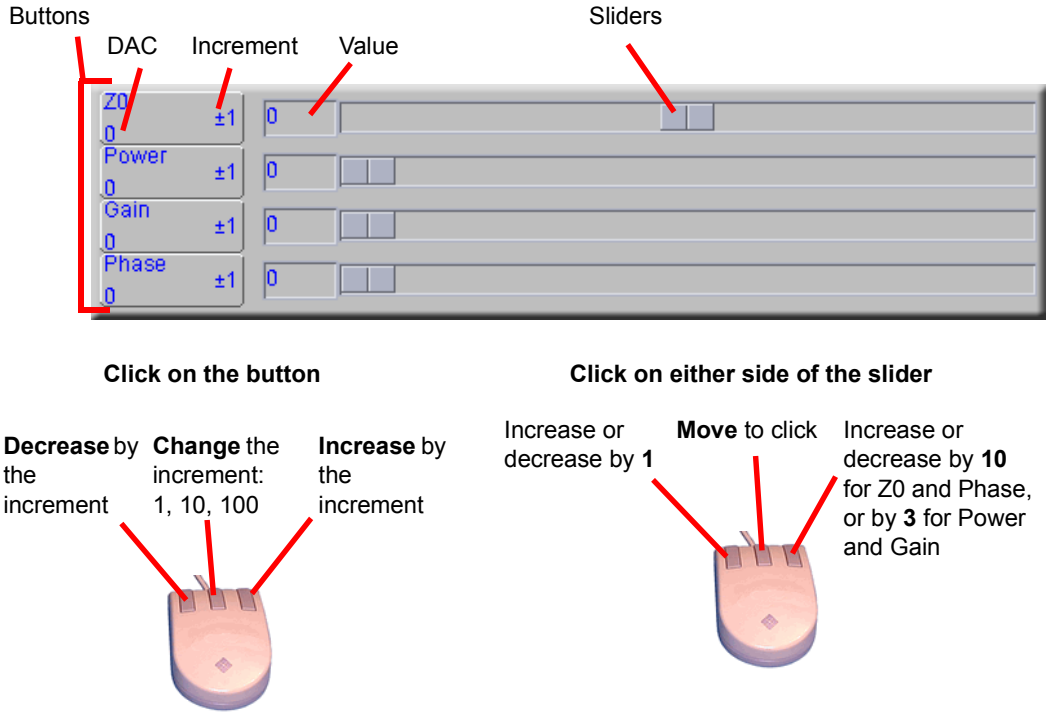
Use the **Spin/Temp** page of the **Start** tab to select spin and temperature error handling. The provided choices specify the action to be taken based on spinner and temperature failure. Also use the Spin/Temp page to specify whether spinning and temperature can be controlled on panels other than the Start tab.



- `Ignore spinner/temperature error` – stops any system checking so that acquisition continues regardless of the spin speed or temperature.
- `Warn after spinner/temperature error` – makes the system check the spin speed and temperature. A warning message is added to the log file if the if the spin speed or temperature is set to a particular value and the spin speed or temperature goes out of regulation; however, acquisition is not stopped.
- `Abort after spinner/temperature error` – makes the system check the spin speed and temperature. Acquisition is halted if spin speed or temperature is set to a particular value and the spin speed or temperature goes out of regulation.

3.6 Working with the Lock and Shim Pages

This section provides some hints for working with the Lock and Shim buttons and Lock sliders. The Lock and Shim buttons (z0, Lk Power, Lk Gain, Lk Phase, and z, x, and y shims) provide on-the-fly configuration. The slider values can be moved with the mouse or entered directly.



To Change the Increment

1. **Middle-click** the button until the increment value that you want to change is displayed. The defaults are 1, 10, and 100.
2. **Shift-middle-click** the button, enter a new value, and press **Return**.

To Change the DAC Value

1. **Shift-left-click** on the button.
2. Enter a new value and press **Return**.

3.7 Optimizing Lock

Under computer control, the lock system maintains a constant field at the sample as the static field generated by the superconducting magnet drifts slowly with time or changes due to external interference. Locking makes the resonance field of the deuterium in the deuterated solvent coincide with the lock frequency.

The lock level can be viewed by clicking the Lock button on the hardware bar:

The entire lock optimization process can be skipped if optimum lock parameters are already known for a particular solvent and probe combination. Values for these parameters can be entered as part of a macro or using normal parameter entry (e.g., by entering

`lockgain=30 lockpower=24`). These parameters do not take effect until an `su`, `go`, or equivalent command is given.

If automatic shimming is to be used, it is important to obtain an optimal lock signal. Manual adjustment often is done to achieve the maximum lock amplitude. This can result in a partly saturating condition, and a true non-saturating power is usually 6 to 10 dB lower. The response of the lock level is governed by the T_1 of the deuterated lock solvent as well as the magnet-determined or chemical exchange-determined T_2^* of the solvent. This T_1 can vary widely, from about 6 seconds for acetone-d_6 to about 1.5 seconds for $CDCl_3$ and lower for more viscous solvents. To allow a reliable, repeatable selection of lock power, automatic optimization may be used.

Finding Z0 and Establishing Lock

You can find Z0 and establish the lock either manually or using Autolock. Both methods are accessed through the **Standard** page of the **Start** tab, see Figure 6.

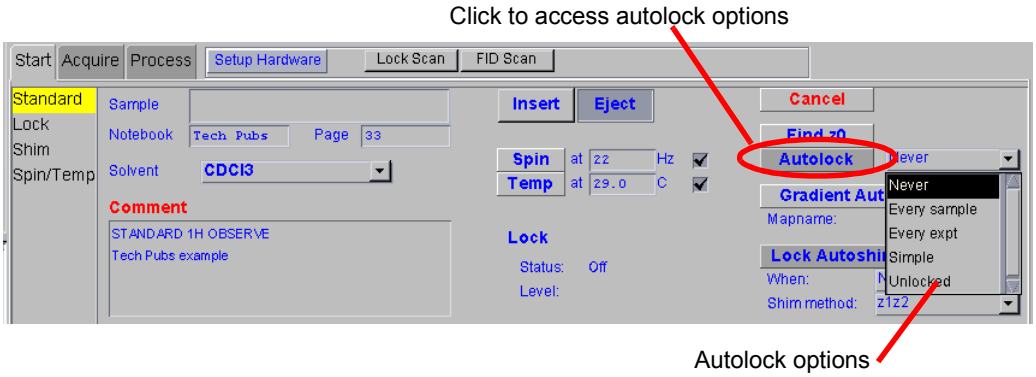
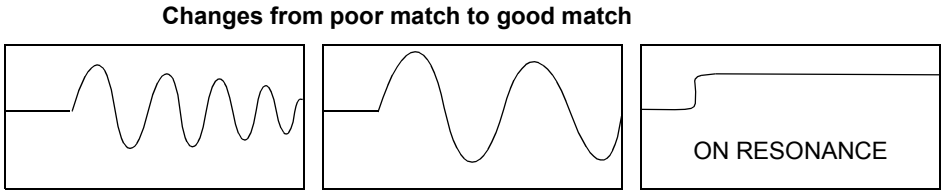


Figure 5. Autolock Options

Manual or Simple Method

Establishing lock using simple or manual locking on the Lock page. The line that crosses the spectral window represents how close the deuterium resonance field is to the lock frequency. When the two are matched, the line should be flat (with perhaps some noise,

depending on the lock gain and lock power). The poorer the match, the greater the number of sine waves in the line.



1. Click on the **Lock** page in the **Start** tab. The page in Figure 6 appears.
2. Click on either **Spin On** or **Spin Off**.
3. Click **Lock Scan** to open the lock display.
4. Find Z0 by clicking on and dragging the Z0 slider bar until lock signal is on resonance.
5. Adjust the lock power, gain, and phase by clicking on and dragging the slider bars, or click the button, shown in Figure 6.
6. Click the **Lock On** button.
7. Click **Lock Scan** again to close the lock display.

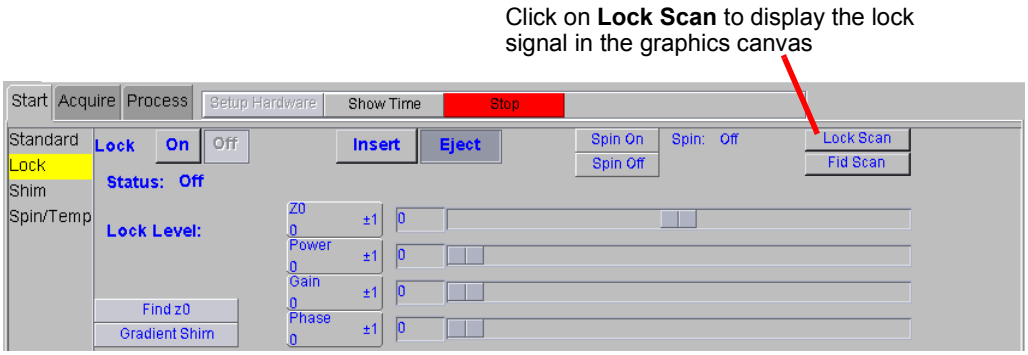


Figure 6. Lock Page

AutoLock

1. Click on the **Standard** page of the **Start** tab. The page in Figure 5 appears.
2. Click on either **Spin On** or **Spin Off**.
3. Choose **Find Z0** or **AutoLock**.

   Clicking on the button next to **Autolock** opens a drop-down menu of options, as shown in Figure 5.
4. Select an option, click on the AutoLock button, and the spectrometer will find Z0 and make all specified adjustments.

Lock Power, Gain, and Phase

Under computer control, lock power, gain, and phase are set by the lock parameters—`lockpower`, `lockgain`, and `lockphase`—with the following limits and step sizes:

- On UNITY*INOVA*, lock power is 0 to 68 dB (68 is full power), lock gain is 0 to 48 dB, and lock phase is 0 to 360 degrees. Step size for power and gain is 1 dB; step size for lock phase is 1.4 degrees.

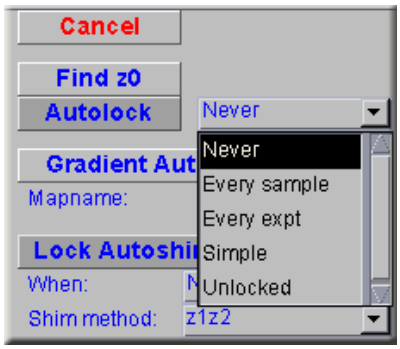
  On *MERCURYplus/-Vx*, lock power is 0 to 40 dB (40 is full power), lock gain is 0 to 39 dB, and lock phase is 0 to 360 degrees. Step size for power and gain is 1 dB; step size for lock phase is 1.4 degrees.

The Z0 field position parameter `z0` holds the current setting of the Z0 setting. The limits of `z0` are –2047 to 2047, in steps of 1, if the parameter `shimset` is set to 1, 2, or 10, and –32767 to +32767 if `shimset` is set to 3 through 9. On *MERCURYplus/-VX* systems, `shimset` is 10.

Lock Control Methods

A number of methods are available for controlling lock on the **Standard** page in the **Start** tab:

- Leave lock in the current state.
- Run an experiment unlocked.
- Use simple autolock.
- Use optimizing autolock.
- Perform full optimization of lock.



Each method is discussed in the following separate sections. Additional sections discuss error handling and lock loop time constant control.

Leaving Lock in the Current State

Set **Autolock** to **Never**.

If simple or optimized Autolock was previously selected, lock is established upon insertion of the new sample. If simple lock was previously selected, the system only locks if the new sample has the same lock solvent.

Running an Experiment Unlocked

Set **Autolock** to **Unlocked**.

Lock is deactivated at the start of acquisition.

Simple Autolock

Set **Autolock** to **Simple**.

Simple Autolock is activated at the start of acquisition if it has not already been activated.

Software simple Autolock searches for the correct Z0 value in software, but does not adjust lock power, gain, or phase.

Optimizing Autolock

Optimizing Autolock uses a sophisticated software algorithm to search the field over the full range of Z0 (as opposed to hardware simple Autolock), captures lock, and automatically adjusts lock power and gain (but not lock phase).

Set **Autolock** to **Simple**.

If **Autolock** is set to **Simple** at the beginning of each experiment (each initiation of an acquisition), the system searches for the lock signal if necessary, and then optimizes lock power and gain (but not phase) whenever an acquisition is initiated with `go`, `ga`, `au` or with any macro or menu button using the `go`, `ga`, or `au`.

If **Autolock** is set to **Every Sample**, the same process as **Simple** occurs but only if the sample has just been changed under computer control and acquisition is started (when manually ejecting or inserting a sample, the software cannot keep track of the action and **Every Sample** has no effect).

If `z0` is inactive and you start an autolock operation, `autolock` searches for the lock signal by changing the lock frequency.

Be aware that spectrometer frequencies are computed from the lock frequency, so if the lock frequency changes as a result of an Autolock operation, frequencies for that acquisition will be off by the amount of that change. Switching from chloroform to acetone requires a change in the lock frequency of about 5 ppm, which can cause problems in precision work. Changing lock frequency is only a problem when you select Autolock with the `alock` parameter. It is *not* a problem for the lock experiment, since, by definition, the lock experiment is complete once the autolock operation is completed.

Full Optimization

Full optimization is the most complete optimization of lock parameters. A fuzzy logic autolock algorithm automates the parameter control process in order to find the exact resonance and the optimum parameters (phase, power, gain) automatically and quickly with high reliability. Fuzzy rules are used in the program to find the exact resonance frequency and for adjusting power and phase. The fuzzy rules are implemented at different stages of the autolock process. First, the software finds the resonance. If the exact resonance cannot be found, phase and power are adjusted and the software looks for the exact resonance again. The software then optimizes the lock power to avoid saturation, optimizes the lock phase, and optimizes the lock gain to about half-range.

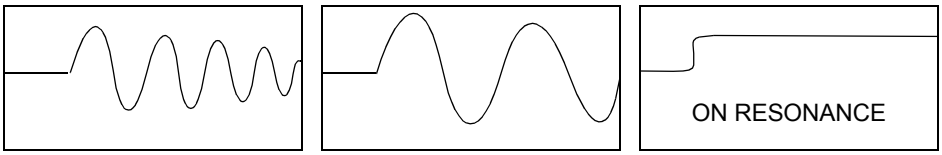
RF frequencies, decoupler status, and temperature are also set during full optimization.

Simple Locking

Establishing lock using simple or manual locking uses the LOCK display. The line that crosses the spectral window represents how close the deuterium resonance field is to the lock frequency. When the two are matched, the line should be flat (with perhaps some noise, depending on the lock gain and lock power). The poorer the match, the greater the number of sine waves in the line.

The following procedure for finding lock manually is typical:

1. Make sure a sample is inserted and seated properly. Spinning helps but is not required.
2. In the LOCK menu at the top of the window, click the off button.



3. Using the slide control, –1+, –4+, –16+, and –64+ buttons, or entering values directly, turn up lockpower and lockgain, and look for some sinusoidal variation in the signal.

   The actual value needed for lockpower and lockgain depends upon the concentration of the deuterated solvent, the nature of the deuterated solvent—the number of deuterium atoms per molecule—and the relaxation time of the deuterium. At this point, do not be too concerned about optimizing power and gain; just look for a sine wave.

4. If you see no sine wave (perhaps just noise), click on the –16+ button for Z0 until some discernible wave appears.

5. If you know the concentration of the lock solvent is high, say greater than 50%, turn down the lock power.

   If the lock power is too high, the deuterium nuclei become “saturated,” the signal oscillates (goes down and then back up), and it is difficult to establish lock. The correct amount of lock power is difficult to determine, but it is helpful to remember that acetone is more easily saturated than most solvents.

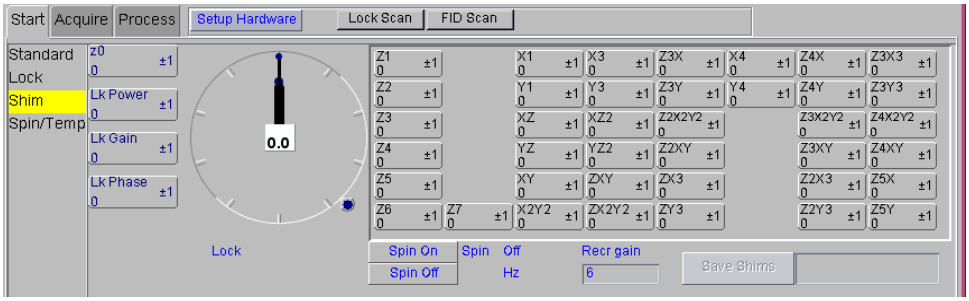
6. Adjust Z0 until the signal changes from a sine wave to an essentially flat line. If the solvent is concentrated, the line may start to move up on the screen as the lock condition is approached.

7. In the LOCK menu, click the on button.

   If the lock signal displays a dip at the point where it starts or the signal slopes downward, incorrect lock phase is the probable cause.

3.8 Adjusting Shims

Shim coils produce small magnetic fields used to cancel out errors in the static field. In shimming, the current to the shim coils is adjusted to make the magnetic field as homogeneous as possible. Computer-controlled digital-to-analog converters (DACs) regulate the room-temperature shim coil currents. Every time a new sample is introduced into the magnet or probe is changed, it is necessary to readjust the shims.



Using the Shim Buttons

This section provides hints for using the shim buttons in the **Study** panel.

- Clicking the left mouse button on the button decrements the value by the amount of the "increment" displayed on the right side of the button.
- Clicking the right mouse button increments the value.
- Clicking the middle mouse button toggles the amount of the increment between one of three values.
- Holding down the **Shift** key and clicking the left mouse button enables you to type in a new current value. Press **Return** to make the value take effect.

Holding down the **Shift** key and clicking the middle mouse button enables you to type in a new value of the increment.

Loading a Shim File

You can load a saved shim set by dragging and dropping a shim file from the Locator to the shim buttons area of the Shim page.

1. Click the **Locator Statements** button (magnifying glass icon).
2. Select **Sort Shimsets**. You can also list the shimsets by probe or filename.
3. Select a shim set and drag-and-drop it onto the shim buttons area or graphics canvas of the Shim page.

Saving a Shim File

Save the shim values by entering a name, pressing return, and clicking Save Shims.

1. Enter a file name in the field next to the Save Shims button, and press **Return**.
2. Click the **Save Shims** button.

Shim Gradients

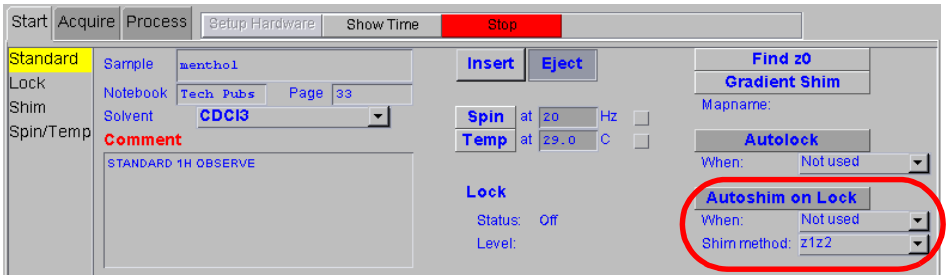
The shims are actually printed coils wrapped round a cylindrical form that encloses the NMR probe. A coil (or sum of coils) whose field is aligned along the axis of the magnet is called a Z axial shim gradient (Z1, Z2, Z3, etc.). Coils whose fields are aligned along the other two orthogonal axes are called X and Y radial shim gradients (X1, XY, X2Y2, Y1, YZ, etc.). The field offset coil Z0 ("zee-zero") alters the total magnetic field.

Each shim gradient is controlled by its own parameter; for example, the X1 shim gradient is controlled by a parameter named `x1`.

Depending on the value of the `shimset` parameter, shim values range from –2047 to +2047 or from –32767 to +32767, with a value of zero producing no current.

Automated Shimming (Autoshim)

Like locking, shimming can be done manually using the controls on the Shim page. Automated shimming is often preferred, however. It can be set up from the **Standard** page of the **Start** tab:



- *Fully Automatic Autoshim* – Under **Lock Autoshim** set **When:** appropriately so that autoshimming starts at the beginning of: **Every sample**, **Every experiment**, **Every FID**. Setting When to Never disables automatic autoshim.
- *Background Autoshim* – Under **Lock Autoshim**, set **Shim method** appropriately, for example, **z1z2**.

Autoshim Control

In the fully automatic Autoshim mode, the Lock Autoshim area of the **Standard** page in the **Start** tab controls the automatic shimming activity. Lock Autoshim can specify no shimming, shimming at the start of data acquisition, etc. Shimming in each case is initiated by some form of data acquisition.

For shims on the lock signal, set When to one of the following:

- Never – no automatic shimming is performed.
- Every sample – automatic shimming only at the beginning of the first experiment, following the change of a sample using the automatic sample changer.
- Every expt – automatic shimming for the experiment prior to data acquisition.

For shims on the FID, set When to one of the following:

- Every FID – automatic shimming is done prior to the data collection of each new array member in a multi-FID experiment (not available on *MERCURYplus/Vx*).

Autoshim Methods

Background Autoshim is controlled by the **Shim method** field. This is a complete background Autoshim method that provides no interaction with the operator whatsoever. The type of automatic shimming to be done during routine sample changes depends on the level of homogeneity required on any particular sample, the change in sample height, and the maximum time desired for shimming.

- For average homogeneity needs with samples which are either long or all of identical height, **z1z2** shimming is usually sufficient.
- If sample height might vary, the method **allzs** has been found to be the most reliable, at the expense of greater time spent in shimming. This method shims first Z1, Z2, and Z4, then Z1, Z2, and Z3, and finally Z1 and Z2.

No matter how the automated shimming is initiated, it is controlled by two aspects of the shimming process (each aspect is discussed in detail in subsequence sections):

- *Quality or criterion for shimming* – The quality of the field homogeneity when the automated shimming is started must be given and the quality of the homogeneity at the conclusion of the shimming must be specified.
- *Method used to shim* – For routine "tweaking" of the resolution, adjustment of Z1 and Z2 is sufficient, and the method z1z2 would be selected.

Automatic shimming relies on a non-saturating lock signal on which an optimizing process can be performed. If too high a lock power is used, the shimming process can become unreliable since it may be chasing a "moving target." Since it is customary to increase lock power until the lock level maximizes, if done manually, it is clear that lock signals will be partially saturated. This follows directly from the shape of a saturation curve where signal amplitude increases linearly with lock power until a point where it flattens out, becomes oscillatory, and eventually declines.

Adjustment for maximum lock level puts the lock power near the top of this curve. The response of the lock level to changes in gradients is not as sensitive as in a non-saturating case, and therefore automatic shimming is not as reliable. Nonsaturating lock power is easily checked by determining if the lock level changes by the proper factor of two upon a change of 6 dB in lock power. Usually, acetone-d_6 must have at least 8 to 15 dB less power than $CDCl_3$, for example, to remain non-saturated.

Which Shims to Use on a Routine Basis

The following suggestions should assist you in routine shimming, especially on shim systems with a larger number of shim channels:

- *Establish and maintain lineshape* – Use Z to Z5, possibly Z6, X, Y, ZX, ZY, and possibly Z2X and Z2Y. The effects of Z7 and Z8 (and realistically Z6) are too small to see with the lineshape sample.
- *Shim a new lineshape sample of different geometry* – Use Z to Z5, possibly Z6, X, Y, ZX, ZY, and possibly Z2X and Z2Y.
- *Shim a new sample of the same geometry* – Use Z, Z2, and maybe Z3.
- *Shim a new sample of different geometry* – Use Z to Z4 and possibly Z5, X, Y, ZX, ZY.
- *Shim for water suppression* – Start with a shim set that produces a good lineshape for the same sample geometry. Next, tweak Z and Z2, and then vary Z5 and Z7 to minimize the width of the base of the water (Z and Z2 may need to be tweaked if Z5 changes by more than 100 to 200 coarse units). About 80 to 90 percent of the odd-order axial-gradient induced water width is probably dominated by Z5, with Z7 and perhaps some Z3 providing the rest.

  The even-order axial shims (Z2, Z4, Z6, and Z8) affect the asymmetry of the residual water line (using presaturation). All four of these even-order axial shims can affect the final water linewidth, with Z2 and Z4 being at about the 5 mM solute level and above, Z6 being at about the 1 mM solute level, and Z8 being at about the 0.3 mM solute level. The even-order axial shims will perform as you would expect unless the sample is less than 40 mm in length, in which case the shims still control the water linewidth but much less responsively.

  Beware of the use of Z4 to narrow an asymmetric residual water line of a sample shorter than about 40 mm. One is probably destroying the base of the standard lineshape faster than the residual water signal is being narrowed. This is because the residual water resonance width is affected more by magnetic susceptibility interfaces as the sample gets shorter. For samples under 40 mm, the iterative use of Z5-Z7 with

Z6-Z8-Z4-Z2 can narrow the residual water line, but the results obtained may be hard to reproduce on subsequent samples due to an increased sensitivity to slight changes in sample geometry.

Shimming Different Sample Geometries

Some suggestions when moving the sample:

- *Moving the same sample up* – Z, Z3, and Z5 need to become more positive.
- *Shortening and centering (moving up) the sample* – Z2 and Z4 need to become much more positive. The trends for Z and Z3 are mixed and more complex, but they tend to become a little more negative. It appears as if Z and Z3 are driven positive as the sample is pulled up, but they are driven negative faster as the sample shortens. When shimming a lineshape sample, plan on the following changes (starting from lineshape shims for a 700 µL sample at a depth 67-68 mm):

  700 µL to 600 µL: move Z2 +50 DAC units and move Z4 +250 units.
  700 µL to 500 µL: move Z2 +200 units and Z4 +600 units.

  The Z2 and Z4 changes track well with sample volume, but are relatively independent of tube depth. It is therefore easiest when changing sample geometries to make the appropriate Z2 and Z4 corrections, then adjust the more complex Z1-Z3-Z5 interactions as needed.

3.9 Shimming on the Lock Signal

When shimming on the lock, you monitor the intensity of the lock signal as you adjust the shim settings. Each shim setting controls the current through shim coils that control magnetic field gradients in different directions. It is important to know that the Z direction is parallel to the vertical direction of the probe and it is for this reason that the height of the sample in the NMR tube affects the Z shim settings rather dramatically.

1. The shim settings could be way off the mark (e.g., if the temperature has changed) and in this case the shim settings that have been most recently established for the particular probe you are using should be loaded as a starting point.
2. Click **Setup Hardware**.
3. Make sure the probe has a sample, that it is spinning at the correct speed, and that the system is locked onto the deuterium resonance from the lock solvent.
4. Check that the lock signal is not saturated. The signal is saturated if you change the lock power by 6 units (6 dB) and the lock level changes by more than a factor of two. Set lock gain as necessary.
5. Open the **Shim** page.

   Try a change of +4 or - 4 in the setting for Z1. If the lock level goes up with one of these, continue in that direction until the level is maximized (it no longer increases, but instead begins to fall).
6. Change the setting for Z2C by +4 or - 4 and continue in that direction until the level is maximized.
7. Adjust Z1 for maximized lock level; then adjust Z2 for the same. Continue this iterative process until the lock level goes no higher. If the lock level increases to 100, decrease lock gain and then continue to adjust Z1 and Z2. Lock power can be adjusted as needed.

In most cases, this concludes the shimming; however, some times it is necessary to shim the other Z controls and the non-spin shims. This must not be undertaken in the same way as the procedure above suggests. That is, if you simply go through Z1, Z2, Z3, and Z4 iteratively until the lock signal is maximized you may well find that your signal shape has degraded considerably. Hence, the following procedure is suggested for a second level of shimming:

1. After Z1 and Z2 have been adjusted for maximum lock signal, write down the lock level, adjust Z3 in one direction, say by +4, and then reoptimize Z1 and Z2 (iteratively) until the lock signal is at a maximum. Note this level of the lock signal. If the lock signal is higher than it was before (when you first wrote it down), continue changing Z3 in the same direction. Every change in Z3 must be followed by optimization of Z1 and Z2 until the lock level is at a maximum.
2. Repeat step 1 with Z4. That is, change Z4 in one direction, then optimize Z1 and Z2. If the lock level does not go up, change Z4 in the opposite direction and optimize Z1 and Z2. Continue until the highest possible lock level is obtained.
3. Repeat steps 1 and 2 iteratively until the highest possible lock level is obtained.
4. Turn the spinner off and go through the non-spin shims, one at a time, maximizing the lock level for each one. Then return and go through each again. Continue through all until the lock level is as high as possible. If lock is lost, increase the lock gain.
5. Turn the spinner on and optimize Z1 and Z2 as described above, return to the non-spins (turn the spinner off) and reoptimize these. Continue until the highest lock level is obtained.

For an ultimate check, you can now insert the lineshape sample ($CHCl_3$ in deuteroacetone for ^{1}H and dioxane in deuterobenzene for ^{13}C) and examine the line shape to make certain that you are close to the original specs, especially for the line shape at 0.55% and 0.11% of the total peak height. Also examine the height of the spinning sidebands. Refer to the *Probe Installation* manual that shipped with your probe.

3.10 Shimming PFG Systems

These procedures apply to the Performa I, Performa II, and the Performa XYZ systems. Once in operation, leave the amplifier on while using the gradient system, to allow the amplifier to reach a long-term equilibrium.

Performa I and Performa II

1. Open the **System settings** window (Utilities->System settings).
2. Next to the **Gradient amplifier** label, set **X**, **Y**, and **Z** to **off**. Click **OK**.
3. Click the **Setup Hardware** button. This button is available when the Start tab is open.
4. Verify a drop in the lock level from the small dc zero current from the amplifier.
5. Shim the system to the desired level.
6. Open the **System settings** window and set **Gradient amplifier Z** to **on**. Click **OK**. The shimming changes from the small dc offset current.
7. Click **Setup Hardware**.
8. Adjust Z1 to restore the homogeneity. The lock level should have identical stability on the meter.

This two-stage approach is not strictly necessary, but it does separate any problems that might arise.

Performa XYZ

1. Prepare the amplifier by moving the switch from STANDBY position to ON.
2. Open the **System settings** window (Utilities->System settings) and set **Gradient amplifier X**, **Y**, and **Z** to **on**. Click **OK**.
3. Click the **Setup Hardware** button. This button is available when the Start tab is open.

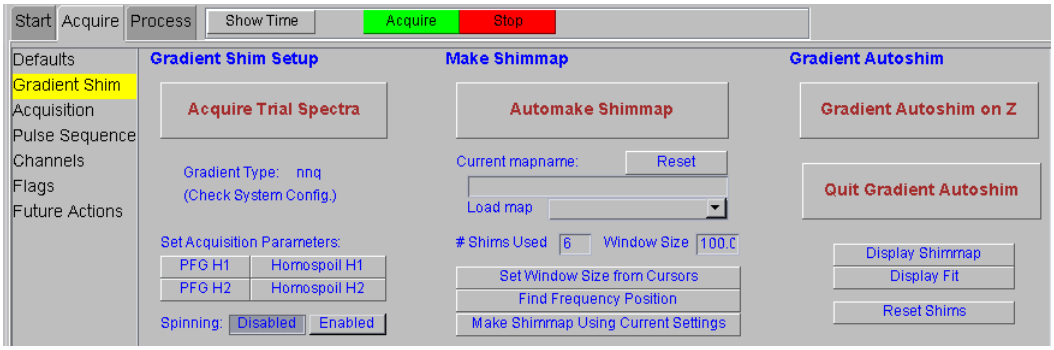
   The yellow RUN lights turns on. Shim the system to the desired level.

Chapter 4. Gradient Shimming

Sections in this chapter:

4.1 Gradient Autoshimming

Gradient autoshimming provides rapid, automatic adjustment of room-temperature shims. It is a very reliable way to set high-order shims, eliminating many hours previously spent on shimming. Typical gradient autoshimming time is only a few minutes, and all steps are done with a few clicks of a mouse button.



Gradient autoshimming is implemented for use with the axial gradients (Z-gradients). For optimal gradient shimming, a PFG amplifier and probe are recommended for their fast gradient recovery performance. However, if a PFG amplifier and probe are not available,

gradient autoshimming can be performed using the homospoil gradient (Z1 room temperature shim coil). For more details on how to set up the homospoil gradient, refer to the section "'Homospoil Gradient Type," page 58".

Gradient autoshimming methods support shimming on a wide variety of samples with different volumes and solvents. For aqueous samples, water protons provide sufficient signal for shimming. For deuterated solvents, gradient shimming can be performed if there is sufficient deuterium signal. Deuterium gradient shimming is feasible on most samples where the lock solvent is a single, strong resonance, which includes the majority of solvents of interest for routine NMR use.

Proton gradient autoshimming with PFG is available on systems configured with a PFG accessory. The Automated Deuterium Gradient Shimming module is required for deuterium gradient shimming with PFG or homospoil.

4.2 Configuring Gradients and Hardware Control

1. Confirm that PFG or homospoil gradients are installed on your system. See the previous sections in this chapter.
2. Confirm that the gradients are active by checking that `gradtype` and `pfgon` are set appropriately for your system. Use `config` to change settings if necessary.
3. If you have the Ultra•nmr shim system, enable control of the shims from the Acquisition window, as described in the Appendix E, "Shimming Using the Ultra•nmr Shim System," page 277.

4.3 Gradient Shimming Method

The full gradient shimming method consists of these steps:

1. "Mapping the Shims," page 50
2. "Starting Gradient Shimming," page 51

The shims must be mapped before autoshimming is used. Mapping the shims is necessary when a new probe is installed, but can be repeated at any time.

Note: Spinning the sample during gradient shimming can cause motion artifacts.

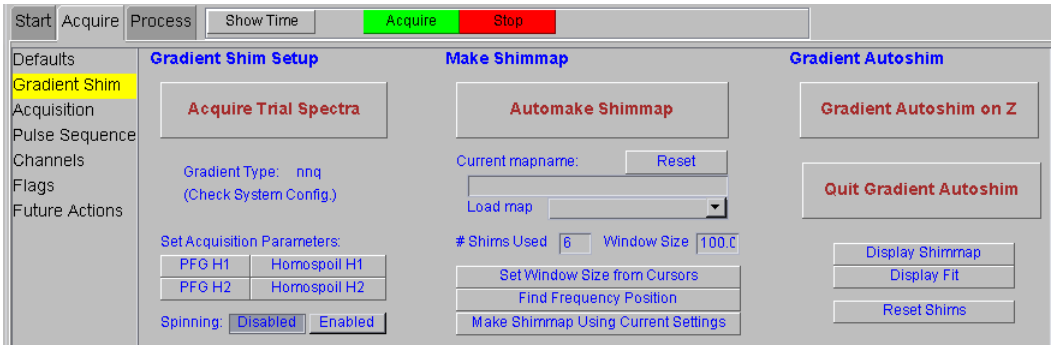
Mapping the Shims

Mapping the shims is necessary after installing a new probe. Recommended samples are listed below:

1H shimming	90% H_2O
2H shimming	1%H_2O/99%D_2O

1. Insert a sample and find lock.
2. Stop sample spinning. Disable sample changer control (**`loc='n'`**).
3. Adjust lock power, lock gain, and lock phase. Make coarse shim adjustments on Z1, Z2, X1, and Y1.
4. Use **`s2pul`** to find the 90° pulse for `tn='H1'` or `tn='lk'`, as appropriate.

5. Enter **gmapsys** to display the Gradient Shimming pages. Or select Utilities->Set Up Gradient Shimming from the menu bar. In the Walkup interface, select Utilities -> Calibration Experiments -> Set Up Gradient Shimming



Standard parameters are retrieved from `gmapz.par` the first time `gmapsys` is entered, or if a shimmap was previously made, parameters are retrieved from the current shimmap. If desired, enter **gmapz** to retrieve standard parameters from `gmapz.par`.

6. To set parameters for a particular gradient and nucleus, click the **Acquire** tab and select the **Gradient Shim** page. Click the appropriate button under **Set Acquisition Parameters**.

7. Set **pw** as follows:
   - For PFG, set `pw` to the 90-degree pulse or less.
   - For homospoil, set `pw` to the 90-degree pulse and `p1` to 180-degree pulse.

8. Test the parameters by clicking **Acquire Trial Spectra** on the Gradient Shim page.

   You should see two profile spectra. If you don't, check that the gradients are active and check `pw`, `tpwr`, and `gain`.

9. Make a shimmap by clicking Automake Shimmap on the Gradient Shim page. Enter a mapname (any string valid for a file name) at the prompt.

Starting Gradient Shimming

To start shimming as a system administrator, click on **Autoshim on Z** on the Gradient Shim page. This button starts gradient shimming using current parameters, and displays the curve fit and shim adjustments for each iteration.

4.4 Shimmap Display, Loading, and Distributing

Displaying the Shimmap

After the shims are mapped, display the shimmap by clicking the **Display Shimmap** button on the Gradient Shim page (under the Acquire tab).

The shimmap display is a multicolored plot of the shimmap, with Z1 as #1 and Z2 as #2, and so on (see Figure 7).

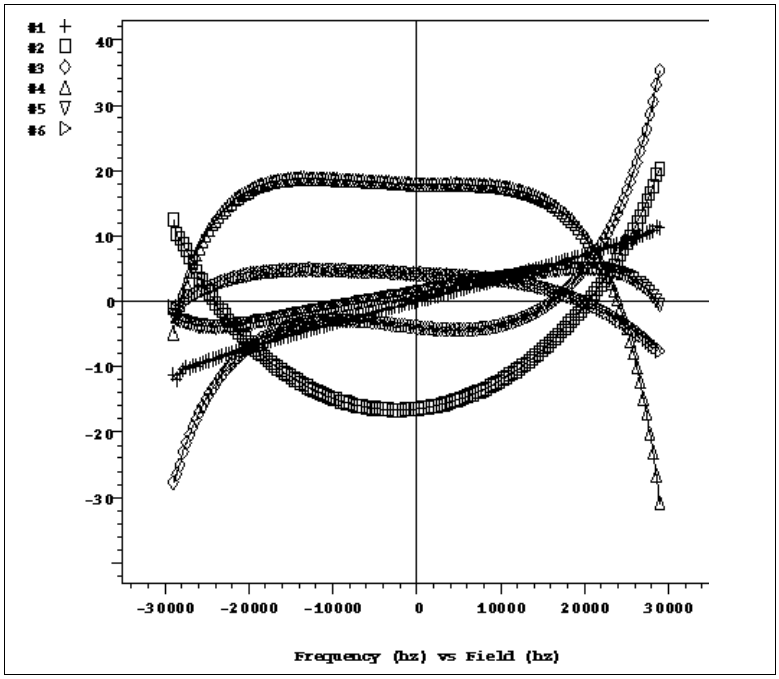


Figure 7. Shimmap Plot

The shimmap is specific to the probe used, and can also be dependent on sample volume for small volumes. The shimmap shows the actual field dependence of the shims, except for a dc offset added for display purposes. Good signal-to-noise in the shimmap is needed for the shimming to work well. Poor signal-to-noise might result in incorrectly set shims.

Loading a Shimmap

To change shimmaps as a system administrator, do the following:

1. Select the Gradient Shim page (under the Acquire tab).
2. Select an entry in the Load map menu. This loads parameters and loads the shimmap files `gshim.list` and `gshim.bas` from `gshimlib/shimmaps/mapname.fid` into `gshimlib/data`.

Distributing a Shimmap

The system administrator can copy a shimmap file from `vnmrsys/gshimlib/shimmaps` into the directory `/vnmr/gshimlib/shimmaps` so that the file is accessible to all users. To copy files, do the following steps:

1. Log in as **`vnmr1`**.

2. In a Terminal window, find the maps, in `vnmrsys/gshimlib/shimmaps`, you wish to copy.
3. Enter **`cd /vnmr`**.
4. If `gshimlib` does not exist, enter **`mkdir gshimlib`**.
5. For each map in `vnmrsys/gshimlib/shimmaps`, enter:
   `cp -r ~/vnmrsys/gshimlib/shimmaps/`*`mapname`*`.fid .`
   Remember the final dot at the end of the command, and remember to substitute a name for *`mapname`*.
6. Select **Utilities** -> **Set Up Gradient Shim**.
7. Select a file from the **Load map** menu on the **Gradient Shim** page.

Shimmap Files and Parameters

The parameters and shimmap files saved under a mapname are retrieved when that mapname is retrieved. When reinserting a probe, reload the shimmap for that probe. If you are unsure if the shimmap is correct, make a new shimmap, which typically only takes a few minutes. The last parameters and files used are automatically retrieved the first time `gmapsys` is entered. If `gmapsys` is entered again, the parameters are not retrieved. Gradient shimming uses the current parameters after the pulse sequence is loaded (`seqfil='gmapz'`).

Standard parameters can be loaded before making a shimmap by entering `gmapz` or by using the PFG H1, PFG H2, Homospoil H1, Homospoil H2 buttons. Parameters and files can also be explicitly loaded and distributed, as described in the following subsections:

4.5 Quitting the Gradient Shimming System Menu

Select **Utilities** -> **Set Up Gradient Shimming** and click the **Quit Gradient Autoshim** button. This also retrieves the previous parameter set and data, including any data processing done on the previous data set.

4.6 General User Gradient Shimming

For the general user, gradient shimming can be run from outside `gmapsys` from any experiment. Any one of the following methods is recommended for routine use:

- Click **Acquisition** -> **Do Gradient Shimming**. Parameters are retrieved from the current mapname, which is displayed at the start of shimming, and the spinner is automatically turned off. The curve fit and shim adjustments are not displayed. The previous parameter set and data are retrieved when shimming is finished. This button only functions after a shimmap is made.
- Under the Start tab, a **Gradient Shim** button is on the Standard and Lock pages.
- Enter **`gmapshim.`** This performs the same action as clicking on **Gradient Autoshim on Z**.
- Within automation parameter sets, use **`wshim='g'`**.
- In the **Walkup** interface, select the **Gradient Shim** checkbox.

To stop gradient shimming before it is completed, use one of the following methods:

- Under the Acquire tab, select the Gradient Shim page and click the **Quit Gradient Autoshim** button. Quitting aborts the experiment and retrieves the previous parameter set and data.
- Abort the acquisition with **aa** and click on **Cancel Cmd**. Then enter **gmapshim('quit')** to retrieve previous data set and parameters.

4.7 Calibrating gzwin (optional)

The parameter gzwin is the percentage of the spectral window used in calculating the field maps. gzwin should be adjusted only when making a new shimmap. If this parameter is not calibrated correctly, you may see excess noise data at the edge of the shimmaps, which corresponds to the region in the profile spectrum where the signal goes to zero. It is normal to have a few noise data points at the edge of the shimmap, but if it is more than a few data points (greater than 25% of the window), gzwin may be miscalibrated. This can occur if there is low signal-to-noise or if gzwin has not previously been calibrated for the current parameter set. If the gain is too high, "wings" will appear on the sides of the spectra and may result in miscalibrated gzwin. This can also occur if there are multiple chemical shifts in the presence of a weak gradient.

Automatic Calibration of gzwin

1. Click the **Find gzwin** button on the Gradient Shim page. This calibrates gzwin and sets tof to center the window used for calculation.
2. Click **Make Shimmap** using the current settings. This makes the shimmap with the current values of gzwin and tof.

You may click through these steps separately to see if gzwin is calibrated correctly. The box cursors at the end of step 1 should be at either edge of the profile.

Manual Calibration of gzwin

Manual calibration of gzwin can be used to avoid noise spikes in the spectrum, or other artifacts. To manually calibrate gzwin, do the following:

1. Click the **Acquire Trial Spectra** button on the Gradient Shim page. Wait until the experiment is done.
2. Display a spectrum using the graphics control buttons. Set the box cursors near the edges of the profile.
3. Click Set **Window Size from Cursors**.
4. Click on **Make Shimmap Using Current Settings**.

The parameter gzwin should be adjusted only when making a new shimmap. The calibrated value of gzwin is saved when the new shimmap is saved at the end of the mapping experiment. The same value of gzwin must be used in shimming as in making a shimmap, and should not be adjusted when shimming.

4.8 Deuterium Gradient Shimming

Deuterium gradient shimming is feasible for most deuterated solvents for which lock solvent has a single, strong deuterium resonance with sufficient signal.

The automated deuterium gradient shimming module is required to run deuterium gradient shimming. If present, this module automatically holds the lock at its current value and switches the transmitter cable to pulse the lock coil when an experiment is run with `tn='lk'`. The module is strongly recommended for all users who wish to run deuterium gradient shimming in automation.

The system administrator must make a shimmap on deuterium before deuterium gradient shimming can be used. Follow the procedure "Gradient Shimming Method," page 50, using the deuterium signal for all steps. The transmitter power (`tpwr`) should be kept low to avoid probe arcing, with a 90° pulse greater than about 200 µs.

The recommended parameters for different solvents are shown in Table 2.

The deuterium parameters are saved for future use when the shimmap is saved, and are used the next time gradient autoshimming is run.

Table 2. Deuterium Parameters

Solvent	*nt*	*d1 (sec)*	*Gain*	
			Inova	*Mercury*
deuterochloroform	8-32	2	36	18
dmso-d6	4-16	2	28	10
D_2O	1-4	2	24	6
deuterobenzene	1-4	2	24	6
deuteroacetone	1-4	6-12	24	6

Note: Actual parameters might vary, depending on solvent concentration, probe, and system hardware.

4.9 Full Deuterium Gradient Shimming Procedure for Lineshape

The automated deuterium gradient shimming module must be installed to use this procedure.

1. Insert the appropriate lineshape sample (chloroform in acetone-d6) and find lock. Turn off spinning and disable sample changer control. Select the Lock page under the Start tab; adjust lock power, lock gain, and lock phase as necessary. Do quick shimming on z1, z2, x1, y1 (use z1c, z2c, if present).
2. Find the 90° pulse on ^2H as follows:
    a. Select the **Proton** protocol. Select the **Channels** page under the Acquire tab and set the **Observe Nucleus** to **lk** and the **90 Degree Pwr** to **42**. On the **Acquisition** page, set the **Observe Pulse** to **200**.
    b. Click the **Acquire** button and wait for acquisition to finish. You should see only a single line.
    c. Click the **Process** tab and select the **Cursors/Integration** page.
    d. Select the Acquisition -> Parameter Arrays menu. enter **pw** for the Param Name, Array size **20**, First Value **100**, Increment **100**, and click **Close**. Select **Acquisition** -> **Acquire and WFT**, and wait for acquisition to complete.

e. In the Acquisition page, click Arrays and click UnArray in the Array window. Set **pw90** to the first maximum.

3. Set up gradient shim parameters. Select **Utilities** -> **Set Up Gradient Shimming**. Select the **Gradient Shim** page.
   - If you have PFG, click on **PFG H2**, set **Observe Pulse** to one half the 90° pulse found in step 2 and set **Relaxation Delay** to **6**.
   - If you have homospoil, click **Homospoil H2**, set **Observe Pulse** to the 90° pulse and **Relaxation Delay** to **6**.
4. Click Acquire Trial Spectrum and wait for acquisition to complete. You should see two profile spectra.
5. Map the shims. Click **Automake Shimmap**. Under **Current mapname**, click the **Reset** button or, enter a file name. Wait for acquisition to complete and the message to be displayed: `shimmap done!`
6. Perform shimming on z's. Click on **Gradient Autoshim on Z**. Wait for the acquisitions to complete, and the message to be displayed: `Gradient Autoshimming on Z done! N iterations.`
7. Click the **Start** tab and select the **Lock** page. Click the **Lock Scan** button and adjust lock phase. Shim only on low-order nonspins (x1, y1, xz, yz, etc.). Do not shim on z's (z1, z2, etc.).
8. Perform shimming on z's. Repeat step 6.
9. Measure proton lineshape. Turn on spinner if appropriate, and click the Lock Scan button to make fine shim adjustment. Shim on all shims as necessary.

4.10 Setting Up Automation

You must have the deuterium gradient shimming module installed to perform deuterium gradient shimming in automation.

1. Insert the appropriate AutoTest sample (chloroform in acetone-d6) and find lock. Turn off spinning and disable sample changer control. Select the Lock page under the Start tab; adjust lock power, lock gain, and lock phase as necessary. Do quick shimming on z1, z2, x1, y1 (use z1c, z2c, if present).
2. Find the 90° pulse on ^{2}H as follows:
   a. Select the **Proton** protocol. Select the **Channels** page under the Acquire tab and set the **Observe Nucleus** to **lk** and the **90 Degree Pwr** to **42**. On the **Acquisition** page, set the **Observe Pulse** to **200**.
   b. Click the **Acquire** button and wait for acquisition to finish. You should see only a single line.
   c. Click the **Process** tab and select the **Cursors/Integration** page.
   d. Select the Acquisition -> Parameter Arrays menu. enter **pw** for the Param Name, Array size **20**, First Value **100**, Increment **100**, and click **Close**. Select **Acquisition** -> **Acquire and WFT**, and wait for acquisition to complete.
   e. In the Acquisition page, click Arrays and click UnArray in the Array window. Set **pw90** to the first maximum.
3. Set the parameters. Select **Utilities** -> **Set Up Gradient Shimming**. Click the **Acquire** tab and select the **Gradient Shim** page.

- If you have PFG, click on **PFG H2** to select the parameter for deuterium with pfg.
- If you have homospoil gradients, select **Homospoil H2** instead.

4. Set Observe Power and Pulse for 90° pulse and set Receiver Gain as appropriate for your solvent.
5. Click **Acquire Trial Spectra** to test the parameters.

   You should see two top-hat profile spectra. Adjust `gain` and `nt` to see good signal-to-noise with no ADC overflow.

   The recommended parameters for different solvents are listed in Table 2.
6. Make a shimmap for a particular solvent using parameters for good signal-to-noise. If the solvent you most often use has a weak signal, make the map on a solvent with more signal. To make the map, click Reset or enter a map name in Current mapname. Then click **Automake Shimmap**.
7. Test autoshimming by clicking on **Gradient Autoshim on Z**.
8. When you are satisfied that autoshiming works well for your particular solvent, open a shell window and edit the `gmapz` macro. Go to the bottom of the macro and uncomment the section appropriate for your solvents. Use parameters as in step 3. Repeat step 3 for all solvents of interest.

4.11 Homospoil Gradient Shimming

Homospoil Gradient Shimming for ^{1}H or ^{2}H

It is also possible to use the Z1 room temperature shim as a homospoil gradient, instead of using a pulsed field gradient or PFG. Use of this option is recommended only if a PFG amplifier or probe is not available. The system administrator must make a shimmap using homospoil before homospoil gradient shimming can be used. Follow the procedure in 4.3, "Gradient Shimming Method," on page 50.

At step 4 select `tn` as appropriate, set `tof` on resonance, and then find the 90° pulse. If deuterium is used, `tpwr` should be kept low, with a 90° pulse greater than about 200 µs.

At step 6, select either **Homospoil H1** for proton parameters or **Homospoil H2** for deuterium parameters, as appropriate. Homospoil gradients must be configured at this step (use `config` or set `gradtype='nnh'`).

At step 7, further testing of the gradient shimming parameters for homospoil can be done as follows:

1. Click the Acquire Trial Spectrum button on the Gradient Shim page.
2. Using the 90° pulse from step 4, calibrate the 90° and 180° pulses to obtain an echo. Enter **`df`** to display the FID. You should see an echo forming in the middle of the FID.
3. If needed, adjust **`sw`** so that the gradient covers at least 10% of the spectral window. Increase **`np`** to 512 to improve Hz/point resolution. However, `np` should be adjusted so that `at` is not longer than the homospoil time limit (20 ms on standard

UNITY*INOVA*). The acquisition time (`at`) should also be shorter than T_2. Set **`d2=at/4`**.

Once all the parameters are set, click on **Automake Shimmap** (step 8). The parameters are saved when the shimmap is done and are used the next time gradient autoshimming is run.

To use homospoil deuterium gradient shimming with different solvents, use Find z0 before gradient shimming.

Homospoil Gradient Type

VnmrJ allows you to use homospoil (room temperature Z1 shim coil) as a general gradient type. It does not require the use of a pulsed field gradient module and thus is available on systems without PFG. Homospoil gradients are implemented only on the Z axis.

When homospoil is switched on in a pulse sequence, the shim current is set to maximum for a given period of time. Homospoil control within a pulse sequence is done in the following manner:

- To use homospoil as a quick homogeneity spoil, use `hsdelay`. This is the traditional homospoil method, and is usually done at the beginning of a relaxation recovery delay (e.g., `hsdelay(d1)`). The parameter `gradtype` is ignored. See the *User Programming manual* for details of how to use `hsdelay`.
- To use homospoil as a general gradient type, first select the homospoil gradient type. Enter **`config`** and under **Gradients** select **Homospoil** (this sets `gradtype='nnh'`). The parameter `pfgon` is ignored, since a separate gradient amplifier is not needed. Homospoil is then triggered by gradient statements such as `rgradient('z',gzlvl1)`. If the value of `gzlvl1` is non-zero, homospoil is switched on; if the value of `gzlvl1` is zero, homospoil is switched off. Only one sign and strength of gradient current is available during a pulse sequence, and is set by hardware.

Homospoil gradients may be switched on only for a limited period of time, usually 20 ms. This time limit is determined by hardware in UNITY*INOVA* systems (see Table 3 for system configurations). Check your pulse sequences to ensure this time limit is not exceeded.

Table 3. Homospoil Control

System	*Shim Supply*	*Homospoil Time Limit*
UNITY*INOVA*	Varian 14	20 ms/200 ms[a]
UNITY*INOVA*	Varian 18 to 40	20 ms/200 ms[a,b]
UNITY*INOVA*	RRI Ultrashims	20 ms/200 ms[a]
MERCURYplus/-Vx	Varian 14	No time limit[c]

a. Hardware upgrade to 200 ms with the Automated Deuterium Gradient Shimming module is required for compatibility with ^{2}H gradient autoshimming.
b. Hardware adjustment required for both ^{1}H and ^{2}H gradient autoshimming. Adjust homospoil potentiometer resistor labeled HOMO (blue square) on front of Z0/Z1 board to maximum in either direction for maximum homospoil gradient strength.
c. A homospoil time limit of 20 ms is set by software for `hsdelay`.

The behavior of homospoil gradients is quite different from that of a pulsed field gradient. The gradient strength is much weaker than the traditional PFG, and the recovery time is

much longer because of eddy currents. The strength and recovery of the gradient depends on the shim coils and system hardware. Typically, these gradients are suitable only for profile-type experiments and unsuitable for gradient coherence-selection experiments such as GCOSY and GNOESY. For all gradient experiments, pulsed field gradients are preferred if available.

Homospoil gradients are suitable for ^{1}H and ^{2}H gradient shimming on some systems (see Table 3 for system configurations). The Automated Deuterium Gradient Shimming module is required on UNITY*INOVA* systems to upgrade the homospoil hardware for compatibility with deuterium gradient shimming.

4.12 Suggestions for Improving Results

Calibrate the 90° pulse and adjust `tpwr`, `pw`, and `gain` to optimize signal-to-noise. Reduce gain if ADC overflow occurs, which may appear as wings on the profile.

For solvents with long T_1, set **`d1`** to 3 to 5 times T_1, or use a small flip angle for `pw`. Or, use the Ernst angle. Stimulated echoes may otherwise result, which may appear as excess noise or a beat pattern in the spectrum, or as secondary echoes in the FID (use `df` to observe this).

The phase encode delay `d3` is arrayed to two values, the first of which is zero. The second value can be increased for better signal-to-noise in the phase maps, up to about the point where the amplitude of the second profile is half that of the first (about 2/3 T_2 without radiation damping; radiation damping can be severe in water ^{1}H). However, longer `d3` values increase the phase excursion, and can make it difficult to shim large shim corrections (especially Z1). Typical ^{1}H values are 5 to 30 ms, and typical ^{2}H values are 30 to 200 ms. If the shims are far off when making a shimmap, the second value of d3 might be too small. If this problem occurs, decrease the second value of d3 to temporarily one-half to one-quarter its value.

When reinstalling a probe, make sure it is in the same vertical position in the magnet barrel as when the shimmap was made. If you are unsure, make a new shimmap, which typically takes only a few minutes.

Alternate between z-axis gradient shimming and shimming the low-order x- and y-axis shims by other methods (e.g., on lock level). The z-axis shims account for the majority of sample volume changes (changes in height), and the x- and y-shims are relatively insensitive to change in height. Evaluate shimming for a particular application, since the ideal lineshape may vary with the application.

The high-order shims can sometimes be set off-scale during shimming. This may occur if the sample is short, or if the sample is improperly seated in the probe, or if the high-order to shims are weak or other effects. In such cases, the off-scale shim is set to maximum, and shimming continues with lower-order shims. Superior results can be obtained in some cases by first setting # Shims Used (gsize) to 4 and clicking on Autoshim on Z to shim on `z1–z4`, and then shimming the low-order transverse shims, and then increasing `gzsize` and shimming again. This may also be done using `gmap_z1z4='y'`. On a short sample it also can be useful to remap the shims.

Some shim systems may need additional time when running the shim mapping experiment to allow the shims to settle. The added time is especially noticeable on some systems for Z4. To account for added time, lengthen the `d1` delay or add dummy scans in between each array element (e.g., `ss=-2`). Decreasing the amount a shim is offset also allows the shim to settle more quickly. Enter `gmapsys('vi')` to edit the values in the Offset column, and then click Make Shimmap using Current Settings to map the shims with user-defined offsets. A new mapname may also be set using `gmapsys('vi')`.

Coarse shims are used on systems on which they are available. To use fine shims on these systems, enter `gmapsys('vi')` to edit the entries in the shim column (e.g., change `z1c` to `z1`), and then enter `gmapsys('shimmap','manual')` to map the shims.

For samples in H_20, the water protons provide sufficient signal for shimming. For samples other than water, deuterium gradient shimming is strongly recommended if there is sufficient deuterium signal. Proton gradient shimming can be made to work in samples other than water if there is sufficient proton signal and the signal is well-resolved (does not overlap with other strong resonances). Gradient shimming can also be done on a water sample of equal height of the sample of interest, and then the sample of interest can be inserted.

For further information, refer to the entries for `gmapshim`, `gmapsys`, and `gmapz` in the *Command and Parameter Reference*.

4.13 How Gradient Shimming Works

- "How Making a Shimmap Works," page 61
- "How Automated Shimming Works," page 62

The basis of gradient shimming is differential phase accumulation from shim gradients during an arrayed delay. The phase is spatially encoded by a pulsed field gradient. Figure 8 shows the gradient shimming pulse sequence.

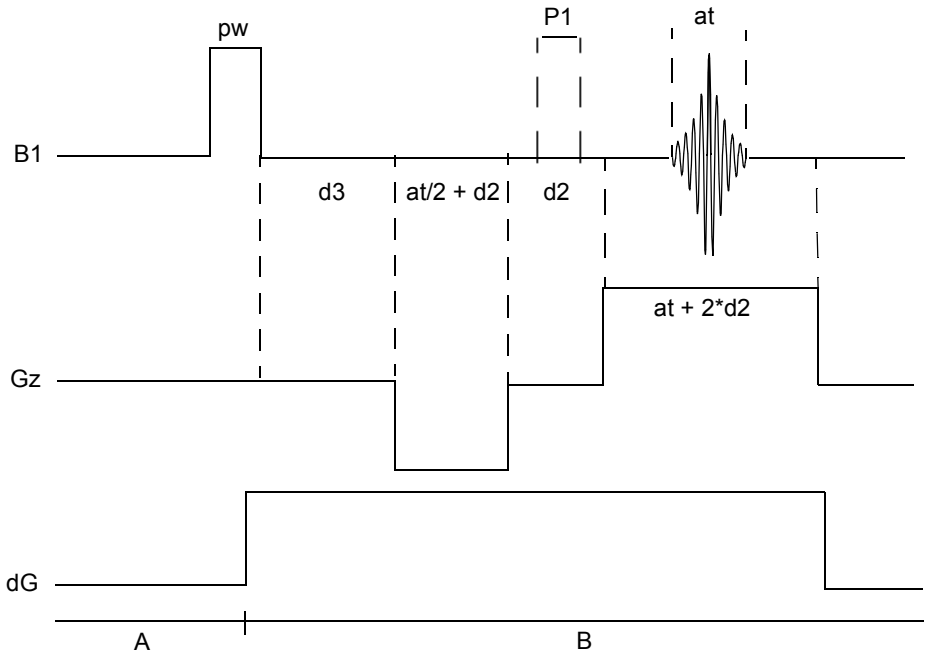


Figure 8. Gradient Shimming Pulse Sequence

The gradient shimming pulse sequence in Figure 8 is shown with `p1=0`, in which case `pw` can be set to a small flip angle. If `p1>0`, the pulse field gradients are both set to the same sign, and `p1` should be set to 180° and `pw` to 90°, so that rf inhomogeneities are refocused. `p1=0` is usually sufficient for most cases.

Phase accumulation from all gradients present is as follows:

ϕ = z Gz(-at/2 + t) + dG(d3 + at/2 + 3*d2 + t)

where t is the time during acquisition at, Gz is the z-axis pulsed field gradient strength, and dG is the sum of the shim gradient fields, shown as being on during relevant times in the pulse sequence.

The effect of the shim gradients dG can be isolated by arraying d3 and taking the difference in the phases:

$\Delta\phi$ = ϕ2 - ϕ1 = dG*(d3[2] - d3[1])

For example, at a particular point, $\Delta\phi$ can be 2π * 100 Hz * 10 ms, or 2π radians. Thus, a pair of profiles with different d3 values can be used to calculate the B_0 field along z.

The effect of any one shim gradient can be isolated by arraying the shim value, represented by dG, and taking the difference in the phase differences:

$\Delta(\Delta\phi)$ = $\Delta\phi$2 - $\Delta\phi$1 = dG2*(d3[2]-d3[1]) - dG1*(d3[2]-d3[1])
=(dG2 - dG1)*(d3[2] - d3[1])

Therefore, two pairs of profiles can be used to map out the effect of a shim. By arraying all the shim values, a set of phase difference maps or shim field maps can be constructed for a given shim set. Shimming can then be performed by constructing a background field map for the starting shim values ($\Delta\phi$) and fitting the result to the shim field maps. The calculations are quite fast, so the entire shimming process is usually limited by the data acquisition time, typically taking only a few minutes.

In practice, the phase is calculated from ϕ=arctan(x,y) from the real and imaginary values at each point in the spectrum, and $\Delta\phi$ is calculated from the difference in the phases of a pair of spectra with d3 arrayed. Figure 9 shows an example of mapping the z1 shim.

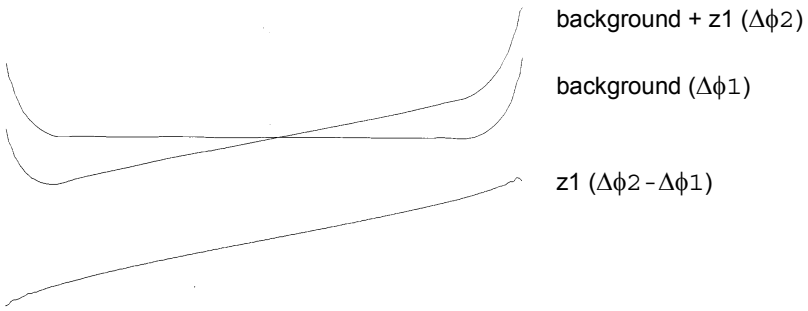


Figure 9. Mapping the z1 Shim

How Making a Shimmap Works

Automake Shimmap first runs an experiment that calibrates gzwin and tof to set the spectral window. Next, it runs an experiment with the shims arrayed to map the shims, and processes the experiment when done. Coarse shims are used if present. The parameters and data for the shimmap are stored in the file userdir + '/gshimlib/shimmaps/' + mapname + '.fid'. These parameters are retrieved the next time gradient shimming is run if the gradient shimming system menu is exited.

How Automated Shimming Works

The shims must be mapped before gradient autoshimming is used. See "Gradient Shimming Method," page 50 for details.

When gradient shimming is run from the Gradient Shim page, the curve fit plot is displayed for each iteration. The plot shows the raw data as #1 and the curve fit as #2 (see Figure 10).

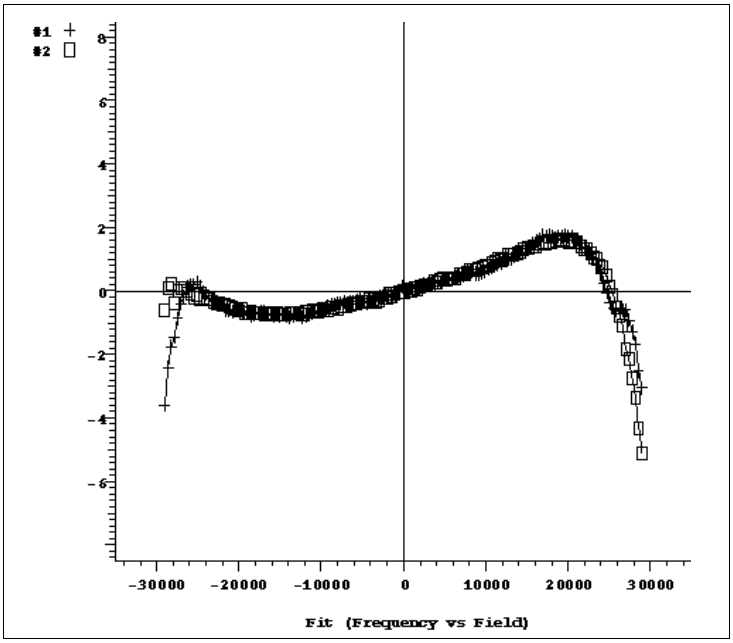


Figure 10. Curve Fit Plot

Shim adjustments for each iteration are also displayed (see Figure 11) and have converged when the rms error number is less than 1.0. Gradient shimming continues until convergence or until a maximum of 10 iterations are reached.

```
mapname 5mm_Triax_01
shimset 4  gzsize 6                          rms err 1.892

Shim     Offset      Old       New        Diff       Error
-----------------------------------------------------------
z1       800        -9405     -9269      -136       48
z2       800        -3118     -3104      -14        13
z3       3200       -4356     -4321      -35        37
z4       -3200      4049      4885       -836       104
z5       -3200      13443     14537      -1094      322
z6       3200       -15619    -12568     -3051      467
z7       3200       0         0          0          0
z8       3200       0         0          0          0
-----------------------------------------------------------
```

Figure 11. Display of Shim Adjustments for Each Iteration

If a shim goes out of range, the shim is set to maximum and shimming continues with the remaining shims. If convergence is then reached, shimming is tried once more with all Z shims and continues unless a shim goes out of range again.

If the parameter `gmap_z1z4` is set to `'y'`, then if `gzsize` is greater than 4, shimming is done first on Z1–Z4 and then proceeds with all shims specified by `gzsize`. Gradient shimming takes longer and goes through more iterations, but this may avoid the problem on some systems where a high-order shim (i.e. Z5, Z6) goes out of range because it contains impurities from lower-order shims. This parameter may be set at any time while shimming from `gmapsys`. In order to use this parameter in user autoshimming, set it before making a shimmap, or in the corresponding parameter set in `gshimlib/shimmaps`.

4.14 References

Van Zijl, P. C. M., et al. *J. Magn. Reson.* **1994**, *111* (Series A), 203–207.

Sukumar, S., et al. *J. Magn. Reson.* **1997**, *125* (Series A), 159–162.

Barjat, H., et al., *J. Magn. Reson.* **1997**, *125* (Series A), 197–201.

Chapter 5. Data Acquisition

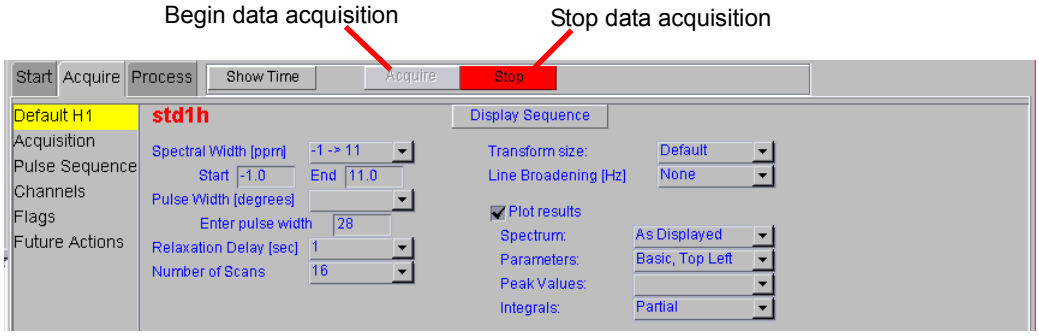
Sections in this chapter:

With a spectrometer configured to perform the proper experiment, and a sample in place, spinning, locked, and shimmed, you are ready to select parameters to acquire data. There are two aspects to selecting parameters. The first is the frequency-related aspect—setting the position and size of the spectral window. The second is the pulse sequence-related aspect.

5.1 Acquiring a Spectrum

During acquisition VnmrJ reads the probe file and sets up the experiment. The default experiment is determined by the type of probe that is being used. If a gradient probe is installed, the default experiments are gradient-based experiments.

You can start an acquisition from the **Acquisition** menu, or you can use the pages under the **Acquire** tab.



1. You can accept the default settings or set acquisition parameters by clicking on the **Acquisition**, **Pulse Sequence**, or **Channels** page.

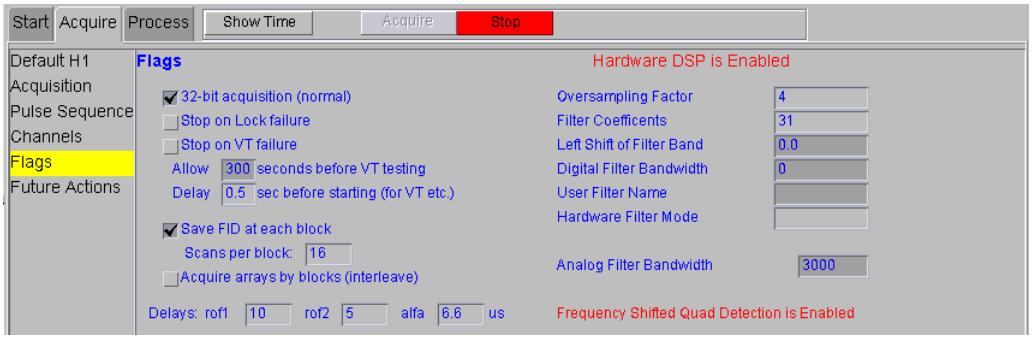
2. When you are finished setting the parameters of your experiment, click the green **Acquire** button.
3. Click the **Stop** button to stop the acquisition.

5.2 Acquisition Settings

Use the pages under the Acquire tab to adjust acquisition settings.

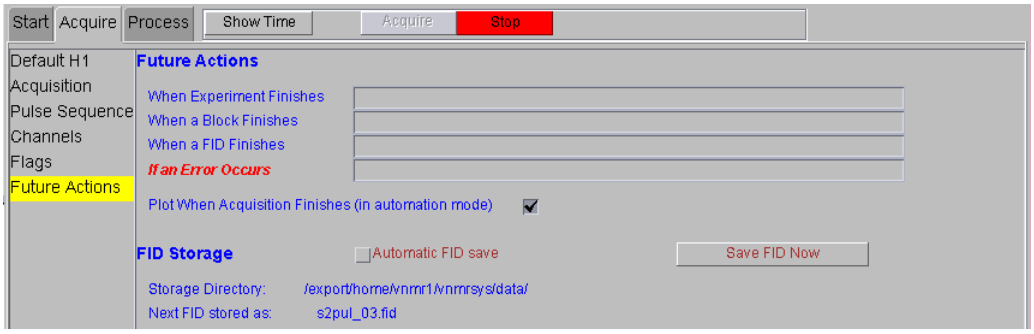
Acquisition and Post Acquisition Actions

Use the **Flags** page of the **Acquire** tab to determine the data acquisition precision (32-bit or double precision is normal), what actions to take during acquisition if the spinner or VT fail to remain in regulation, how often to save the FID during the acquisition (block size), and other acquisition related actions.



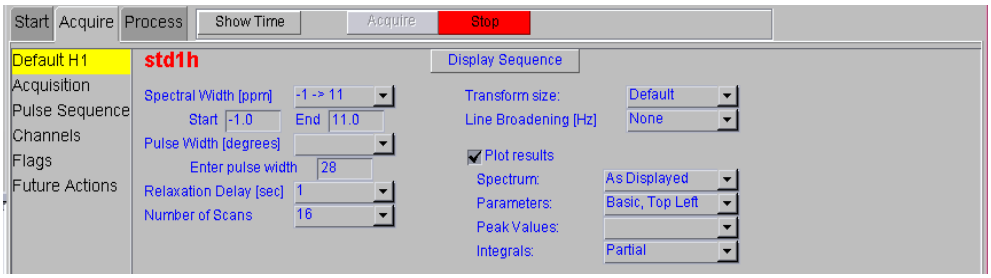
Preacquisition delay	Usually set to 0.5 seconds. This delay allows the hardware to set up at the beginning of the experiment. A principal use of this parameter is for kinetics experiments. `Delay __ sec before starting (for VT etc.)`
Analog Filter Bandwidth	Sets the audio filters, which prevent noise of higher frequency than the spectral limits from "folding in" to the spectrum. The standard value is 10% more than half of the **Spectral width** (set on the **Acquisition** page). Analog Filter Bandwidth is automatically changed whenever the spectral width is changed. After Spectral width has been change, Analog Filter Bandwidth can be changed.
Delays: rof1, rof2, alpha	**rof1** is normally fixed as 10 µs. After the final pulse in each pulse sequence, the receiver is gated off for **rof2** µs before the acquisition begins. If "pulse breakthrough" effects are seen (spike in the beginning of the FID), increasing **rof2** can reduce or eliminate the problem. **alfa** and **rof2** can be important where the flatness of the baseline is of concern.

Specify actions that are to occur automatically after acquisition finishes on the **Future Actions** page For example, you can Save FIDs or set automatic FID saving.



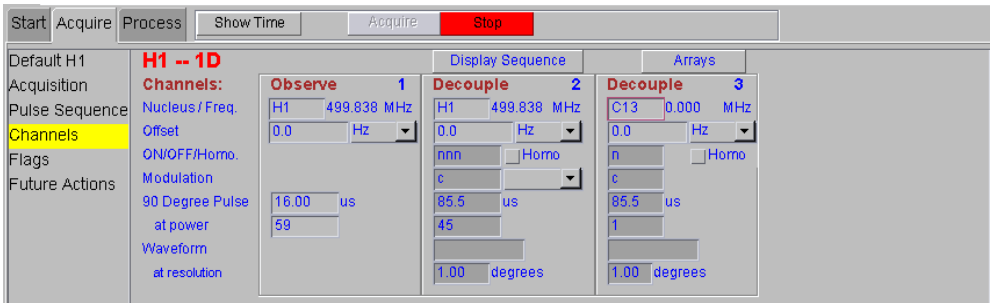
Nucleus-Specific Frequency Settings

Adjust nucleus-specific settings in the **Default*nucleus*** page. The example window below shows the Default H1 page, which is used for setting the proton frequency. Other Default pages available are: Default C13, Default F19, and Default P31.



Transmitter and Decoupler Positioning

Set transmitter and decoupler values in the Channels page.

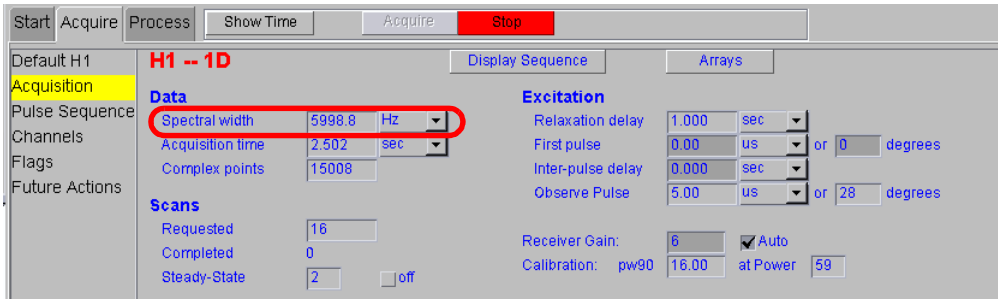


The **Observe Offset** field moves the observe transmitter offset so that the current cursor position becomes the center of the spectrum. If referencing was used, referencing is maintained. If you wish to specify the transmitter frequency directly, rather than using the cursor position, enter a value in the **Offset** field. This provides a convenient method of moving the transmitter frequency outside the current spectral window.

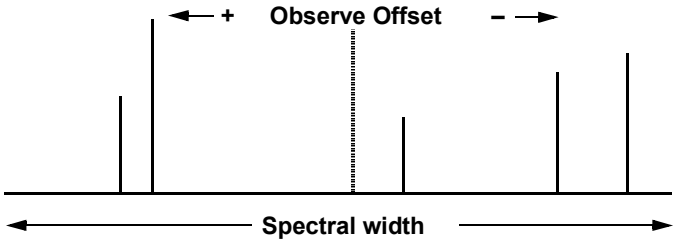
For systems with 3 or 4 channels, **Decouple 3** and **Decouple 4** input fields become available for the second and third decouplers.

Spectral Window

Set the Spectral window size in the **Spectral width** field in the **Acquisition** page.



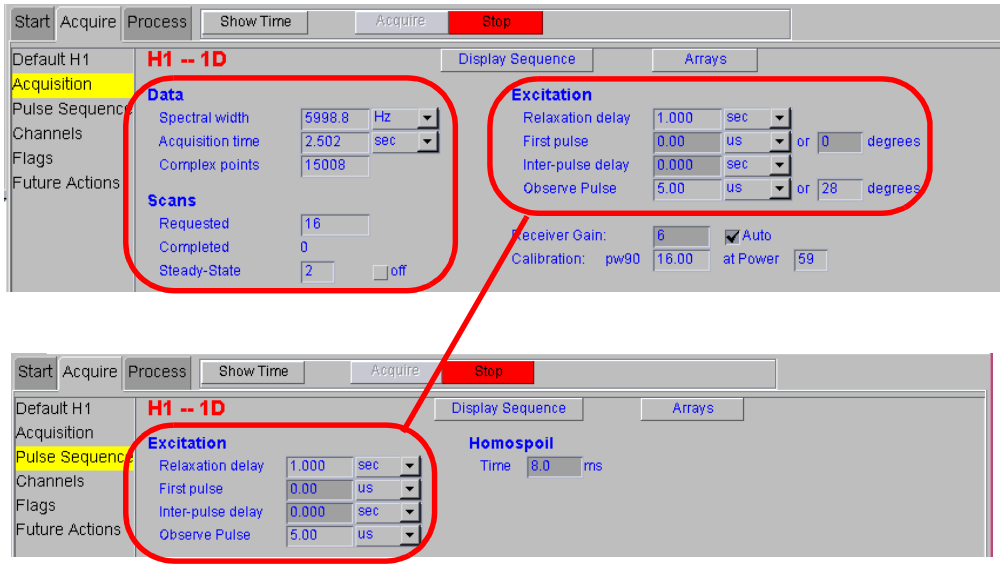
Spectral width determines the sampling rate for data, which occurs at a rate of `2*` (spectral width complex points per second).



With two cursors displayed, the value entered in the **Spectral width** field calculates a new spectral width and a new **Observe Offset**. Referencing is also adjusted, if used.

Pulse Sequence Settings (Standard Two-Pulse)

Use the **Acquisition** and **Pulse Sequence** pages to set the values particular to the pulse sequence.

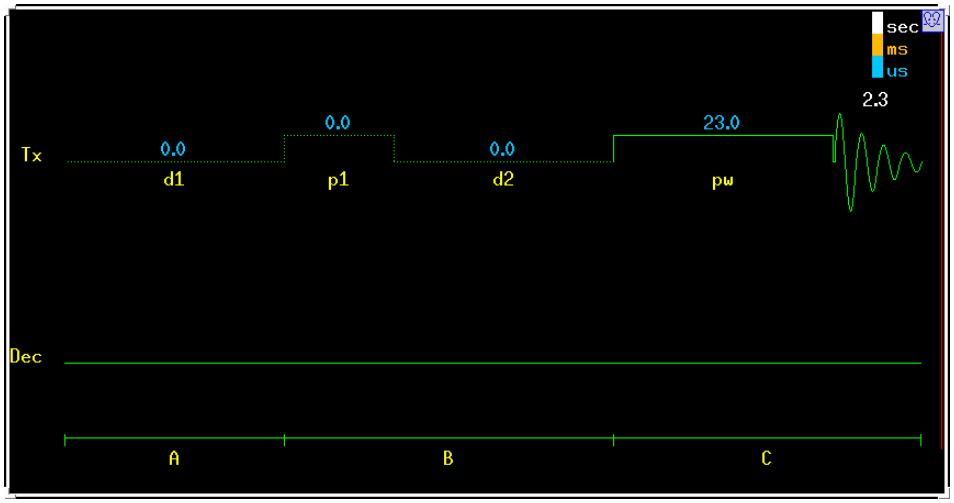


Complex points	Generally calculated automatically when spectral width or acquisition time is changed. If you enter a value in the Complex points field, VnmrJ calculates a new spectral width value.
Acquisition time	The length of time during which each FID is acquired. Acquisition time values that give a number of data points not a multiple of 2 for UNITY*INOVA*, or 64 for *MERCURYplus/-VX*, are readjusted automatically.
Steady-State	The number of complete executions of the pulse sequence not accompanied by data collection prior to the acquisition of the real data. In a multi-FID experiment, if Steady-State is a positive value, the steady-state pulses are applied at the start of the first FID only; if Steady-State is a negative value, the steady-state pulses are applied at the start of every FID.
Scans Requested	The number of repetitions or scans performed to make up the experiment—`the` number of transients acquired. To set up an indefinite acquisition, set Scans Requested to a very large number, (e.g., 1e9). The **Scans Completed** field changes during the course of an experiment to reflect the number of completed transients.
Relaxation delay, **First pulse**, **Inter-pulse delay**	For "normal" 1D NMR, **First pulse** (p1) and **Inter-pulse delay** (d2) are zero. The **Relaxation delay** (d1, used to allow recovery of magnetization back to equilibrium) is often zero as well, reducing the total pulse sequence to a pulse of the time entered in the **Observe Pulse** field (pw), followed by the **Acquisition time** (at).
Homospoil	Homospoil is a process by which the homogeneity is temporarily made very bad ("spoiled") to cause any transverse magnetizations present at that time to decay rapidly to zero.
Receiver Gain	Low gain in multiline, high-dynamic range samples can cause a number of problems, including intermodulation distortions, lower sensitivity, and extra lines in the spectrum. Too high a gain, on the other hand, can cause receiver overload and consequent baseline distortion. Autogain capability allows the observe channel to be set optimally for detecting and digitizing NMR signals from a wide variety of samples. With `gain=60` representing the highest possible actual receiver gain and `gain=0` the lowest. (On *MERCURYplus/-VX* systems, gain range is 0 to 38 dB; step size for gain is 2 dB. On UNITY*INOVA* 500-, 600-, and 750-MHz systems only, the controllable usable range of `gain` is 18 to 60 when using low-band observe nuclei.) `gain` increases in steps of 2 dB. `gain='n'` activates Autogain, in which the gain is automatically adjusted at the start of acquisition for an optimum value. After the acquisition is finished, setting `gain='y'` then allows the value of `gain` to be read by typing `gain` followed by a question mark (i.e., `gain?`).
Calibration: pw90	field displays the length of the 90° pulse, in µs. This value is determined when the probe is installed, calibrated and tested as described in the probe's installation manual.

5.3 Pulse Sequences

Display the Pulse Sequence

Click the **Display Sequence** button to display the pulse sequence in the graphics window.



Standard Two-Pulse Parameters

Most experiments will be acquired using a pulse sequence known as the standard two-pulse, or S2PUL.Figure 12 shows a two-pulse sequence and the associated labels from the Acquire pages.

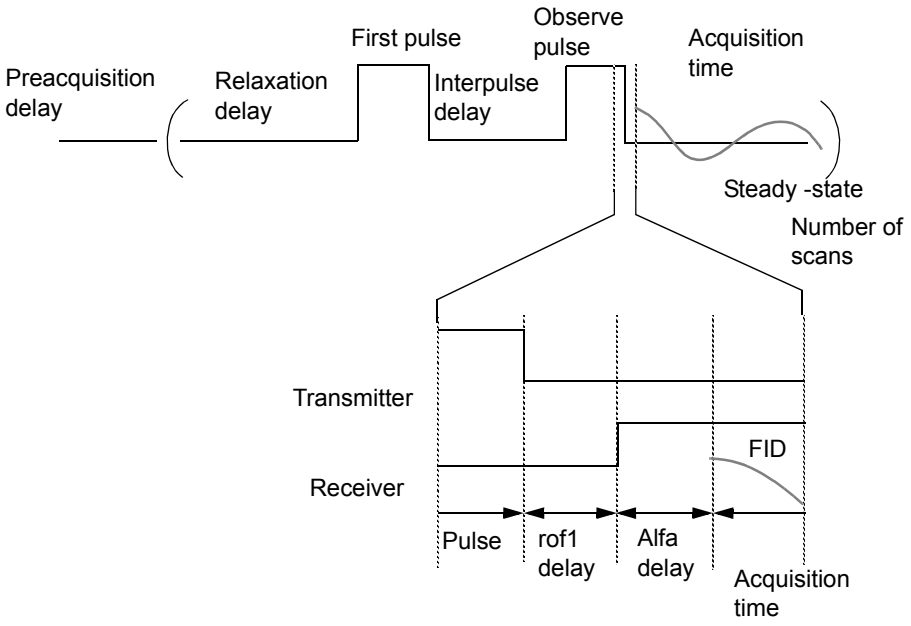


Figure 12. Acquisition Parameters for Standard Two-Pulse Sequence

A preacquisition delay is usually set to 0.5 seconds at the beginning of the experiment.

Following the preacquisition delay are:

1. Relaxation delay
2. First pulse
3. Inter-pulse delay
4. Observe pulse

Dead times `rof2` (with receiver off) and `alfa` (with receiver on) are put into the observe pulse. The complex data points are acquired during the acquisition time.

This process is repeated the number of times set steady-state plus the Requested number of scans field. Data is actually acquired only during the number of scans and not during the first steady-state transients.

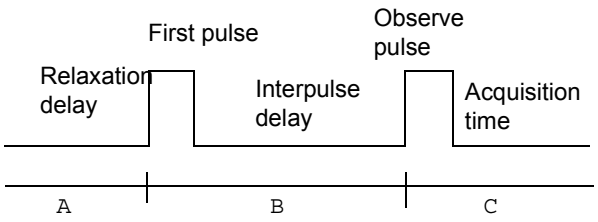
On *UNITYINOVA* systems, the receiver is off during the pulse sequence and on only during **Acquisition time**. The amplifier can be unblanked at any time but no longer than 10 ms. Blanking and unblanking are implicitly done around pulses.

After the final pulse in each pulse sequence, the receiver is gated off for **rof2** µs before the acquisition begins. If "pulse breakthrough" effects are seen (spike in the beginning of the FID), increasing **rof2** can reduce or eliminate the problem.

The "Status" Concept

Every pulse sequence can be divided logically into "periods" of time. The standard two-pulse sequence, for example, can be divided as shown below. This sequence has three logical periods, referred to in the diagram as A, B, and C. These periods are used in controlling the decoupler "status" (as well as the "homospoil" status, discussed later in this chapter).

Logical Periods A, B, C in Standard Two-Pulse Sequence



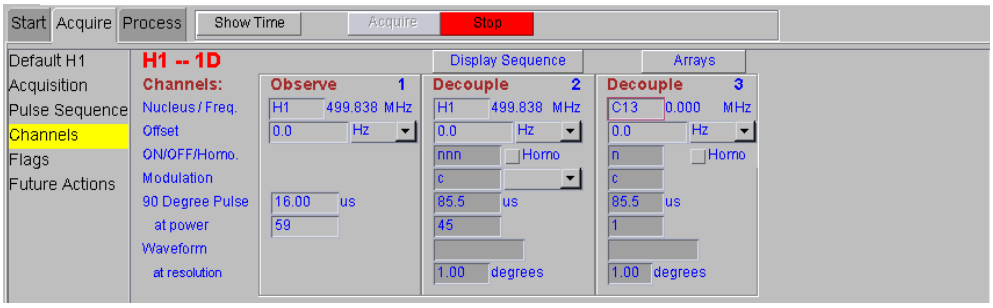
In the **ON/OFF/Homo** field in the **Channels** page, use the following letters:

n	no, or an off status
y	yes, or an on status

For example, say we want the decoupler to be on during period A, on during period B, and off during period C. We could then describe the desired decoupler status as **yyn**. Setting the **ON/OFF/Homo field** to **yyn** will select this experiment, which in the heteronuclear case might produce a coupled spectrum with NOE, or in the homonuclear case might be used for solvent presaturation experiments. Setting nny would give us an experiment with the decoupler only on during period C, the acquisition time, which in the heteronuclear case would be a decoupled spectrum without NOE.

Observe Transmitter and Decoupling Settings

Transmitter power levels are set through attenuators, which are in turn controlled through fields in the **Channels** page. Figure 13 shows schematics for the attenuator configuration for different systems.



The observe transmitter power, which is under computer control on systems with linear amplifiers, is set in the Observe at power field. The power can be set to a value from 0 to 63, or from –16 to 63, depending on the range of attenuators present in the system. In both cases, 63 is the maximum possible power.

The **Decouple ON/OFF/Homo** field determines first decoupler output:

- **y**, **ynn**, **yyn**, etc turns the **first decoupler on**.
- **n** or **nnn** turns the **first decoupler off**.

On UNITY*INOVA* systems, entering a or y (setting `dm` to `'a'` or `'y'`) specifies the asynchronous mode. In this mode, the decoupler rf is gated on and modulation is started at random places in the modulation sequence. Similarly, entering s specifies the synchronous mode in which the decoupler rf is gated on and modulation is started at the beginning of the modulation sequence. The `'s'` and `'a'` values have meaning only on UNITY*INOVA* systems. The second, third and fourth decouplers function analogously to the first decoupler.

A UNITY*INOVA* system is configured with transmitter fine power control. This fine power level is controlled by the `tpwrf` (or `tpwrm`) parameter. The attenuation is linear and spans 60 dB. If no fine attenuator is present, the value offsets the coarse power, simulating fine power.

The Homo selection box sets the homonuclear decoupling control for the decoupler. On UNITY*INOVA* systems, selecting Homo specifies that the receiver is gated, which is done by controlling the observe L.O. (local oscillator) line. The first decoupler rf, amplifier, and preamplifier are gated only if first decoupler is on. If the first decoupler is off, no gating of these signals takes place. When Homo is selected, Modulation should be set to c for continuous wave (CW) modulation.

On UNITY*INOVA* systems only, homonuclear decoupling for the second, third, and the deuterium decouplers is set the same way as the first decoupler.

On *MERCURYplus/Vx*, Homo has no meaning. Gated (homo) decoupling is used if the transmitter nucleus `tn` is ^{1}H or ^{19}F, and the decoupler mode is turned on.

CAUTION: **Decoupler power greater than 2 watts in a switchable probe will damage the probe. Always carefully calibrate high-power decoupling to avoid exceeding 2 watts of power. The maximum value for `dpwr` on a 200-, 300-, or 400-MHz system with a linear amplifier on the decoupler channel has been set to 49, corresponding to approximately**

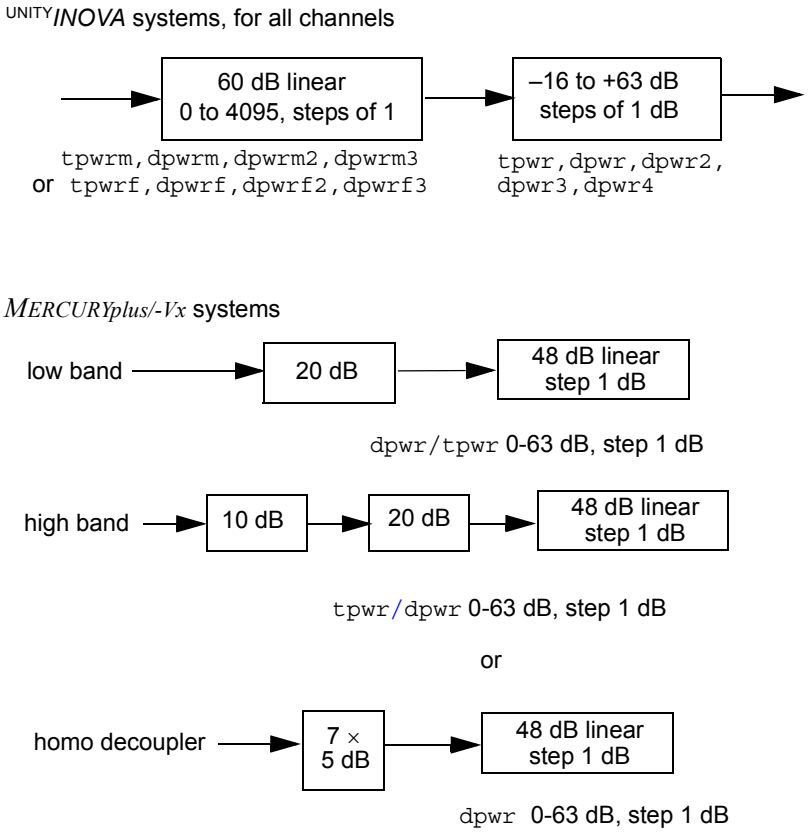


Figure 13. Attenuator Configurations

2 watts of power. Before using `dpwr=49` for continuous decoupling, ensure safe operation by measuring the output power. This safety maximum may be adjusted in the `config` program.

On systems equipped with a linear amplifier on the first decoupler channel, the decoupler power is set in the Decouple at power field, which is under computer control. This field be given values from 0 to 63, or from –16 to 63, depending on the range of attenuators present in the system. In both cases, 63 is the absolute maximum power. However, the output power should be measured to make sure a maximum of 2 watts is applied to switchable probes. This safety maximum, which limits the value that can be entered can be adjusted in the System Settings and System Configuration windows. The decoupler power for the second, third, and fourth decoupler channels, respectively. also have safety maximums.

Decoupler Modes

Several other efficient decoupling schemes are available from the Modulation pull-down menu, including GARP decoupling, MLEV-16 decoupling, and XY32 decoupling. Refer to the description of `dmm` in the *Command and Parameter Reference* for other modulation modes available.

In the standard two-pulse sequence, modulation normally has just a single “state,” since the decoupler modulation remains normally unchanged during the pulse sequence. Multiple states are possible; for example, `'ccw'` gives single-frequency decoupling during the first part of the pulse sequence, and WALTZ-16 decoupling during acquisition.

For systems with a waveform generator on a decoupling channel, set `dmm` to `'p'` to select programmable decoupling using that waveform generator. To specify the decoupling sequence during any period of waveform generator programmable decoupling, use the `dseq` parameter for the first decoupler, `dseq2` for the second decoupler, and `dseq3` for the third decoupler. The parameters `dres`, `dres2`, `dres3`, and `dres4` control the tip-angle resolution used within a programmable decoupling sequence on the first, second, third, and fourth decouplers, respectively. See the manual *User Programming* for further information on pulse control of waveform generators.

The following values are typical for decoupling:

- Homonuclear decoupling with linear amplifiers:

`dm='y'`	Decoupler mode on
`homo='y'`	Homonuclear decoupling on (INOVA only)
`dmm='c'`	Decoupler modulation mode is continuous wave
`dpwr=5–15`	Decoupler power level range (`dlp`, `dhp` nonfunctional)

- Heteronuclear decoupling with linear amplifiers:

`dm='y'`	Decoupler mode on
`homo='n'`	Homonuclear decoupling off (INOVA only)
`dmm='w'`	WALTZ-16 decoupling
`dpwr=40`	Decoupler power level (`dlp`, `dhp` nonfunctional)
`dmf=10000`	Decoupler modulation frequency

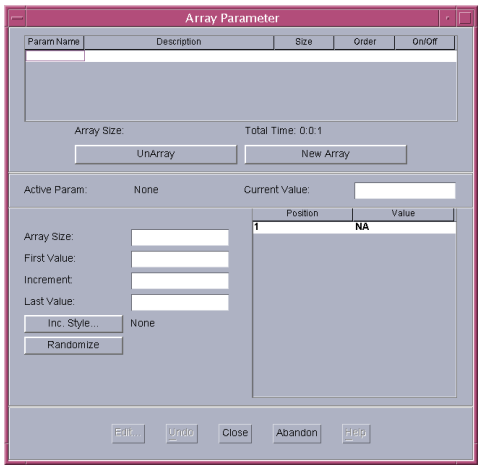
- Homonuclear decoupling:

`dm='y'`	Decoupler mode on
`dmm='c'`	Decoupler modulation mode is continuous wave
`dpwr=6–20`	Decoupler power level range

- Heteronuclear decoupling:

`dm='y'`	Decoupler mode on
`dmm='w'`	WALTZ-16 decoupling
`dpwr=45`	Decoupler power level
`dmf=10000`	Decoupler modulation frequency

5.4 Parameter Arrays



Use this function to open the Array Parameter window set arrayed parameters. This window can be used both within or outside of a study.

The top panel in the window is a table of currently arrayed parameters. The **Parameter Name**, **Description**, array **Size**, array **Order** and **On/Off** status are displayed. You can add parameters with the **New Array** button and remove parameters with **Unarray**. You can turn on or off an arrayed parameter by double-clicking **On/Off**. You can edit the array order (except for implicit arrays `ni, ni2`, etc.) to enable nested or parallel arrays. Only parameters of the same array size can be parallel, i.e., have the same array order.

You can highlight the array parameter table one row at a time with single clicks. The values of the highlighted array parameter (**Active Param**) are displayed in a table in the middle panel, along with editable entries for **Array Size**, **First Value**, **Increment**, and **Last Value**. You can also edit the parameter values. There are also buttons for increment style (**Inc. Style**), linear or exponential, and randomizing (**Randomize**) the order of array values. Specific values can also be entered manually for every element in the list of values.

The **Current Value** of the parameter is displayed above the array parameter values. You can select this value from the arrayed values by double-clicking the position number in the list of values or you can manually change it. When the parameter is unarrayed or turned off, it is set to the current value.

Abandon restores the original state of the window (the state it was in when it opened) and closes the window. **Close** keeps the changes and closes the window.

5.5 Stopping and Resuming Acquisition

Click the Stop button  or the Stop button next to the Acquire button to stop an acquisition experiment that has been submitted for acquisition.

If the experiment is waiting for execution, no action is taken. If the experiment is active, it is stopped and data is retained. Options are available for stopping the experiment at several user-specified places during acquisition:

- At the end of the next data accumulation.
- At a multiple of the value of the completed transients—can be used to complete a phase cycle before stopping.
- At the end of the next block size.
- At the end of the next complete FID.
- At the end of the next complete interleave cycle, i.e. the latest block size has been complete for all FIDs in the interleave cycle.

Clicking the Acquire button or selecting acquisition->Resume acquisition resumes a stopped acquisition.

Once an acquisition is in progress, it generally continues to completion; however, several situations can stop acquisition early. The system may detect an error, it may detect an overflow, or the operator may stop the system with an `aa` or an `sa` command.

If an acquisition is to be prematurely terminated (because sufficient signal-to-noise has been obtained or because the experiment has proved useless), select **Acquisition->Abort Acquisition**. If the experiment is active, it is aborted immediately, all data is discarded, and the experiment is interpreted as an error. Any data collected from an earlier block size transfer is retained. If any **Future Actions** for errors were defined, that processing occurs, followed by any queued experiments.

5.6 Data Precision and Overflow

DSP (digital signal processing) is set in the **System Settings** window. **Inline** enables software DSP and **Realtime** enables hardware DSP. **Precision** is set in the **Flags** page of the Acquire panel.

Single precision (*uncheck* **32-bit acquisitively** on the **Flags** page) is mainly designed for a single application related to imaging. Because DSP (digital signal processing) can give you 20 bits of data in a single acquisition, the 16-bit data size is usually not desirable. The console will detect a numeric overflow in hardware and post an error. Even without DSP, the standard 16-bit ADC boards can theoretically overflow after one transient. Therefore, *checking* **32-bit acquisitively** on the **Flags** page is the preferred setting.

Overflowing **32-bit acquisitively** real-time DSP is possible with greater than 4000 transients. Hardware DSP scales to 16 bits with single precision so that it will not overflow. As a result, many of advantages of hardware DSP are discarded.

Under some conditions, single precision is useful. The main example is “flash” imaging, where only one transient is typically taken, using very large data sets under fast conditions. In these experiments, the DTM memory (data-to-memory board memory, typically 16 Mbytes) can be filled up with rapidly acquired single-shot acquisitions. The single precision mode doubles the capacity of the DTM memory, increasing the number of increments in the experiment. If a cancellation experiment is carefully designed to avoid overflowing 16 bits, the single precision mode can cut the storage data size by a factor of 2. No down scaling is performed on UNITY*INOVA* so that the full available signal is always used. This use requires care in the cancellation cycle and an estimate when averaging will overflow 16 bits, so that **32-bit acquisitively** is still preferable for robust operation.

5.7 Automatic Processing

Set up automatic processing on the **Future Actions** page.

To examine data from the experiment in progress, the concept of the “block-size” is provided. The data system uses two independent computers, the host computer and the acquisition computer. When the parameter `bs` is set to some number, say 64, the acquisition computer is instructed, after every 64 transients, to provide the accumulated data up to that point to the host computer to be stored in an appropriate disk file (overwriting earlier data). Thus, every block size transients, an updated version of the experiment in progress is available for viewing by the user, who is communicating with the host computer. Weighting and transforming the data processes the current FID as of the last block size transients and display the resulting spectrum on the screen.

This process can be made automatic, because the host computer can detect whenever new data is present on the disk, using the Future Actions page (When Block Finishes field). If you enter `wft` and then click Acquire, you are telling the computer "When block size (that is, when `bs` transients are completed), perform the action `wft`." Now every 64 (or whatever the value of `bs` is) transients, the FID is automatically transformed and the spectrum displayed on the screen. Any command or macro can be invoked to occur automatically using the When Block Finishes field. If acquisition is started, `wbs` processing may still be set by using the `wbs` command, (e.g., `wbs('wft')`). `wbs` processing may also be disabled by entering `wbs('stop')`. Setting `bs='n'` before starting the acquisition disables this block-size storage. *If bs='n', data are stored on disk only at the end of the experiment, and. if the experiment is aborted prior to termination, data will be lost.*

There are other times when automatic processing is desirable:

- When an FID is finished, frequently you want it to be automatically transformed. You can accomplish this with the `wnt` (for "when number of transients") parameter, (e.g., `wnt='wft'`). This particular action, in fact, is automatically performed by the `ga` command, since it is so common.
- When more than one FID is being accumulated, we may want to reserve one particular action to occur at the end of all of the FIDs. We might be performing a 2D experiment and, after all the data have been accumulated, we want to perform a 2D transform. For this we use the `wexp` (for "when experiment") parameter, (e.g., `wexp='wft2d'`).
- When an acquisition error occurs, some corrective action may be desirable. You can accomplish this with the `werr` (when error) parameter, (e.g., `werr='react'`).

5.8 Acquisition Status Window

The Acquisition Status window normally appears when you click the black triangle next to the status display at the bottom of the VnmrJ interface.

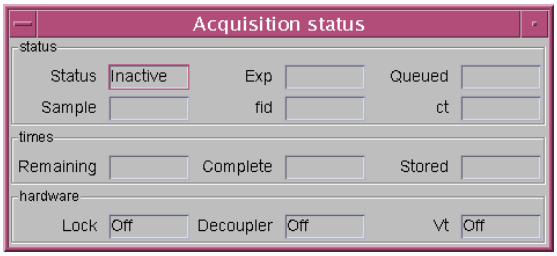


Figure 14. Acquisition Status Window

Figure 14 shows a typical Acquisition Status window when first opened. The display can contain 19 fields of acquisition status information, but all fields are not always displayed due to the hardware configuration of the system or the parameters set on the system.

Table 4 lists the possible fields, with a description of each field.

5.9 Applying Digital Filtering

The digital signal processing (DSP) provides many benefits, such as constant noise level across the spectrum, improved integral accuracy, increased dynamic range, and flatter baselines. DSP usually involves the following three steps, all of which are automatically performed by the DSP software:

- The first step is to *oversample* the data. Oversampling means acquiring data with larger spectral width using a larger number of data points. For example, instead of collecting an 8-Kword data point set with a 5-kHz spectral width, a 160-Kword data set is

Table 4. Fields in the Acquisition Status Window

Field	*Description*
Status	Present status of acquisition The values displayed should be self-explanatory (e. g., "Shimming") with two exceptions: "Active" means that the acquisition computer started but the console is not active yet, and "Inactive" means that `acqstat` cannot communicate with the acquisition computer or that the acquisition computer is not executing.
Queued	Number of experiments queued by multiple `go` commands
Exp	Number of the active experiment (e. g., `exp1`, `exp2`, `exp3`).
fid	Number of the FID being acquired if in an arrayed experiment
ct	Number of completed transients
Decoupler	Decoupler state: On, Off, Gated
Sample	Sample number in magnet if in automation mode
Lock	Lock status: Off, Regulated, Not Regulated
Complete	Estimated time when experiment will complete
Vt	Variable temperature unit status: Off, Regulated, or Not Regulated (if VT is set as present and `vttype=2`)
Stored	Last time data was transferred to disk.

acquired at 100 kHz. In DSP terms, this represents 20 times oversampling. An advantage of oversampling is that noise is reduced in situations where the noise in the time-domain FID is predominantly from round-off errors ("digitization errors") in the analog-to-digital converter (ADC). This happens at low spectrometer gain settings, where the "real" noise being sampled by the ADC is small, perhaps less than 3 bits. Another advantage is that digital filters should cause less distortion of the FID, leading to many of the benefits of DSP mentioned in the introductory paragraph of this section.

- After the oversampled data is acquired, the next step is to apply a *digital filter* to the time-domain signal, or FID, to remove signals and noise at frequencies outside the final desired spectral width. Digital filters are defined by the filtering algorithm and the number of coefficients for the filter. The more coefficients, the sharper the filter cutoff; however, the more complex the filter function and the greater the number of coefficients, the longer the calculation time required. In the example of oversampling above, the digital filter would be used to "cut-off" all frequencies outside the 5 kHz spectral width.
- The final step is to *downsample* the data. Downsampling (sometimes referred to as "decimation") means reducing the number of data points in the FID to the number actually required for spectral analysis at the chosen spectral width (the same number that you would have chosen if you had not used DSP). Again referring to the example above, the final FID would have 8-Kword data points and a final downsampled spectral width of 5 kHz.

Types of Digital Filtering

The main types of DSP provide digital filtering during the acquisition of data:

- Inline DSP uses software on the workstation to perform digital filtering and downsampling immediately after each oversampled FID is transferred from the console. Only the digitally filtered and downsampled data is written to disk. The

advantages of the inline DSP include no increase in disk storage and user-defined filter functions. A potential disadvantage is some additional load on the workstation. Inline DSP software is compatible with all systems with Sun host computers.

- Real-time DSP uses a dedicated embedded processor chip installed on the input board of certain systems (such as the UNITY*INOVA*) to filter the data prior to time averaging. The advantages of real-time DSP include no increase in data storage and compatibility with ultra-rapid experiments (because there is no additional loading on the workstation). A disadvantage is that fewer parameters are available to control real-time digital filtering.

Another type of DSP, postacquisition DSP, integrates the digital filtering and downsampling process into the Fourier transform commands `ft` and `ft2d`. The digitally filtered and downsampled FID can then be saved to disk. The advantage is that the original FID is not altered in the original experiment, multiple applications of digital filtering are possible, no limitations exist on filter shape complexity and filter cutoff, and user-defined filter functions are possible. The main disadvantage is the large disk storage required. The usual disk storage requirements are directly multiplied by the oversampling factor, which ranges from 2 through 68. Postacquisition DSP software is compatible with all systems with Sun host computers.

In all types of DSP, the signal is initially filtered at the oversampled spectral width by an analog anti-aliasing filter.

Control of DSP

The value of parameter `dsp` specifies the type of DSP for data acquisition: `'i'` for inline DSP, `'r'` for real-time DSP, or `'n'` for none. As a global parameter, `dsp` affects DSP operation in all experiments and should be thought of as an hardware configuration parameter, because if DSP hardware is available (such as on the UNITY*INOVA*), that hardware is generally the method of choice.

Normally, DSP works quite invisibly to the user. Regardless of whether `dsp` is set to `'i'` or `'r'`, the oversampling factor (the parameter `oversamp`) is automatically set to the maximum allowed value whenever `sw` is entered. Thus, after the user enters `sw`, `at`, and/or `np` in the normal manner, the software and hardware automatically oversamples at the maximum rate, and then digitally filters and downsamples the data according to the selected `sw` and `np` parameters.

If the user wishes to disable DSP for a particular experiment, the `oversamp` parameter can be set to `'n'` and oversampling and DSP are not be used. Or if desired, the `oversamp` parameter can be set by direct numeric entry to a value less than the maximum (e.g., `oversamp=4`). Be warned, however, that `oversamp` is reset to its maximum value the next time `sw` is entered in that experiment.

Inline DSP

Inline DSP applies digital filtering and downsampling to the acquired data prior to storage to disk on the host computer. Only the downsampled data set is stored using this method. DSP prior to data storage to disk has a time constraint: the digital filtering and downsampling must be completed within the time between transfer of successive data blocks (or increments of a 2D experiment) to the host computer disk. Additional processing done during acquisition, such as `wnt`, also adds to the time constraints. This processing can limit the speed of rapid arrayed experiments and depends on the type of Sun host computer and the parameters used.

Inline DSP is activated by setting the global parameter dsp to 'i'. If dsp is not present or is set to 'n', DSP is disabled. If dsp='i', setting the oversamp parameter to a value greater than 1 in a particular experiment causes the next experiment run to be oversampled, digitally filtered, and downsampled back to the selected sw prior to saving it to disk.

To Apply Inline DSP

1. Open **System->System Settings**. Set **Type of digital signal processing** to **Inline**.
2. Set acquisition parameters to the values desired for the final spectrum.
3. As required, adjust the values of the oversampling parameters **oversamp**, **oscoef**, **oslsfrq**, **osfb**, and **filtfile**:
   - oversamp specifies the oversampling factor (68 or less) for the acquisition. As a result, np*oversamp data points are acquired at a rate of sw*oversamp. Once the data has been transferred to the host computer, it is digitally filtered and downsampled to give np points and a spectral width of sw. sw*oversamp and np*oversamp are limited by the values given in Table 5.:

     Table 5. Maximum Values for sw*oversamp and np*oversamp

System	*Maximum* sw*oversamp	*Maximum* np*oversamp
UNITY*INOVA*	500 kHz	2M
MERCURYplus/-VX	100 kHz	256 K

     The maximum np*oversamp is given for double precision data (dp='y'). For dp='n', multiply the value by 2. The value of oversamp might need to be decreased further for rapid arrayed experiments, because of host computer memory and speed limitations. Setting oversamp to 'n' causes normal acquisition to be done without digital filtering
   - oscoef specifies the number of coefficients used in the digital filter. The default is 7.5*oversamp+1. A larger number of coefficients gives a filter with sharper cutoffs; a smaller number of coefficients gives a filter with more gradual cutoffs. The value of oscoef does not need to be changed when oversamp is changed because oscoef is automatically adjusted by VnmrJ to give filter cutoffs that are the same regardless of the value of oversamp.
   - oslsfrq is used to select a bandpass filter that is not centered about the transmitter frequency. oslsfrq is specified in Hz and works much like lsfrq. A positive value of oslsfrq selects a region upfield from the transmitter frequency, and a negative value selects a downfield region. The oslsfrq parameter can be used to perform frequency-shifted quadrature detection (see section “Removing Quadrature Artifacts Using DSP” on page 85).
   - osfb specifies the digital filter bandwidth. If osfb='n', the bandwidth defaults to sw/2. A value less than sw/2 rejects frequencies at the edges of the spectrum; a value more than sw/2 aliases noise and signals at frequencies outside of ±sw/2.
   - filtfile specifies the name of a file of finite impulse response (FIR) digital filter coefficients. The file must be in the user's vnmrsys/filtlib directory. The filter coefficient file is a text file with one real filter coefficient per line. Complex filters are not currently supported. To use the default filter

coefficients calculated by VnmrJ, `filtfile` should be set to the empty string (`''`, i.e., two single quotes with no space between them).

4. If `oversamp` is set to a value greater than 1, the next experiment is oversampled, digitally filtered, and downsampled to the `sw` selected prior to saving it to disk.

After acquiring a data set without digital filtering, the `moveossw` macro can be used to set `oslsfrq` and `sw` to appropriate values for oversampling and digitally filtering for the region of the spectrum selected between the cursors in the `ds` display. You must manually set `oversamp` to an appropriate value.

Real-Time DSP

Real-time DSP, available on certain systems (such as the UNITY*INOVA*), applies digital filtering during data acquisition, prior to storing the data in the memory of the acquisition computer. Data sampling is performed at a maximum rate of 400 kHz, with a maximum oversampling factor of 68. Thus, a typical 7-kHz spectrum is oversampled at a factor of 57; a 25-kHz spectrum is oversampled at a factor of 16. Above `sw=200000`, oversampling (and hence real-time DSP) is not possible and is automatically deactivated.

Oversampling lessens the effect of digitization noise on the spectrum, more so the more oversampling is done. At low gain, this can produce a marked improvement in the obtainable signal-to-noise (S/N) ratio. Equivalently, the use of oversampling and digital filtering will produce the same S/N at lower receiver gain values.

To Apply Real-Time DSP

When real-time DSP is first installed on a system, each user should enter `dsp?` to check that real-time DSP (`dsp='r'`) is set on the system. From that point on, the software automatically calculates oversampling factors and performs experiments using real-time DSP in a manner totally transparent to the user.

- To turn on Real-Time DSP for the system, open **System->System Settings**. Set **Type of digital signal processing** to **Inline**.
- To turn DSP off in a single experiment, set `oversamp='n'`; to turn DSP back on in that experiment, set `oversamp='y'`.
- To turn off DSP "permanently," for all future experiments (until you decide to turn it back on again), open **System->System Settings**. Set **Type of digital signal processing** to **None**.

Types of Real-Time Digital Filters

Two different digital filters are supplied with real-time DSP. The first type, the Analog*Plus*™ filter, was designed to have similar characteristics to traditional analog filters, while using digital technology to improve on the analog filter in every way. The Analog*Plus* digital filter is flatter in the passband (the spectral region of interest) than an analog filter, and has sharper cutoff in the stopband (the region outside the spectrum to be filtered out). This gives better quantitation across more of the spectrum and reduced "noise fold-in" compared with analog filters, in addition to the improvement in S/N from the removal of digitization noise.

When comparing Analog*Plus* digital filters with analog filters, note that when using a "real" analog filter, VnmrJ increases the filter bandwidth `fb` by 10%, compared with half the spectrum width, in order to provide better filter flatness across the spectral region of interest. This increase, however, causes significant noise to fold in to the spectrum. With

the Analog*Plus* digital filter, however, the filter bandwidth (the 3-dB point) is set to exactly `sw/2` to ensure the best possible S/N across the spectrum.

The second type of digital filter provided, the Brickwall filter, has much sharper cutoff characteristics (as implied by its name) than the Analog*Plus* filter and is flatter even closer to the edges of the spectrum. This enhanced filtering may come at the expense of baseline performance, however. Users working with "simple" spectra, such as of organic compounds, should not notice this at all, but for work with high-dynamic range spectra or spectra of proteins, the baselines obtainable with the Brickwall filter may not be as good as the baselines obtained with the Analog*Plus* digital filter.

The global parameter `def_osfilt` specifies whether the digital filter you normally prefer is `'a'` (Analog*Plus*) or `'b'` (Brickwall). Once you set `def_osfilt`, you need not change it again. You can set a local parameter in each experiment, `osfilt`, to `'a'` or `'b'` to run a specific type of digital filter in that experiment, without changing your default choice.

The amount of oversampling performed by the system is contained in the parameter `oversamp`, which is normally calculated by the software to be the maximum possible for any given spectral width. You can change `oversamp` (to smaller values only) if you want to, but since the maximum benefit of DSP is only obtained with the maximum possible oversampling, there is little reason to do so.

For Brickwall filters, which use more coefficients than Analog*Plus* filters, some difference appears in the steepness of the filter as a function of the `oversamp` parameter. At `oversamp` set from 2 through 7, the cutoff is the steepest; as `oversamp` increases, the filter becomes slightly less steep and approaches Analog*plus* filters at `oversamp=50`. Thus, if you want the flattest (in the passband) and sharpest (in the stopband) possible filter, and if you are not operating at low gain where oversampling is important to S/N, you may wish to use minimum oversampling (`oversamp` set from 2 through 7). Brickwall filters at oversampling factors of 20 to 40 make a nice compromise filter with better amplitude flatness than Analog*plus* and better baselines than Brickwall set at lower oversampling factors.

Real-Time DSP Details

Real-time DSP is not compatible with pulse sequences that use explicit acquisition to acquire less than the full number of data points (`np`) in a single `acquire` statement (e.g., solids sequences such as `br24` and `flipflop`). This incompatibility is taken care of automatically by the software, which turns off DSP and sets `oversamp='n'` if you attempt to acquire data using an incompatible pulse sequence. If you want to obtain the benefits of DSP in such experiments, use inline or postacquisition DSP.

Preserving the full potential dynamic range benefit of DSP for `dp='y'`, real-time DSP contains an inherent "gain" of 16 or the equivalent of 20 bits of data. A consequence of this gain is that if you look at the output of a single transient, the largest possible signal is no longer ±32768 (2^{16}) but instead ±524288 (2^{20}). In other words, the system behaves as if it has a 20-bit digitizer instead of a 16-bit digitizer.

For `dp='n'`, the "gain" of 16 mentioned above is disabled and, therefore, the main data value is 32767.

Another consequence of the gain with `dp='y'` is that if the ADC is filled on a single transient and the signals add coherently on successive transients (as they do in most experiments but not in, say, an indirect detection experiment), after a minimum of 4096 transients ($2^{32}/2^{20}$), the accumulating signal can overflow the available memory (the largest signal becomes greater than 2^{32} and cannot be stored in memory). The hardware

does not prevent this overflow, and it is possible to obtain data that is useless in such a case. If you need to run more than 4096 transients, use inline or postacquisition DSP.

One subtle point involves ADC overflow on a single transient. When DSP and hence oversampling are activated, two changes occur that affect the maximum signal seen by the ADC. First, the initial sampling occurs earlier in time. The system may have acquired 10 points of an oversampled FID before it would have acquired a single point in a "normal" FID. Thus, any transient signals (e.g., probe background, pulse breakthrough) that occur at the front of the FID are more likely to cause ADC overflow in an oversampled FID. Second, the analog filter bandwidth is now set to a larger value—it may have gone from 2.5 kHz up to 50 kHz. Any large "out-of-band" signal that was being filtered out by the analog filter appears at the ADC to be digitized (and then to be subsequently filtered out by the digital filter). Thus again, ADC overflow may occur under conditions (i.e., identical pulse width and gain) where it did not occur when DSP was not used.

Even more subtly, because of the digital filtering that occurs, the output of the digital filter may actually have a value less than the maximum possible value, even though the input to the digital filter did indeed exceed the ADC limit. Thus, you should not be surprised if you occasionally need a very slightly lower gain to avoid ADC overflow when using DSP, and you should also not be surprised if you examine the (output) signal and do not see any evidence of ADC overflow, despite having been told by the software that ADC overflow did occur.

As explained earlier however, one of the real advantage of DSP with significant oversampling is the ability to work at lower gain settings while maintaining full signal-to-noise. This "headroom" afforded by DSP makes it is far less important to carefully adjust the gain setting and fill the ADC. Thus, gain settings 6 to 10, or even 20 dB below ADC overflow, are likely to give perfectly acceptable results.

Data Format Issues

The output of inline and real-time DSP is a "normal" FID, without the distortion associated with the large frequency-dependent phaseshift associated with some digital filters, and with characteristics (such as `np`) that are identical to an FID obtained without DSP. As such, the output can be processed by any software (VnmrJ or third party) that can process standard VnmrJ FIDs. Real-time DSP FIDs are always in fixed point format (16- or 32-bit, depending on the value of the parameter `dp`).

The output of inline DSP is also a "normal" FID that can be processed in standard ways. If `dp='n'`, the FID is in a 16-bit fixed point format; however, if `dp='y'`, the FID is in 32-bit floating point format, not 32-bit fixed point.

VnmrJ processes such FIDs transparently, but some third-party software may not be compatible with this mode

Obtaining Good Baselines with Inline and Real-time DSP

The algorithms used by inline and real-time DSP processing now contain Varian's time-corrected zero-phase digital filters. These filters allow very flat baselines to be obtained with no frequency dependent phase shift across the spectra. Getting these flat baselines does require some changes in setup and acquisition parameters from those used for analog filters.

Many users working with spectra of proteins are accustomed to using the `hoult` and `calfa` macros to adjust the acquisition conditions such that spectra are obtained with a frequency-dependent phase shift (`lp`) of zero and with minimal distortions of the second

and subsequent data points with analog filters. These conditions are typically satisfied with negative values of alfa.

The same acquisition conditions do not result in the flattest possible baselines when using real-time DSP. The following procedure is recommended to set alfa and rof2:

1. Start by using "normal" positive values of alfa and rof2, (e.g., alfa=6 rof2=2; in fact, the software automatically sets these values the first time you activate DSP).
2. Obtain a spectrum and phase it properly.
3. Enter **crof2** to recalculate rof2 so that lp will be zero.
4. Reacquire the spectrum to verify that lp is now zero.

After the last step, the baseline should be reasonably good. To improve it even further, you can make fine adjustments to alfa and rof2, keeping the sum of the two constant (e.g., enter alfa=alfa-0.5 rof2=rof2+0.5 ga). Once good values are obtained, you should find them relatively invariant with sw (as long as maximum oversampling is used).

The calfa macro has also been modified for use with DSP. calfa now sets alfa to the default value of about 6 µs and then adjusts rof2 to set the appropriate timing for lp=0. If a value of alfa other than the default value is found to be preferable using the above methods, use crof2 to adjust acquisition timing for the lp=0 condition without changing the preferred value of alfa.

Postacquisition DSP

The software allows postacquisition digital filtering and downsampling to selectively detect a region of a spectrum. The digital filtered and downsampled FID can then be saved to disk. The digital filtering and downsampling processes are integrated into the ft and ft2d commands and occur when these commands are executed as specified by the parameters below. The digital filtering and downsampling are done just prior to the Fourier transform, so all apodization, linear prediction, solvent suppression, etc. are done prior to digital filtering.

Postacquisition digital filtering uses the same algorithm as inline DSP, with a transition bandwidth correction enhancement to minimize baseline distortion.

Application of postacquisition DSP takes the following steps:

1. Acquire a data set with sw = N*(final desired sw) and np = N*[(final desired np) + dscoef/2)], with N the oversampling factor.

   In most situations, you can use np = N*(final desired np) because the final np is usually much larger than dscoef/2.
2. After a data set has been acquired, enter the macro pards to create additional parameters downsamp, dscoef, dslsfrq, dsfb, and filtfile used by downsampling.
3. Setting up of the parameters can be made easier if an initial Fourier transform spectrum exists already. In this case, the macro movedssw can be used to set the parameters by using cursors in the ds spectral display. Position the vertical cursors around the region of interest and enter movedssw. Otherwise, set the parameters as follows:
   - downsamp specifies the downsampling factor applied after digital filtering. For example, starting with a spectral width of 100 kHz and downsamp set to 20, downsampling reduces the final spectral width to 5 kHz. The spectral width

sw of the data set after digital filtering and downsampling is (acquired sw)/ downsamp. Setting downsamp to 1 allows digital filtering with a filter bandwidth specified by dsfb without downsampling. Setting downsamp to 'n' allows normal data processing in VnmrJ without digital filtering.

- dscoef specifies the number of coefficients used for filter computation. The default of 61 is usually a good choice. A larger number of coefficients gives a filter with sharper cutoffs, and a smaller number of coefficients gives a filter with more gradual cutoffs. Larger values in the range of 199 to 399 coefficients may have to be used to prevent aliasing of large peaks just outside the downsampled window. dscoef does not need to be changed as downsamp is changed, because dscoef is automatically adjusted by VnmrJ to give filter cutoffs that are the same regardless of the value of downsamp. This is done by actually using dscoef*downsamp/2 coefficients in the digital filter. VnmrJ always rounds dscoef*downsamp/2 to an odd number.
- dslsfrq is used to select a bandpass filter that is not centered about the transmitter frequency (tof). dslsfrq is specified in Hz and works much like lsfrq. A positive value of dslsfrq selects a region upfield from the transmitter frequency and a negative value selects a downfield region. Bandpass filters are used to select regions away from the transmitter frequency.
- dsfb specifies the digital filter bandwidth, which is set to half of the downsampled spectral width by default. If dsfb='n', the default value for the filter bandwidth is used. A smaller value rejects frequencies at the edges of the spectrum, and a larger value aliases noise and signals at frequencies outside of ± sw/2.
- filtfile specifies the name of a file of finite impulse response (FIR) digital filter coefficients. The file must be in the user's vnmrsys/filtlib directory. The filter coefficient file is a text file with one real filter coefficient per line. Complex filters are not currently supported. To use the default filter coefficients calculated by VnmrJ, filtfile should be set to the empty string ('', i.e., two single quotes with no space between them).

4. Once the parameters have been set, the filtered and downsampled dataset can be saved by using the macro digfilt(exp_number<,option>) to write the digitally filtered FIDs to another experiment. The possible options available with the digfilt macro are 'nodc', 'zero', and 't2dc'. Use these options if you used the same option when processing the data with ft, wft, ft2d, or wft2d. If ct=1, it may also be useful to use dcrmv='y' during data processing. If proc='lp', linear prediction will be done prior to digital filtering. Apodization will also be done prior to digital filtering.
5. Carry out the digital filtering and Fourier transformation by entering wft, or in the case of 2D datasets, by entering wft2d. The digital filtering and downsampling step takes place after all other processing on the FID (dc, solvent suppression, linear prediction, apodization, etc.).

Removing Quadrature Artifacts Using DSP

Normally, NMR spectra are acquired with the receiver in the center of the spectrum, and the center (zero) frequency glitch and quadrature artifacts fall within the desired spectral width. The use of the oslsfrq parameter with inline DSP allows these artifacts to be removed by the digital filter before downsampling so that they are not present in the downsampled FID that is stored on the disk. This technique is called frequency-shifted, or “digital,” quadrature detection.

Frequency-shifted quadrature detection is performed by moving `tof` to just outside the desired spectral width and then using `oslsfrq` to offset the digital filter center frequency by the same amount, thus keeping the region of interest after digital filtering. Frequency-offset filtering is done in the inline DSP algorithm.

- If real-time DSP is used (`dsp='r'`, `fsq='y'`), `oversamp` is set to a multiple of 4, and downsampling of a factor of `oversamp/4` is then done in real-time DSP. The remaining factor of 4 is then done in inline DSP, during which frequency-shifting by `oslsfrq` is also done. This feature is available only on UNITY*INOVA* systems.
- If pure inline DSP is used (`dsp='i'`, `fsq='y'`), filtering and frequency-shifting is done in a single stage.

To Apply Frequency-Shifted Quadrature Detection

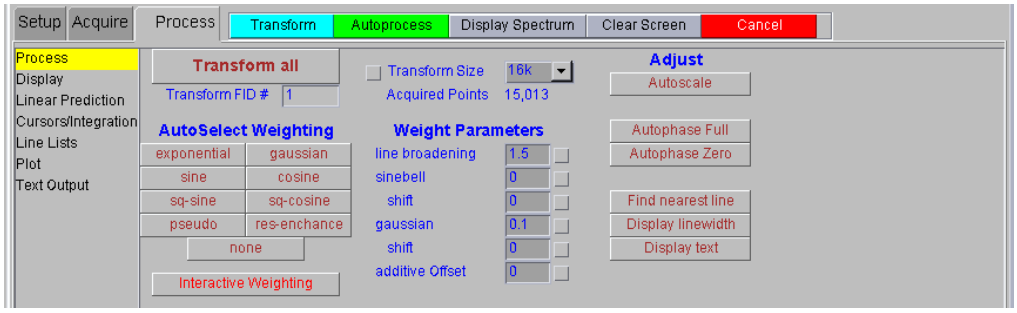
1. Open **System->System Settings**.
2. Set **Type of digital signal processing** to **Inline** or **Realtime**.
3. Click the selection box next to **Frequency-shifted quadrature detection**.

Doing these steps sets `oslsfrq` to `1.25*sw` and offsets the transmitter frequency to `tof+oslsfrq` just before acquisition. If a different value of `oslsfrq` is desired, it can be entered after `fsq` is set to `'y'`.

Chapter 6. Processing Data

Sections in this chapter:

After data are acquired, the next step in the process is applying a “weighting function” to the FID, which is an optional part of the process, and Fourier transformation, which is not. Both operations are done using the Process page on the Process panel.



6.1 Weighting Function

The weighting function used is governed by the following parameters:

- **exponential** – A positive value gives the desired line broadening in Hz, which is then used to calculate a decaying exponential function. A negative value gives a resolution enhancement function.
- **gaussian** – Time constant, in seconds, and defines a Gaussian function of the form `exp(-(t/gf)2)`.

  shift – shifts the center of the Gaussian function `exp(-((t-gfs)/gf)2)`.
- **sinebell** – A positive value, in seconds, applies a sinebell of the form `sin(t*p/(2*sb))`. A negative value applies a squared sinebell function of the form `sin2(t*p/(2*sb))`.

  shift – a sinebell shift constant, in seconds. It allows shifting the origin of the sinebell function according to the formula `sin((t-sbs)*p/(2*sb))`. Again, the square of this function is applied if `sb` is negative.

- **additive Offset** – an additive weighting constant that adds the constant `awc` to each value of the weighting function. It is applied *after* the sinebell and exponential function but *before* the Gaussian function.

All these weighting functions can be applied simultaneously to the data. That is, one does not first apply a decreasing exponential function, then apply a convolution difference function, etc. Instead, all weighting functions to be used are set, and then are applied simultaneously as part of the Transform process. To remove a particular weighting function from use, deselect its check box.

Although the system allows the combination of sinebell, exponential and Gaussian weighting, a combination of those can be difficult to understand and should only be used after experimenting with the individual parameters. The use of either Gaussian apodization, which leads to Gaussian line shapes, or line broadening (greater than 0), which leads to Lorentzian lineshapes, is especially critical for deconvolution.

Other line shapes cannot be handled by the deconvolution program, but may be appropriate for 1D resolution enhancement or in absolute-value 2D experiments. In any case, weighting affects the integrals of different lines in different ways, and should be used with great care if quantitative results are requested.

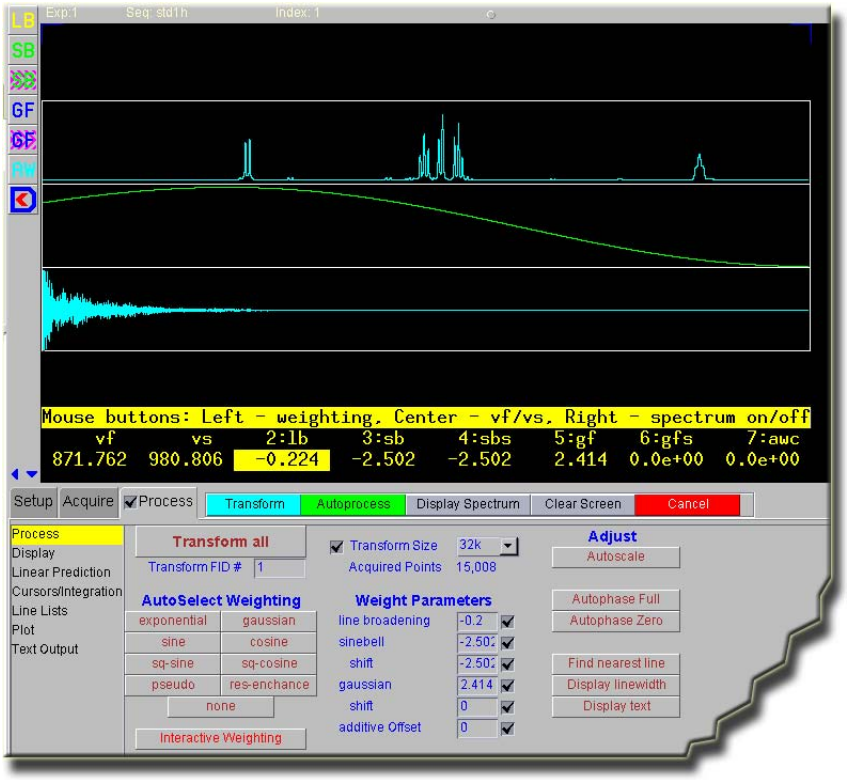
The res-enhance button sets defaults of `a` equal to 0.1 and `b` equal to 0.3 into the formulas `lb=-0.318/(a*sw)`, and `gf=b*sw`, thereby calculating “reasonable” values for the resolution enhancement parameters `lb` and `gf`. The arguments `a` and `b` can also be selected by the user.

Several macros exist that set weighting parameters to give certain window functions. These include `gaussian`, `pi3ssbsq`, `pi4ssbsq`, `sqcosin`, and `sqsinebell`.

The parameter `wtfile` is available for handling user-written weighting functions; see the manual *User Programming* for details

6.2 Interactive Weighting

To start interactive weighting, click the Interactive Weighting button on the Process panel.



The Interactive Weighting window provides button to the left of the graphics window:

LB	Selects line broadening or exponential weighting. A negative value gives resolution enhancement.
SB	Selects the sinebell constant. A negative value gives squared sinebell.
SB >>	Selects the sinebell shift constant (only if sinebell is active).
GF	Selects the Gaussian time constant.
GF >>	Selects the Gaussian time constant shift (only if Gaussian time constant is active).
AW	Selects the additive weighting constant.
return	Returns to the last menu before interactive weighting was entered.

Currently active weighting parameters can be changed by moving the mouse cursor to the appropriate field in the weighting function box and pressing the left mouse button. New values for weighting parameters can also be typed in. Note that all other parameters, unless set to “not used”, are also used to calculate the weighting function.

To adjust FID intensity (parameter `vf`), use the center mouse button within the FID box. To adjust spectrum intensity (`vs`), use the center mouse button within the spectrum box.

The right mouse button turns off and on the display of the transformed spectrum, useful for modifying the weighting function on slow terminals or large spectra.

6.3 Fourier Transformation

The Transform button Fourier transforms one or more FIDs without weighting applied to the FID.

To perform the same Fourier transform with weighting, select one or more of the weighting functions and click the Transform button.

The Transform button performs a shift and phase rotation according to the parameters set on the Linear Prediction page, if these are set. Any baseline drift correction in the FID is automatically calculated and removed. Baseline Correct buttons are located on the Display page.

For the Fourier transformation process, the Transform Size field is the number of points to be Fourier transformed (`fn`). Because of the type of algorithm used, this number must be a power of two; typical numbers are 16384, 32768, or 65536 (listed as 16K, 32K, and 64K, where K is equivalent to multiplying the number by 1024). The most common entry for Transform Size is Default. This value specifies that however many data points (`np`) were acquired, the first power of two greater than or equal to `np` will be used as `fn`. If `fn` is greater than `np`, or if `fn` is `'n'` and `np` is not a power of two, the remaining points in the transform are filled in with values of zero (*zero-filling*). Thus there is no explicit zero-filling command; this process is an implicit one governed by `fn`.

6.4 Phasing

Phasing spectra may be considered part of either data processing or data display. Performing a complex Fourier transformation produces two sets of data, referred to as the *cosine* and *sine* transforms, or the *real* and *imaginary* channels, respectively. In almost all cases, the absorption spectrum (peaks “in-phase”) and the dispersion spectrum (peaks “out-of-phase”) do not coincide with the real and imaginary channels, but must instead be produced from a linear combination of the two channels.

Phasing can be adjusted using Phase button  for interactive phasing, or using the Autophase functions on the Process page.

Phase Parameters

The process of phasing a spectrum requires the determination of an angle θ that can be used to “mix” these two data sets to produce one data set, according to the formula:

$$\textit{absorption spectrum}_{\omega} = \textit{real channel}_{\omega} * \cos\theta + \textit{imaginary channel}_{\omega} * \sin\theta \qquad \text{[Eq. 1]}$$

The process is complicated by the fact that phase angle θ is a function of frequency:

$$\theta = \texttt{rp} + (\omega - \omega_0) * \texttt{lp} \qquad \text{[Eq. 2]}$$

where `lp` (left or first-order phase) and `rp` (right or zero-order phase) are constants that must be determined.

The following is clear about the terms in Equation 2:

- `rp` is *frequency independent*. Changes in `rp` affect all peaks in the spectrum equally.
- `lp` is *frequency dependent*. Changes in `lp` affect peaks with a differing amount as a function of frequency.

There are several ways in which `lp` and `rp` can be adjusted:

- Like any parameter, they can be recalled with a particular parameter set. Once entered, they can also be entered directly (e.g., `lp=-150`).
- Fully automatic phasing is also provided with the `aph` command, which optimizes both the frequency-dependent (`lp`) and the frequency-independent (`rp`) parameters, and is independent of the starting point. The `aph0` command only adjusts `rp`. The `aphx` macro optimizes parameters and arguments for the `aph` command. `aphx` first performs an `aph` then calculates a theoretical value for `lp`. If `lp` set by the `aph` is different from the calculated value by 10 per cent, the calculated value is used and an `aph0` is performed.

The command `phase(phase_change)` changes the phase of all peaks in the spectrum by adding `phase_change` to the current value of `rp`, then removing any excess in `rp` more than 360°.

Autophase Algorithm

The automatic phasing algorithms `aph` and `aph0` have been enhanced in several ways:

- Weighting parameters no longer affect the algorithms.
- Spectra with very low signal-to-noise can be phased.
- In vivo spectra can be phased. These spectra are very difficult for most autophasing algorithms.
- Spectra with inverted lines can be phased. Such spectra includes DEPT experiments or selectively inverted lines obtained with shaped pulses. This type of phasing is difficult for traditional autophasing algorithms, which cannot distinguish when a line is inverted and when a line is normal.

The autophasing algorithm uses many rules that are used in a manual phasing procedure. First, it finds the peak areas. Then, it estimates the correct phase for each peak. An initial guess of the first order phasing parameter `lp` is made based on the estimated phases of two "normal" peaks. The peaks are categorized into three classes: normal, inverted, and bad. The peaks in the normal and inverted group will be used to find the optimal values for the phasing parameters `lp` and `rp`. A final check is made to determine whether autophasing was successful or unsuccessful.

Algorithms are complicated but fairly "intelligent." The key point of an algorithm is to use a set of fuzzy rules to estimate the correct phase for each peak. The use of these rules makes an algorithm less sensitive to the signal-to-noise ratio, weighting parameters, and the base line quality. Fuzzy logic also makes it possible to do the classifications on the peaks.

The command `aphb` autophases Bruker data. Refer to the *Command and Parameter Reference* for more information about this command.

Spectrum Display

The displayed spectrum is calculated in one of four *mutually exclusive* modes:

- The *phase-sensitive mode* is selected by the command `ph`. In this mode, the displayed spectrum is calculated using the phase parameters `lp` and `rp`.
- The *absolute-value mode* is selected by the command `av`. In this mode, the displayed spectrum is calculated according to the equation

  $$absorption\ spectrum(\omega) = (real\ channel^2(\omega) + imaginary\ channel^2(\omega))^{1/2}$$

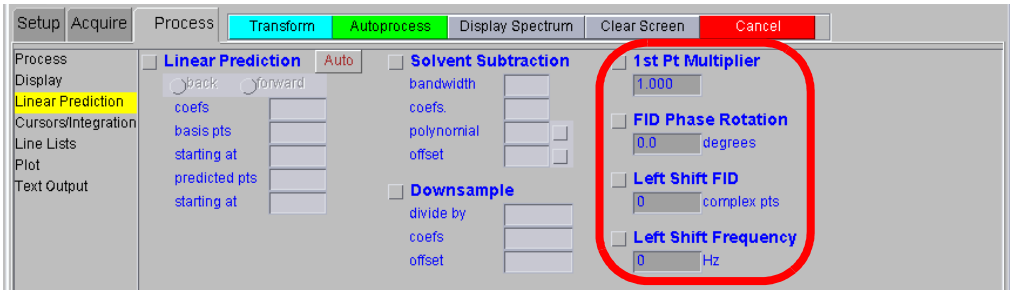
- The *power mode* is selected by the command `pwr`. In this mode, the displayed spectrum is the square of the displayed spectrum calculated in the absolute value mode.

- The *phase-angle mode* is selected by the command `pa`. In this mode, each point in the displayed spectrum is the arctangent of the phase angle of the real and imaginary point.

Once a spectrum is displayed using the interactive display command `ds`, the spectrum can be interactively phased by selecting the Phase button from the menu. Any integral and cursors displayed along with the spectrum are removed.

6.5 Advanced Data Processing

This section covers advanced data processing, including phase rotation, frequency shifting, linear prediction, and interleaving FIDs. These functions are available on the Linear Prediction page of the Process panel.



FID Phase Rotation

1st Pt Multiplier	Allows error correction if the first point of the FID is misadjusted. Refer to the `fpmult` parameter in the *Command and Parameter Reference*.
FID Phase Rotation	The parameter `phfid` is a zero-order FID phasing constant. If `phfid` is set to a value other than `'n'`, the FID is phase rotated by `phfid` degrees before weighting or Fourier transformation is performed.
Left Shift FID	The parameter `lsfid` is a constant used in left-shifting the FID. If `lsfid` is set to a value other than `'n'`, the FID is left-shifted by `lsfid` complex points before weighting or Fourier transformation is performed. The value for `lsfid` must lie between `0` and `np/2`. The `tmove` macro provides a method of setting the parameter `lsfid`—position the right time cursor at the place that should be the start of the FID, then enter `tmove` to adjust the parameter `lsfid`.
Left Shift Frequency	Sets the frequency shift of spectral data, in Hz. Refer to `lsfrq` in the *Command and Parameter Reference*.

Frequency Shifting

Left Shift Frequency	Sets a frequency shift of spectral data, in Hz, with a negative value resulting in peaks being shifted upfield (to the right) and a positive value in peaks being shifted downfield (to the left). `lsfrq` operates in the time domain on complex FID data, and thus must be entered

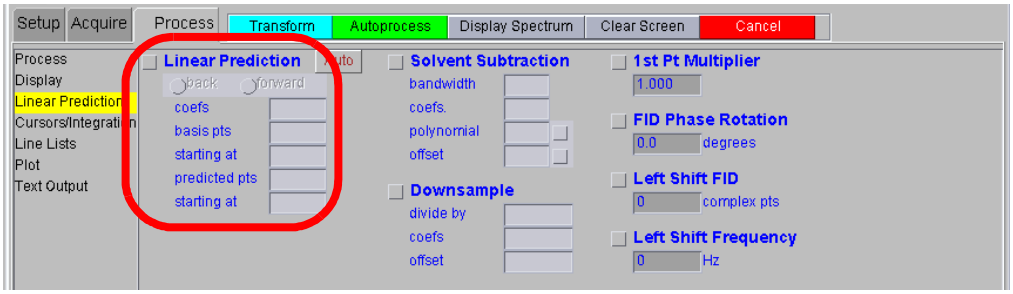
Data Processing Methods

All data processed in VnmrJ is processed using the method of Fourier transformation, but there are three variations that are governed by the `proc` parameter:

- Most NMR data is acquired using *simultaneously* sampled ("true") quadrature detection. This means that two orthogonal (real and imaginary, or *x* and *y*, as you prefer) data points are sampled at the same time and form a single complex data point in the FID. Such data are processed using a normal complex Fourier transformation, using `proc='ft'`.
- Some spectrometers, notably those from Bruker Instruments, acquire pseudo-quadrature data by sampling two orthogonal data points *sequentially*, rather than simultaneously. Such data must be processed using a real Fourier transformation, with `proc='rft'`.
- For simultaneously sampled data only, it is possible to include as part of the Fourier transform process a "linear prediction," described in the next section. `proc='lp'` is used to trigger this operation.

Linear Prediction

Use the Linear Prediction page to activate (default) or deactivate linear prediction and to adjust linear prediction parameters.



Linear Prediction in VnmrJ

In VnmrJ, linear prediction is incorporated directly into the Fourier transform routine, so that normally one does not see the "improved" FID, but merely the spectrum which results from Fourier transforming the linear predicted FID. This is accomplished by selecting the Linear Prediction check box in the Linear Prediction panel and clicking the Transform button.

If you do wish to see the linear predicted FID, it is possible to do so by entering `ft('noft')`, which performs all the steps of the Fourier transform routine except the actual Fourier transformation. You can now see the real points of the FID by setting `lp=0 rp=0`, or see the imaginary points by setting `lp=0 rp=90`.

Since linear prediction involves solving a series of equations for appropriate coefficients based on the actual FID, it involves quite a number of parameters and can be somewhat tricky to optimize (if not optimized properly, or if the data are not amenable, the analysis may simply fail, just like any least-squares fit process may fail to converge).

For more complex problems, linear prediction can even be run in a iterative fashion—first extending backward, then forward, and perhaps again backward.

Why Use Linear Prediction

Raw time-domain data acquired during a pulsed NMR experiment can have two flaws:

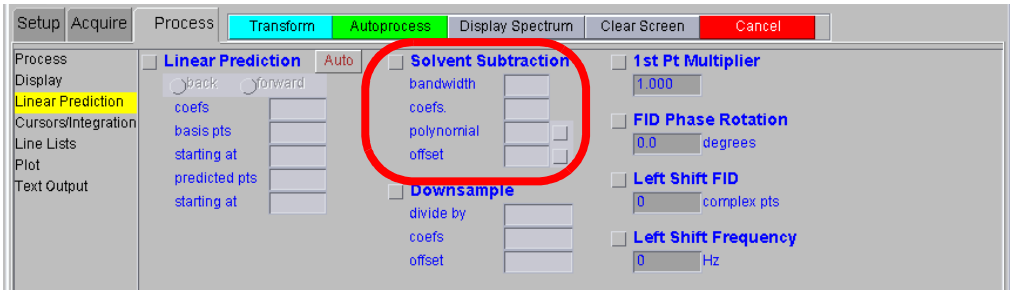
- Early points in the FID may be distorted due to a host of hardware characteristics, such as preamplifier saturation, probe ringing, and filter non-linearities. Even on a perfect spectrometer, these distortions cannot always be avoided.
- The acquisition time of each FID may have been too short to allow for full decay of the signal, leading to distortion in the Fourier transformed spectrum.

Both types of distortions can be solved using *linear prediction*. This uses the "good" part of the FID to analyze for the frequencies that are present in the signal, and then uses that information to extend the FID either in a reverse direction (to "fix" the first few "bad" points) or in a forward direction (to eliminate truncation problems). Following this process, the "new, improved" FID is then Fourier transformed in the usual way.

For more information on the algorithm implemented in the software, and on linear prediction in general, refer to H. Barkhuijsen, R. de Beer, W.M.M.J. Bovée, and D. van Ormondt, *J. Magn. Reson.*, **61**, 465-481 (1985).

Solvent Subtraction Filtering

Numerous solvent suppression pulse sequences exist that reduce the signal from a large solvent peak to a level where the desired resonances can be observed. Often, however, experimental solvent suppression does not entirely eliminate an unwanted solvent peak. Digital filtering of the data can further suppress or eliminate a solvent peak.



VnmrJ incorporates two algorithms for solvent subtraction by digital filtering:

- In the first, called lfs (low-frequency suppression), a low-pass digital filter is applied to the acquired FID. This filter severely attenuates all signals that lie outside the passband of the filter, leaving only the on-resonance solvent signal and other low-frequency signals that fall within the filter bandwidth. This filtered FID is then subtracted from the original FID to remove the solvent peak contribution. The Fourier transform of this FID gives the solvent-subtracted spectrum.
- In the second, called zfs (zero-frequency suppression), the acquired FID is also low-pass filtered, but then the filtered FID is fit with a polynomial (specified by the parameter `ssorder`), and the polynomial is subtracted from the original FID. This has the effect of removing from the FID only the signal that is exactly on-resonance. The Fourier transform of this FID produces the solvent-subtracted spectrum.

The solvent subtraction parameters `ssfilter`, `sslsfrq`, `ssntaps`, and `ssorder` control processing.

The parameters `ssfilter` and `ssorder` select the processing option as follows:

- The zfs (zero-frequency suppression) option is selected if both `ssfilter` and `ssorder` are set to a value other than "Not Used."
- The lfs (low-frequency suppression) option is selected if `ssfilter` is set to a value other than "Not Used" and `ssorder` is set to "Not Used."

- The zfs and lfs options are both turned off if `ssfilter` is set to “Not Used.”

The characteristics of the low-pass digital filter used with the lfs and zfs options can be modified by changing the parameters `ssfilter`, `sslsfrq`, and `ssntaps`:

- The value of `ssfilter` specifies the full bandwidth of the low-pass filter applied to the original FID to yield a filtered FID. Its default value is 100 Hz.
- The value of `sslsfrq` specifies the location of the center of the solvent-suppressed region of the spectrum. Setting `sslsfrq` to a non-zero value shifts the solvent-suppressed region by `sslsfrq` Hz. Setting `sslsfrq` to `'n'` (the default value) solvent suppresses a region centered about the transmitter frequency.
- The value of `ssntaps` specifies the number of taps (coefficients) used for the digital filter. The default value is 121 but the value can range from `1` to `np/4`. The more taps in a filter, the flatter the passband response and the steeper the transition from passband to stopband, giving a more rectangular filter. For the lfs (low-frequency suppression) option, the default is suitable. For the zfs (zero-frequency suppression) option, a value between 3 and 21 usually works better.

Additionally, if the zfs option is selected, the parameter `ssorder` specifies the order of the polynomial used to fit the digitally filtered FID. The order can range from 1 to 20. The default value is “Not Active.” If the lfs option is selected, `ssorder` is not used.

The quality of filtering with zfs diminishes rapidly as the solvent peak moves off the exact center of the digital filter. It may be necessary to adjust `lsfrq` or `sslsfrq` to move the solvent peak to within ± 0.2 Hz of the center of the filter to obtain optimal solvent suppression. The lfs option is less sensitive to small offsets, but typically removes or distorts peaks near to the solvent peak.

Interleave FIDs

The `ilfid` command converts a multiple FID element into a single FID by interleaving the FIDs. When invoked in an experiment of `nf` FIDs, each of `np` points, `ilfid` sorts the data into a single FID of `np*nf` points that can then be transformed. The interleaving takes the first complex point of each of the `nf` FIDs and places them in sequential order in the new FID. It then takes the second complex point from each of the `nf` FIDs and appends them sequentially to the new FID. This operation is repeated for all complex points. Although `ilfid` adjusts `np` and `nf`, it does not alter other parameters such as `sw`.

CAUTION: **Because ilfid alters the data irrevocably, it is strongly recommended that you save the FID before using ilfid.**

For further information on `ilfid`, including an example, refer to the *Command and Parameter Reference*.

Chapter 7. Displaying FIDs and Spectra

Sections in this chapter:

7.1 Displaying a FID or 1D Spectrum

Click the Display FID graphics control button to display a FID. Click the 1D Spectrum graphics control button to display a 1D spectrum.

FID Display
1D Spectrum

- “FID Display,” page 97
- “1D Spectrum Display,” page 98

FID Display

After data is acquired, a FID becomes available for displaying. Clicking the FID button on the graphics control bar displays a FID and enables interactive manipulation of the FID display.

Click to display a FID

The FID display graphics buttons change to show that multiple FIDs can be viewed.

The FID is left shifted by the number of complex data points specified in the **Left Shift FID** field on the **Linear Prediction** page.

Figure 15 shows a typical display with a FID and two vertical cursors (box mode).

The FID is also phase rotated (zero-order only) by the number of degrees specified in the **FID Phase Rotation** field on the **Linear Prediction** page.

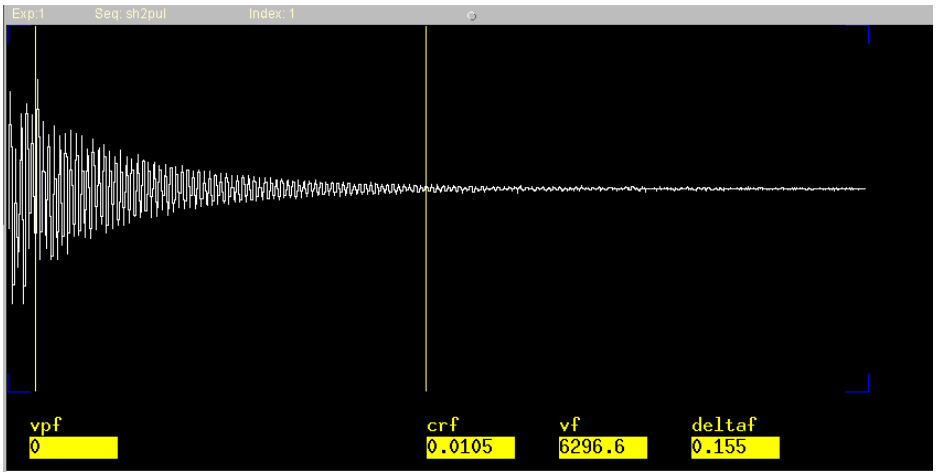


Figure 15. Interactive FID Display

1D Spectrum Display

After data is transformed, a spectrum becomes available for display and plotting.

The normal spectrum display enables interactive manipulation of a single 1D spectrum. A spectrum is displayed by clicking the 1D Spectrum graphics control button or by transforming a data set.


Click to display a spectrum

A spectrum displays in the graphics window similar to Figure 16.

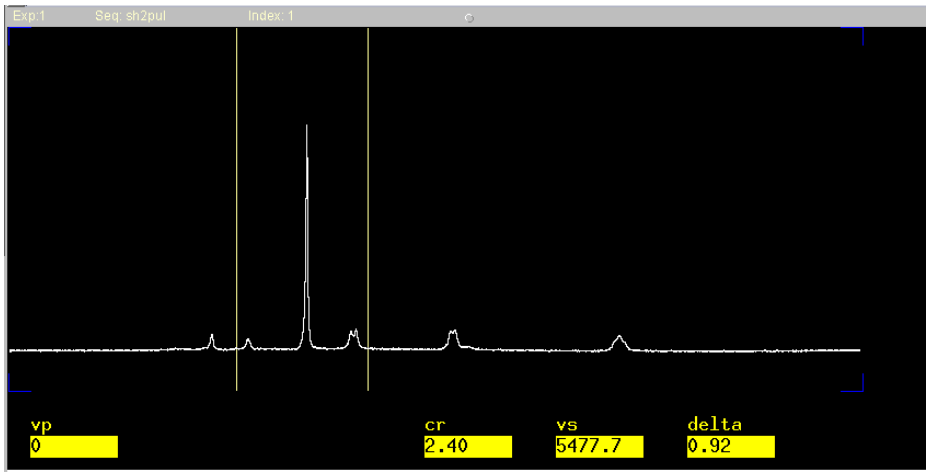


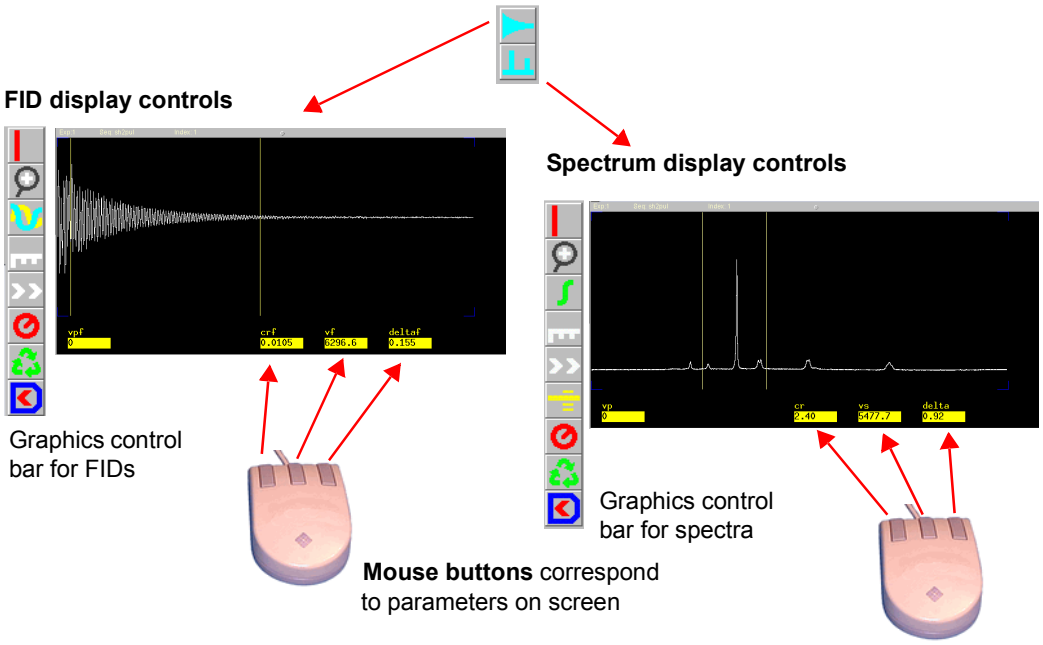
Figure 16. Interactive Spectrum Display

7.2 Display Tools

VnmrJ provides interactive tools for creating highly individualized display of NMR data.

Interactive Display Tools

Figure 17 shows the VnmrJ tools used for the interactive display of FIDs and spectra



Display page in Process panel

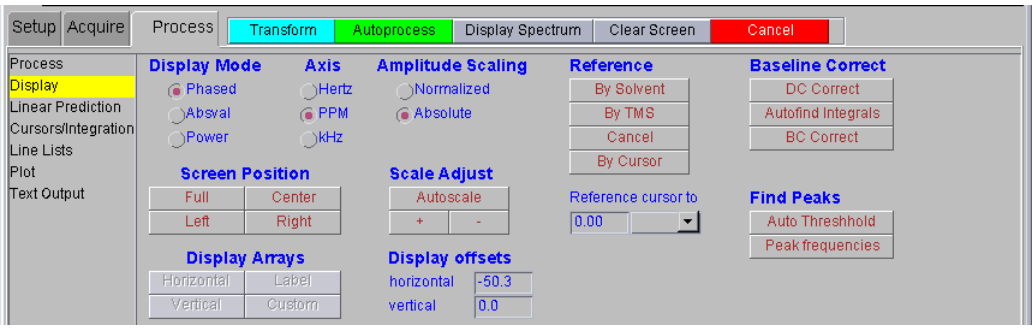


Figure 17. Display Controls

These tools are described below:

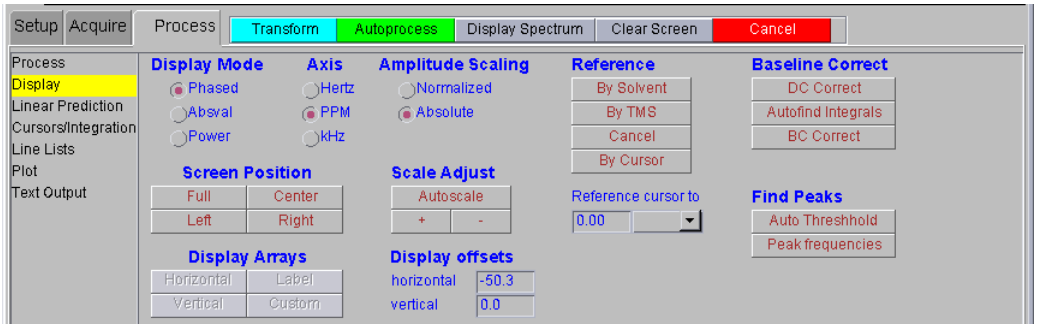
Mouse buttons	The mouse button correspond to the display parameters shown on the lower right part of the graphics window. These display parameter change as you select the different graphics control functions. Typically, the left button controls the left cursor position, the middle button controls vertical scaling, and the right button controls the right cursor or delta between the two cursors.
Graphics control buttons	The graphics control bar is arranged vertically to the left of the graphics canvas. The graphics control buttons provide controls for cursors, zooming, scales, grab & move, threshold, phasing, and refresh. Different functions appear for FID or spectrum display.
Display page	The Display page on the Process tab provides appropriate display parameters, including display mode, axis, and amplitude scaling.
Display menu	The Display menu provides tools for displaying multiple spectra, plotting, and creating insets.

A typical use of these tools might be expanding a region on a spectrum:

1. Display the spectrum -- click the spectrum icon on the graphics control bar.
2. Select the region to expand -- left click on the spectrum to place the cursor on the left boundary of the region of interest, and right click to designate the right boundary. Use the left mouse button to drag the left cursor and right button to drag the right cursor until the region you want to expand is between the cursors.
3. Expand the region -- click the magnifying glass icon on the graphics control bar.

Display Parameters

FID and spectral display is governed by parameters on the Display page.



Display Mode

The Display Mode parameters set the display mode along the directly or indirectly detected dimension.

Phased	each real point in the displayed spectrum is calculated from a linear combination of real and imaginary points comprising each respective complex data point.

Absval	(Absolute Value mode) each real point in the displayed spectrum is calculated as a square root of the sum of squares of the real and imaginary points comprising each respective complex data point.
Power	each real point in the displayed spectrum is calculated as a sum of squares of real and imaginary points comprising each respective complex data point.

Axis

The Axis parameters set the labeling of plot scales, peak frequencies, etc. Typically, FID display is in seconds and spectrum display is in PPM, Hz, or kHz.

Amplitude Scaling

The amplitude scaling, or vertical scale, parameters set the scale intensities for the display:

Normalized	the largest peak in the spectrum is automatically found, then the display is normalized to make the peak vertical scale on the plot in millimeters.
Absolute	the appearance on the display screen is used as a guide to adjust the vertical scale to produce the desired height. This mode enables comparing intensity from one experiment to another, a necessity for *all* arrayed experiments.

For vertical scaling, full scale on the screen represents full scale on the plotter. This relationship is used to adjust the vertical scale in Absval display mode, since in that case vertical scale is not the height of the largest peak. In Normalized amplitude scaling mode, this is also used when the largest peak is desired to be off-scale.

An exception to the general rule of plotting is provided by the `wysiwyg` parameter. This parameter is set in the **Utilities** -> **System settings** window, on the **Display/Plot** tab.
`Set display from plotter aspect ratio (wysiwyg)`

Checked	scales the image to the current plotter setting (wysiwyg).
Unchecked	scales the image to the full window, which is easier to view. This option scales the window but does not change the ratio of the image.

Screen Position

The screen position parameters set the horizontal position of the display on the screen and the plotter. Clicking one of the buttons updates the display:

Full	display or plot on the entire screen or page.
Center	display or plot in the center of the screen or page.
Left	display or plot in the left half of the screen or page.
Right	display or plot in the right portion of the screen or page.

Controlling Cursors and Vertical Scale

Click the mouse buttons in the graphics display window to position cursors and adjust the FID or spectral vertical scale and position.

Left cursor	Click the left mouse button to position the cursor and update the value displayed for the `crf` or `cr` parameter (`crf` for a FID or `cr` for a spectrum).
Right cursor (box)	Click the right mouse button to display and position a second cursor to the right of the original cursor. The value of the parameter `deltaf` for a FID or `delta` for a spectra changes with the position of the right cursor and is the difference in seconds between the two cursors.
Two cursors	If both cursors are displayed, the left mouse button moves both cursors simultaneously, leaving the distance between them (`deltaf` or `delta`) unchanged.
Vertical scale	Click the middle mouse button to adjust the vertical scale of the FID (`vf` parameter) or spectrum (`vs` parameter).
Vertical position	To adjust the vertical position of the FID, click and hold the middle mouse button near the left edge of the graphics display and slide the FID or spectrum up or down. The value of `vpf` or `vp` (or `vpfi` if the imaginary channel) is will change.

Display Limits

The **Screen Position** buttons (Full, Enter, Left, Right) on the **Display** page place the display and plot in the desired portion of the page.

The `wysiwyg` parameter is useful for scaling the image to a full window instead of the same size as the plot. This parameter is set in the **Utilities** -> **System settings** window, on the **Display/Plot** tab.

`Set display from plotter aspect ratio (wysiwyg)`

Checked	scales the image to the current plotter setting (wysiwyg).
Unchecked	scales the image to the full window, which is easier to view. This option scales the window but does not change the ratio of the image.

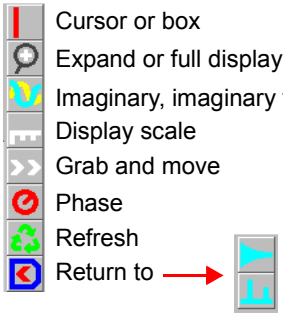
7.3 Graphics Control Buttons

Use the Graphics control buttons in combination with the mouse buttons for complete interactive control of the displayed FID or spectrum.

- "FID Display Buttons," page 103
- "1D Spectrum Display Buttons," page 103

FID Display Buttons

To the left of the display is the FID Display Menu with the following buttons (note that the labels change on some of the buttons according to the mode the program is in):



These buttons function as follows:

Cursor or Box

Box	Change to the box mode with two cursors.
Cursor	Change to the cursor mode with one cursor.

Expand or full display

Expand	Expand the area between the cursors.
Full	Display the full area.

Imaginary, imaginary with all zeros, or no imaginary

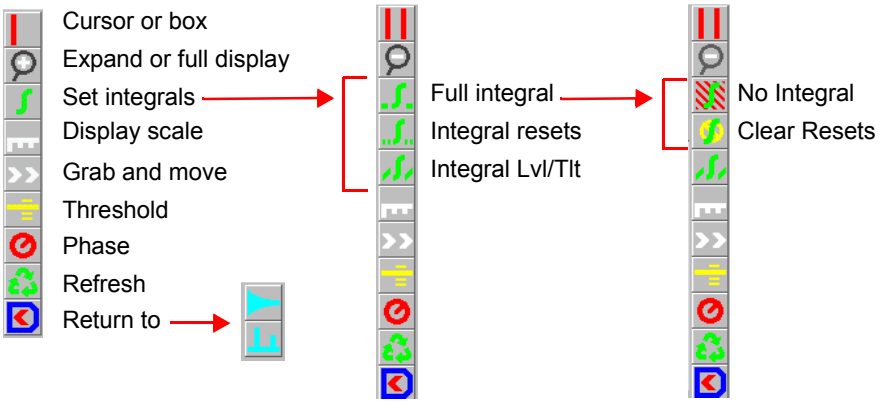
Imaginary	Display the imaginary FID.
Zero Imag	Display the imaginary FID as all zero.
No Imag	Remove the imaginary FID display.

Scale	Display a time scale under the FID in units specified on the Display page.
Grab and Move	Opens the interactive FID windowing mode. Use the left mouse button to adjust the starting time of the display and move the display left or right. Use the right mouse button to adjust the width of the display.
Phase	Opens the interactive phasing mode. Use the left mouse button for course adjustments (180°) and right button for fine adjustments (20°), where full scale corresponds to 180°. See "FID Phasing". Use the middle button to adjust the vertical scale of the display and to apply the latest phase correction to the entire FID.
Refresh	Refresh the display.
Return	Returns to the last menu.

1D Spectrum Display Buttons

Clicking the spectrum button on the graphics control bar displays a spectrum and enables interactive manipulation of the FID.

To the left of the display is the spectral display graphics control buttons (note that the labels change on some of the buttons according to the mode the program is in):



These buttons function as follows:

Cursor or Box

Box	Change to the box mode with two cursors.
Cursor	Change to the cursor mode with one cursor.

Expand or full display

Expand	Expand the area between the cursors.
Full	Display the full area.

Set integrals	The first click displays the intregal function buttons. The second click displays the No Integral and Clear Resets buttons.
Full integral	Display all integral regions.
Integral resets	Open an interactive integral reset mode. The left mouse button defines an integral reset at the current mouse position. The right mouse button removes an integral reset closest to the current mouse position. The middle mouse button adjusts the scale. The integral does not have to be displayed. However, if the integral is displayed in the “partial” mode, the normally blanked regions are displayed as dotted lines. To clear the integral reset points before beginning, click the Clear Resets button.
Integral Lvl/Tlt	Open interactive zero- and first-order baseline correction mode, see See “Interactive Zero- and First-Order Baseline Correction Mode”.
Scale	Display a time scale under the spectrum in units specified on the Display page.
Grab and Move	Opens the interactive spectral windowing mode. Use the left mouse button to adjust the starting time of the display and move the display left or right. Use the right mouse button to adjust the width of the display.
Threshold	Toggles the display of a horizontal cursor. The left mouse button positions this cursor at the mouse arrow position. The middle mouse button adjusts the scale.
Phase	Opens the interactive phasing mode. Use the left mouse button for course adjustments (180°) and right button for fine adjustments (20°), where full scale corresponds to 180°. See “Spectrum Phasing”. Use the middle button to adjust the vertical scale of the display and to apply the latest phase correction to the entire spectrum.
Refresh	Open an interactive integral reset mode, see below.
Return	Returns to the last menu.

7.4 Phasing

The Phase button starts the interactive phasing mode. Any integral and cursors that are displayed along with the spectrum are removed. The width of the update region is set by the **Spectrum updating during phasing (0-100)** field in **Utilities->System settings->Display/Plot** tab, which sets the percentage of the screen display to be updated:

- "FID Phasing," page 105
- "Spectrum Phasing," page 105

FID Phasing

The Phase button activates the interactive phasing mode:

1. Position the mouse arrow on a FID region of interest, about halfway vertically up the screen, and click the left mouse button.

   A horizontal cursor intersects at the mouse arrow and two vertical cursors are placed on either side of the mouse arrow. A small region of FID is displayed in a different color if a color display is present; only this spectral region is interactively updated.

2. Move the mouse above or below the horizontal cursor, but within the two vertical cursors. Click the left or right button to adjust the FID phase parameter `phfid`.

   Click the mouse above the horizontal cursor to increase `phfid`. Click below the horizontal cursor to decrease `phfid`. Place the mouse arrow right on the horizontal cursor and click the left button to restore the initial phase.

3. To exit the interactive phasing mode, make another selection from the menu. Select the Cursor or Box button if no other choice is desirable.

Spectrum Phasing

1. Position the mouse arrow on a spectral region of interest toward the right side of the spectrum, about halfway vertically up the screen, and click the left mouse button.

   A horizontal cursor will intersect at the mouse arrow. Two vertical cursors will be placed on either side of the mouse arrow. A small region of the spectrum will be displayed in a different color, if a color display is present, and only this spectral region will be interactively updated.

2. Move the mouse above or below the horizontal cursor, but within the two vertical cursors. Click the left or right button to adjust the zero-order or frequency-independent phase parameter `rp`.

   - Click above the horizontal cursor to increase `rp` (cause a clockwise rotation of the peaks).
   - Click below the horizontal cursor to decrease `rp` (and cause a counter-clockwise rotation).
   - Place the arrow on the horizontal cursor and click the left button to restore the initial phase.

   The left and right button of the mouse differ only in their sensitivity. Full scale (top to bottom of the screen) corresponds to approximately 180° for the left button, and 20° for the right button, and hence you can consider the left button the "coarse" adjust and the right button the "fine" adjust.

3. Move the mouse arrow to another region of the spectrum, near the left edge of the display, outside the vertical cursors, and click the left mouse button again.

The frequency-independent phase-correction made so far is first applied to the entire spectrum. A new horizontal cursor is displayed at the mouse arrow, and two new vertical cursors are displayed on either side of the mouse arrows. The mouse now controls the first-order or frequency-dependent phase parameter `lp`.

4. Click the left or right button above or below the horizontal cursor to increase or decrease `lp`, and change `rp` so that the phase at the center of the previous region bracketed by the vertical cursors is held constant.

   This process eliminates or substantially reduces the necessity to iteratively adjust the two parameters `rp` and `lp`. As with the zero-order correction, the left button acts as a “coarse” adjust, and the right button as a “fine.”

   Define a new update region by clicking the mouse outside the two vertical cursors.

   Subsequent first-order phase changes causes the zero-order phase to be adjusted such that the phase angle at the center of the previous region bracketed by the vertical cursors remains constant. If you wish to return to the zero-order phase correction, click the Phase button again.

   Adjust the vertical scale and apply the latest phase correction by clicking the middle mouse button at the top of a peak that is on scale. This leaves the vertical scale unaffected but recalculates the phase of the entire spectrum. Clicking the center button above or below the peak raises or lowers the vertical scale.

5. Exit the interactive phasing mode by clicking another graphics control button.

7.5 Line Tools

VnmrJ provides tools for finding the nearest line, measuring line resolution, and displaying line lists.

- “Find Nearest Line and Line Resolution,” page 106
- “Display Line List,” page 106

Find Nearest Line and Line Resolution

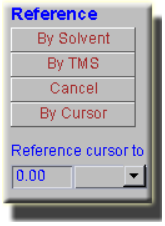
1. Place a cursor near the line of interest.
2. Select the **Process** page and click the **Find nearest line** button. The cursor moves to the nearest line and displays its height and frequency (in Hz and ppm) in the message window.
3. Click **Display linewidth** to display the resolution of a line, as well as the limiting digital resolution of the spectrum.The resolution is determined by a width at half-height algorithm and not by least-squares.

Display Line List

1. Click the Threshold graphics control button and use the middle mouse button to vertically position the yellow threshold line.
2. Select the **Line List** page and click the **Display Line List** button. This process displays line frequencies and intensities that are above a threshold.

7.6 Spectral Referencing

Frequency referencing is set on the **Display** page.



By Solvent	Reference the spectrum to a selected solvent line. Use the **Find nearest line** button on the **Process** page to place the cursor.
By TMS	Reference the spectrum to a TMS line. In the case of other signals (e.g., from silicon grease) immediately to the left of the TMS line (even if they are higher than the reference line), `tmsref` tries avoiding those signals by taking the line furthest to the right in that area, as long as it is at least 10% of the main Si-CH_3 signal. Large signals within 0.6 ppm for ^{1}H (or 6 ppm for ^{13}C) to the right of TMS might lead to misreferencing.
Cancel	Clears the reference line by removing any spectral referencing present, and turns off referencing.
By Cursor	References the spectrum based on the current cursor position. If you want to reference the spectrum based on a line position in the spectrum, first use the **Find nearest line** button on the **Process** page, then click **By Cursor**.

Terms used in spectral referencing.:

Reference line	The distance, in Hz, of the reference line from the right edge of the spectral window. This line is the spectral position used to set the referencing. It can be the signal of a frequency standard (such as TMS), or any line (such as a solvent signal) with a known chemical shift (in ppm), or a position in the spectrum where you expect such a line to appear.
Reference position	The difference between the reference line and the reference frequency (zero position of the scale), in Hz. If you reference a spectrum using the signal of a frequency standard, such as TMS, then reference position is 0. The distance of the reference frequency from the right edge of the spectrum is *reference line - reference position*.
Spectrometer frequency	The absolute frequency, in MHz, of the center of the spectrum (the transmitter position). In order to see the accurate value of the spectrometer frequency (`sfrq` parameter), you should use the `spcfrq` command.
Reference frequency	The frequency, in MHz, of the frequency standard, i.e., the zero position of the frequency scale, *and* the divider (unit) for the calculation of ppm scales.

The By Solvent and By TMS buttons assume that the system is locked (and that the lock solvent is defined in `/vnmr/solvents`). If you are working without lock and still want to use these buttons, you must ensure that the field offset has been adjusted so that the lock frequency is on resonance with a sample of similar susceptibility. To ensure that the field offset is adjusted, do the following procedure:

1. Insert a sample with deuterated solvent.

2. Adjust `z0` (or `lkof`) in `acqi` so that the lock frequency is on resonance.
3. Switch off the lock.
4. Insert the nondeuterated sample.

The accuracy of the solvent and TMS referencing buttons is mostly limited by the accuracy of the chemical shift of the lock resonance line, which may depend on the concentration and the chemical properties (acidity/basicity) of the components in the sample. But they should normally be accurate enough to find an actual reference line close to its predicted position.

Estimating the position if the reference frequency in spectra from unlocked samples, provided the spectrometer is first locked on a sample with similar susceptibility, then the lock is disengaged and the field offset adjusted such that the lock signal is on-resonance. Now, you can acquire spectra without lock and calculate their (estimated) referencing using `setref`, provided the `solvent` parameter is set to the solvent that was last locked on.

7.7 Stacked FID Display

FIDs can be displayed (and plotted) in a stacked or whitewashed view.

The **Display** menu provides tools for displaying one or more FIDs as a *stacked display*. Each FID is offset horizontally and vertically from the previous FID. The position of the first FID is set using the graphic control buttons and the mouse buttons. Each subsequent FID is positioned relative to the preceding FID by the menu choices.

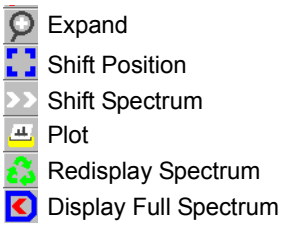
Use the **Plot** page to setup plots of one or more FIDs as a stacked display.

7.8 Inset Display

An inset display is a part of a spectrum superimposed on the display of the entire spectrum. The vertical position of the inset spectrum is shifted up about one-quarter of the height of the whole display window. The old spectrum remains on the screen, but the parameters shown at the bottom are now relevant to the inset display. If present, the integral trace and the scale are duplicated with the `inset` spectrum.

1. Place cursors around the part of the spectrum you want to display as an inset.
2. Open the **Cursors/Integration** page and click **Inset spectrum**. Additional inset display tools are available under the Display menu.

   The inset is placed above the selected part of the spectrum.



3. Shift, expand, or contract the inset using the mouse buttons and graphics control buttons.

The graphics control buttons function as follows:

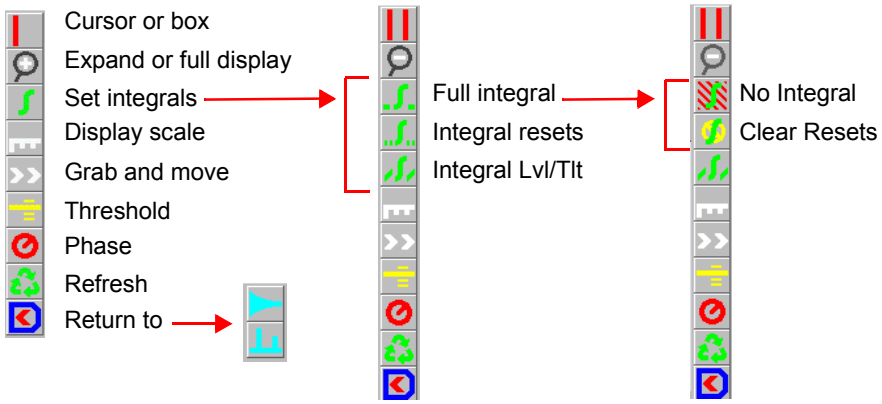
Expand	Expand the area between the cursors.
Shift Position	Shift the inset spectrum left or right. Adjust the start of the display with he left mouse button and the width on the screen with the right button.
Shift Spectrum	Interactively adjust the starting frequency with the left mouse button and width of frequencies displayed in the spectrum with the right button.

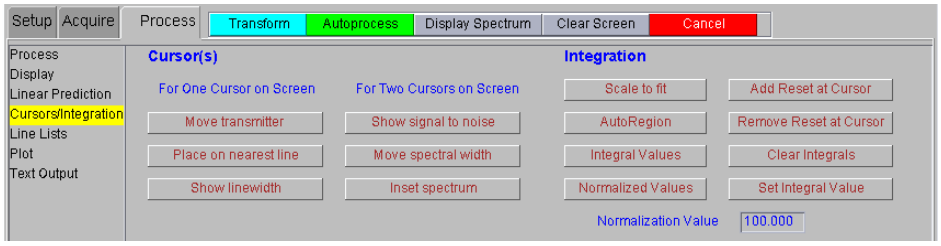
Plot	Plot the spectrum, and if displayed, the integral and scale
Redisplay Spectrum	Return to graphics control buttons for spectral display
Display Full Spectrum	Return to graphics control buttons and erase inset.

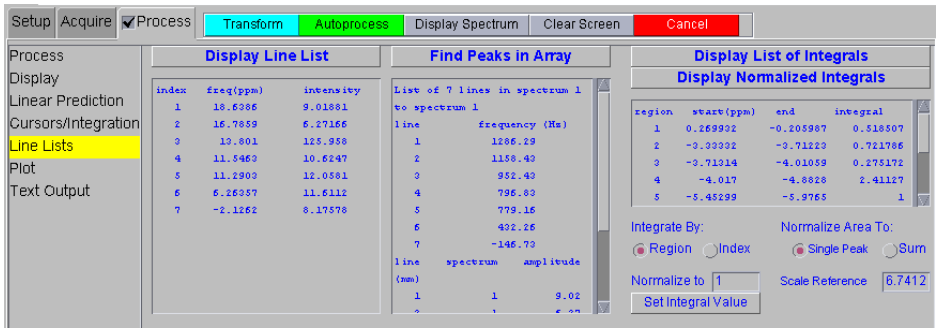
Use the mouse left and right mouse buttons to place cursors and the middle button to adjust vertical scale of the spectrum or integral.

7.9 Integration

This section describes methods and tools for displaying and plotting integrals.







Interactive Zero- and First-Order Baseline Correction Mode

The Integral Lvl/Tlt button activates interactive zero and first order baseline correction mode. The zero order correction is represented by the `lvl` parameter; the first order correction is represented by the `tlt` parameter. If no integral is displayed when the Integral Lvl/Tlt button is activated, the integral is automatically displayed.

1. Left click on an integral region of interest, about halfway vertically up the screen.

   A horizontal cursor intersects at the mouse arrow. Two vertical cursors are placed on either side of the mouse arrow.

2. Right or left click above or below the horizontal cursor, but within the two vertical cursors, to adjust the zero-order baseline correction parameter `lvl`.

   - Clicking the above the horizontal cursor increases `lvl`.
   - Clicking below the horizontal cursor decreases `lvl`.
   - Clicking on the horizontal cursor restores the initial baseline correction value.

3. Left click on another region of the spectrum, outside the vertical cursors.

   A new horizontal cursor displays at the mouse arrow, two new vertical cursors display on either side of the mouse arrow, and a single vertical cursor displays in the middle of the region where `lvl` was being updated. The mouse now controls the first-order baseline correction parameter `tlt`.

4. Right or left click above or below the horizontal cursor to increase or decrease `tlt`, and change `lvl` so that the total drift correction at the single vertical cursor in the middle of the previous region is held constant.

   This process eliminates or substantially reduces the necessity to iteratively adjust the two parameters `lvl` and `tlt`. As with the zero-order correction, placing the mouse arrow right on the horizontal cursor and clicking the mouse button will restore the initial baseline correction values.

   Each time the mouse is clicked outside the two vertical cursors, new vertical and horizontal cursors display.

   The left and right mouse buttons both adjust the baseline correction parameters and differ only in their sensitivity. The left button causes changes a factor of eight times larger than the right button, and hence you can consider the left button the “coarse” adjust and the right button the “fine” adjust. The overall sensitivity of these adjustments can also be controlled by the parameter `lvltlt`. This parameter is a multiplier, with a default value of 1.0, for the size of the changes. To make larger changes, make `lvltlt` larger than 1.0. To have finer control, set `lvltlt` to be between 0.0 and 1.0.

   The middle mouse button adjusts the integral scale (parameter `is`) or the integral offset (parameter `io`), exactly as whenever an integral is displayed.

5. Exit the interactive baseline correction mode by clicking on another graphics control button.

Displaying Integrals Step-by-Step

The following methods should give you an opportunity to compare procedures. Before starting each procedure, be sure to obtain a typical spectrum by entering:

1. Click the magnifying glass button on the **Locator**.
2. Select **Sort by Directory**.
3. Drag **fid1d** to the graphics window.
4. **Transform** the data if necessary.

Fully Automated Method

1. Enter **`intmod='partial' region.`**

   The integral display mode is changed so that only every other integral region is displayed, and the spectrum is automatically broken into integral regions.

2. (Optional) Enter **`bc.`**

   A spline-fit baseline correction is performed to produce the flattest possible baseline.

3. Enter **`isadj`**.

   The largest integral is adjusted to a reasonable size.

4. Enter **`dli`**.

   The text window displays a list of integral intensities.

Manual Method

1. Enter **`cz`**.

   Any currently defined integral reset points are cleared.

2. Enter **`intmod='partial'`**.

   The integral display mode is changed so that only every other integral region is displayed.

3. Click the Set Integrals icon; then click the Integrals rests icon.

4. Click the **left button** slightly to the left of the left-most group of peaks.

   This establishes the end of the first (from the left end) section of baseline. You can position the mouse cursor anywhere vertically that seems most comfortable.

5. Click the **left button** slightly to the right of the left-most group of peaks.

   This establishes the end of the first section of peaks.

6. Repeat steps 4 and 5 for each group of peaks across the spectrum. The reset points must alternately separate baseline and peaks. If two peaks are adjacent to each other but you want a reset between them, click the button `twice` at the same place. This establishes a “baseline” region of zero length.

   Note that you can also `add` additional resets in this way to resets that were established automatically by the `region` command.

7. Enter **`vp=12`**.

   The spectrum moves up to allow space for a numerical display of integrals.

8. Click the **center** mouse button above the right end of any displayed integral.

   This adjusts the integral vertical scale

9. Enter **`ins=x`**, where *x* is the value you wish to assign to the sum of the integrals.

   The value entered affects only printed output, not the trace of the integral.

10. Enter **`dpirn`**.

    The text window displays a list of integral amplitudes. The sum of the integrals is normalized to the value of the parameter `ins`.

Baseline Correction

Almost all of the operations performed on spectra assume a “good” baseline. Line lists, integrations, resolution measurements, 2D volume integrations, etc., all measure intensities from “zero” and do not perform any baseline adjustments. If the baseline in your spectrum is not “good,” you should first perform a baseline correction operation before performing further data reduction. Two types of baseline correction are provided: linear and non-linear.

The `dc` command turns on a linear baseline correction, using the beginning and end of the displayed spectrum to define a straight line to be used for baseline correction. The result is to calculate a zero-order baseline correction parameter `lvl` and a first-order baseline correction parameter `tlt`. The `cdc` command turns off this correction. The results of the `dc` or `cdc` command is stored in the `dcg` parameter, which can be queried (`dcg?`) to determine whether drift correction is active. If active, `dcg=''`; if inactive, `dcg='cdc'`.

The `bc` command turns on 1D and 2D baseline correction. The 1D baseline correction uses spline or second to twentieth order polynomial fitting of predefined baseline regions. `bc` defines every other integral, that is, those integrals that disappear when `intmod='partial'` as baseline and attempts to correct these points to zero. A variety of parameters can be used to control the effect of the `bc` command.

For more information about the `bc` command, refer to the entry for `bc` in the *Command and Parameter Reference*.

Integral Reset Points

The `z` command (or the equivalent function key) resets the integral to zero at the point marked by the displayed cursor. `z(reset1,reset2,...)` allows the input of the reset points as part of the command, instead of using the position of the cursor. Reset points do not have to be entered in order. The resets are stored as frequencies and will not change if the parameter `fn` is changed. The command `cz` removes all such integral resets. `cz(reset1,reset2,...)` clears specific integral resets.

For a list of integrals, the `liamp` parameter stores the integral amplitudes at the integral resets points and the `lifrq` parameter stores the frequencies of integral reset points. To display the values of `liamp`, enter `display('liamp')`. Frequencies are stored in Hz and are not adjusted by the reference parameters `rfl` and `rfp`.

Integral Regions

The `region` command divides a spectrum up into regions containing peaks. A variety of parameters can be used to control the effect of the `region` command; see the *Command and Parameter Reference* for details.

Integral Display and Plotting

Display and plotting of the integral trace is independent of the values of the integrals. The height of the trace is controlled by the parameter `is` and can be interactively adjusted with the `ds` command. Also, the macro `isadj(height)` adjusts the integral height so that largest integral fits the paper or is `height` mm tall if an argument is provided, for example, `isadj(100)`.

The command `dli` displays a list of integral values at the integral reset points. The frequency units of the reset points are defined by the parameter `axis`. The reset points are stored as Hz and are not referenced to `rfl` and `rfp`. The amplitudes are stored as actual values; they are not scaled. The integral values are scaled by the parameters `ins` and

insref and the Fourier number. Typically, ins is set to the number of nuclei in a given region. For example, if a region represented a single methyl group, the following procedure would scale the integral values of that region:

1. Set **ins=3**.
2. Set **insref** to the Fourier-number-scaled-values of that integral.
3. Enter **dli**. The integral value of that region is displayed as 3 and all other integral values are accordingly scaled.

Integral value scaling can be interactively set with the ds command. The setint macro can also be used to adjust integral value scaling. setint sets the value of an integral and is used in conjunction with the command dli to scale integral values.Normalized integral values can also be selected. In this case, ins represents the total number of nuclei. The individual integral values will be scaled so that their sum is equal to ins. The normalized mode may be selected by setting insref to "not used." The integral is scaled by ins and insref.

Two commands are closely related to dli:

- nli is equivalent to dli except that no screen display is produced.
- dlni normalizes the values from dli using the integral normalization scale parameter ins and then displays the list.

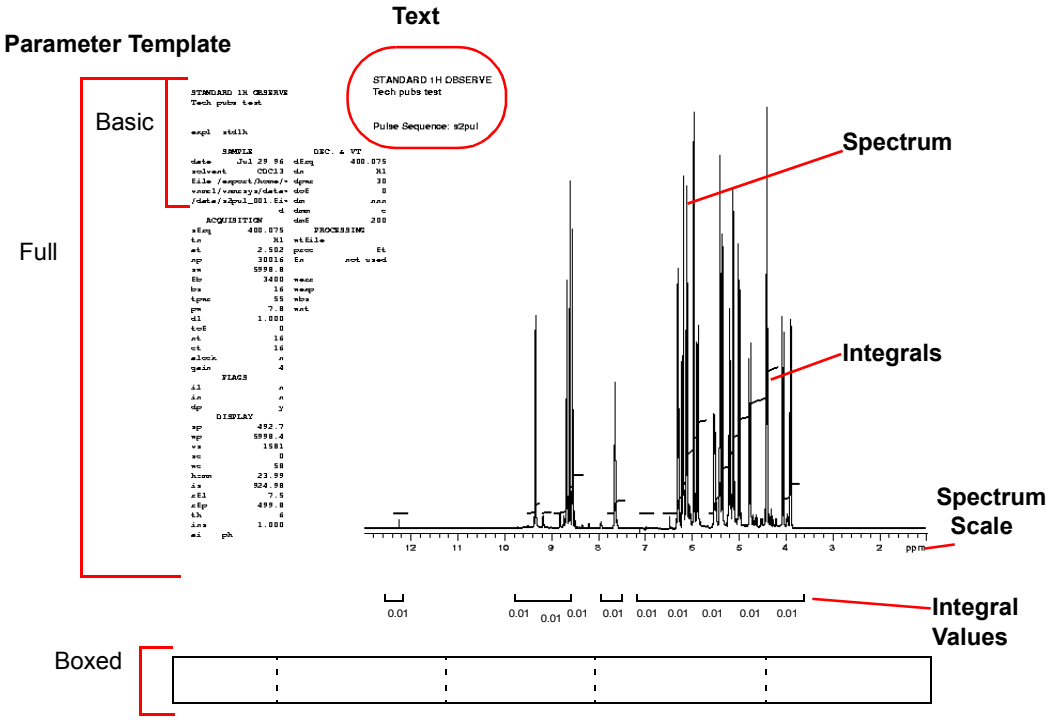
The dpir command displays numerical integral values below the appropriate spectral regions, using the integral blanking mode in which only every other integral is plotted. The command dpirn shows the normalized integral values in an analogous fashion.

The pir command plots digital integral values below the spectrum, using the integral blanking mode in which only every other integral is plotted. The command pirn plots the normalized integral values in an analogous fashion.

Chapter 8. Plotting and Printing

Sections in this chapter:

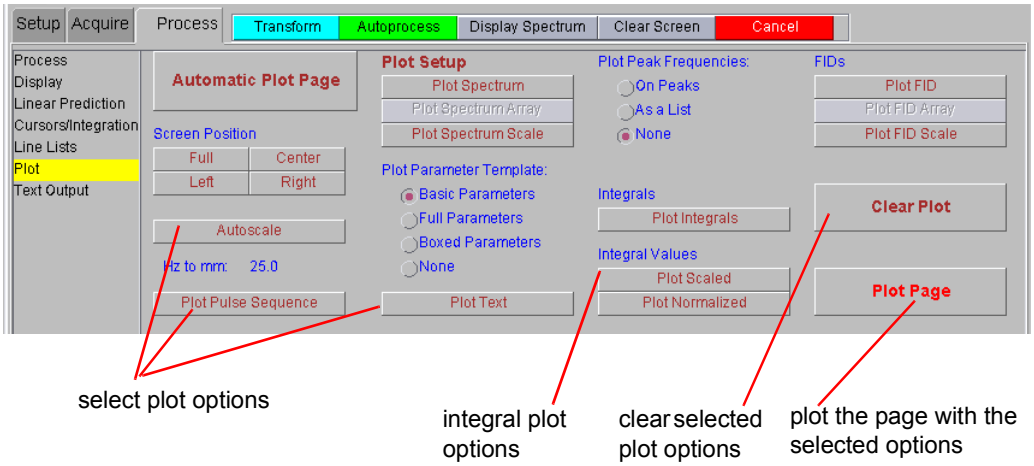
Plotting and printing of data are highly individualized activities. Each user has their own ideas about proper formats, necessary expansions, etc.



8.1 Plotting

Plotting is based around the concept of a plot file. Items selected on the Plot page are added to a temporary plot file and the Plot Page button submits the plot file to the plotter. The Clear Plot button removes the plot file

After the spectrum or FID is displayed, you can set up and submit a plot using the selections on the **Plot** page under the **Process** tab.



To Plot:	*Select:*	*Click:*
Spectrum and scale		Automatic Plot Page
Pulse sequence		Plot Pulse Sequence
FID		Plot FID, Plot Page
FID and scale		Plot FID, Plot FID Scale, Plot Page
Spectrum		Plot Spectrum, Plot Page, Clear Plot
Spectrum and scale		Plot Spectrum, Plot Spectrum Scale, Plot Page
Spectrum, scale, and text		Plot Text, Plot Spectrum, Plot Spectrum Scale, Plot Page
Spectrum, scale, and parameters	Parameter Template option	Plot Spectrum, Plot Spectrum Scale, Plot Page
Spectrum, scale, and peak frequencies	Peak Frequencies option	Plot Spectrum, Plot Spectrum Scale, Plot Page
Spectrum, scale, and integrals		Plot Spectrum, Plot Spectrum Scale, Plot Integrals, Plot Page
Spectrum, scale, and integrals, integral values		Plot Spectrum, Plot Spectrum Scale, Plot Integrals, Plot Page
Parameters only	Parameter Template option	Plot Page
Text only		Plot Text, Plot Page
Peak frequencies only	Peak Frequencies option	Plot Page
Integrals only		Plot Integrals, Plot Page
Scaled integral values only		Plot Scaled, Plot Page
Normalized integral values only		Plot Normalized, Plot Page

8.2 Plot Designer

Plot Designer allows you to see and design a plot before you print it, and provides templates, drawing tools, and a text editor that give you the capability of positioning spectra, parameters, axes, and other plot output on a page. Plot Designer gives you the following capabilities:

- Interactively compose a plot and interactively fine-tune the layout on the screen without having to make sample plots.
- Plot layouts and templates can be saved and used again.
- Label spectra with text in various fonts and draw lines, boxes, and arrows.
- Plots can be exported for further annotation and incorporation into reports and publications.

This section contains the following:

System Requirements

Plot Designer is a Java-based application.You must have Solaris 2.6 or later installed in order to use Plot Designer. The Java Runtime Environment (JRE) for Solaris from Sun Microsystems provides an environment in which you can run Java applications. Plot Designer requires at least JRE 1.1.6. You can download the latest version of JRE for Solaris from the Sun Microsystems Web site at http://www.sun.com/solaris/jre/index.html.

Starting Plot Designer

Start the Plot Designer program from the Display menu: **Display -> Create a Plot Design**.

The Plot Designer window, shown in Figure 18, opens.

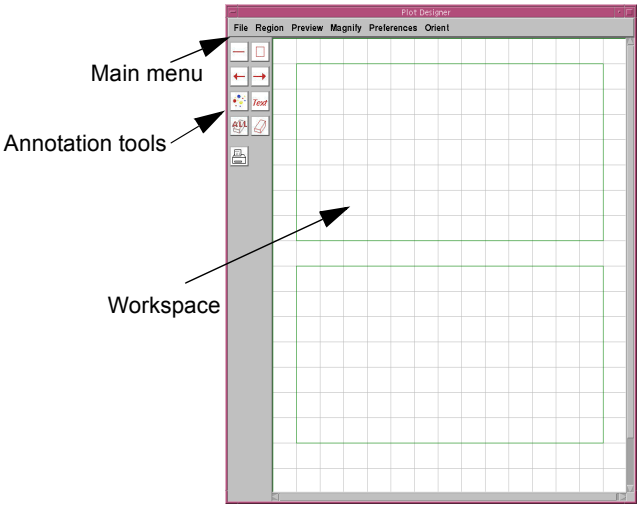
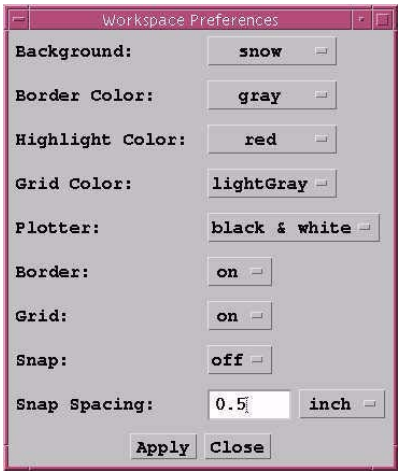


Figure 18. Plot Designer Window

Customizing the Plot Designer Window



You can easily change the size and appearance of the Plot Designer window by doing the following procedure:

1. Click on **Preferences** in the main menu, then **Set Up** to open the Workspace Preferences panel.
2. To change an aspect of, or property in, the Plot Designer window, click on its corresponding button to open a pull-down menu.

   See Table 6 for a description of each control.

   Figure 19 is an example of the window without visible region borders and without a grid.
3. After you have entered all of your preferences, click **Apply** to execute the changes.
4. Click **Close** to exit the window.

Customizing a Plot

You can add simple graphics and text to a plot and change its size and appearance by using the tools listed in Table 7. To use a drawing tool, press and hold down the left mouse button and drag the cursor in the workspace.

Using Templates

You can create your own templates. After you have created a design, do the following procedure to save your design as a template:

Table 6. Workspace Preference Controls

Control	*Function*
Background	Changes the background color of the window.
Border Color	Changes the color of the border surrounding the workspace.
Highlight Color	When you double-click on an object, its color changes to indicate that it is highlighted. This option controls the highlight color.
Grid Color	Changes the color of the grid.
Plotter	Allows you to choose a black and white or color plotter.
Border	Shows (**on**) and hides (**off**) region borders.
Grid	Shows (**on**) and hides (**off**) grid in the workspace.
Snap	The grid has magnetic properties. When snap is turned **on**, the path of an object (the center of its border) automatically snaps to the grid whenever you draw or move the object or change its size or shape. Turning **off** Snap demagnetizes the grid.
Snap Spacing	Controls the amount of space on the grid to which an object snaps. Spacing can be in inches, centimeters, or points.

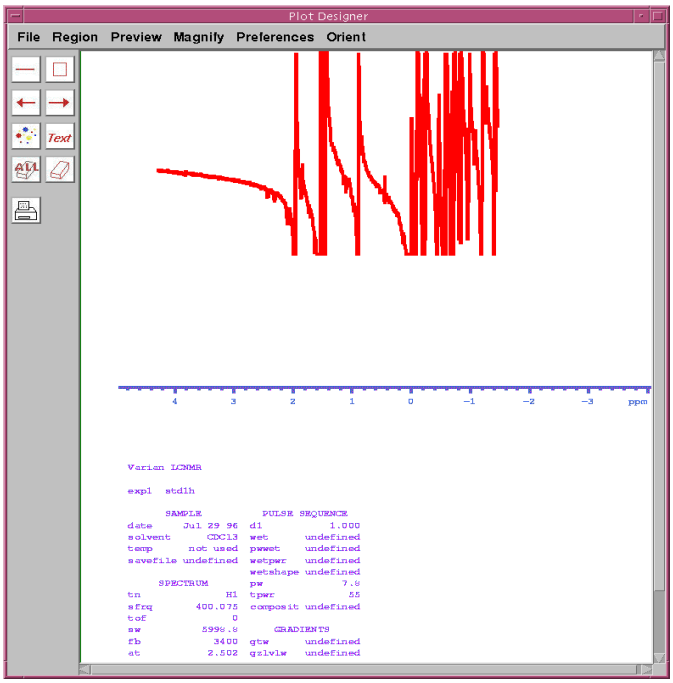


Figure 19. Window with Data and Without Borders and a Grid

1. Click **File-Templates** to open the Plot Templates window.
2. Enter a name in the **Template** field. If you want the file to be the default template, click the box next to **Use this template as default**. After you select a file as a template, the next time that you start Plot Designer, it will automatically open with the template.
3. Click **Save** to store the template in `$vnmruser/templates/plot` directory.

Table 7. Plot Designer Tools

	**Line Drawing**	Draws a line.
	**Box**	Draws a box.
	**Arrows**	Draws an arrow; places the arrowhead at the point in which you START to draw the arrow.
		Draws an arrow; places the arrowhead at the point in which you END drawing the arrow.
	**Item Preferences**	Sets the color and size of lines and fonts. To edit an object, highlight it by double-clicking on it. For a description of its properties, see page 123. You can also open this tool by clicking on **Region-Preferences**.
	**Text Input**	Allows you to add text into your design. Several options allow you to control the size and appearance of the text. To use this tool, see "Adding Text," page 124.
	**Erasers**	The **ALL** eraser removes all objects. You can also remove selected objects by using the **Region-Delete All** option described on page 123
		The eraser tool removes only selected objects.
	**Print**	Prints a file.

If you try to save a template with the same name as an already existing template, a warning notifying you that the file will be overwritten appears. If you do not want the file replaced, click **Cancel**.

4. Quit the Plot Templates window by clicking on **Close**.

Using Saved Templates

After you have created templates, you can plot a page with a specific template by typing the `jplot` command and the template name. For example, entering `jplot('t1')` starts a plot with the t1 template automatically loaded.

If you opened Plot Designer with the `jdesign` macro, the workspace will either be empty or contain the design that was being worked on the previous time Plot Designer was used. Do the following procedure to load a template file:

1. Click on **File** in the main menu, then **Templates** to open the Plot Templates window.
2. Highlight (select) a template by either DOUBLE-CLICKING on a file in the list in the upper region of the window or by entering the file name in the **Template** field.

If you want the file to be the default template, click on the field **Use this template as default**.

3. To insert the template into the Plot Designer window, click on **Open**.
4. Click **Close** to exit the window.

Removing Templates

To remove a template from the list in the Plot Templates window, click on **Delete**. A warning appears notifying you that the template will be deleted. Click **Cancel** if you do not want to delete the template.

Importing a Plot

To import a plot from the VnmrJ graphics window onto the Plot Designer workspace, you must first create a region. Regions are smaller workspaces in which you can customize a plot. Create a region by doing the following procedure:

1. Click on **Region** in the main menu, then **New** to create a region on the workspace.
2. Move the cursor into the workspace. The cursor becomes a cross-hair. Next, move the cross-hair to the approximate location of a corner of the region you want to create.
3. Press and hold down the left mouse button and drag the cross-hair to the approximate opposite corner of the new region, and then release the mouse button. The rectangle you created is a region.
4. If a region is not already selected, click the middle mouse button inside it, then click **Region-Edit** to open the Region Editor window shown in Figure 20. Region Editor is a text editor in which you can enter commands to change an imported plot.

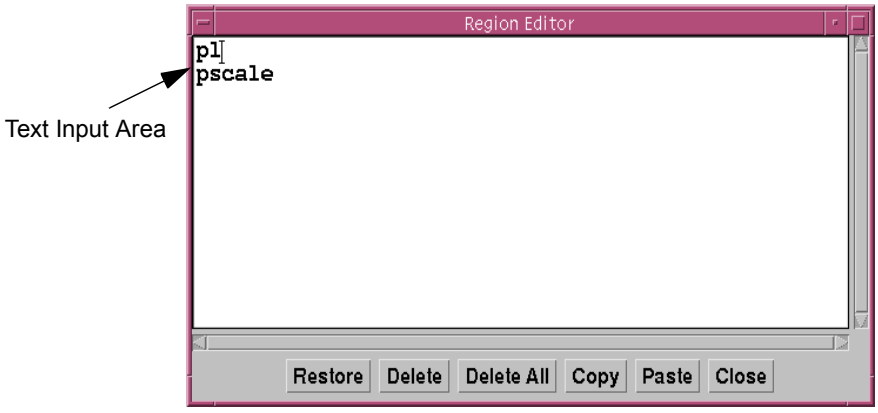


Figure 20. Region Editor Window

5. Enter a plot command (`pl`, `pscale`, `ppa`, `pltext`) in the command line.

   Plot commands are stored in the `/$vnmruser/templates/plot/menu` file or `/$vnmrsystem/user_templates/plot/menu` file. You can edit both of these files to add or delete commands. In the `menu` file, the command is indicated by two lines:

   - The first line is the label of the command that appears in the plot menu window.
   - The second line is the command itself.

In Figure 21, the label `pl` identifies the command line `pl pscale`. The label `PAP` identifies the `pap` command.

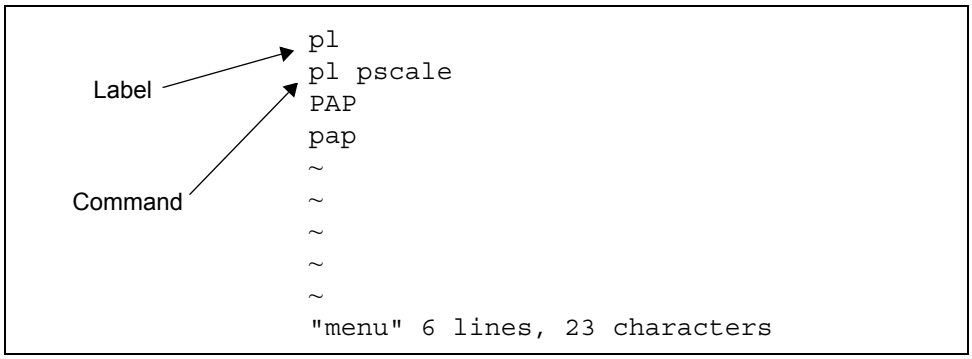


Figure 21. `menu` File

6. Click **Preview-Selected** to import the plot into the region. Click **Preview-All** to import plots into multiple regions.

You can also import a plot into a region by doing the following procedure:

1. Draw a region.
2. Click on **Region-Edit** to open the Region Editor.
3. Press the right mouse button to open the plot menu window.
4. Choose a selection to import into the Region Editor.

Editing a Plot

To edit a plot, do the following procedure:

1. Select a region.
2. Click **Region-Edit** to open the Region Editor window.
3. Enter a command (such as `pl` or `pscale`) in the text input area. Use the buttons listed in Table 8 to edit text.

Table 8. Region Editor Buttons

Button	*Function*
Restore	Applies the original template to a region. If you opened a template and made changes to it, you can restore it to its original design by using this button.
Delete	Removes text. This option is not similar to **Copy**. Deleted text is not stored in a buffer; do not use **Delete** to cut and paste text.
Delete all	Clears all text from the input area.
Copy	Duplicates text.
Paste	Inserts copied text in the input area.

4. Exit the Region Editor by clicking **Close**.

Deleting a Region

To delete a region from the workspace, highlight the region, then click **Region-Delete**. Click **Region-Delete All** to remove all regions.

Note: Regions removed with **Delete All** are not stored in a buffer and cannot be restored to the workspace.

Restoring a Deleted Region

To restore a *single* region deleted from the workspace, click **Region-Undelete**. Regions removed with **Delete All** cannot be restored with **Undelete**.

Clearing the Workspace

To *permanently* remove all regions from the workspace, click **Delete All**. Remember, if you remove all regions, you cannot restore them with **Undelete**.

Customizing Objects in a Region

You can change the size and color of objects in a region with the Item Preferences window, shown in Figure 22. Click on **Region-Preferences** to open this window. You can also open the window by clicking on the Item Preferences tool  , described on page 120.

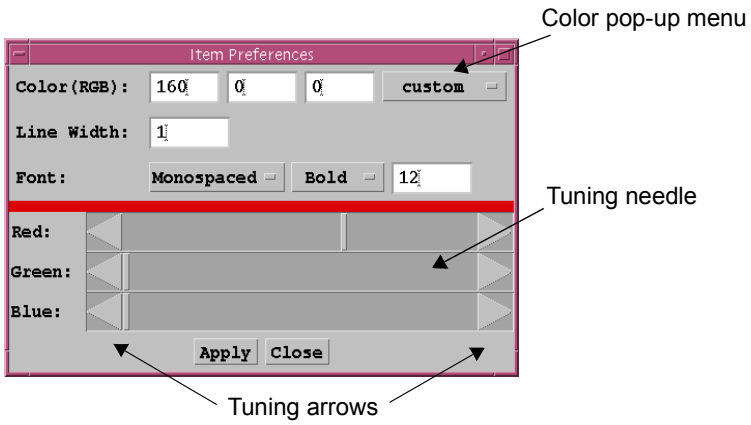


Figure 22. Item Preferences Window

Changing Line Width

Change the width of a line by doing the following procedure:

1. Highlight the line or region by double-clicking on it.
2. Enter a new width in the **Line Width** field.
3. Click **Apply** to change the line.
4. Click anywhere in the workspace to deselect the line.

Changing Fonts

Plot Designer has three font families: SansSerif, `Monospaced`, and Serif. Fonts can be Plain, **Bold**, or *Italic*. To change the family, style, and size of a font, do the following procedure:

1. Highlight the text or region.
2. Click on the Item Preferences tool  to open the Item Preferences window.
3. Choose a family, style, and enter a size in the **Font** field.
4. Click **Apply** to change the text.

Changing Line Color

You can change the color of a line by doing the following procedure:

1. Highlight the line or region.
2. In the Item Preferences window, click on the color button 

to open a pop-up menu showing a range of colors.
3. Move either the tuning needle left or right to change a color. You can also change a color by clicking on the left or right arrows in the **Red**, **Green**, and **Blue** fields; the values in the **Color(RGB)** field automatically change as you move a needle.
4. When you are satisfied with a color, click **Apply**.
5. Place the cursor anywhere in the workspace and click once to see the color change.

Adding Text

To add text into your design, do the following procedure:

1. Click on the text input tool 

to open the text input window.
2. Type text in the field at the top of the window.

   You can customize your text by clicking on the desired options and entering a font size in the indicated field.
3. Click on **Put** and drag the cursor into the workspace, then click once to paste in the text.

To copy text that is already on the workspace and paste it in different font styles, do the following procedure:

1. Highlight the text.
2. Open the Text Input window shown in Figure 23 by clicking the text input tool 

.

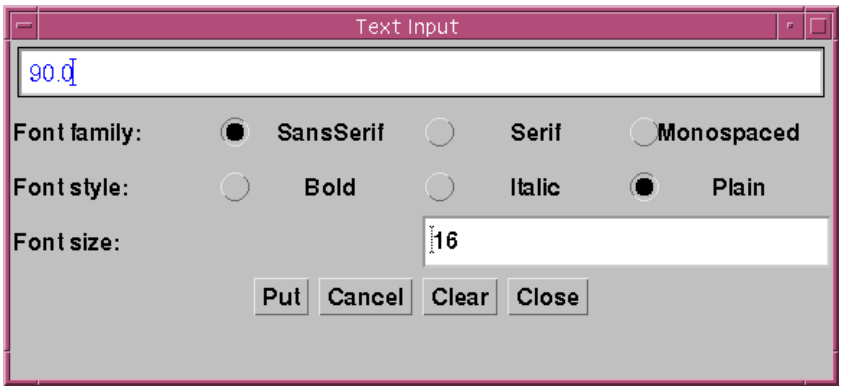


Figure 23. Text Input Window

3. Select a **Font family** and **Font style**, and enter a **Font size**.
4. Click **Put** to paste the text in the workspace.

Changing Font Color

To change the color of fonts, repeat the procedure given in "Changing Line Color."

Moving Objects and Changing Object Size

You can move an object by double-clicking on it and dragging the mouse across the workspace. To move a region, click anywhere inside the region or on its border. You can also use arrow keys to move objects.

You can shrink or enlarge a region by double-clicking on it, placing the cursor on a border anchor, and dragging it.

Changing the Shape of the Plot Designer Window

Plot Designer can be viewed in two orientations, **Landscape** or **Portrait** (which is the default orientation). You can change the shape of the Plot Designer window in the **Orient** menu.

Changing the Size of the Plot Designer Window

Increase or decrease the size of the Plot Designer window by clicking on the sizes listed in the **Magnify** menu.

Saving Your Plot

After you are satisfied with the plot that you have created, do the following procedure to store your file:

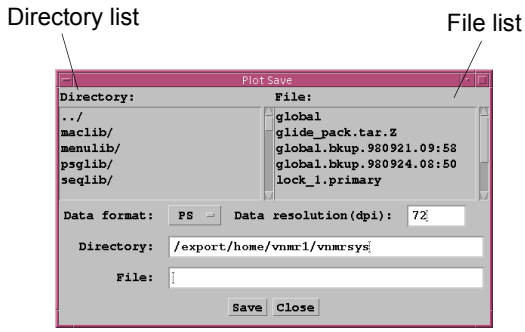


Figure 24. Plot Save Window

1. Click on **File** in the Main Menu, then **Save Data** to open the Plot Save window shown in Figure 24.
2. Scroll down the list of directories and choose a directory. You can also enter a path for your file in the **Directory** field.
3. Select a **Data format** for your file and enter a **Data resolution**. Table 9 lists the formats that are available.
4. Label your file by entering a name in the **File** field.
5. Click **Close** to exit the window.

Printing Your Plot

To print your plot, click on the print tool. 

Table 9. Formats Available in Plot Designer

Format	*Description*
AVS	AVS X image file
BMP	Microsoft Windows bitmap image file
EPS	Adobe Encapsulated PostScript file
FAX	Group 3 FAX
FITS	Flexible Image Transport System
GIF	Compuserve Graphics Interchange Format (version 89a)
GIF87	Compuserve Graphics Interchange Format (version 87a)
JPEG	Compressed format from Joint Photographic Experts Group
MIFF	Magick image file format
PCD	Photo CD
PCX	ZSoft IBM PC Paintbrush file
PDF	Portable Document Format
PICT	Apple Macintosh QuickDraw/PICT file
PGM	Portable gray map
PNG	Portable Network Graphics
PS	Adobe PostScript file
PS2	Adobe Level II PostScript file
SGI	Irix RGB image file
SUN	Sun Rasterfile
TGA	Truevision Targa image file
TIFF	Tagged Image File Format
VIFF	Khoros Visualization image file
XBM	X11 bitmap file
XPM	X11 pixmap file
XWD	X Window System window dump image file

Exiting Plot Designer

To exit the program, click on **File-Quit**. If you leave a design in the window when you exit Plot Designer, your design will automatically appear in the workspace the next time that you use the program.

8.3 Color Printing

The `color` program allows you to change the colors on the VnmrJ display screen and color print to a plotter.

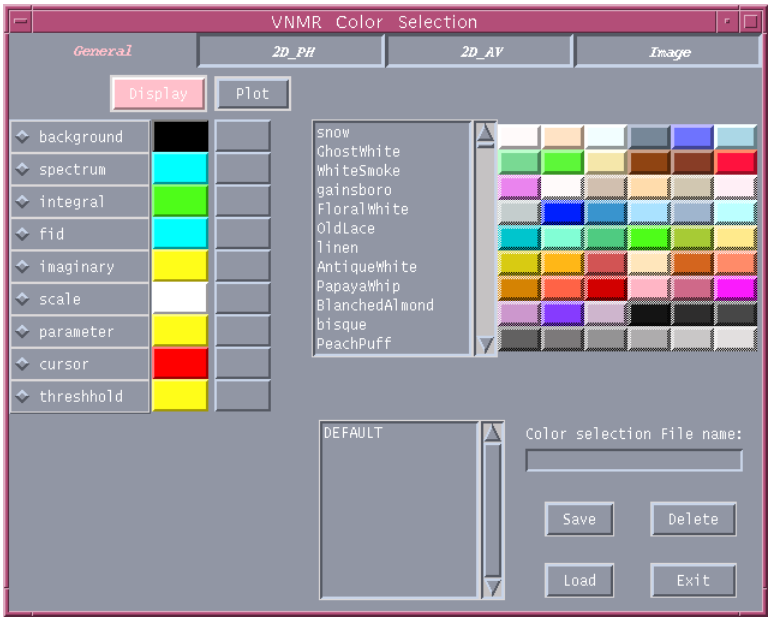
- "Starting the Color Program," this page
- "Setting Colors," page 129
- "Loading A Color File," page 129
- "Changing or Renaming A Color File," page 130

Starting the Color Program

To start the program, enter `color` on the VnmrJ command line.

A color selection window opens.

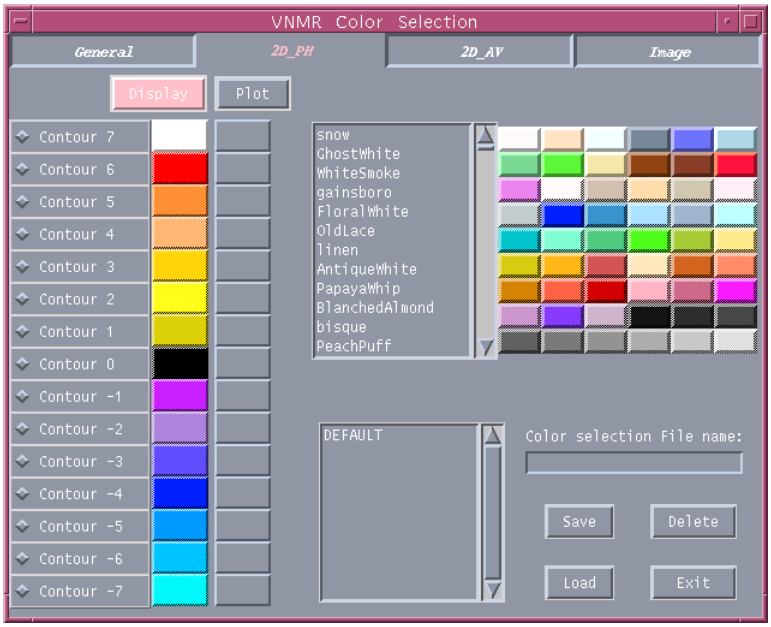
Color Selection Window (`color` Program)



The default window is for General (or 1D phase) color selection for the graphics window (Display). To change to other color selection windows, click on the buttons near the top of the VnmrJ Color Selection window to display the 2D Phase, 2D Absolute Value, and Image color selection windows:

- The General window has buttons along the left side that list the areas of the graphics window for which you can set colors: background, spectrum, integral, fid, imaginary, scale, parameter, cursor, and threshold. To the right of each button is the color currently assigned for that area of the graphics window.
- The buttons for the 2D Phase window for the display allow you to set colors for the contours of the display.
- The buttons in the 2D Absolute Value window for the display allow you to assign colors to the contours of the display.
- The Image window for the display allows you to set the colors for the display background and foreground.

2D Phase Color Selection Window (`color` Program)



2D Absolute Value Selection Window

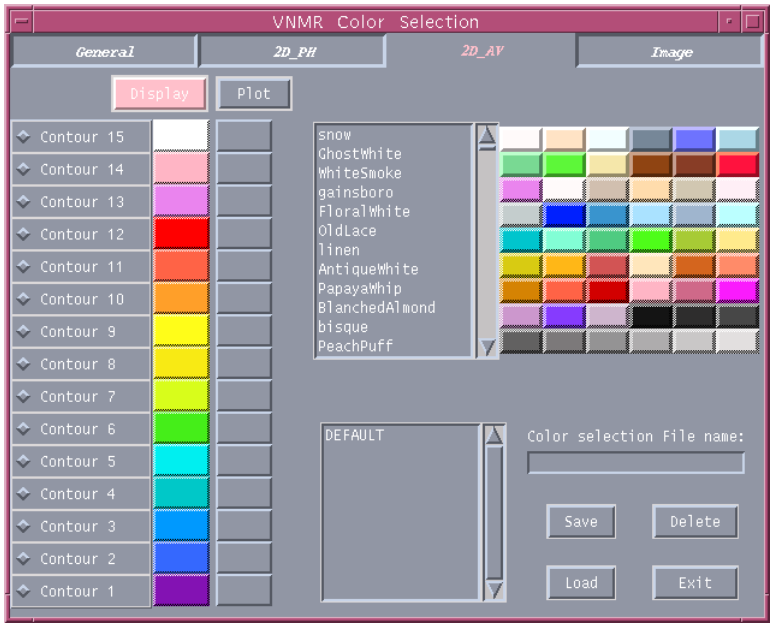
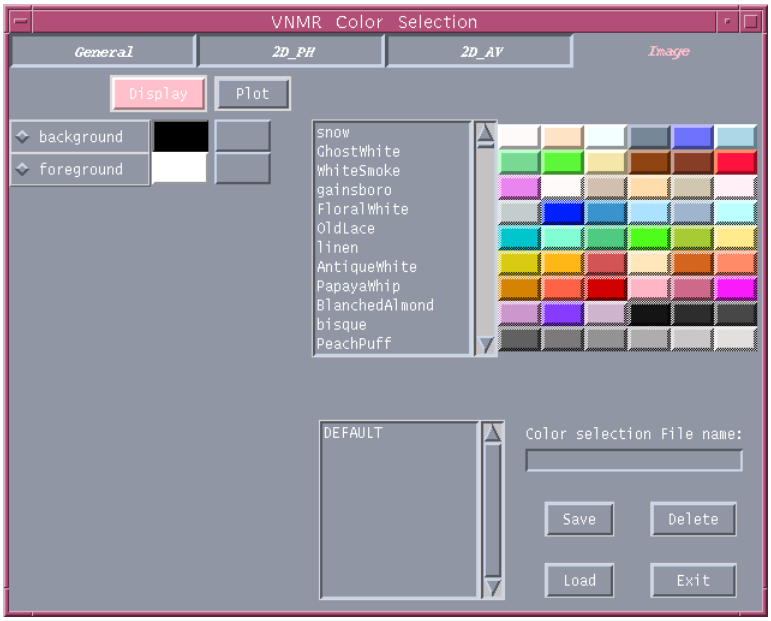


Image Color Selection Window



Setting Colors

You can select colors from either the color list box (in the center of the window) or the color palette. For every color on the palette, there are ten values in the list box. To set the background color of the graphics window to gray, for example, do the following steps:

1. Click on the background button.
2. Click on a gray button in the color palette.

   The name of the button and its values appear in the scrollable list box directly to the left of the palette. To change the color shade, click on a value in the list (e.g., gray13).

To set colors for the remaining eight areas of the display, repeat steps 1 and 2.

Saving A Color File

When you are satisfied with the color assignments, you can save them in a file as follows:

1. Enter a name in the Color Selection File Name field in the bottom right-hand corner of the Color Selection window. The bottom list box is updated with the new file name.
2. Click on the Save button. The color file is saved in the designated place in the VnmrJ system file.

Loading A Color File

To retrieve a color file:

1. Click on the color file name in the scrollable list box at the bottom of the window. The color file name appears on the entry area of the Color Selection File Name field.

2. Click on the Load button.

Changing or Renaming A Color File

To change the colors in a file:

1. Load the file.
2. Enter new color assignments.
3. Save the file.

To change the name of a color file:

1. Load the file.
2. Save the file with a new name.
3. Delete the file with the old name.

Removing A Color File

To remove a color file from the list:

1. Click on the file name.
2. Click on the Delete button. The deleted file is removed from the bottom list box.
3. When you are prompted, choose OK to delete the file, or Cancel to keep the file.

Assigning Colors to A Plotter

To assign colors to a plotter:

1. Click on the Plot button.
2. Choose the type of plotter that you have.
   Use the previously listed procedures to save, load, change, rename, and remove files.

To use normal plotting commands:

1. Exit the Color Selection window.
2. Enter `setcolor` on a command line.

Closing the Color Selection Window

When you have finished using the program, click on Exit to close the window.

Color Table Loader

The macro `loadcolors<(color_file)>` loads the color table for the graphics window and plotters. `loadcolors` is generated by the `colors` program and includes a series of `setcolor` commands. On bootup, the `bootup` macro calls `loadcolors` to set the graphics window and plotter colors.

Chapter 9. Advanced 1D NMR

Sections in this chapter:

This chapter describes working with 1D NMR liquids experiments.

9.1 Working with Experiments

In the Locator, experiments are contained in Workspaces.

- To view the experiments in the Locator, click the **Locator Statements** icon () and select **Sort Workspaces**.
- To open an experiment, drag it to the graphics window.
- To create a new experiment, click Utilities -> Create a Workspace.
- To delete an experiment, drag the experiment from the Locator to the trash can icon.

To view the experiments on a system in the Process tab Text Output page, enter `explib`. The monitor displays the experiment library of the currently available experiment files (`exp1`, `exp2`, ..., `exp9999`).

When multiple experiments are created, an issue arises concerning how to individually work with each experiment. To handle this matter, only one experiment is allowed at a time to be currently active (i.e., in the foreground for manipulation), although background processing can be occurring in other experiments at the same time.

The `mp`, `mf`, and `md` commands move FIDs and parameters between experiments:

- `mp(<n,>m)` moves parameters from experiment `n` to experiment `m`, for example, `mp(4,5)`. If `n` is omitted, parameters are moved from the currently active experiment to experiment `m`.
- `mf(<n,>m)` moves the last acquired FID and the associated parameters.
- `md(<n,>m)` moves only those "saved display" parameters associated with the commands `s1` through `s9`.

9.2 Multi-FID (Arrayed) Spectra

Many experiments require obtaining a series of FIDs, related to each other through the variation of one or more parameters. For example, suppose it is necessary to run a series of spectra at four different temperatures: 30°C, 50°C, 70°C, and 90°C. Instead of acquiring four separate sets of data, it is possible to create an array in which the `temp` parameter is given four successively different values. These four subexperiments are now all treated as a single experiment. Entering `go` begins successive acquisition of all four experiments. One command can be used to transform all the spectra, one command to display all the spectra on the screen simultaneously, one command to plot all the spectra, and one command to save all the spectra.

Arrayed Parameters

To create an array for a numeric parameter, enter the arrayed values separated by commas, (e.g., `temp=30,50,70,90` or `pw=5,10,15,20,25`). Alphanumeric parameters can also be arrayed. To perform two experiments in which the decoupler is off in one case and on in the other, for example, you can use `dm='n','y'`.

Not all parameters can be arrayed. Non-arrayable acquisition parameters include processing parameters, display parameters, and any parameter that changes the number of data points to be acquired, such as `np`, `sw`, `dp`, and `at`.

To display the values of the arrayed parameter, the `da` command is used. `da` displays all values of arrayed parameters if entered without an argument. If one or more parameters are listed as an argument, `da` displays only the specified parameters.

Multiple Arrays

Two or more parameters can be arrayed in an experiment. For example, an experiment to perform a series of decoupling experiments using an array of decoupler power levels and an array of decoupler frequencies might be set up with `dpwr=17,20,23` and `dof=295.1,345.6,507.2,1245.5`. In this example, *twelve* experiments are performed (i.e., three different values of decoupler power `dpwr` are used), and for each of those values, four different values of the decoupler offset `dof` are used.

Setting Array Order and Precedence

Whenever an array of one or more parameters is set up, the parameter `array` becomes important. This parameter tells the system the name of the parameter or parameters that are arrayed, and the order and precedence in which the arraying is to take place.

The string parameter `array` can have one of several forms:

- `array=''` means no parameter is arrayed (this value is two single quotation marks with no space between, not a double quotation mark).
- `array='x'` means parameter `x` is arrayed.
- `array='y,x'` means parameters `x` and `y` are arrayed, with `x` taking precedence. The order of the experiments is x_1y_1, x_2y_1, ...x_ny_1, x_1y_2, x_2y_2, ...x_my_2 ...x_my_n, with a total of `m×n` experiments being performed.
- `array='x,y'` means parameters `x` and `y` are arrayed, with `y` taking precedence. The order of the experiments is x_1y_1, x_1y_2,...x_1y_n, x_2y_1, x_2y_2,...x_2y_n,...x_my_n, with a total of `m×n` experiments being performed.
- `array='(x,y)'` means parameters `x` and `y` are jointly ("diagonally") arrayed. The number of elements of the parameters `x` and `y` must be identical, and the order of experiments is x_1y_1, x_2y_2, ...x_ny_n, with `n` experiments being performed.

As you enter one or more arrayed parameters, `array` is automatically set for you. Only if you want to change the order or precedence is it necessary to enter `array` directly.

Interactively Arraying Parameters

Separate from the `array` *parameter* is the `array` *macro*. If you enter the `array` macro without an argument, an interactive mode is started in which you are asked for the following information, in this order:

- The name of the parameter to be arrayed.
- The number of values of the parameter.
- The starting value.
- The magnitude of the difference between elements in the array.

Using the information you provide, an arrayed parameter is set up. The restrictions are that only numeric parameters can be arrayed and all values of the array must satisfy the limits of the parameter.

Entering `array` with a parameter name as an argument, (e.g., `array('pw')`) still starts an interactive mode but the program only asks for the remaining three items of information.

If you enter the macro with all four pieces of information as arguments (in this order—parameter name, number of steps, starting value, and step size), `array` bypasses the interactive mode completely. For example, entering `array('tof',5,1000,-50)` sets the `tof` parameter to have 5 elements with the values in the order 1000, 950, 900, 850, 800.

Resetting an Array

Once an array is created, it is possible to change the value of a single element of the array by typing, for example, `pw[2]=11.3`, where the `2` enclosed in brackets indicates which element of the array to modify (array elements are counted starting at 1).

To reset an arrayed parameter to a single value, enter a single value for the parameter (e.g., `pw=10`). The `array` parameter is automatically modified to reflect this change.

Array Limitations

Regular multiple arrays can include up to 20 parameters, each of which can be a simple parameter or a diagonal array (a set of parameters), which can include up to 10 parameters. The total number of elements of all arrays is essentially unlimited (2^{32}–1).

Acquiring Data

Once any parameter is an array, entering `go` (or related commands and macros) generates not just one, but an entire array of spectra. These spectra can then be examined either individually or as a group, as described below.

Autogain cannot be used in an arrayed experiment. You can either use `gain='y'`, which sets the gain to the previously determined value, or set `gain` equal to a fixed value.

Arrayed acquisitions can be interleaved, in which a part of each experiment is done in turn rather than starting and finishing each experiment sequentially. The interleave function is controlled by the parameter `il`.

- If `il='y'`, experiments are interleaved. `bs` transients are performed for each member of the array, followed by `bs` more transients for each member of the array, and so on until `nt` transients are collected for each member of the array. Thus, `il` is relevant only if `bs` (block size) is less than `nt` (number of transients).
- If `il='n'`, all transients are acquired for the first experiment in the array, then all transients for the second experiment, etc.

Processing

The command `ft` or `wft` is used to transform all of the spectra. Both commands take the same arguments and options:

- `'acq'` does not transform elements that have already been transformed.
- `'nodc'` does not perform FID drift correction.
- `'nods'` prevents an automatic spectral display (sane as `ds` command).
- `'zero'` zeroes the imaginary channel of the FID before Fourier transform.

Phasing can be done on any spectrum. Only one set of phase correction parameters exists, so all spectra have the same phase at any one time (although the phase can of course be changed when examining different spectra).

Display and Plotting

The command `ds(index)` displays interactively the requested spectrum from the array. The index can have one, two, or three numbers, depending on the dimensionality of the spectral array. Spectra are always scaled according to the number of completed transients `ct`; if `nt` is arrayed (`nt=1,2,4,8`), each spectrum is scaled by its *own* `ct`.

Other spectra display commands are `dss`, `dssn`, `dssa`, `dssan`, `dssh`, `dsshn` and `dssl`. These are not interactive like the `ds` command. They display stacked spectra in which each spectrum is offset with respect to the previous spectrum. The order of stacking can be left to right, right to left, top to bottom, or bottom to top, depending on whether the horizontal offset (`ho`) and vertical offset (`vo`) parameters are positive or negative. Some of these commands set `ho` and `vo` automatically.

The spectra display commands function as follows:

- `dss` displays stacked spectra using the current values of `ho` and `vo` to set the order of stacking.
- `dssn` displays stacked spectra the same as `dss`, but the graphics window is not erased before starting the display. This allows composite displays of many spectra to be created.
- `dssa` displays stacked spectra automatically (i.e., `vo` and `ho` are automatically adjusted to fill the screen in a lower left to upper right presentation).
- `dssan` displays stacked spectra automatically the same as `dssa`, but the graphics window is not erased before starting the display.
- `dssh` displays stacked spectra horizontally (i.e., `vo` is set to zero and `ho` is adjusted to fill the screen from left to right).
- `dsshn` displays spectra horizontally the same as `dssh`, but the graphics window is not erased before starting the display.
- `dssl` displays a label for each element in a set of stacked spectra. The label is an integer value starting with 1 and extending up to the number of spectra in the display.

The command `pl` plots stacked spectra with the same format as displayed by `dss`.

The argument syntax `<(start,finish<,step>)><,options>)>` is used by the `dss` command, variants of `dss`, and by the `pl` command. The arguments are the following:

- `start` is the index of the first spectra when displaying multiple spectra. It is also the index number of a particular trace to be viewed when displaying arrayed 1D spectra or 2D spectra.
- `finish` is the index of the last spectra when displaying multiple spectra. Because the parameter `arraydim` is automatically set to the total number of spectra, it can be used to set `finish` to include all spectra.
- `step` is the increment for the spectral index when displaying multiple spectra. The default step is 1.
- `options` can be any of the following:

  `'all'` is a keyword to display all of the spectra.

  `'int'` is a keyword to only display the integral, independently of the value of the parameter `intmod`.

  `'top'` or `'side'` are keywords that cause the spectrum to be displayed either above or at the left edge, respectively, of a contour plot. This assumes that the parameters `sc`, `wc`, `sc2`, and `wc2` are those used to position the contour plot. This option does not apply to `dssa`, `dssan`, `dssh`, or `dsshn`.

  `'dodc'` is a keyword for all spectra to be drift corrected independently.

  `'red'`, `'green'`, `'blue'`, `'cyan'`, `'magenta'`, `'yellow'`, `'black'`, and `'white'` are keywords that select a color. This option does not apply to `dssa`, `dssan`, `dssh`, `dsshn`, or `pl`.

  `'pen1'`, `'pen2'`, `'pen3'`, etc. specify a pen number on a plotter. This option does not apply to `dss` or any of its variants.

Pulse Width Calibration Step-by-Step

To illustrate using arrays, note how the following steps perform a pulse width calibration:

1. Set up parameters and obtain a normal spectrum of any sample. For best results, one or more intense signals should appear near the center of the spectrum.

2. Enter **pw=5**. You can use some other small value if you wish.
3. Enter **nt=1**.
4. Obtain a spectrum and phase it properly. Set **d1** to 5*T_1.
5. Enter **pw=5,10,15,20,25,30 ai go**.

   You can use some other set of suitable values for the pw array.
6. After the experiment finishes acquisition, enter **wft dssh**.
7. Find the experiment where the signal goes through its 180° or 360° null. Enter **da** to remind yourself of the values of the pw array.
8. To reset the array, enter **pw=10**.

9.3 T_1 and T_2 Analysis

One relatively common form of arrayed experiment is the inversion-recovery T_1 experiment. In this experiment, the nuclei are allowed to relax to equilibrium (d1), then inverted with a 180° pulse (p1), given a variable time to return to equilibrium (d2), and finally given a monitoring 90° pulse (pw) to measure their peak height as a function of d2. Under most circumstances, the behavior of the peak heights as a function of d2 will be exponential, and this exponential time is the T_1.

- "Setting Up The Experiment," this page
- "Processing the Data," page 136
- "Analyzing the Data," page 137
- "Exponential Analysis Menu," page 137
- "T_1 Data Workup: Step-by-Step," page 137

Setting Up The Experiment

The standard two-pulse sequence is set up to perform the T_1 experiment. You can start if you wish by entering appropriate values for p1, pw, d1, and an array of values for d2.

Alternatively, you can use the dot1 macro. dot1 sets up all parameters to perform a T_1 experiment, including d1, pw, p1, nt, and an array of d2 values, based on information you enter. The three arguments that can be input are the minimum expected T_1, the maximum expected T_1, and the total time in hours the experiment should take. If no arguments are provided, dot1 prompts the user for the information.

Be sure that the parameter pw90 is set properly and contains the correctly calibrated 90-degree pulse width, because dot1 uses this information.

Processing the Data

Once the data is acquired, process the data as follows:

1. Enter **wft ds(arraydim)** to display the last spectrum (or **ds(1)** for a T_2 experiment to display the first spectrum).
2. Phase this spectrum properly.
3. Select a threshold and adjust the threshold line position.
4. Enter **dpf** or **dll** to display a line list and locate lines for the system.

5. Enter **fp** to measure the peak height of each peak in an array of spectra. If optional line indexes are supplied to fp as arguments (e.g., fp(1,3)), only the peak heights of the corresponding lines are measured.

   The npoint parameter (if defined and set "on") determines the range of data points over which the fp command searches for a maximum for each peak.

Analyzing the Data

T_1 and T_2 analysis is performed by the t1 and t2 macros, respectively. t1 and t2 measure relaxation times for all lines in the line listing and display an extended listing of observed and predicted peak intensities. t1s and t2s perform the same calculation as t1 and t2 but produce a shorter output, showing only a summary of the measured relaxation times.

The command expl displays exponential/polynomial curves resulting from T_1, T_2, or kinetic analysis. Optional input of line numbers as arguments allows displaying only selected lines. Similarly, the command pexpl plots the same curves.

The macro autoscale returns the command expl to autoscaling in which scale limits (set by scalelimits) are determined that will display all the data in the expl input file. The macro scalelimits causes the command expl to use typed-in scale limits. If no arguments are given, scalelimits asks for the desired limits. The limits are retained as long as an expl display is retained.

To delete spectra from the t1 or t2 analysis (or from t1s or t2s), enter dels(index1<, index2>...). This command deletes the spectra selected by the indexes from the output file fp.out of the fp command used by the t1 or t2 analysis. Spectra can be restored by rerunning fp.

Exponential Analysis Menu

Most of the commands for working with T_1 and T_2 analysis are available by clicking on Main Menu button, followed by the Analyze button, and then the Exponential button. The following menu, called the Exponential Analysis menu, is displayed

T_1 Data Workup: Step-by-Step

The following procedures accomplish the same result.

1. Enter **rt('/vnmr/fidlib/t1data.fid')**.
2. Enter **wft dssh full ds(arraydim) aph**.
3. Click on **Next** > **Th**. Use the left mouse button to set the threshold.
4. Enter **dll fp t1 center expl**.

9.4 Kinetics

The arraying capability of the VnmrJ software provides for the acquisition of data for the study of kinetics.

- "Setting Up the Experiment," this page
- "Processing the Data," this page
- "Kinetics Step-by-Step," this page

Setting Up the Experiment

Usually, the best procedure is to array the preacquisition delay parameter `pad`. For example, if `pad=0,3600,3600,3600,3600`, the system acquires the first spectrum immediately (`pad[1]=0`), waits 3600 seconds (`pad[2]=3600`), acquires the second spectrum, waits another 3600 seconds, etc. Because 3600 seconds is 1 hour, this inserts a wait of one hour between acquisitions. After all the spectra have been obtained, they are processed much like T_1 or T_2 data.

Processing the Data

If the signal decreases exponentially with time, the output is matched to the equation `I=A1*EXP(-T/TAU)+A3`. The analysis is done by the macro `kind`, or by macro `kinds` if a short output table is desired.

If the signal increases exponentially with time, the output is matched to the equation `I=-A1*EXP(-T/TAU)+A3-A1` with analysis done by the macro `kini`, or by the macro `kinis` for a short output table.

Kinetics Step-by-Step

The following steps are typical in processing a kinetics experiment:

1. Enter **`wft dssh full ds aph`**.
2. Click on **Threshold** icon in the graphics control menu. Use the left mouse button to set the threshold.
3. Enter **`dll fp`**.
4. Enter **`kind`**, **`kini`**, **`kinds`**, or **`kinis`**, as appropriate.
5. If desired, adjust `sc`, `wc`, `sc2`, and `wc2` by entering **`center`** or **`full`**.
6. Enter **`expl`**.

9.5 Diffusion Experiments

Software for diffusion measurements using a pulsed Z-gradient probe is an optional feature of Varian NMR systems. This section describes the pulsed gradient experiment and its analysis.

Pulsed Gradient Experiments

The pulse sequences diagrammed in Figure 25 illustrate the two pulsed gradient experiments. These experiments are fully described in the literature by Stejskal and Tanner (Stejskal, E. O.; Tanner, J. E. *J. Chem. Phys.* **1965**, ***42***, 288-292, and Tanner, J. E. *J. Chem. Phys.* **1970**, ***52***, 2523–2526).

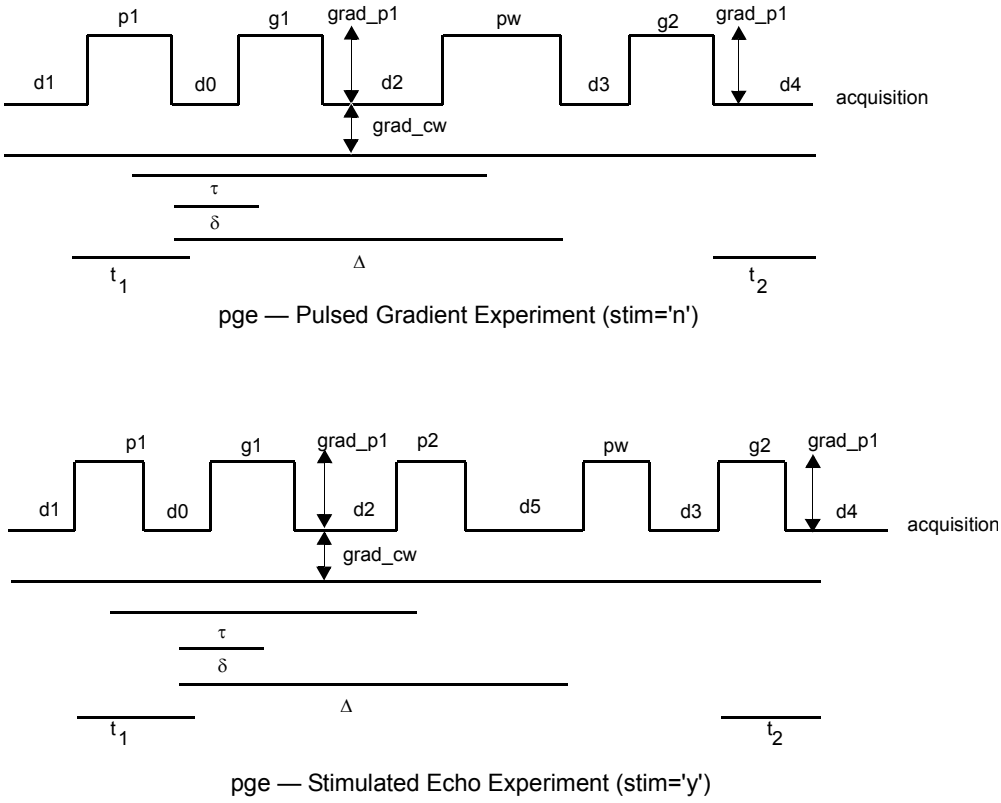


Figure 25. PGE Pulse Sequences

A pulse sequence named PGE and its associated help file are provided. A parameter set `/vnmr/parlib/pge` is used as the master parameter set.

The parameters `p2` and `d5` are only used for the stimulated echo experiment (`stim='y'`). The parameter `p1` defines the first rf pulse, `pw` defines the last rf pulse. Delay `d1` occurs before the first rf pulse.

With the current pulse sequence, `grad_p1` defines the amplitudes of both pulsed gradients: `g1` and `g2`. Unused parameters `grad_p2` and `dac_p2` are provided if you wish to modify the pulse sequence to use different gradients. These parameters are unused by the `pge` pulse sequence that is provided.

The PGE pulse sequence checks the average power output of the gradient coils for a safe value. Maximum value is 100 gauss/cm for a continuous wave gradient. This value is hard-

coded into the pulse sequence by defining a numerical value in the pulse sequence for the variable `dutycycle`. The value is calculated from the following expression:

$$\texttt{dutycycle} = \sqrt{\frac{(\texttt{grad\_sw})^2 \times \texttt{time} + (\texttt{grad\_p1})^2 \times (\texttt{g1} + \texttt{g2})}{\texttt{time}}}$$

where `time=g1+g2+d0+d1+d2+d4+d5+p1+p2+pw+at`.

The user should change this to fit the needs of a specific probe. Comments in the pulse sequence file `/vnmr/psglib/pge.c` contain more details about the sequence.

Pulsed Gradient Experiment Setup

There are two ways to retrieve a set of parameters for the pulsed gradient experiment:

- Enter `rt('/vnmr/parlib/pge')`. This command returns a complete parameter set into the active experiment.
- Enter `pge(macro_name)`. This appends the required parameters to those already present in the active experiment.

For the data from a diffusion experiment to be analyzed properly, it is necessary to define integral regions in the Fourier transformed spectra. The simplest way to define these regions is to run the pulsed gradient experiment as a simple echo experiment (`grad_p1=0`): first, to check that the values of τ and 2τ start the data acquisition at the top of the echo, and second, to define integral regions in the experiment where the pulsed field gradient experiment will be subsequently done. The phase parameters (`rp` and `tp`) should also be determined for this spectrum. If data acquisition was started at the top of the echo, the `lp` phase parameter should be zero. The easiest way to define integrals is by using the RESETS button of the `ds` command.

Once the parameters are in the experiment, those associated with gradient control can be set by entering `pge_setup`. This macro has a single optional argument to turn off interactive questioning; any argument can suffice (e.g., `pge_setup('no')`).

The macro `pge_setup` performs the following three tasks.

1. Sets the gradient (`grad_p1`) array.

   If the interactive mode is used, `pge_setup` prompts for the values of `g_max`, `g_min`, `g_steps`, and `g_array`. These parameters are used to calculate the gradient amplitude array. Manual override is provided by typing in each value. The value of `g_array` can be set to `'linear'` if the gradient values are equally spaced between `g_min` and `g_max`, or set to `'square'` if the square of the gradient values is equally spaced between the square of `g_min` and the square of `g_max`.

2. Sets the number of transients (`nt`) array.

   If the interactive mode is used, `pge_setup` prompts for the values of `nt_array` and `nt_first`. The values of `nt_array` are `'same'` and `'fraction'`. The parameter `nt_first` is used to set to the value of the first `nt` array element. If `'same'` is selected, all elements of the `nt` array are set to `nt_first`. If `'fraction'` is selected, `nt_fract` is set so that elements of the `nt` array are calculated according to the equation:

   $\texttt{nt}_i = \texttt{nt\_first} \times [1 - i \times (1 - \texttt{nt\_fract}) / \texttt{g\_steps}]^2$

   Manual override is provided by typing in each value.

3. Does necessary housekeeping.

The array parameter is set so that nt and dac_p1 form a "diagonal array." The time macro is executed to display the experiment duration. The average gradient level of the particular parameter combination is checked when acquisition is attempted. The wexp parameter is set to 'pge_process' to perform appropriate data processing at the end of the experiment.

Other acquisition parameters can be altered by typing in new values. Once a "good" set of parameters is entered, it may be saved for future recall with the svp (save parameters) command. The automatic processing of the diffusion data following data acquisition is initiated with the au command if wexp is set to 'pge_process'.

Gradient Calibration

Calibration constants, which relate DAC values (in DAC units) to resulting values (in gauss/cm), are stored in conpar with names grad_cw_coef and grad_p_coef. These coefficients are in units of (gauss/cm)/(dac unit). The DAC values are whole numbers while gradient values are real numbers that may contain fractional parts.

Whenever the dac_x or grad_x parameters are changed, where x is cw, p1, or p2, macros are available to adjust the dependent parameter, taking into account possible minimum and maximum values and housekeeping in case the parameters are arrayed. The mathematical relationships are defined as follows:

- When dac_x is changed: grad_x=dac_x*grad_x_coef
- When grad_x is changed: dac_x=grad_x/grad_x_coef
  grad_x=dac_x*grad_x_coef

The second step taken, when grad_x is changed, is necessary because the calculation of dac_x is rounded to the nearest integer, which necessitates that grad_x then be recalculated so that it corresponds to dac_x.

A macro pge_calib is provided to assist in the calibration of acquisition parameters that control the gradient power levels (i.e., the DAC values). After phasing and selecting the integral region of the standard sample, run a pulsed gradient experiment and process data with the pge_process macro. The calculated diffusion constant is displayed in the text window. Then run the pge_calib macro to recalculate the coefficient. This macro resets the set of data, followed by processing with the pge_process macro, and should now give the diffusion constant that was selected with the pge_calib macro. After running pge_calib, run pge_setup again to calculate DAC values with the new coefficient.

Data Reduction

The pge_process macro performs several tasks (see Figure 26), calling macros as appropriate.

1. Transforms the data with the specified weighting function.
2. For each array element (i.e., spectrum), writes into a text file the following information: the gradient value for that spectrum, the integral amplitudes, and the spectral parameters needed to reproduce the data.

   The file is stored in the current experiment directory and its name reflects which element of the array this information pertains to (e.g., info_1 to info_n). The macro pge_data is called by pge_process to do this for each array element.

   The pge_data macro has the element number passed as an argument. This feature allows an operator to manually adjust the spectral parameters for a single element

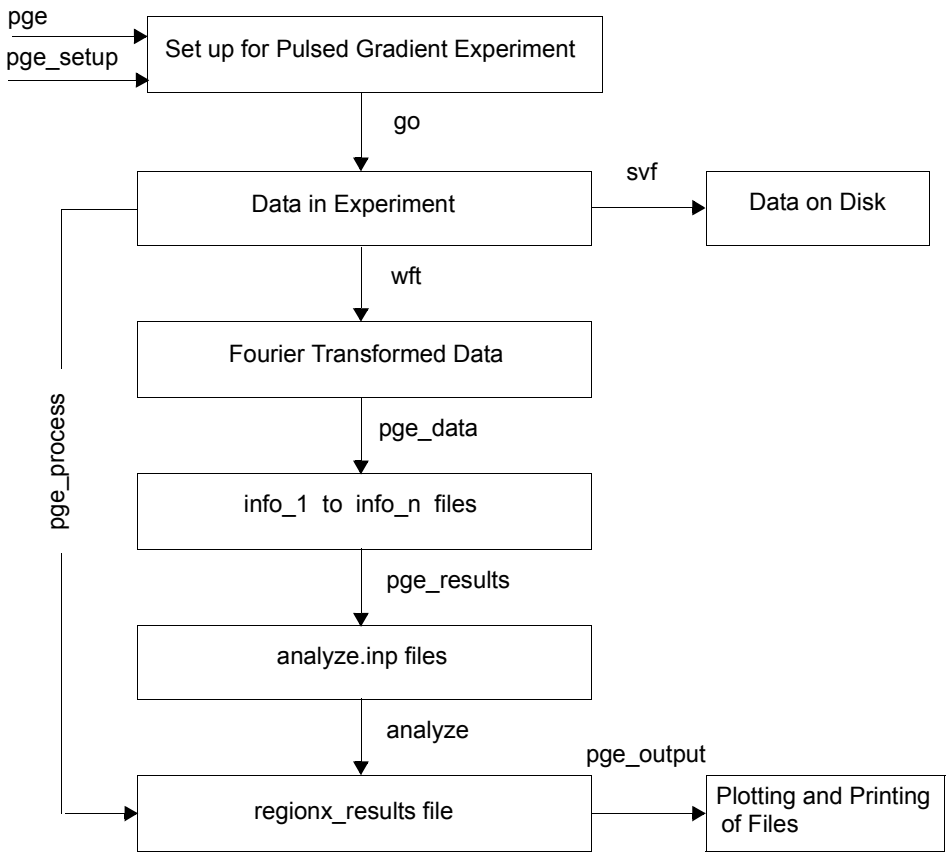


Figure 26. Data Processing Macros Flowchart

and then invoke the `pge_data` macro to update the raw information in the appropriate text file.

A sample information file for the second element of an array is the following:

```
Gradient amplitude for spectrum 2 is 63.9715
Spectral parameters:
rp= 344.604 lp= 0 lvl= 0 tlt= 0
sp= -5000 wp= 10000 is= 691.5 ins= 1 fn= 32768
2 Integral Regions          Value
7440186      -46.5117       4.58914
-325.582     -1604.65       36.8508
```

To manually correct the data from a single element of the data array, the following sequence can be used: select the desired array element with `ds` (e.g., `ds(5)`); manually adjust the phase or drift corrections; then, type `pge_data(5)` to write the new integral information into the `info_5` text file.

3. The `pge_process` macro calls the `pge_results` macro. `pge_results` collects the necessary information from the `info_n` files and constructs a text file `analyze.inp` that is used to calculate the diffusion coefficient. `pge_results` requires a single argument that indicates which integral region to use when recalculating the diffusion coefficient with the information from `info_n` files.

   The `pge_process` macro calls `pge_results` once for each of the defined integral regions. Figure 27 is a sample `analyze.inp` text file.

```
List of 8 x-y data
pairs
diffusion data for   \
integral region 2
Grad*Grad     ln(Amp)
0             4.23277
4092.35       3.60688
8192.57       3.01594
12305.2       2.37857
16369.4       1.72547
20619.8       1.13322
```

Figure 27. Sample `analyze.inp` File

The diffusion coefficient is calculated using Equation 1:

$$A_i = A_0 \exp -\gamma^2 D\left\{\frac{2}{3}t^3 \times \mathtt{grad\_cw}^2 + \delta^2\left(\Delta - \frac{1}{3}\delta\right)\mathtt{grad\_p1}i^2 - \delta\left[(t1^2 + t2^2) + \delta(t1 + t2) + \frac{2}{3}\delta^2 - 2\tau^2\right]\mathtt{grad\_p1}i \times \mathtt{grad\_cw}\right\}$$

where A_i is the observed integral value of an NMR resonance for the *i*th element of the gradient array. A_0 is the integral value of an NMR resonance just after the first 90° pulse in the pulse sequence. Δ is the self-diffusion coefficient. In the case of the two-pulse echo sequence, the variables τ, δ, Δ, *t1*, and *t2* (refer to Figure 25) are calculated from the pulse sequence variables as follows:

$$\tau = \frac{(\mathtt{p1})}{2} + \mathtt{d0} + \mathtt{g1} + \mathtt{d2} + \frac{\mathtt{pw}}{2}$$

$$\delta = \mathtt{g1}$$

$$\Delta = \mathtt{g1} + \mathtt{d2} + \mathtt{pw} + \mathtt{d3}$$

$$t1 = \frac{\mathtt{p1}}{2} + \mathtt{d0}$$

$$t2 = \mathtt{d4}$$

In the case of the stimulated-echo sequence, the equation for Δ is as follows:

$$D = \mathtt{g1} + \mathtt{d2} + \mathtt{p2} + \mathtt{d5} + \mathtt{pw} + \mathtt{d3}$$

Equation 1 can be recast as the following, which becomes Equation 2:

$$ln(A_i) = ln(A_0) + D \times C0 + D \times C1 \times \mathtt{grad\_p1}i + D \times C2 \times \mathtt{grad\_p1}i \times \mathtt{grad\_p1}i$$

where:

$$C0 = \frac{2}{3}\gamma^2\tau^3\,\mathtt{grad\_cw}^2$$

$$C1 = \gamma^2\delta\left[(t1^2 + t2^2) + \delta(t1 + t2) + \frac{2}{3}\delta^2 - 2\tau^2\right] \times \mathtt{grad\_cw}$$

$$C2 = -\gamma^2\delta\left(\Delta - \frac{1}{3}\delta\right)$$

The fitting program `analyze` accepts two arguments that instruct it to perform a polynomial fit. The selection of which polynomial to fit depends on whether `grad_cw` is zero. If it is zero, the second and third terms of Equation 2 vanish and a first-order polynomial $y = c0 + c1 \times x$ is used where:

$y = ln(A_i)$
$c0 = ln(A_0) + D \times C0$
$c1 = D \times C2$
$x =$ `grad_p1`i

Otherwise, a second-order polynomial $y = c0 + c1 \times x + c2 \times x \times x$ is used, where:

$y = ln(A_i)$
$c0 = ln(A0) + D \times C0$
$c1 = D \times C0$
$c2 = D \times C2$
$x =$ `grad_p1`i

Another argument of `analyze` is the complete name of a text file (`analyze.inp`) that contains the x–y data pairs. The output of this calculation is written into a text file in the current experiment directory. The name of this text file reflects the integral region on which the analysis was performed. This name has the form `regionx_results`, where the `x` is the integral region number. Using the experimental delays, `grad_cw` and `gamma`, the `pge_results` macro calculates the diffusion constant and the time-zero integral amplitude from the fitting parameters `c0`, `c1`, and `c2`. The results of these calculations are appended to the text file that contains the least-squares analysis results.

The diffusion coefficients of both components of a two component mixture can be calculated assuming the following condition is met. It is possible to find one integral region where the NMR resonance is due to only one component of the mixture. The diffusion coefficient is calculated using that integral region with the processing already described. For integral regions, where the NMR intensity results from both components of a two component mixture, Equation 1 transforms to the following, which becomes Equation 3:

$A_i = a0 \times exp[D \times (C0 + C1 \times$ `grad_p1`$i + C2 \times$ `grad_p1`$i \times$ `grad_p1`$i)]$
$+ a2 \times exp[a1 \times D \times (C0 + C1 \times$ `grad_p1`$i + C2 \times$ `grad_p1`$i \times$ `grad_p1`$i)]$

The diffusion coefficient D is available from the separate reference integral region. The constants $C0$, $C1$, and $C2$ are defined in Equation 2. The fitting parameters are $a0$, $a1$, and $a2$. In order to perform the non-linear least squares analysis of Equation 3, the `pge_results` macro is supplied with two arguments (e.g., `pge_results(1,3)`). The first argument is the region on which to perform the analysis (just as for the single-component analysis case) and the second argument is the integral region used to get the value of D. The fitting parameter a0 corresponds to the time-zero integral amplitude of the reference component; $a2$ corresponds to the time-zero amplitude of the other component; $a1$ corresponds to the ratio of the two diffusion coefficients.

Data Display

The macro `pge_output` prints the experimental parameters and the results of the diffusion calculations. It also prints graphs of the line fitting results and the spectra.

As with any printing operation, the `pge_output` macro calls `printon`, does a `cat` of the `regionx_results` files, and then calls `printoff`. The plotting is done with the `pexpl` command. The analogous `expl` command displays graphs on the screen.

Two macros are supplied that add the results of the separate calculation to an existing graph. These are called `expladd` and `pexpladd`, for graphics display and plotting, respectively.

Each requires a single argument that specifies the number of the region whose results are to be added to the existing plot or graph.

To plot or display the results of a two-component analysis, the commands `pexpl` and `expl`, respectively, are provided. For example, to plot the results of a two-component analysis, enter `pexpl('square','log')`. This command makes a plot of the square of the gradient versus the natural logarithm of the amplitude.

Variations on the pge Pulse Sequence

In addition to the basic pulse sequence `pge` for diffusion measurements, there are pulse sequence variations on `pge`:

`pgeramp`	Ramps gradients, unlike `pge`. Once the `pge` parameter set has been recalled, set `seqfil='pgeramp'` to execute `pgeramp`. Use this pulse sequence when probe impedance is highly mismatched to the gradient amplifier output. `pgeramp` determines ramp length (defined in μs). When executed, this pulse sequence determines the number of steps in ramping the gradient based on the value of `tramp` (default value is 200 μs) and the gradient strength. In arrayed series of gradients, lower gradients have fewer steps and higher gradients have more steps.
`g2pulramp`	Analogous to `g2pul` except that the gradients are ramped and ramp time is determined by `tramp`. `g2pulramp` is executed by setting `seqfil='g2pulramp'`. It determines the number of steps in ramping the gradient based on the value of `tramp` (the default value is 200 μs) and the gradient strength.

9.6 DOSY Experiments

This section contains the following:

The DOSY (**D**iffusion **O**rdered **S**pectroscop**Y**) application separates the NMR signals of mixture components based on different diffusion coefficients. Generally speaking, DOSY increases the dimensionality of an NMR experiment by one. In 2D DOSY the initial diffusion weighted NMR spectra are one-dimensional; adding diffusion weighting to a 2D NMR experiment such as COSY or HMQC gives 3D DOSY spectra.

The DOSY analyzes involves two steps. These steps are executed by the `dosy` macro.

1. Set up and acquire a DOSY spectrum.
2. Determine the diffusion coefficients for each line (or cross-peak) in the spectrum. Take line (or cross-peak) positions and diffusion coefficients and show the results in a DOSY plot.

Table 10 shows the available tools for DOSY.

Table 10. Tools for the DOSY Experiment

Command	Function
cleardosy	Delete any temporarily saved data in the current (sub) experiment.
ddif	Synthesize and display DOSY plot.
Dgcstecosy	Set up parameters for the Dgcstecosy.c pulse sequence.
Dgcstehmqc	Set up parameters for the Dgcstehmqc.c pulse sequence.
DgsteSL	Set up parameters for the DgsteSL.c pulse sequence.
Doneshot	Set up parameters for the Doneshot.c pulse sequence.
dosy	Process DOSY experiments.
Dbppste	Set up parameters for the Dbppste.c pulse sequence.
Dbppsteinept	Set up parameters for the Dbppsteinept.c pulse sequence.
fbc	Apply baseline correction for each spectrum in the array.
fiddle*	Perform reference deconvolution.
makedosyparams	Create DOSY-related parameters (called by setup macros).
makeslice	Synthesize 2D projection of a 3D DOSY spectrum in diffusion limits.
redosy	Restore the previous 2D DOSY display from the subexperiment.
setup_dosy	Start dialog to set up gradient levels for DOSY experiments.
sdp	Show diffusion projection.
setgcal	Set the gradient calibration constant.
showoriginal	Restore the first 2D spectrum in a 3D DOSY experiment.
undosy	Restore the original 1D NMR data from the subexperiment.

* fiddle(option<,file><,option<,file>><,start><,finish><,increment>)

DOSY Pulse Sequences

Previous DOSY pulse sequences used an unhelpful choice of parameter names. These names have been corrected, but compatibility with old data has largely been maintained. The `dosy` macro attempts to identify the relevant information from the parameters and the pulse sequence name; if it fails, `dosy` starts a dialog asking for three pieces of required information:

- The width of the gradient pulse(s) used for dephasing before diffusion.
- The diffusion delay between dephasing and rephasing.
- For bipolar sequences, the time between the positive and negative gradient pulses.

New sequences always start with "D" and are supplied with this version of the DOSY software. The sequences calculate the time portion of the exponent governing diffusional attenuation, storing the calculation as `dosytimecubed`, and the Larmor frequency of the diffusing spins, storing that calculation as `dosyfrq`.

General Considerations

The DOSY experiments are probably the most demanding gradient sequences in NMR. In conventional coherence-pathway-selected-experiments, you can optimize the experimental conditions for a given gradient setting. However, in DOSY, very often the whole scale of available gradient power is used and the high-resolution NMR conditions still must be maintained. Convection, i.e., moving liquid columns along the sample axis (primarily due to temperature gradients), does not seriously hurt coherence-pathway-selected-experiments (apart from the obvious intensity losses), but, it can make the DOSY analysis completely useless.

DOSY pulse sequences use the gradient stimulated echo element (or one of its modifications), shown in Figure 28.

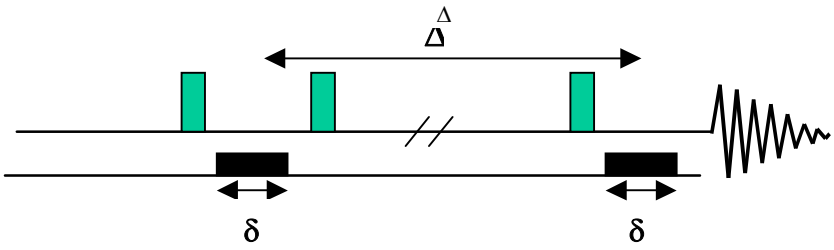


Figure 28. Gradient Stimulated Echo Element

In the DOSY experiments the strength of the diffusion encoding gradient is arrayed and the diffusion coefficients are calculated according to the Stejskal-Tanner formula:

$$S(G_{zi}) = S(0)\exp(-D_i\gamma^2\delta^2(G_{zi})^2(\Delta - \delta/3))$$

where $S(G_{zi})$ and $S(0)$ are the signal intensities obtained with the respective gradients strengths of G_{zi} and 0. D is the diffusion coefficient. γ is the gyromagnetic constant, δ is the gradient pulse duration and Δ is the diffusion delay.

The formula itself provides valuable hints on how to set DOSY-related parameters in different pulse sequences.

- $(\gamma*\delta*G_{zi})^2$ is the gradient pulse area.

Nuclei with higher γ are more sensitive to diffusion than the low-γ nuclei. If possible, observe ^{1}H, ^{19}F, or at least do the diffusion encoding step on the high-γ nucleus. See "Dbppsteinept (DOSY Bipolar Pulse Pair Stimulated Echo INEPT) Experiment," page 153.

- δ is the gradient pulse duration.

  During δ (and the subsequent gradient stabilization delay, `gstab`) the magnetization is transverse and subject to T_2 relaxation and homonuclear J-evolution. Do not use long δ values in the presence of large homonuclear couplings or short T_2 relaxation times ($\delta << T_2$ or 1/J).
- G_z is the gradient strength. Use as high values as possible provided high-resolution NMR conditions are still maintained (no phase, amplitude, and line shape distortions).
- Δ is the diffusion delay. Convection can always be an unwanted competitor to diffusion and T_1 relaxation attenuates the signal intensities. Do not use unnecessarily long diffusion delays ($\Delta < T_1$).

Some of the previous recommendations might seem contradictory. Of course, in real cases you need to find an acceptable compromise between them.

The separation efficiency in the diffusion domain is determined by the accuracy of the measured diffusion coefficients. DOSY does not necessarily intend to get absolute diffusion coefficients (in mixtures, it is difficult to discuss “absolute” numbers anyway); the relative differences in the D values might be adequate for separation.

Note: Changing the solvent of a DOSY mixture might change the diffusion coefficients and the separation power of the method. The solvent might play a similar role in DOSY as the different columns in HPLC.

Diffusion coefficients errors can either be statistical or systematic. The most obvious source of statistical errors is inappropriate signal-to-noise ratio; therefore in DOSY experiments, relatively high S/N values must be reached even with the strongest phase encoding gradients. Systematic errors are primarily caused by instrumental imperfections (such as gradient nonlinearity over the active sample volume, phase distortions, changes in experimental lineshape as a function of gradient amplitude). Systematic errors can be minimized by careful pulse sequence design (see *Magn. Reson. Chem.* (1998), **36:** 706) and by adding a suitable internal reference to the sample (a component producing a strong, well isolated singlet peak in the spectrum) suitable for reference deconvolution (FIDDLE) when processing DOSY.

When setting up DOSY experiments, consider the following recommendations:

- Be sure that the `probe` parameter is set to the probe you intend to use and `Probegcal` has the right value (the setup macros extract the gradient strength, `gcal`, from the probe file and store it in the local parameter DAC_to_G.) Pulse power levels and pw90 values are also read from the probe calibration file.
- Set z0 precisely on resonance and carefully adjust the lock phase. Incorrect adjustment might cause progressive phase errors with increasing gradient power.
- Do not spin the sample.
- Use an adequate number of data points for proper spectral digitization.
- When running long experiments, use interleaved acquisition.
- To minimize temperature gradients (and convection), avoid using extreme (low and high) temperatures. For solutions with very low viscosity, you might prefer to completely switch off the VT controller.

- If you can find a substance suitable for reference deconvolution, add it to the mixture before running DOSY (in proton spectra, TMS might be an ideal candidate).

2D-DOSY Spectroscopy

The current DOSY package includes four 2D DOSY sequences: Dbppste, DgcsteSL, Oneshot, Dbppsteinept.

- "Setting Up 2D-DOSY Experiments," page 149
- "Dbppste (DOSY Bipolar Pulse Pair Stimulated Echo Experiment)," page 151
- "DgcsteSL (DOSY Gradient Compensated Stimulated Echo with Spin Lock) Experiment," page 152
- "The “Oneshot” Experiment," page 152
- "Dbppsteinept (DOSY Bipolar Pulse Pair Stimulated Echo INEPT) Experiment," page 153
- "Processing 2D-DOSY Experiments," page 153

Setting Up 2D-DOSY Experiments

1. Start setting up any of the four experiments by recording a normal s2pul spectrum on the nucleus to be observed.
2. Calibrate (or check) pulse widths if necessary.
3. Before calling the `setup` macro, which always has the same name as the pulse sequence itself, reduce the spectral window to the region of interest.
4. Each sequence has a parameter called `delflag`. By setting `delflag='y'`, the actual DOSY sequence is activated. Setting `delflag='n'` enables you to go back to the basic s2pul (Dbppste, DgcsteSL, Doneshot) or INEPT (Dbppsteinept) sequence without changing the experiment or the parameter set.
5. In all of the sequences, the phase encoding gradient duration is defined by the `gt1` parameter (the total defocusing time). Its strength is defined by the `gzlvl1` parameter and the diffusion delay by the `del` parameter. The actual DOSY setup determines the proper relationship among these three parameters. The best setting primarily depends on the sample itself (e.g., solvent, viscosity, molecular size and shape, the isotope to be detected) and on the experimental conditions (e.g., temperature). Therefore, it is recommended that you use the DOSY sample to optimize the experimental parameters. For small or medium sized molecules, it might be useful to set `gt1=0.002` and `del=0.05` sec and to array the gradient strength:

   `gzlvl1=500,5000,15000,20000,25000,30000` for Performa II

   `gzlvl1=50,500,1000,1500,2000` for Performa I gradient systems
6. For the maximum gradient power used in the DOSY experiment, select the `gzlvl1` value, which attenuates the signal intensities to 5% to 15% of the intensities obtained with the weakest gradient pulse. If the intensity drop is not sufficient at the end of the array, you can increase `del` or `gt1`. If no signal is detected towards the end of the array, decrease `del` or `gt1` and repeat the procedure again.
7. Before the final setup, optimize the alfa delay to reach ideal baseline performance.
8. After having determined suitable values for `gt1`, `del`, and the maximum gradient power, call the `setup_dosy` macro.

`setup_dosy` asks for the number of gradient levels, for the weakest and strongest gradient power to be used in the experiment and sets up a range of `gzlvl1` values with their squares evenly spaced. The minimum gradient strength may be set to 0.3-0.5 G/cm.The number of different pulse areas to use depends on the range of diffusion coefficients to be covered and the balance between systematic and random errors but typically is in the range of 10 to 30. As in any quantitative experiment, there is a balance to be struck when choosing a repetition rate between signal-to-noise and accuracy. But in DOSY experiments, a delay of 1-2 T_1 suffices, provided that care is taken to establish a steady state before acquiring data. It is recommended to set `ss<0` to have steady-state pulses at every new array element and run the acquisition interleaved (`il='y'`).

Each sequence is equipped with a Tcl-Tk acquisition panel, which provides direct access to parameters and setup related commands. Figure 29 shows the acquisition panel of the Doneshot sequence.

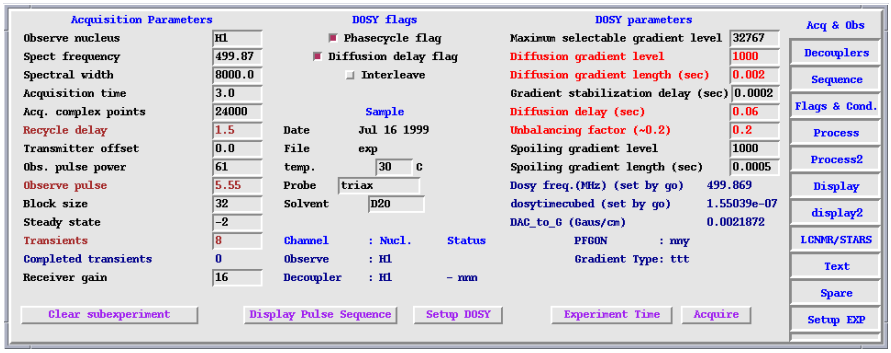


Figure 29. Tcl-Tk Acquisition Panel of Doneshot Pulse Sequence

*Dbppste (**D**OSY **B**ipolar **P**ulse **P**air **St**imulated **E**cho Experiment)*

Figure 30. Dbppste Experiment

Table 11. Dbppste Parameters

Parameters	Function
calibflag	Correct systematic errors in DOSY experiments.
DAC_to_G	Store gradient calibration value in DOSY sequences.
del	Actual diffusion delay.
delflag	y runs the Dbppste sequence. n runs the normal s2pul sequence.
fn2D	Fourier number to build up the 2D display in F2.
gstab	Gradient stabilization delay (~200-300 us)fn2D.
gt1	Total diffusion-encoding pulse width.
gzlvll	Diffusion-encoding pulse strength.

*DgcsteSL (**D**OSY **G**radient **C**ompensated **St**imulated **E**cho with **S**pin **L**ock) Experiment*

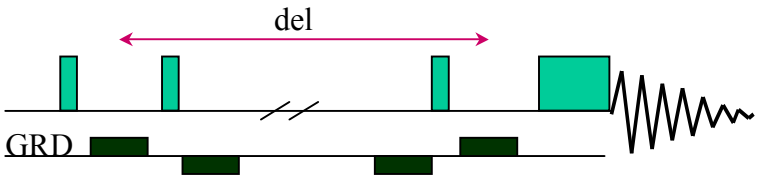


Figure 31. DgcsteSL Experiment

Table 12. DgcsteSL Parameters

Parameters	Function
calibflag	Correct systematic errors in DOSY experiments.
DAC_to_G	Store gradient calibration value in DOSY sequences.
del	Actual diffusion delay.
delflag	`y` runs the DgcsteSL sequence. `n` runs the normal s2pul sequence.
fn2D	Fourier number to build up the 2D display in F2.
gstab	Gradient stabilization delay (~200-300 us).
gt1	Total diffusion-encoding pulse width.
gz_alt	Flag to invert the gradient sign on alternating scans (default is `n`).
gzlvl1	Diffusion-encoding pulse strength.
`prg_flg`*	`y` selects purging trim pulse (default). `n` omits purging pulse.
prgpwr*	Power level for the purge pulse.
prgtime*	Purging pulse length (in second).
tweek	Tuning factor to limit eddy currents. It can be set between 0 and 1, usually set to 0.0.

* The optional purging pulse can effectively eliminate the dispersion signal components. Be careful not to create convection in the sample by the trim pulse.

The “Oneshot” Experiment

The total gradient power transmitted to the sample remains independent on the phase encoding gradient power. Although the sequence design makes phase cycling unnecessary and, unlike other DOSY sequences, the Doneshot sequence can be run with a single transient per array element, it is recommended to turn on the cyclops cycle:

```
phasecycleflag='y'
```

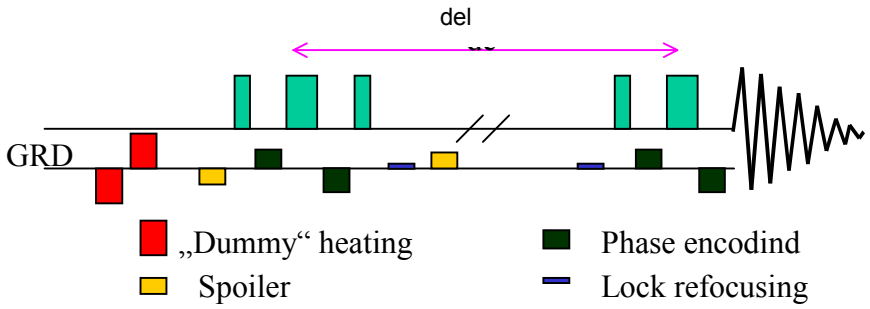


Figure 32. Oneshot DOSY Experiment

The lock refocusing gradient is determined by `kappa` and `gzlvl1`. The dummy heating gradients are automatically adjusted by the sequence. For the maximum gradient power available in the experiment, use:

```
gzlvl_max > gzlvl1*(1+kappa)
```

Table 13. Oneshot DOSY Parameters

Parameters	Function
calibflag	Correct systematic errors in DOSY experiments.
DAC_to_G	Store gradient calibration value in DOSY sequences.
del	Actual diffusion delay.
delflag	`y` runs the Doneshot sequence. `n` runs the normal s2pul sequence.
fn2D	Fourier number to build up the 2D display in F2.
gstab	Gradient stabilization delay (~200-300 µs).
gt1	Total diffusion-encoding pulse width.
gt3	Spoiling gradient duration (sec).
gzlvl1	Diffusion-encoding pulse strength.
gzlvl3	Spoiling gradient strength.
gzlvl_max	Maximum gradient strength accepted. (32767 with Performa II or III, 2047 with Performa I)
kappa	Unbalancing factor between bipolar pulses as a proportion of gradient strength (recommended:~0.2).
phasecycleflag	Flag to turn on or off the phase cycle.

*Dbppsteinept (**D**OSY **B**ipolar **P**ulse **P**air **St**imulated **E**cho **INEPT**) Experiment*

This sequence uses the higher “resolving power” of the wide ^{13}C chemical shift range, while the phase encoding and decoding step is more effectively done on the ^{1}H magnetization.

Processing 2D-DOSY Experiments

After DOSY data has been acquired, it must be processed to give a 2D DOSY spectrum. Processing data involves the following stages:

Table 14. Dbppsteinept Parameters

Parameters	Function
`calibflag`	Correct systematic errors in DOSY experiments.
DAC_to_G	Store gradient calibration value in DOSY sequences.
`del`	Actual diffusion delay.
`delflag`	`y` runs the Dbppsteinept sequence. `n` runs the normal INEPT sequence.
`fn2D`	Fourier number to build up the 2D display in F2.
`gt0`	Gradient duration of the spoiler gradient (for `satflag='y'`).
`gt1`	Total diffusion-encoding pulse width.
`gzlvl0`	Spoiling gradient power.
`gzlvl1`	Diffusion-encoding pulse strength.
`j1xh`	One-bond H-X coupling.
`mult`	Carbon multiplicity. 2 selects CHs (doublets). 3 gives CH2s down and CHs CH3s up. 4 enhances all protonated carbons.
`pp`	90° hard ^{1}H pulse.
`pplvl`	Decoupler power level for pp pulses.
`satflag`	`y` gives a grad-90(X) grad sequence during the diffusion delay to destroy the X-magnetization not originated from INEPT transfer.

1. Reference deconvolution with the command `fiddle` (optional, but useful if there is a suitable reference line that slowly diffuses).
2. Baseline correction with the macro `fbc` (also optional, but strongly recommended).
3. Extraction of diffusion data from the spectra and synthesis of a 2D DOSY spectrum with the macro `dosy`.

fiddle

The `fiddle` program enables you to use reference deconvolution to correct the line shapes, frequencies, phases, etc. of the signals caused by instrumental imperfections. Full instructions for its use are given in section 6.2, “Deconvolution,” on page 207 and in the *Command and Parameter Reference* manual. Reference deconvolution of DOSY spectra removes systematic errors caused by the disturbance of the magnetic field and field/frequency lock caused by gradient pulses. It is best to use `fiddle` with the `writefid` option to store the corrected data, recall the corrected data, and set all the weighting parameters to `n` before Fourier transforming and proceeding to the next step.

fbc

The `fbc` macro applies `bc` type baseline correction to all spectra in an array. Use the partial integral mode to set integral regions to include all significant signals. Leave blank as large an area of baseline as you possible can; this minimizes systematic errors in diffusion coefficient fits caused by baseline errors.

dosy

This macro uses the commands `dll` and `fp` to determine the heights of all signals above the threshold defined by the parameter `th`. Then it fits the decay curve for each signal to a Gaussian using the program `dosyfit`, storing a summary of all diffusion coefficients and their estimated standard errors and various other results as follows:

In the directory `$HOME/vnmrsys/Dosy`:

- diffusion_display.inp
- general_dosy_stats
- calibrated_gradients
- fit_errors
- diffusion_spectrum

In the current experiment:

- A second copy of diffusion_display.inp.

The `dll` program is limited to handling 512 lines, so very crowded spectra might need to be processed in sections by appropriately choosing `sp` and `wp`. `dosy` then runs the command `ddif` to synthesize the 2D DOSY spectrum.

The peak representation and the accuracy of the peak heights might increase with higher digital resolution, i.e., zero-filling the FIDs (`fn>np`) can occasionally improve the quality of the DOSY data. In extreme cases, even `fn=512k` is allowed by the software. Building up a 2D data set (and a 2D display) with this data size would not make sense; therefore a new parameter, `fn2D` (with a maximum limit of 64k), is introduced in the 2D-DOSY sequences, replacing `fn` when setting up the 2D display.

Note: The 2D DOSY display is set up in the same experiment where the data processing takes place.

The synthesized spectrum contains `fn1/2` traces in the diffusion domain (`f1`), and `fn2D` real data points in the spectral domain (`f2`); `fn1` is limited to the range 128-1024. Normally `fn2D` of 16k suffices. If `fn2D*fn1` is too large, spectral synthesis and display will be slow and/or you might run out of disk space.

Note: After displaying a 2D spectrum, the variable `ni` is set to `fn1/2` (this setting is required by `dconi`). So if more data is to be acquired or the sequence is to be displayed (`dps`) you must set `ni` back to zero.

By default, `dosy` uses all the experimental spectra and covers the whole diffusion range seen in the experimental peaks. Either one or three arguments, shown in Table 15, can be supplied to `dosy` to change the defaults.

Table 15. `dosy` Commands

Command	Function
`dosy('prune')`	Start a dialog to allow one or more spectra to be omitted from the analysis.
`dosy(d1,d2)`	`d1` and `d2` are numbers causing the diffusion range of the synthesized spectrum to be limited to $d1*10^{-10}$ m^2/sec and $d2*10^{-10}$ m^2/sec.
`dosy('prune',d1,d2)`	Combine the previously described arguments.

The message `Systematic Gz deviations` indicates that the random errors in the data are sufficiently small. It might be worthwhile correcting for the small systematic errors in the field gradients, produced by the spectrometer hardware, by using the decay curves of selected signals to provide an internal calibration of the relative gradient strengths. To correct for systematic gradient errors, do the following procedure:

1. Set the display/threshold parameters to select a few strong, well-resolved signals, which are known to arise from single species (i.e., the signals are not composites of overlapping signals from species with different diffusion coefficients). Enter `dosy`

to perform the analysis a first time, readjust the display and threshold to contain all the signals of interest.

2. Enter **`undosy calibflag='y' dosy`**. The second analysis uses the shapes of the decay curves in the first analysis to correct for systematic errors. Remember to set `calibflag` back to `'n'` if you wish to stop using the internal gradient calibration.

WARNING: **If the argument `prune` was used for the initial run of the `dosy` macro, you must ensure that the same increments are deleted in the second run. Use `undosy calibflag='y' dosy('prune')` and specify the same increment number(s).**

The two-dimensional DOSY display (and plot) is constructed by taking the bandshape of a given signal from the first (lowest gradient area) spectrum and convoluting it in a second dimension with a Gaussian line centred at the calculated diffusion coefficient and with a width determined by the estimated error of the diffusion coefficient obtained from the fitting process.

To extract spectra of the mixture components separated along the diffusion axis, select the region of interest using the two cursors in the interactive 2D display (`dconi`) mode and click on `Proj` (projection) and `Hproj (sum)` (horizontal projection). The spectrum can be plotted with the `Plot` menu.

When processing 2D DOSY spectra, you might find the Tcl-Tk process panel, shown in Figure 33, useful.

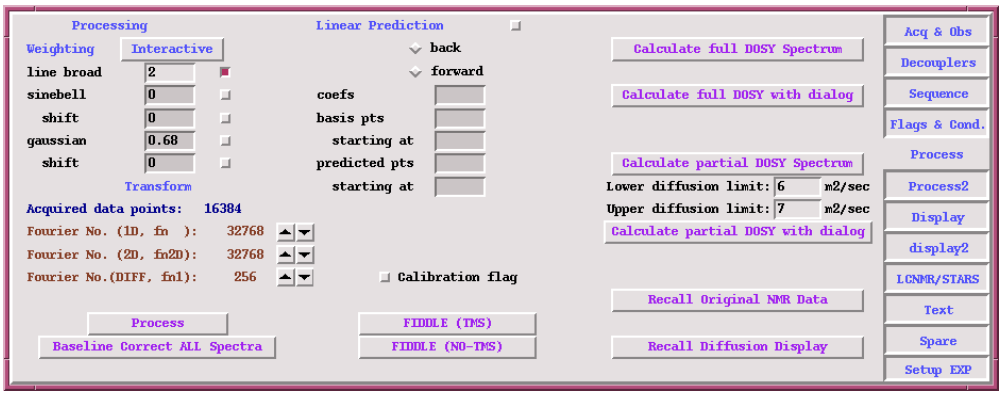


Figure 33. Tcl-Tk Process Panel for 2D_DOSY Pulse Sequences

WARNING: **Do not process the data with the `dosy` macro until the acquisition has been completed.**

sdp

The `sdp` macro (show diffusion projection) displays the integral projection of a DOSY dataset onto the diffusion axis. `sdp` uses the file `userdir+'/Dosy/diffusion_spectrum'` as input for the `sdp` command. Only use `sdp` in an experiment in which data can be overwritten because it modifies parameters such as `sw` and `at`.

3D-DOSY Experiments

3D DOSY adds a diffusion domain to “conventional” 2D experiments such as COSY or HMQC. The package contains sequences for DOSY-COSY (Dgcstecosy) and DOSY-HMQC (Dgcstehmqc), but it is easy to add diffusion encoding to many other 2D experiments. The 3D DOSY sequences provide better resolving power than the 2D counterparts (the probability of overlapping cross-peaks in 2D is much lower than the probability of overlapping lines in 1D proton detected experiments), at the expense of data size and experiment time.

An arrayed set of 2D experiments is performed using different values of gradient strength (`gzlvl1`). The data is doubly Fourier transformed, and the first 2D spectrum is used to manually define 2D integral regions. `dosy` analyzes then fits the integral volumes in successive increments to Gaussians and synthesizes 2D integral projections of the 3D dataset between defined diffusion limits. Full 3D display is not implemented; although with patience, you can achieve a similar effect by performing a series of projections.

Setting Up 3D-DOSY Experiments

1. Make sure that the “conventional” parameters of the COSY / HMQC experiment, such as pulse widths, transmitter offset, spectral window are correctly set.
2. As with 2D DOSY, try to find suitable lower and upper bounds for the gradient strength `gzlvl1`. There is no need to run 2D experiments for this purpose; the first increment from a 2D run is normally adequate (`ni=1`).
3. In a COSY experiment with higher quantum filter (`qlvl`>1), the first increment does not contain signals. Set the incremented delay (`d2`) to 0.05-0.1 during the gradient optimization process. Set `d2` back to zero when starting the DOSY-COSY experiment.
4. Use the `setup_dosy` macro to set up an array of 5 to 10 different `gzlvl1` values.

   The full 3D experiments then can be acquired. Note the total experiment time when choosing the number of `gzlvl1` values, `ni` and `nt`. Both sequences are equipped with a Tcl-Tk acquisition panel.

*Dgcstecosy (**D**OSY **G**radient **C**ompensated **S**timulated **E**cho **COSY**) Experiment (AV mode)*

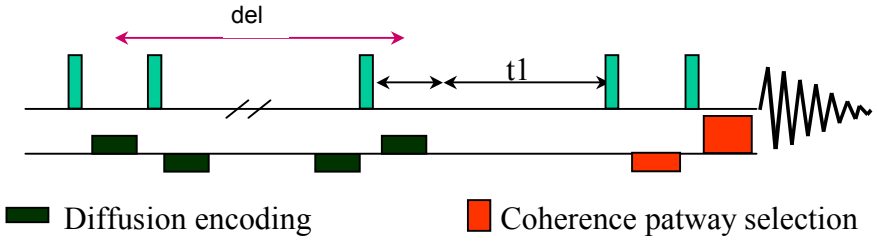


Figure 34. Dgcstecosy (AV Mode) Experiment

Using the `('t2dc')` argument to `wft2d` can be useful.

Table 16. Dgcstecosy Parameters

Parameters	Function
calibflag	Correct systematic errors in DOSY experiments.
DAC_to_G	Store gradient calibration value in DOSY sequences.
del	Actual diffusion delay.
gt1	Total diffusion-encoding pulse width.
gzlvl1	Diffusion-encoding pulse strength.
gstab	Gradient stabilization delay (~200-300 us).
tweek	Tuning factor to limit eddy currents. (can be set between 0 and 1, usually set to 0.0).
gt2	Gradient duration for pathway selection.
gzlvl2	Gradient power for pathway selection.
qlvl	Quantum filter level (1= single quantum, 2=double quantum).

*Dgcstehmqc (**D**OSY **G**radient **C**ompensated **S**timulated **E**cho **HMQC**) Experiment (AV mode)*

Process N-type data with `wft2d(1)` to process the first 2D experiment.

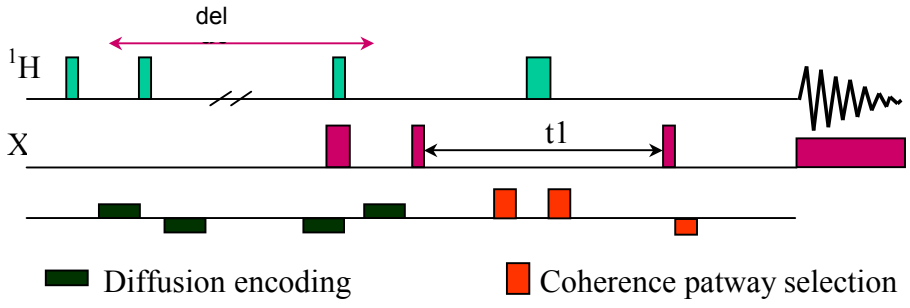


Figure 35. Dgcstehmqc Experiment (AV Mode)

Table 17. Dgcstehmqc Parameters

Parameters	Function
del	Actual diffusion delay.
gt1	Total diffusion-encoding pulse width.
gzlvl1	Diffusion-encoding pulse strength.
gt2	First coherence pathway selection gradient in HMQC.
gzlvl2	Gradient power for gt2.
gt3	Second coherence pathway selection gradient in HMQC.
gzlvl3	Gradient power for gt3.
gt4	Refocusing gradient in HMQC.
gzlvl4	Gradient power for gt4.
gstab	Gradient stabilization delay (~200-300 us).
pwx	90° X pulse.
pwxlvl	Power level for `pwx`.
j	One-bond H-X coupling.
mbond	Flag to select multiple-bond correlation (HMBC).
taumb	Delay for magnetization transfer for `mbond='y'`.
c180	Flag to use the 180° X pulse a composite pulse.

If `gzlvl4` has the same sign as `gzlvl2` and `gzlvl3` (N-type selection), use `wft2d` to process the data.

If `gzlvl4` has opposite sign to `gzlvl2` and `gzlvl3` (P-type selection), use `wft2d('ptype',1)` to process the data. Using the (`'t2dc'`) argument to `wft2d` can be useful.

Before using the sequence for the first time the coherence pathway selected gradients needs to be calibrated for a given probe and gradient amplifier.

The choice of decoupling method in the DOSY-HMQC experiment is crucial, as even relatively low values of `dpwr` can cause sufficient convection currents to invalidate DOSY results. On UNITY*INOVA* systems equipped with a PPM module in the ^{13}C channel adiabatic decoupling schemes (WURST, STUD) is recommended.

Processing 3D-DOSY Experiments

In order to analyze 3D results, it is necessary to manually define the individual signal regions in the 2D spectrum.

1. 2D Fourier transform the first increment of the 3D data set (i.e., the increment with the lowest `gzlvl1` value), using proper weighting functions in both dimensions:
   ```
   wft2d(1) for COSY
   wft2d('ptype',1) for HMQC
   ```
2. Correctly set `vs2d` and `th`, then define the signal regions in the first spectrum using the standard `ll2d` command and its options (e.g., `'reset'`, `'volume'`, `'clear'`, `'combine'`) The options are easily accessible via the `dconi/Peak/Edit` menu set.
3. Include all the components of a given multiplet (cross-peak) in a single integral region, provided that there is no contamination by other signals. Grouping signals in this way maximizes the signal-to-noise ratio available for data fitting.

This step offers you the unique opportunity to exclude apparent spectral artifacts (t1-noise, decoupling sidebands, spurious peaks, etc. from the DOSY analysis.) Because the manual peak selection is probably the most boring and time-consuming step of the whole procedure, after it is completed, it is worth storing the file (using the command `ll2dbackup`) in the same directory where the FID is stored for later processing.

4. After the signal regions have been defined, enter the command `dosy`.

   The macro extracts the volume of each region for every value of `gzlvl1` (this step involves, among other things, as many 2D Fourier transforms as there are `gzlvl1` increments). `dosy` then fits the volumes as functions of `gzlvl1`, returning with a display in which each signal region is labelled with its diffusion coefficient (10^{-10} m^2/sec) and with its standard error in brackets. The coefficients are automatically displayed when the `dosy` macro is completed using the `label` facility of the `ll2d` command. Thus, 6.05(0.05) means a diffusion coefficient of $6.05*10^{-10}$ m^2/sec (+/- $0.05*10^{-10}$ m^2/sec). The 2D spectrum on which the display is based is that of the first 2D increment of the 3D experiment. A copy of the diffusion results is available from the file `userdir+/Dosy/diffusion_display_3D.inp`. This file contains three columns:

   - The peak number (as obtained by `ll2dmode='nynn'`)
   - The diffusion coefficient
   - The standard error

5. The display of diffusion coefficients as numbers on the screen can result in very crowded display. You can change the type of information shown by using the `ll2dmode` parameter (for details see the *Command and Parameter Reference Manual* and Figure 36).

6. In order to make the analysis easier, use `sdp` to obtain the integral projection of the 3D data set onto the diffusion axis. You can use this diffusion spectrum to choose suitable diffusion regions for which to examine 2D projections of the 3D DOSY data.

WARNING: **Warning: Be sure to use `sdp` in an experiment in which data can be overwritten!**

7. In the experiment containing the 3D data, enter the command:

   `makeslice(d1,d2)`

   where `d1` and `d2` are the diffusion limits (in units of 10^{-10} m^2/s) between which the 2D projection of the 3D DOSY spectrum is required. The `makeslice` macro builds the slice and displays it after a few seconds. `makeslice` uses, among other things, the diffusion information in the file `userdir+'/Dosy/diffusion_display_3D.inp'`.

8. To return to the original spectrum, enter `showoriginal`. This command reverts to the original 2D spectrum for the first value of `gzlvl1`.

9. You can repeat the sequence `makeslice` – `showoriginal` as needed with different diffusion values (or slice thicknesses), but you must use `showoriginal` in between the display of two slices.

Both sequences are equipped with a Tcl-Tk Process2 panel, shown in Figure 36, providing access to necessary functions and parameters to process 3D DOSY data.

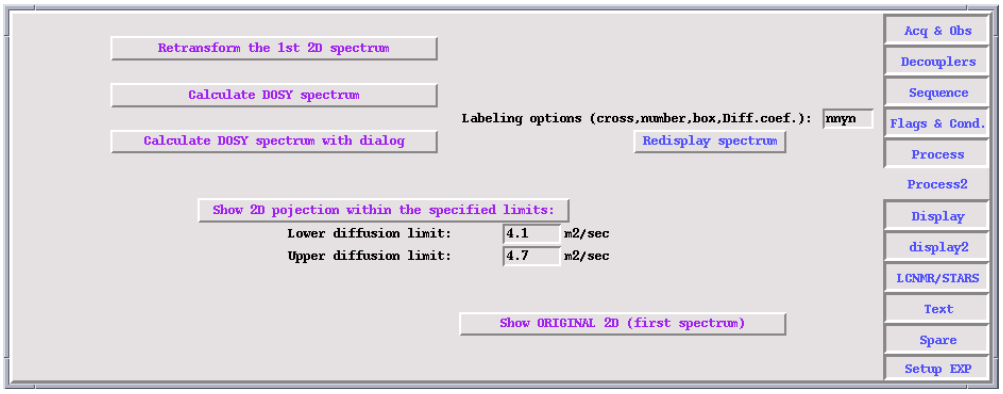


Figure 36. Tcl-Tk Process2 Panel for the 3D-DOSY Pulse Sequences

Sample FIDs to Practice DOSY Processing

The package includes a few 2D and 3D FIDs (in `/vnmr/fidlib/Dosy`) to practice DOSY processing. Except for Doneshot, every pulse sequence has an example. Processing Doneshot data is not unique; it requires the same procedure as the Dbppste or the DgcsteSL sequences.

Dbppste.fid

The sample is a mixture of three dipeptides (Phe-Val, Phe-Glu, and Phe-Gly) and 3 (trimethylsilyl)-1-propane-sulfonic acid dissolved in D_2O.

1. Load the data file into the experiment:
   `cd('/vnmr/fidlib/Dosy') rt('Dbppste.fid')`
2. Enter **`ft`** and adjust the phase of the first spectrum.
3. Set the cursor to the TSP singlet and enter **`nl rl(0)`**.
4. Change to a directory (`cd`) in which you have write permission.
5. Set the cursors 30 Hz either side of the TSP singlet, set **`lb='n'`** and **`gf=0.75`**, and enter the command
   `fiddle('satellites','TMS','writefid','temp')`
   to perform reference deconvolution on all the data, regularizing the lineshapes so that the peak heights in successive spectra accurately reflect the signal integrals.
6. Recall the FID created in the previous step and retransform it:
   `rt('temp') gf='n' lb='n' ft`
7. The integral regions have already been set in the supplied parameters. Display the integral to see where the resets have been positioned.
8. Enter **`fbc`** to perform baseline correction.
9. Set the threshold below the peaks of interest:
   `(vs=500, th=3)`
10. Enter **`dosy`**.

11. To zoom into the diffusion region of interest, enter **`undosy dosy(4,7)`**.

The following examples describe how to process DOSY data in the command mode (Commands column) or by using the Tcl-Tk Process (2D-DOSY) or Process2 (3D-DOSY) panels (Buttons in the Process Window column). The middle column has instructions or comments about both types of operations.

DgcsteSL.fid

The sample is a mixture of adenosine mono-, di-, tri-phosphate (AMP, ADP, ATP) and K_2HPO_4 in D_2O (pH=7). The data was acquired in a 3mm probe with direct ^{31}P observe.

Commands	*Comments, Instructions for Both*	*Buttons in the Process Window*

	Recall the FID:
	`cd('/vnmr/fidlib/Dosy')`
	`rt('DgcsteSL.fid')`	

`lb=2 wft`	Fourier transform	**Process**
`fbc`	Do baseline correction	**Baseline correct All spectra**
`dosy`	Execute dosy.	**Calculate full DOSY spectrum**
	To have better diffusion resolution, calculate a partial dosy spectrum:	
`undosy`		**Recall original NMR data**
`dosy(6.1,7.1)`		**Calculate partial DOSY spectrum**
	To display (and plot) the diffusion spectrum, join another experiment and execute `sdp`.	

Dbppsteinept.fid

The sample is a mixture of sucrose, methyl-alfa-D-glucopyranosid, 1,3,5,-O-methylidene-mio-inosytol, and dioxane (as internal reference) in D_2O. The experiment was run using an AutoSwitchable gradient probe.

Command	*Comments, Instructions for Both*	*Buttons in the Process Window*

	Recall the FID:
	`cd('/vnmr/fidlib/Dosy')`
	`rt('Dbppsteinept.fid')`	

`ft`	Fourier transform	**Process** (`unset lb` and `gf`)
`fbc`	Do baseline correction	**Baseline correct All spectra**
`lb=-0.4 gf=0.7`	Set weighting functions. Expand the spectrum and put the two cursors around the most intense line dixonane ± 15 Hz.	(Activate `lb` and `gf`)
`fiddle`	Execute fiddle.	**FIDDLE (No TMS)**
	Display full spectrum and set threshold.	
`dosy`	Execute dosy.	**Calculate full DOSY spectrum**
	To have better diffusion resolution, calculate a partial dosy spectrum:	
`undosy`		**Recall original NMR data**

Command	*Comments, Instructions for Both*	*Buttons in the Process Window*
`dosy(2.0,5.0)`		**Calculate partial DOSY spectrum**
	To display (and plot) the diffusion spectrum, join another experiment and execute `sdp`.	

Dgcstecosy.fid

The sample is a mixture of sucrose, methyl-alfa-D-glucopyranosid, and 1,3,5,-O-methylidene-mio-inosytol in D_2O. The experiment was run using an AutoSwitchable gradient probe.

Command	*Comments, Instructions for Both*	*Buttons in the Process2 Window*

	Recall the FID:
	`cd('/vnmr/fidlib/Dosy')`
	`rt('Dgcstecosy.fid')`	

`wft2d(1)`	Fourier transform.	**Retransform the 1st 2D spectrum**
	Signal regions for this file have already been saved. Recall `112d` file: `112d 112d('read' 'Dgcstecosy.112d')`	
`112dmode='nnyn'`	Check preset regions. Each cross peak of interest is boxed.	**Labelling options: nnyn**
`dconi`		**Redisplay spectrum**
`dosy`	Execute dosy.	**Calculate DOSY spectrum**
	When ready, COSY spectrum is displayed again with each cross peak labelled by its diffusion coefficient and its error.	
	Join another experiment and display diffusion projection: `sdp`	
	A set of signals appear:	
	4.1-4.8–1,3,5,-O-methylidene-mio-inosytol	
	3.6-3.9–methyl-alpha-D-glucopyranosid	
	2.8-3.1–sucrose	
	Other three lines between 3.2 and 3.6 D (10^{-10} m^2/sec) are overlapping diagonal peaks.	
	Rejoin DOSY experiment.	
`112dmode='nnnn'`	Reset peak labels.	**Labelling options: nnnn**
`dconi`		**Redisplay spectrum**
	To display inosytol spectrum:	

Command	*Comments, Instructions for Both*	*Buttons in the Process2 Window*
`makeslice(4.1,4.8)`		**Low Lim.: 4.1, Up. Lim: 4.8** **Show 2D projection within . . .**
	To display glucopyranosid projection, recall original 2D.	
`showoriginal`		**Show original 2D (first spectrum)**
`makeslice(3.6,3.9)`	Display glucopyranosid. To display the last projection, recall the original 2D.	**Low. Lim.: 3.6, Up. Lim: 3.9**
`showoriginal`		**Show original 2D (first spectrum)**
`makeslice(2.8,3.1)`	Display sucrose.	**Low Lim.: 2.8, Up. Lim: 3.1** **Show 2D projection within . . .**

Note: By accident, this cosy spectrum was run with an unusual parameter setting (`sw<>sw1`). The setting was absolutely unintended and should not affect the DOSY processing. The operator (P.Sandor, Darmstadt) assures you that the sequence also gives proper results with adequate parametrization.

Dgcstehmqc.fid

The sample is a mixture of quinine, geraniol, and camphene (and TMS) in deutero-methanol. See *J. Magn. Reson*. (1998), **131:** 131-138.

Commands	*Comments, Instructions for Both*	*Buttons in the Process2 Window*

	Recall FID:
	
	`cd('/vnmr/fidlib/Dosy')`
	`rt('Dgcstehmqc.fid')`	

`wft2d('ptype',1)`	Fourier transform.
	
	Signal regions for this file have already been saved. Recall ll2d file:
	
	`ll2d`
	`ll2d('read','Dgcstehmqc.ll2d')`
		Retransform the first 2D spectrum

`ll2dmode='nnyn'`	Check preset signal regions. Each cross peak of interest is boxed	**`Labelling options: nnyn`**
`dconi`		**Redisplay spectrum**
`dosy`	Execute dosy.	**Calculate DOSY spectrum**

Commands	*Comments, Instructions for Both*	*Buttons in the Process2 Window*
	When ready, HMQC spectrum is displayed again with each cross peak labelled by its diffusion coefficient and its error. Join another experiment and display diffusion projection: `sdp` A set of signals appears: 7.0-8.5 – quinine 10.0-11.6 – geraniol 14.0-15.4 – camphene Other lines around 18 D (10^{-10} m^2/sec) are methanol and TMS. Rejoin DOSY experiment.	
`ll2dmode='nnnn'`		**Labelling options: nnnn**
`dconi`		**Redisplay spectrum**
	To display quinine spectrum:	
`makeslice(7.0,8.5)`		**Low. Lim.: 7.0, Up. Lim: 8.5** **Show 2D projection within . . .**
	To display another projection, you need to recall the original 2D.	
`showoriginal`		**Show original 2D (first spectrum)**
`makeslice(10.0,11.6)`	Display geraniol.	**Low. Lim.: 10.0, Up. Lim: 11.6** **Show 2D projection within . . .**
	To display last projection, recall original 2D.	
`showoriginal`		**Show original 2D (first spectrum)**
`makeslice(14.0,15.4)`	Display camphene.	**Low. Lim.: 14.0, Up. Lim: 15.4** **Show 2D projection within …**

DOSY-Related Literature

Morris, K.F.; Johnson, C.S., Jr. “Resolution of Discrete and Continuos Molecular Size Distributions by Means of Diffusion-Ordered 2D NMR Spectroscopy,” *J. Am. Chem. Soc.* (1993), **115:** 4291-4299.

Wider, G.; Dötsch, V.; Wütrich, K. “Self-Compensating Pulsed Magnetic-Field Gradients for Short Recovery Times,” *J. Magn. Reson.* (1994), **108 (Series A):** 255-258.

Barjat, H.; Morris, G.A.; Smart, S.; Swanson, A.G.; Williams, S.C.R. “High-Resolution Diffusion-Ordered 2D Spectroscopy (HR-DOSY) – A New Tool for the Analysis of Complex Mixtures,” *J. Magn. Reson.* (1995), **108 (Series B):** 170-172.

Wu, D.; Chen, A.; Johnson, C.S.,Jr. “An Improved Diffusion-Ordered Spectroscopy Experiment Incorporating Bipolar-Gradient Pulses,” *J. Magn. Reson.* (1995), **115 (Series A:** 260-264.

Gozansky, E.K.; Gorenstein, D.G. “DOSY-NOESY: Diffusion-Ordered NOESY,” *J. Magn. Reson.* (1996), **111, (Series B):** 94-96.

Wu, D.; Chen, A.; Johnson, C.S.,Jr. “Three-Dimensional Diffusion-Ordered NMR Spectroscopy: The Homonuclear COSY-DOSY Experiment,” *J. Magn. Reson.* (1996), **121 (Series A):** 88-91.

Wu, D.; Chen, A.; Johnson, C.S.,Jr. “Heteronuclear-Detected Diffusion-Ordered NMR Spectroscopy through Coherence Transfer,” *J. Magn. Reson.* (1996), **123 (Series A):** 215-218.

Jerschow, A.; Müller, N. “3D Diffusion-Ordered TOCSY for Slowly Diffusing Molecules,” *J. Magn. Reson.* (1996), **123 (Series A):** 222-225.

Birlikaris, N,; Guittet, E. “A New Approach in the Use of Gradients for Size-Resolved 2D-NMR Experiments,” *J. Am. Chem. Soc.* (1996), **118:** 13083-13084.

Jerschow, A.; Müller, N. “Suppression of Convection Artifacts in Stimulated Echo Diffusion Experiments. Double-Stimulated-Echo Experiments*,” J. Magn. Reson.* (1997), **125:** 372-375.

Barjat, H.; Morris, G.A.; Swanson, A.G., *“*A Three-Dimensional DOSY-HMQC Experiment for the High-Resolution Analysis of Complex Mixtures,” *J. Magn. Reson.* (1998) **131:** 131-138.

Pelta, M.D.; Barjat, H.; Morris, G.A.; Davis, A.L., Hammond, S.J. “Pulse Sequences for High Resolution Diffusion-Ordered Spectroscopy (HR-DOSY),” *Magn. Reson. Chem.* (1998), **36:** 706.

Tillett, M.L.; Lian, L.Y.; Norwood, T.J. “Practical Aspects of the Measurement of the Diffusion of Proteins in Aqueous Solution.,” *J. Magn. Reson.* (1998), **133:** 379-384.

Gounarides, J.S.; Chen, A.; Shapiro, M.J. “Nuclear Magnetic Resonance Chromatography: Applications of Pulse Field Gradient Diffusion NMR to Mixture Analysis and Ligand-Receptor Interactions,” *Journal of Chromatography B* (1999), **725:** 79-90.

DOSY Review Papers

Morris, G.A.; Barjat, H., “High Resolution Diffusion Ordered Spectroscopy,” *Methods for Structure Elucidation by High Resolution NMR,* ed. K. Kövér, Gy. Batta, Cs. Szántay, Jr. (Amsterdam: 1997), pp. 209-226.

Morris, G.A.; Barjat, H.; Horne, T.J. “Reference Deconvolution Methods (FIDDLE),” *Progress in Nuclear Magnetic Resonance Spectroscopy* (1997), **31:** 197-257.

Johnson C.S. Jr., “Diffusion-Ordered Nuclear Magnetic Resonance Spectroscopy: Principles and Applications,” *Progress in Nuclear Magnetic Resonance Spectroscopy* (1999), **34:** 203-256.

9.7 Filter Diagonalization Method (FDM)

This section contains the following:

Filter Diagonalization Method (FDM) is a non-Fourier data processing method that extracts spectral parameters (peak positions, line widths, amplitudes, and phases) of Lorentzian lines directly from the time-domain signal by fitting FID data to a sum of damped complex sinusoids. The spectral parameters (saved in `curexp/datdir/fdm1.parm`) are also called “line list” and are used to construct an “ersatz” spectrum of the NMR data.

FDM is slower than Fast Fourier Transform, but it offers better resolution in the case of truncated signals and the option of processing only a selected spectrum region. FDM has the potential to work well with corrupted data, and the potential to produce a line list with each line represents a true NMR peak.

FDM reads input parameters from a file created by the `fdm1` macro, using default (optimal) values. You can change any of the parameters from the command line. Table 18 lists `fdm1` parameters. If the spectrum is not referenced with `rl`, the reference `rfl` is also read from `curpar`. The section "Changing Local FDM Variables," page 169 describes how you can override the default setting.

Table 18. fdm1 Parameters

Parameter	Description
`cheat`	No cheat if `cheat=1`, lines are narrower if `cheat<1`.
`cheatmore`	No cheatmore if `cheatmore=0`.
`error`	Error threshold for throwing away poles.
`fdm`	1 for `fdm`, –1 for `dft`.
`Gamm*`	Smoothing width (line broadening).
`Gcut`	Maximum width for a pole.
`idat`	–4 for ASCII complex FID file, –5 for FID file.
`kcoef`	`kcoef>0`; use “complicated” `dk(k)`. –1 always preferred.
`Nb*`	Number of basis function in a single window.
`Nbc*`	Number of coarse basis vectors.
`Nsig*`	Number of points to use, 3000 is ok.
`Nskip*`	Number of points to skip.
`rho`	`rho=1` is optimal.
`ssw`	A test parameter.
`t0`	Delay of the first point.
`theta`	Overall phase of FID (`rp` in radians).
`wmin`	Minimum spectrum frequency in hertz.
`wmax`	Maximum spectrum frequency in hertz.

* Global; see "FDM Global Parameters," page 168 for more information.

In most cases, you only need to decide the number of data points to be used and the spectrum window to be processed. By default, half of the FID data or 3000 data points, whichever is smaller, is used.

Using FDM

The following steps describe how to do normal activities such as phasing, zooming in, zooming out, and processing a spectrum window with the `fdm1` macro.

1. Display the FID data and use the right mouse button to select the data points to be used by FDM.
2. Process the data with `ft` (it uses all FID points), then display and reference the spectrum.
3. Place the cursor on a region of interest, zoom in on it, then type `fdm1` or select the **Process2** panel and click on **1D FDM** to process the data. If you select but do not zoom in on a region, the whole spectrum in display is processed.

   A new menu appears with **Stop FDM** and **Display** buttons. The calculation might take a few seconds to a few minutes depending on the number of data points used and the size of spectrum window to be processed. To abort the process, click on **Stop FDM**. To check if the process is finished, click on **Display**. If the process is finished, display the spectrum.

Reprocessing a Spectrum

The **1D FDM** button is displayed on the **Process2** panel. Use this button to reprocess a spectrum.

Changing Parameters

Relevant `fdm1` global parameters are displayed on the **Process2** panel with current values. You can change these parameters. The value of a global parameter is saved to `curpar` and it remains the same until you change it from the parameter panel or make a new assignment using the command line. You can also change the parameters from the `fdm1` command line as described in the section "Changing Local FDM Variables," page 169.

FDM Global Parameters

The following FDM parameters are global.

- `Nsig` is the number of FID points to use. You initialize it with the right mouse button position (`crf` + `deltaf`) `*sw`. If `Nsig=0`, half of the FID data points or 3000, whichever is smaller, is used. `Nsig` can be changed from the parameter panel, the command line `Nsig=nnnn`, the right mouse button (when the FID is displayed), or the command line `fdm1('Nsig',nnnn)`. In general, the more peaks you have, the more data points it takes to fit the spectrum. To check the reliability of the FDM method, change `Nsig` a few times and reprocess the data to see if you get the same result.
- `Nskip` is the number of data points to skip at the beginning of a FID. By default, zero points are skipped. In some cases, you can improve baseline by skipping the first one or two points.
- `Nb` is the number of basis functions (poles) used to fit each of the windows in an FDM calculation. The default is 10. FDM breaks down the specified window into smaller

windows. In general, bigger Nb gives better results, especially better baseline. Sensible values for Nb are between 10 and 50.

- Nbc is the number of additional poles (coarse basis functions) to be used. The default is zero, but setting Nbc to an integer larger than zero (typically 4-10) might improve the baseline.
- Gamm is the smoothing width (line broadening). The default is 0.2*sw/Nsig, which is about a tenth of the FT resolution. Typical values are 0.1 to 1.0.

Using bigger Nsig, Nb, Nbc, or a spectral window significantly slows down the calculation.

Changing Local FDM Variables

FMD parameters that are not commonly used are set as fdm1 local variables. These parameters are listed with global parameters in Table 18. You can change local variables only from the fdm1 command line. Parameter values are lost after the completion of the macro. To use a value again, you must reenter it; otherwise, fdm1 sets the value to the default. To change more than one local variable, enter the variables from the same command line.

You can change any of the FDM parameters from the fdm1 command line and you can change both global and local variables. Values entered from the fdm1 command line override the default, the change from the **Process2** panel, and the value that you select with the cursor. Enter command line arguments by giving the parameter name in single quotation marks and a value separated by a comma, for example:

```
fdm1('cheat',0.8)
fdm1('Nsig',3000)
fdm1('Nsig',3050)
fdm1('Nb',20)
fdm1('Nbc',10,'Nb',20)
fdm1('Nsig',3000,'Nb',20,'Gamm',0.5)
fdm1('wmin',-1600,'wmax',1600)
fdm1('wmin',-1600,'wmax',0)
```

cheat is a factor multiplied to the line width. There is no cheat when cheat=1; lines are narrower when cheat<1.

wmin is the minimum spectrum frequency in Hz. The default is sp+rfl-sw/2. wmin is the upper field.

wmax is the maximum spectrum frequency in Hz. The default is wmin+wp. wmax is the lower field.

The center of the full spectrum is zero.

Seeing Parameter Values

Parameters are set to their default values. Normally, you do not need to change these parameters or you might change some of the global parameters. You cannot inquire values of local fdm1 parameters in the same way that you inquire global parameters with echo or ?. To see the values of all parameters used, look in the fdm1.inparm file created by the fdm1 macro in the datdir directory of the current experiment. Figure 37 shows the format of the fdm1.inparm file; the number of spaces and tabs is arbitrary.

```
fid_filename              idat
t0              theta
fdm
par
fn_Sp1D     spectype    axis
wmin            wmax
Nsig
Nskip
rho             Nb
error
Npower          Gamm            Gcut
cheat           cheatmore
Nbc     kcoef
ssw    sw
fidmt    specfmt
i_fid
```

Figure 37. `fdm1.inparm` File

FDM References

J. Chen and V. A. Mandelshtam, *J. Chem. Phys.* (2000) **112:** 4429-4437.

V. A. Mandelshtam, *J. Magn. Reson.* (2000) **144:** 343-356.

A. A. De Angelis, H. Hu, V. A. Mandelshtam and A.J. Shaka, *J. Magn. Reson.* (2000) **144:** 357-366.

Chapter 10. Multidimensional NMR

Sections in this chapter:

In some respects, 2D NMR is similar to an arrayed 1D experiment. In both, as a function of time (one of the time variables in the pulse sequence), we obtain a series of FIDs that we then transform to become a series of spectra. For 2D experiments, however, the times for each experiment are *not* explicitly specified. Instead, two new parameters are used: `sw1`, which describes our “2D” spectral width (to be discussed shortly), and `ni`, the number of increments, which sets the number of different experiments we will do. The implicit time variable will then be updated from experiment to experiment as determined by `sw1`.

Real-Time 2D

Real-time 2D performs 2D actions while the experiment is still in progress. Once eight or more increments have been completed, you can perform the full 2D transform on the data that exists up to that point.

For some experiments, such as heteronuclear chemical shift correlation and homonuclear 2D-J experiments, you will be surprised at how few increments are needed to resolve the resonances of interest. Others may require more increments. In any case, if you find you have sufficient data to solve the problem, you can always stop the experiment, so that the remaining increments are not performed, and proceed to the next problem.

Interferograms

Once the data are obtained and transformed along the acquisition dimension, we have a series of spectra. If this were a 1D arrayed experiment, like an inversion-recovery T_1 experiment, we would see that the peak heights behave exponentially as a function of time.

In 2D experiments, however, the peaks heights will oscillate as a function of time, and that oscillation is the information of interest. To unravel this information, we first transpose the matrix to form a series of *interferograms*.

Each interferogram contains a series of points that represent the peak height at a particular frequency in the original spectrum as a function of time. Of course, most of these interferograms contain only noise, because many of the frequencies in the original spectrum also contained noise. However, some interferograms, namely those corresponding to the peaks in the original spectrum, contain useful information.

The time that is varied in a 2D experiment is known as the *evolution time* or t_1, because it is the first of two key time periods in the 2D experiment. Evolution time is controlled by the parameter `d2`. This time is normally calculated by setting the number of increments to the value of the parameter `ni` and the increment value to `1/sw1`. The value of `ni` determines if a 2D experiment will be run. Initially, `d2` can be set to any value but is usually set to zero.

The `d2` array does not appear in the display `da` (i.e., `d2` is "implicitly" arrayed). Only the first value of `d2` appears as the parameter value in the display `dg`. A minimum of eight increments must be used to do a 2D transform. Typical range is 32 to 512.

The time during which the signal is detected is known as the *detection time* or t_2, because it is the second of the two key time periods. After transform of the signals detected during the time t_2, the "normal" spectrum appears along the f_2 axis. The second transform reveals information about the frequencies of oscillations during the t_1 time period along the f_1 axis.

Many parameters that refer to the new f_1 axis in a 2D experiment are identified by the number 1 (e.g. `sw1`, `lb1`, `fn1`), whereas the normal 1D parameters control f_2.

The process of transformation, transposition to interferograms, and second transformation may seem complicated, however, it can all be reduced to literally a single command, or even a single menu choice, that starts an acquisition of a 2D experiment and performs all the necessary processing when the experiment is done. So the process can be fairly simple.

Phase-Sensitive 2D NMR

Just as in 1D NMR, phase-sensitive processing and display offers better sensitivity and resolution. Phase-sensitive 2D NMR by itself is simply the ability to display and plot *phased data*, as opposed to absolute-value data. There are four kinds of experiments in which a user might want to examine phase-sensitive data:

- A 2D experiment in which the data are not expected to appear in absorption mode in both directions, but in which it is nonetheless desirable to observe the data in a phase-sensitive presentation.
- A 2D experiment in which the data, processed in a suitable way, *are* expected to appear in absorption mode in both directions. Heteronuclear 2D-J is such an experiment.
- An experiment in which two different experiments are performed for each value of t_1, typically using different phase cycles, producing a full complex data set for the second transformation. We shall refer to this method, popularized by States, Haberkorn, and Ruben (*J. Magn. Reson.* **1982**, *48*, 286), as the *hypercomplex method*
- An experiment in which the phase of the excitation pulse is updated as a function of t_1 (TPPI or Time Proportional Phase Incrementation, see Marion and Wuthrich, *Biochem. Biophys. Res. Commun.* **1983**, *113*, 967) and which produces real data along the t_1 axis.

Complex transforms are usually performed along t_1, which is the ideal situation for the hypercomplex method. TPPI data can be processed along t_1 with either a complex FT or a real FT, depending upon the method of data collection.

In general, the hypercomplex method is the method of choice. A natural first reaction, since this method requires *two* data tables instead of one, is to assume that it requires twice as much storage as the TPPI method. This is untrue, however, for the same reason that a real 1D transform covering a given spectral width requires exactly as much data as a complex 1D transform— the sampling rate must be twice as high in the real case to produce the same result. In the same way, the TPPI method requires only one data table, but requires sampling to occur twice as frequently along t_1, thereby incurring twice the data size per data table to produce the same real resolution. So in this sense the two experiments are equivalent in data storage requirements and experimental time.

10.1 2D Experiment Set Up

1. Use the Locator to list the available 2D experiments:
    a. Click the Locator Statements button (magnifying glass icon), and select Sort Protocols for experiments.
    b. Set the columns to name, apptype, and seqfil.
    c. Modify the Locator Statement to show hetero2d (heteronuclear 2D) or homo2d (homonuclear 2D).
2. Drag-and-drop or double-click the desired experiment.

10.2 Data Acquisition: Arrayed 2D

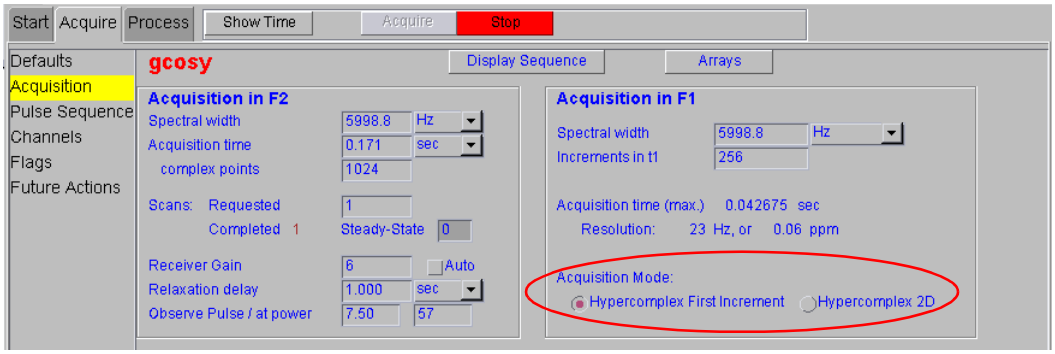
2D experiments have one implicitly "arrayed" parameter, `d2`. Like 1D experiments, however, 2D experiments can also have other parameters explicitly arrayed. This feature can be used, of course, for purposes that have nothing to do with phase-sensitive 2D, such as running a series of 2D-NOE experiments using different mixing times. As we shall see below in discussing the processing of such data, this feature alone opens up a variety of experiments, including addition/subtraction of two or more 2D experiments.

- "Hypercomplex Method," this page
- "TPPI Method," page 174

Hypercomplex Method

The hypercomplex method of phase-sensitive 2D NMR requires the use of two data tables. Appropriate pulse sequences must be created (see below for more details of this point) which, as a function of some parameter, generate a different sequence of pulses or pulse

phases suitable for generating the two component experiments of the hypercomplex method.



Any parameter may be used for this purpose. As a convention, we use the parameter `phase`, which takes on values of 0, 1, or 2:

- A value of `phase=0` can be used to produce a phase cycle suitable for a conventional (non-phase-sensitive) 2D experiment.
- Running instead an array of experiments with `phase=1,2` produces the two experiments suitable for the hypercomplex method.

You may ascertain the possible values of `phase` by reading the source code in the `psglib` directory for any particular pulse sequence.

TPPI Method

The TPPI method of phase-sensitive 2D NMR requires one data table when `phase=3`. The data must be processed along t_1 with a complex Fourier transform by setting `proc1` (which sets the type of data processing to be performed on the t_1 interferogram) to `'ft'`. This manner of implementing TPPI leads to a doubling of the f_1 frequency axis.

When an arrayed 2D experiment is run in this manner, there is in reality a double array: `d2` (the evolution time) and `phase`. The order of these arrays is such that the `phase` array is cycled the most rapidly, so that the order of the experiments is, for example:

Evolution Time	*phase*	*Method*
`d2=0`	`phase=1`	
`d2=0`	`phase=2`	States-Haberkorn
`d2=1/sw1`	`phase=1`	
`d2=1/sw1`	`phase=2`	
`d2=0`	`phase=3`	
`d2=1/sw1`	`phase=3`	TPPI (non-arrayed)
`d2=2/sw1`	`phase=3`	
`d2=3/sw1`	`phase=3`	

Not all pulse sequences have the TPPI method incorporated.

When an experiment is in progress, the acquisition status window displays a count of the current FID and the number of completed transients (`ct`) in that FID. As indeed happens with a 1D arrayed experiment, the current FID number is actually the *total* count of the completed FIDs to this point, including all arrays. Since the `phase` parameter is cycling the most rapidly, and since typically `phase` is an array of two values, the current FID number is typically *twice* the number of the current increment. For example, when the

counter reads FID 54, this means that 27 FIDs of the first type of experiment have been completed, 26 of the second type, and the system is working on the 27th experiment of the second type.

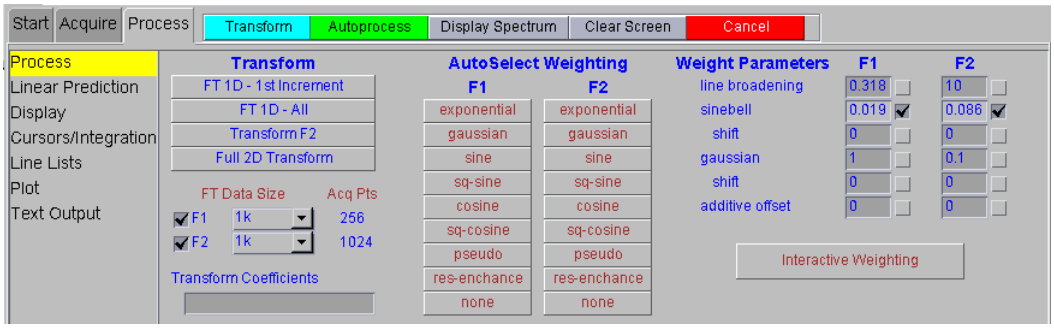
10.3 Weighting

This section describes weighting functions for processing 2D experiments (in the `ni` and `ni2` dimensions).

- "2D Weighting Parameters," this page
- "Interactive Weighting," page 175

2D Weighting Parameters

The 2D weighting parameters are set on the Process page in the Process panel. The 2D weighting parameters are analogous to weighting parameters for 1D experiments.



Parameters for 2D experiments are used for processing the t_1 domain (the interferograms) or first indirectly detected dimension (`ni`).

In **non-phase-sensitive (absolute-value and power) 2D experiments**, "pseudo-echo," sinebell, or sinebell-squared weighting is typically used to attenuate long dispersion tails. This weighting is often responsible for a significant loss in sensitivity in such 2D experiments.

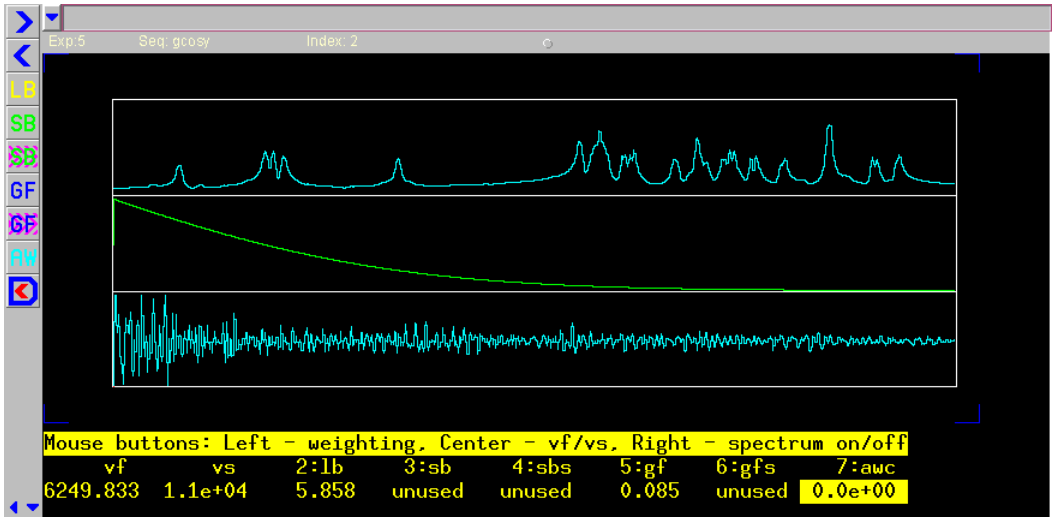
Weighting	*AutoSelect Weighting button*
pseudo-echo	pseudo
sinebell	sine
sinebell-squared	sq-sine

In **phase-sensitive 2D experiments**, the key in using weighting functions is to ensure that the weighted FID or interferogram decays to zero by the end to avoid "truncation wiggles." The Gaussian function is ideally suited for this; typical values might be `gf=0.6*at`, `gf1=0.6*ni/sw1` (`=0.6*at1`), which are entered in the **Weight Parameters** fields. Resolution enhancement (using negative line broadening) may be helpful in cases of spectral overlap, but can also be dangerous, since the "dips" that it can induce around the sides of peaks show up as peaks of opposite sign in the 2D plot, complicating analysis.

Interactive Weighting

The **Interactive Weighting** button on the **Process** page (or the `wti` command) allows interactive setting of weighting parameters for both t_2 FIDs and t_1 interferograms (both the

`ni` and `ni2` dimension). The currently active element or trace is used in adjusting the weighting parameters.



The following parameters are used with interactive weighting:

Graphics button	*Screen parameter*	*Function*
LB	`lb`	**Line broadening** -- line broadening factor, in Hz; a positive value gives sensitivity enhancement; a negative value gives resolution enhancement.
SB	`sb`	**Sinebell** -- sinebell time period, in seconds; a negative value gives a sine squared bell.
SB>>	`sbs`	**Sinebell shift** -- sinebell shift, in seconds; shifts the origin of the sinebell; active only if sinebell is active.
GF	`gf`	**Gaussian** -- Gaussian apodization constant, in seconds.
GF>>	`gfs`	**Gaussian shift** -- set the Gaussian function shift, in seconds. This shifts the origin of the Gaussian function; active only if Gaussian is active.
AW	`awc`	**Additive width correction** -- adjusts the additive weighting constant; added to the weighting function after the line broadening and sinebell (sinebell shift) contributions, but before the Gaussian (Gaussian shift).

The values displayed in the graphics display window correspond to the values displayed in the Weight Parameters fields on the Process page. Clicking a graphics control button toggles on and off the weighting function.

You can enter values in the fields next to the Weight Parameters, and check the box to activate the parameter. Press Return to enter the value.

The left mouse button also changes the selected parameters. The right mouse button turns off the spectrum for a faster response to changes in the weighting function.

10.4 Baseline and Drift Correction

Baseline and drift correction are done using the Linear Prediction and the Display pages under the Process panel.

Calculating the Preacquisition Delay

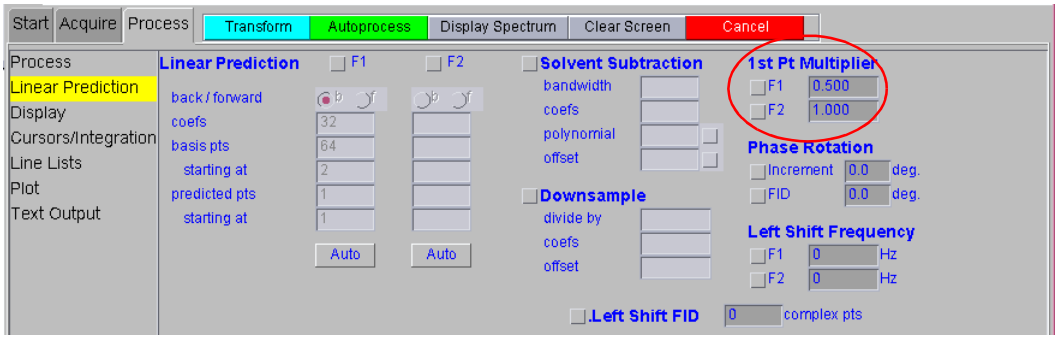
Unless first-order phase in the directly-detected dimension is approximately zero, the **F2** value **1st Pt Multiplier** will affect both the spectral drift correction (`dc`) offset and the curvature of the spectrum during 2D data processing. Delay values are shown on the Flags page under the Acquire panel.

1. Obtain a trial spectrum and phase it to pure absorption. This spectrum provides the current preacquisition delay and first-order phase values. Using these values, the `calfa` macro can calculate a new value for `alfa` so that `lp` is rendered approximately 0.
2. Enter **`calfa`** to calculate a new value for the preacquisition delay (`alfa`) so that `lp` is rendered approximately 0. If `dsp='i'` or `'r'`, calfa adjusts both `rof2` and `alfa`.

Calculating Receiver Gating Time

1. Obtain a trial spectrum and phase it to pure absorption. This spectrum provides the current `rof2` and `lp` values for `crof2`. The value of the `alfa` delay is left constant, provided `rof2` does not become less than 1 µs.
2. Enter **`crof2`** to calculate a new value for `rof2` (receiver gating time following a pulse) based upon the current `rof2` and `lp` (first-order phase) values, so that `lp` is rendered approximately 0.

First-Point Multiplier



The fields under **1st Pt Multiplier** (on the Linear Prediction page) multiply the first point of each FID by the **F2** value (the default value is 1.0, except that if the processing involves backward extension of the time-domain data with linear prediction, the default value is then 0.5) and the first point of each interferogram by the **F1** value (default value is 0.5) for the indirectly-detected dimension. **1st Pt Multiplier** attempts to compensate for the first point distortion caused by analog filters (see Otting, Widmer, Wagner and Wüthrich, *J. Magn. Reson.* **1986**, *66*, 187).

The effect of using the **F2** value in **1st Pt Multiplier** is to perform a linear baseline correction on all f_2 data, reducing negative-going ridges along f_2 in phase-sensitive 2D data.This correction is not needed in experiments such as COSY where the FID *starts* at zero and grows or in absolute-value mode presentation if pseudo-echo or sinebell processing is used, because the processing function goes to zero at t_2=0, forcing all FIDs to start at zero amplitude.

The best value of 1st Pt Multiplier is a function of the filter setting and should be determined empirically. It can be determined before, during, or after the 2D experiment by using **FT 1D -1st Increment** button on the Process page.

1. With a properly phased first increment spectrum on the screen, enter **`dc`**.
2. Position the mouse at the right edge of spectrum baseline (to keep track of the ideal baseline position).
3. Enter **`cdc`** and observe the new position of the baseline. It typically drops.
   - If the baseline goes negative, set the **F2 1st Pt Multiplier** to greater than 1.0 (try 1.5) and click **FT 1D -1st Increment** on the Process page.
   - If the baseline rises but does not return to the position indicated by the mouse arrow, increase the value of **F2 1st Pt Multiplier** and click **FT 1D -1st Increment** again. If in doing so the baseline rises above the ideal level, reduce value and try again.

   Only a few tries are required before the proper value of F2 1st Pt Multiplier is found.

Normally, no correction for 1st Pt Multiplier is necessary. Unchecking the boxes next to F1 and F2 disables the first-point multiplier feature. This would be the usual value for sinebell or pseudo-echo processing.

Baseline Correction

For baseline correction, an alternative to the use of 1st Pt Multiplier is using the **BC F1** and **BC F2** buttons on the **Display** page. Baseline correction in 2D processing uses the spline or second to twentieth order polynomial fitting of predefined baseline regions. These

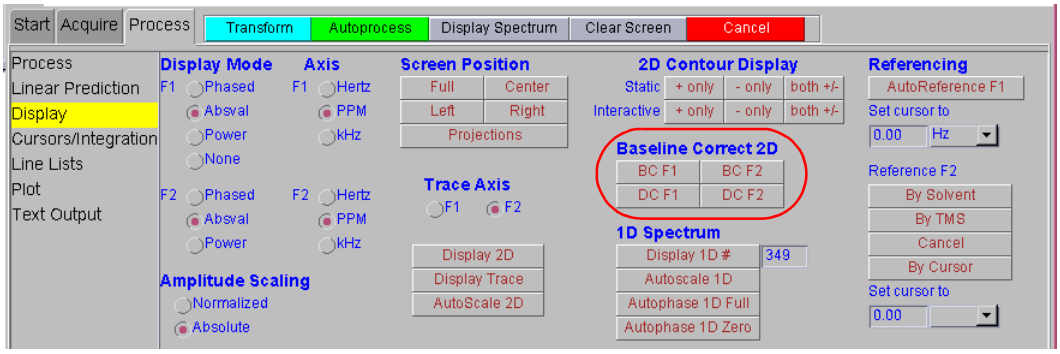
regions are set up prior to the use of `bc` by setting integral resets so that integrals appear only over regions of the spectrum with signals present. These can be set after a clicking **FT 1D - 1st Increment**. The quality of the baseline correction can be assessed by `bc(1)`. In setting baseline regions near the ends of the spectrum, the `bc` operation does essentially the equivalent of 1st Pt Multiplier because this represents a simple spectral drift correction.

FID Drift Correction

A `dc` offset in time-domain data transforms into a “center glitch” in the frequency spectrum. For 1D data, the `ft` program automatically applies a `dc` correction to the FID. Such a correction is not applied to 2D FIDs or interferograms unless explicitly requested. The command `ft2d('t2dc')` causes a `dc` correction to be applied to each t_2 FID before the first FT and `ft2d('t1dc')` causes a `dc` correction to be applied to each t_1 interferogram prior to the second FT. In both cases, the last one-sixteenth of the time-domain data is used to calculate the `dc` correction.

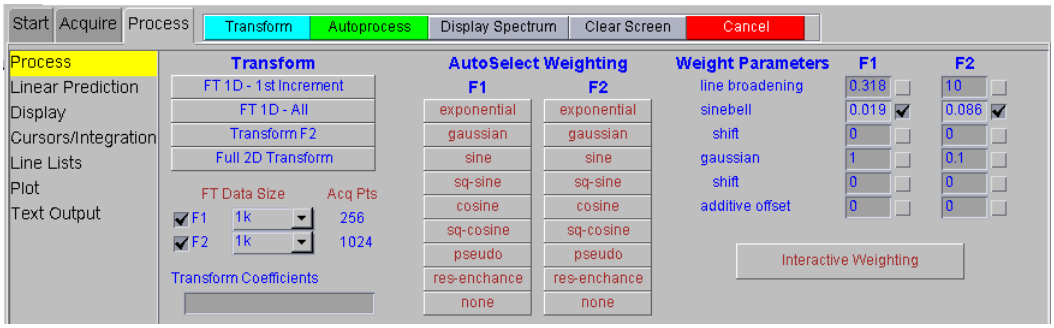
Spectral Drift Correction

Use the **DC F1** and **DC F2** buttons on the Display page only after the 2D transform. Use **DC F1** for corrections along f_1 and **DC F2** for corrections along f_2. The drift correction calculation is done separately for each trace in the 2D data set.



10.5 Processing Phase-Sensitive 2D and 3D Data

After 2D data has been acquired, the complete 2D transformation can be performed with one of the Transform buttons on the Process page.



A series of complex FIDs, obtained as a function of t_1, are transformed to become a series of spectra. Each spectrum consists of a real and imaginary part. Each spectrum is then phase rotated, according to the phase correction determined from an individual spectrum. We now have a series of spectra, each consisting of an absorptive and a dispersive part, formed as linear combinations of the original real and imaginary parts. Complex interferograms are then formed out of corresponding points along the frequency axis from each of the spectra, and transformed to produce the final 2D spectrum.

The real and imaginary part of the interferograms can be formed from any linear combination of the real and imaginary parts of one or more spectral sets after the first Fourier transformation. We refer to these coefficients below according to the following scheme: `RR1` is the coefficient used to multiply the real part (first `R`) of spectra in set `1` (the `1`) before it is added to the real part (second `R`) of the interferogram. `IR2` would thus represent the contribution from the imaginary part of spectra in set 2 to the real part of the interferogram, and so on.

For some experiments, another set of complex interferograms are formed from these two sets of f_2 spectra. This set of interferograms is 90° out-of-phase in f_2 to the previous set and can be constructed without any *additional* coefficients.

Different experiments will require different coefficients. Some, such as heteronuclear 2D-J experiments, consist of only one FID and spectral set, and hence there will be a total of four coefficients. Others, including hypercomplex 2D experiments, will consist of two original data sets and hence a total of eight coefficients. Other experiments are possible with three or even more data sets, requiring in each case four times as many coefficients as the number of data sets (see the macro `wft2dac`).

If there are *n* data sets to be transformed, as in typical phase-sensitive experiments, 4*n* coefficients must be supplied. The first 2*n* coefficients are the contributions to the real part of the interferogram, alternating between real and imaginary parts of the successive data sets. The next 2*n* coefficients are the contributions to the imaginary part of the interferogram, in the same order.

Thus, using the definition that the first letter refers to the source data set, the second letter refers to the interferogram, and the number identifies the source data set, we have the cases shown in the table on the right.

Data Sets	*Coefficient Order*
1	RR1, IR1, RI1, II1.
2	RR1, IR1, RR2, IR2, RI1, II1, RI2, II2.
3	RR1, IR1, RR2, IR2, RR3, IR3, RI1, II1, RI2, II2, RI3, II3.
. . .	...

The coefficients are generally 1, 0, or –1, but other coefficients are acceptable. Any *real* coefficient can be used, and as many coefficients can be non-zero as is desired. Up to 32 coefficients can be supplied, which at four per data set allows the addition, subtraction, etc., of eight 2D data sets (that is, eight different phase cycles). See the macro `wft2dac` for more information.

Processing Programs

A number of processing programs are available:

- `ft1d(coefficients)` performs only the first Fourier transformation along the f_2 dimension (without weighting) and matrix transposition, allowing the display of interferograms with the `wti`, `dcon`, and `dconi` commands.
- `wft1d(coefficients)` functions the same as `ft1d` except weighting is included.
- `ft2d<(<option,>coefficients)>` performs a complete transformation in 2D, without weighting, after 2D data has been acquired. If the first Fourier transformation has already been done using `ft1d`, `wft1d`, `ft1da`, or `wft1da`, then `ft2d` performs only the second (t_1) transform. `'ptype'` or `'ntype'` can be used as the first argument to select P-type or N-type peak selection. The `coefficients` argument are discussed below.
- `wft2d<(<option,>coefficients)>` performs the same as `ft2d` except weighting is included. To perform a normal 2D transform on the `n`-th element in an arrayed 2D experiment, type `wft2d(n)`.
- `ft2da<('bc',polynomial_order)>` runs complete phase-sensitive Fourier transform after the 2D FID data has been acquired. `'bc'` is a keyword to perform a baseline correction on the f_2 spectra prior to the Fourier transform along f_1. `polynomial_order` is the order of the polynomial used in the baseline correction.
- `wft2da<('bc',polynomial_order)>` functions the same as `ft2da` except weighting is included.
- `ft1da` functions the same as `ft2da` except a Fourier transform along f_1 is omitted.
- `wft1da` functions the same as `ft1da` except weighting is included.

For some 2D data sets, you can save much time by selectively transforming the t_1 interferograms. `ft2d('f2sel')` allows only preselected f_2 regions to be transformed along t_1; the t_1 interferograms in the non-selected f_2 regions are zeroed but *not* transformed. The same mechanism used to select baseline regions for baseline correction (`bc`) is used to select the f_2 regions that are to be transformed along t_1. Partition the integral of the spectrum into several regions. The even numbered f_2 regions, e.g., 2, 4, etc., will be transformed along t_1; the odd numbered ones will not be transformed along t_1.

Unreliable peak heights can be caused by Fourier transformation of truncated time-domain data, instead of Fourier numbers `fn` and `fn1` being too low, as might be intuitively expected. To obtain properly defined signals, take one of the following steps:

- Collect data until the signal has decayed to zero in the time domain, or
- Transform the data with zero-filling (`fn>=2*np`, `fn1>=4*ni`).

Taking one of these steps is particularly important in 2D spectra with antiphase or dispersive signals, where underdigitization can lead to signal cancellation.

Common Coefficients for wft2d Processing

To enter process coefficients, use the **Transform Coefficients** field on the Process page. Typically, the coefficients are set in the 2D data (4 coefficients for absolute value mode and 8 coefficients for phase sensitive).

A magnitude-mode transform, in which the real part of the interferogram is formed from the real part of the spectra and the imaginary part of the interferogram is formed from the imaginary part, would require **1,0,0,1**. Changing the sign of the imaginary part of the interferogram serves to change the effective direction of the f_1 frequency axis, as is required for data in which N-type peaks are detected. This can be done with **1,0,0,-1**.

In some experiments, including heteronuclear 2D-J, the basic data are purely amplitude modulated, with a starting amplitude of +1. After the first transformation and phasing operation are complete, the dispersion part of each spectrum serves only to produce a phase-twist in the final spectrum without contributing any information. Setting the imaginary part of the second transform to zero produces a pure absorption display in both domains: **1,0,0,0**.

In the hypercomplex method for pure absorption 2D data, we have two complete sets of spectra and must therefore provide eight coefficients to specify the composition of the interferograms. A typical execution of the method described by States, Haberkorn, and Ruben, assuming that the first spectrum of the first data block has been phased for absorption, requires **1,0,0,0,0,0,1,0** to produce pure absorption spectra.

Other manipulations of two data blocks are formatted similarly. A magnitude-mode 2D experiment that is the sum of the two different experiments can be constructed by **1,0,1,0,0,1,0,1**. For a COSY experiment, this would produce the P-type experiment. Subtracting data block two from block one, which for a COSY experiment gives the N-type COSY, would be accomplished by **1,0,-1,0,0,1,0,-1.** Thus two different absolute-value 2D experiments (P-type and N-type), and a phase-sensitive 2D experiment, can all be produced from the *same* data set, without acquiring the data again.

Different combinations of data sets with appropriate phase cycling might allow selection of various quantum orders in a *single* experiment. Note that since the coefficients may be different from one, it is possible essentially to phase shift each experiment *separately* (phase shift the receiver) *after* the experiment is done. For TPPI data with `phase=3`, only one data set is collected, and the imaginary part of the second transform is set to zero: **1,0,0,0**.

2D Solvent Subtraction Filtering

2D solvent subtraction is set up on the **Linear Prediction** page under the **Process** panel. In a 2D transform, solvent subtraction is invoked on t_2 FIDs. The parameters `ssfilter` and `ssorder` select the processing option as follows:

- The zfs (zero-frequency suppression) option is selected if both bandwidth (`ssfilter`) and polynomial (`ssorder`) are set to a value.
- The lfs (low-frequency suppression) option is selected if bandwidth is set to a value and polynomial is not checked.
- The zfs and lfs options are both turned off if bandwidth is blank.

Left Shift FID, Left Shift Frequency, Phase Rotation

Use the **Linear Prediction** panel to adjust the Left Shift FID, Left Shift Frequency, and the Phase Rotation.

Check **Left Shift FID** to left-shift the interferogram by the entered number of complex (or hypercomplex) points before weighting and Fourier transformation are performed. The value must be between 0 and number of increments minus 1.

To shift the frequency, enter a negative value to shift the peaks upfield (to the right) or a positive value to shift the peaks downfield (to the left). The Left Shift Frequency values operate on complex np FID data, referred to as the t_2 dimension in a 2D experiment.

To phase-rotate the interferogram, check the appropriate box and enter a value in degrees (zero-order phase rotation). This causes zero-order phase rotation before weighting and Fourier transformation are performed.

2D Processing of 3D Data

Acquisition and full processing of 3D data is available provided the parameters ni2 and sw2 have been created (d3 is the delay increment in the ni2 dimension). Also available is 2D processing of "slices" of the 3D data matrix, which can be performed as described below.

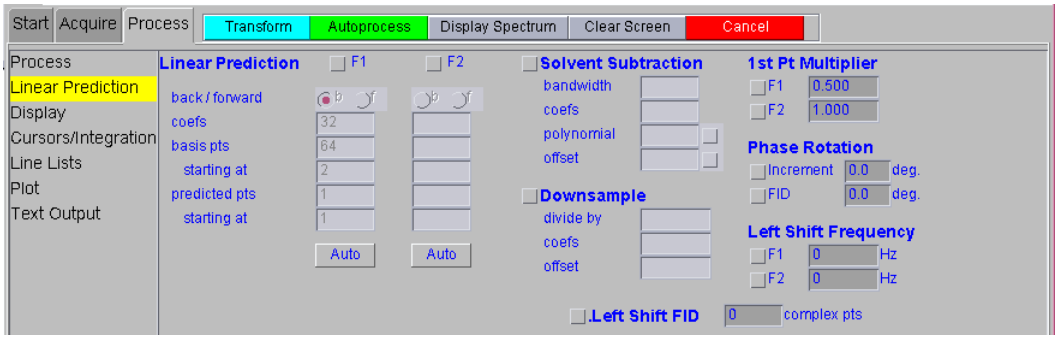
`ft2d('ni2')`	transforms non-arrayed 2D data that have been collected with ni2 and sw2 (instead of ni and sw1). The `addpar('3d')` macro creates the necessary processing parameters for the `ft2d('ni2')` operation (`par3d` functions the same as `addpar('3rf')`).
`ft2d('ni',#)`	selectively transforms a particular np–ni 2D plane within a non-arrayed 3D data set; # is an integer that can range from 1 to ni2 in this example.
`ft2d('ni2',#)`	selectively transforms a particular np–ni2 2D plane within a non-arrayed 3D data set; # is an integer that can range from 1 to ni in this example.

If an arrayed 3D data set is to be selectively processed, the format of the arguments to ft2d changes. For example, `ft2d('ni',#1,#2)` performs a 2D transform along np and ni of the #2-th ni2 increment and the #1-th element within the explicit array. This yields a 2D np–ni frequency plane. #1 ranges from 1 to ni2; #2 ranges from 1 to `[arraydim/(ni*ni2)]`.

Arrayed 3D data sets can also be subjected to 2D processing to yield 2D absorptive spectra. If the States-Haberkorn method is used along both f_1 (ni2 dimension) and f_2 (ni dimension), there will generally be four spectra per (ni,ni2) 3D element. In this case, the command `ft2d('ni2',#1,<16 coefficients>)` would perform a 2D transform along np and ni2 of the #1-th ni increment using the ensuing 16 coefficients to construct the 2D t_1-interferogram from appropriate combinations of the four spectra per (ni,ni2) 3D element. Use the proc2 parameter to specify the type of data processing to be performed on the ni2 interferogram (3D): 'ft' for complex FT, 'rft' for real FT, or 'lp' for linear prediction processing on complex data. The macro dg2 displays 3D processing parameters.

10.6 2D and 3D Linear Prediction

Linear prediction parameters are adjusted on the **Linear Prediction** page.



F1 controls the transformation process along t_1, and F2 controls the transformation process along t_2. Using the same method of transformation is not necessary along two (or three axes). You might, for example, employ a backwards linear prediction in t_2 of a 2D experiment and a forwards linear prediction along t_1, or perhaps a simple Fourier transformation along t_2 and a backwards linear prediction along t_1.

10.7 Phasing the 2D Spectrum

The phase constants `lp1` and `rp1` control the phase correction along f_1 in phase-sensitive data. In most 2D experiments, these should be near zero, but because of finite pulse widths and delays present in the pulse sequence, they may be far from zero. If the pulse sequence properly compensates for these pulse widths and delays, it is possible to have zero `lp1` and `rp1`. Most of the setup macro set `lp1` and `rp1` to zero so that the first display will indicate the need (if any) for phase correction in f_1. The same techniques as used in 1D phasing are employed here, with a minor difference.

1. Enter **`f full`** to display the full data matrix in a full chart display.
2. To phase the 2D spectrum, use the horizontal cursor present in the interactive display to identify a peak toward the right-hand edge of the spectrum. Note the trace number indicated at the top of the display (you can “memorize” this by setting **`r1`** equal to its value.)
3. Select one or more other traces at f_1 values more toward the center and left parts of the spectrum. If there is a diagonal in the spectrum with large peaks, these will be the most sensitive with which to work. Use **`r2`**, **`r3`**, etc. to “memorize” these trace values. A minimum of two is needed, one at the far right and one at the far left.
4. Enter **`ds(r1)`**. Phase this spectrum as you would a 1D spectrum using the Phase button in the displayed menu. Click the mouse on the peak displayed near the right edge of the spectrum. Phase up this spectrum (thus setting `rp1`). Do not “click” in the left part of the spectrum at this time.
5. Enter **`ds(r2)`**. The second trace appears. Click the mouse near the right edge of the spectrum (to fix `rp1` at the previously determined value) and do not rephase. Move the mouse to the peak at the left, click and phase it (thus setting `lp1`).
6. Enter **`ds(r1)`** to recheck `rp1`. Repeat the process again if necessary.

In homonuclear correlation spectra (such as NOESY, TOCSY, and ROESY), use the diagonal peaks for phasing. It there are strong cross-peaks, you can phase an f_1 trace

exactly like a 1D spectrum. Phase HMQC spectra by progressively working from right to left, with several peaks selected along the way to make sure that `lp1` does not go through an extra revolution that would induce some baseline roll.

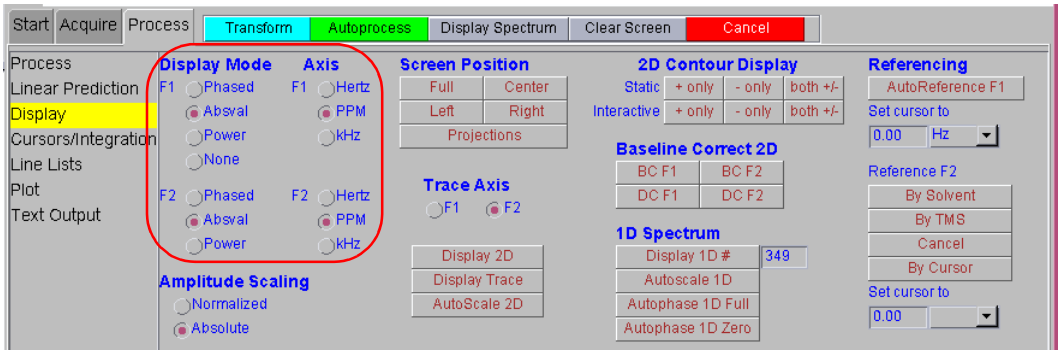
Corrections in f_2 phasing may be obvious in the 2D data when they are not in the first increment 1D spectrum. If `pmode='full'` before the 2D transform, f_2 phasing may be corrected without retransforming by setting `trace='f2'` and using the same approach as described for f_1 phasing. Transformation of the data again is necessary if `pmode=''` or `pmode='partial'`. No f_1 phasing is possible after transformation if `pmode=''`; f_1 rephasing after the transform is possible (but not f_2 rephasing) if `pmode='partial'`. Do baseline corrections such as `dc2d` or `bc` only after data are properly phased in f_1 and f_2.

10.8 Display and Plotting

This section discusses noninteractive 2D display and plotting.

Display Modes

Select display modes in the **Display** page.



Display and Plot Limits

The `center`, `left`, `right`, and `full` set the spectrum to display (and subsequent plot) in the relevant portion of the screen (and page).

Maximum Intensity

The `peak2d` command searches the area defined by `sp`, `wp`, `sp1`, and `wp1` in a 2D data set for a maximum intensity. It returns the maximum intensity value found, the trace number of the maximum, and the data point number of the maximum on that trace.

Display Scaling

The **AutoScale 2D** button on the **Display** page sets up the vertical scaling and threshold for a 2D contour plot and color map display.

Grid Lines

A grid of horizontal and vertical lines over a 2D display can be drawn by the `grid` macro. By default, grid lines are drawn in blue at approximately 1 cm intervals, rounded so that the intervals fall at a multiple of 1, 2, or 5 of Hz or ppm. To change the defaults, enter `grid` with a different spacing (in cm) or a different color (`'red'`, `'green'`, etc.); for example, `grid(2,'white')` gives white grid lines at 2 cm intervals.

The `grid` command also can define a grid, using the following syntax:

`grid<(startf2,incrf2,startf1,incrf1,color)>`

The arguments define the frequency and increments between grid lines in the f_2 and f_1 directions and the color of the grid lines.

The `plgrid` macro uses the same arguments as `grid`, but plots the `grid` instead.

Whitewashed Spectra

The `dsww<(start,finish,step)>` command displays one or more spectra with whitewashing (traces in front "block" the view of traces behind them). Use the argument `'all'` to display all spectra. `plww<(start,finish,step)>` plots the same spectra.

Use the Stacked Plot graphics button to display a stacked display of 2D spectra in the whitewash mode.

Label Display

The `dssl` macro displays a label for each element in a set of stacked spectra. The label is an integer value starting with 1 and ranging to the number of spectra in the display.

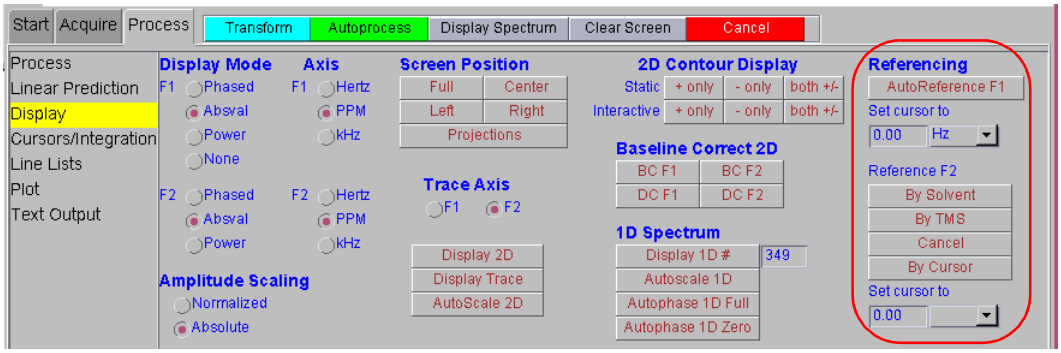
If `wysiwyg='n'`, labels can appear at incorrect positions. The positions were empirically determined for a large screen display and are not guaranteed to be correct for all displays.

The following `options` control the `dssl` display (more than one option can be entered as long as the options do not conflict with each other):

- `'center'`, `'left'`, `'right'`, `'top'`, `'bottom'`, `'above'`, and `'below'` are keywords setting the position of the displayed index relative to each spectrum.
- `'value'` is a keyword that produces a display of the values of each array element, instead of an integer index.
- `'list=xxx'` produces a display of the values contained in the arrayed parameter `xxx`.
- `'format=yyy'` uses the format `yyy` to control the display of each label. See the `write` command for information about formats.

2D Referencing

Use the Referencing buttons on the Display page to set up 2D referencing.



By default, each reference line is set at the cursor position after taking into account any frequency scaling.

To set the reference lines to other than the cursor position, enter a frequency (in Hz or PPM). For example, if you are doing a 2D experiment in which the indirect axis is determined by the decoupler channel (i.e., HMQC or HETCOR experiment), you might enter, `10 ppm`, which is equivalent to 10*decouple frequency.

To clear referencing along f_2 and f_1, click the Cancel button.

To center the cursors in the spectrum, use the `centersw` for the directly detected dimension, and the `centersw1` for the first indirect dimension.

To set the spectral width for a given spectral window, use the macro `setsw1(nucleus,downfieldppm,upfieldppm)`.

Rotating Homonuclear 2D-J Spectra

The `rotate<(angle)>` command rotates homonuclear 2D-J data 45° (rotation in frequency-space) to line up multiplets. Use the `angle` argument to specify other angles.

Setting Negative Intensities to Zero

The command `zeroneg` is used for the projection of proton 2D-J spectra at 45° to strip a high resolution proton spectrum down to a list of chemical shifts. `zeroneg` sets all negative intensities to zero.

10.9 Interactive 2D Color Map Display

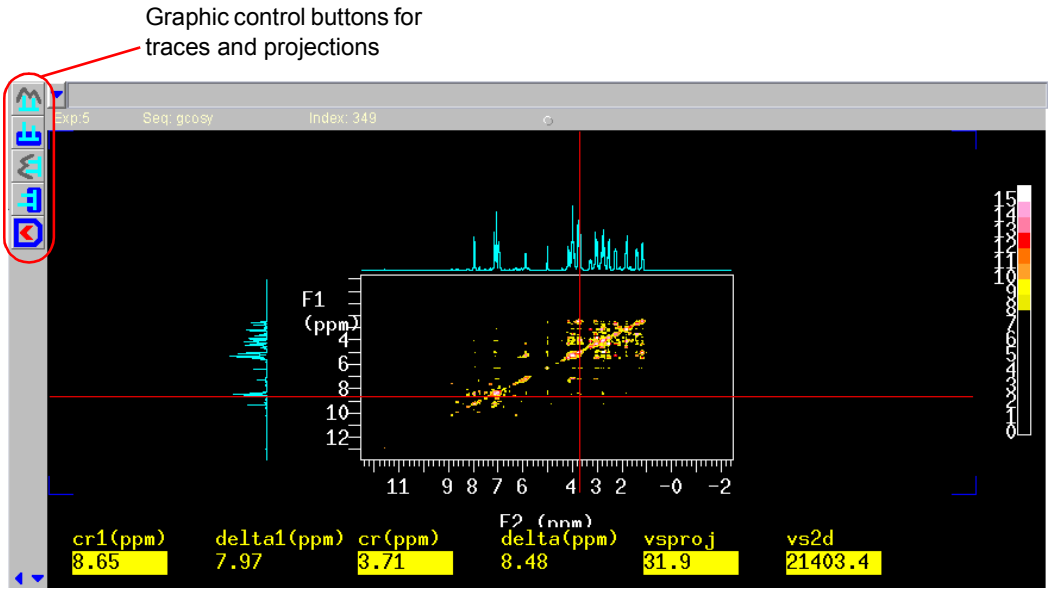
Use the graphics control buttons and the mouse to control the display in the graphics window.

- "2D Display," this page
- "Interactive 2D Display Control Buttons," page 189
- "Cursors," page 190
- "Traces," page 190
- "Projections," page 191
- "Expanding the Display," page 191

- "Setting the Vertical Scale," page 191
- "Adjusting the Threshold," page 191
- "Treating 2D Traces as 1D Spectra," page 191

2D Display

Below is an example of a 2D display with the projection graphics control buttons selected.

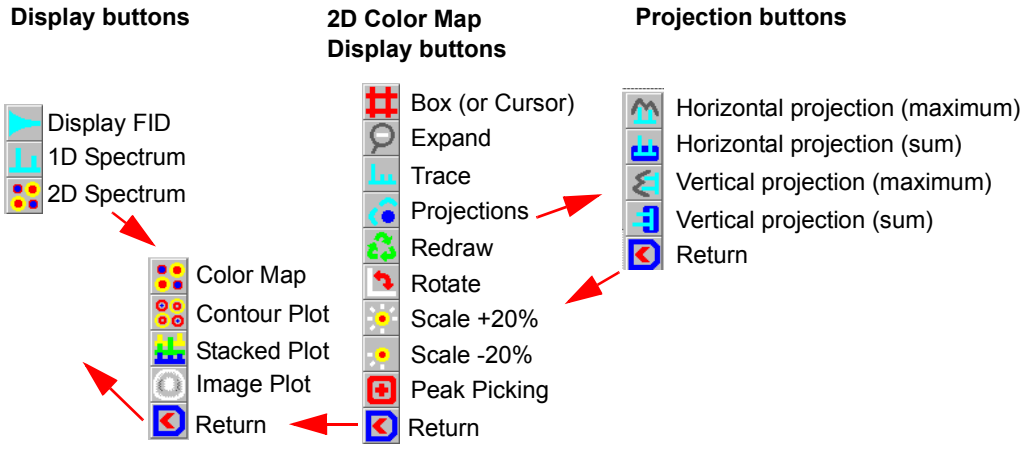


The color/grayscale adjustment appears on the right side of the window. The interactive display parameters are displayed across the bottom of the graphics window:

`cr`	shows the current cursor position.
`cr1`	shows the current cursor position along the first indirectly detected dimension.
`delta`	shows the cursor difference.
`delta1`	shows the cursor difference along the first indirectly detected dimension.
`vs2d`	shows the vertical scale of the display.
`vsproj`	shows the vertical scale of the trace or projection.

Interactive 2D Display Control Buttons

Each of the buttons on graphics control bar is described below.



2D Color Map Display Buttons

The buttons the Interactive 2D Color Map Display Main Menu function as follows:

Box	The first button is labeled Box or Cursor, depending on the display mode you are in. If labeled Box, you are in the cursor mode, and this button changes the display to he box mode with two pairs of cursors.
Cursor	If labeled Cursor, you are in the box mode, and this button changes the display to the cursor mode with one pair of cursors.
Trace	Selects the trace display mode.
Projections	Displays the Interactive 2D Display Projection Menu, see "Projection Buttons," page 190.
Expand	The magnifying glass button is labeled Expand or Full depending on the mode you are in. If labeled Expand, you are in the box mode and this button expands the area between the cursors.
Full	If labeled Full, you are in the cursor mode and this button displays the full area.
Redraw	Repeats the last 2D or image display with current parameters.
Rotate	Flips the F2 and F1 axes.
Scale +20%	Scales the 2D display in + 20% increments.
Scale -20%	Scales the 2D display in -20% increments.
Peak Picking	Displays the Interactive 2D Peak Picking Main Menu—`ll2d` program (see page 192).
Return	Returns to the 2D display menu.

Projection Buttons

Use the graphics control buttons to interactively display 2D traces and projections, which can also be plotted. You can display a maximum (or skyline) projection or a summing projection:

Horizontal projection (max)	Displays a horizontal projection of the maximum intensity at each frequency.
Horizontal projection (sum)	Displays a horizontal projection of the summed intensity at each frequency.
Vertical projection (max)	Displays a vertical projection of the maximum intensity at each frequency.
Vertical projection (sum)	Displays a vertical projection of the summed intensity at each frequency.
Return	Returns to the Interactive 2D Color Map Display Menu.

Cursors

Use the left and right mouse buttons to move cursors, the center button to adjust the vertical scale of traces, projections, and contour maps, as well as to adjust the threshold in the color bar. The cursors can be used to select regions for expansions of the display. The cursors can also be used to select positions to "mark" using the `ll2d('mark')` command, which displays and records spectral frequencies, maxima, intensities, and volumes.

The left mouse button adjusts the position of the 2D cursor. The corresponding frequencies are displayed at the bottom of the graphics window. Both the horizontal and vertical cursors move if the left mouse button is pressed within the 2D display box.

Above and below the box, only the vertical cursor can be moved; at the left and the right of the box, only the horizontal cursor. In addition, holding the mouse button down and then moving the mouse moves the cursor with the mouse.

The function of the center mouse button depends on the location of the cursor:

- If the cursor is within the 2D display box, in gray scale images, pressing the center button sets the point to medium gray. Otherwise, for color map and contour displays, if there is no intensity at that point, the center button changes vertical scale to show intensity at that point. If there is intensity at the point, the center button changes the scale to show no intensity, then changes the parameter `vs` and redraws.
- If the cursor is near an active trace and active horizontal or vertical projection, pressing the center button changes the vertical scale of trace or projection, so that spectrum goes through the current mouse position.
- If the cursor is near the color/grayscale bar and in the color mode, pressing the center button sets the threshold to remove low intensity peaks. If in the grayscale mode, pressing the center button sets the grayscale intensity (the right button adjusts contrast).

A second cursor pair is displayed with the right mouse button. The second pair can be moved in exactly the same way as the first pair, and is used to select a box within the 2D display. The right mouse button also switches the display into the box mode, the same as clicking on the Box button in the menu.

Traces

1. Click the Trace button. A trace displays for the position of the horizontal cursor.

2. Move the horizontal cursor to changed the displayed trace.
3. Adjust the vertical scale of the trace by clicking the middle mouse button on the trace, not in the 2D spectrum.

Exit the trace mode by displaying a box with the right mouse button, or by selecting another display mode.

Projections

1. Click the Projection button to open the Projection graphic control buttons.
2. Select type of direction (horizontal or vertical) and mode (maximum or summary).
3. Adjust the vertical scale of the projections with the middle mouse button.

Expanding the Display

1. Use the Box cursors to select the region in the spectrum that you want to expand.
2. Click on the Expand button to obtain an expanded display.

Setting the Vertical Scale

If a peak is expected at a certain position in the spectrum but is not visible, click once at that position with the middle mouse button. This selects a new vertical scale, so that the intensity at that point is by a factor of 2 above the threshold, and the display is redrawn.

Adjusting the Threshold

If noise is visible at a certain position in a spectrum, but should be suppressed below the threshold, click once at that position with the middle mouse button. A vertical scale is calculated so that this intensity falls by a factor of 2 below the threshold, and again the spectrum is redrawn.

Treating 2D Traces as 1D Spectra

After a trace has been selected in the interactive 2D display program, entering the command `ds` allows the trace to be displayed as if it were a simple 1D spectrum. All standard 1D data manipulations, including line listing, integration, etc., are then accessible for that trace. The command `ds(tracenumber)` also can be used to display an f_1 or f_2 trace, depending on the value of `trace`.

10.10 Interactive 2D Peak Picking

The `ll2d` program is used to automatically or interactively pick peaks in 2D spectra or 2D planes of 3D spectra. The peaks can be displayed on top of the spectrum in the `dconi` display or can be plotted using the `pll2d` command.

The results of all peak picking operations are stored in a binary file in the `ll2d` subdirectory of the current experiment directory:

- For 2D spectra, the results are stored in the file `peaks.bin`.

- For 2D planes of 3D spectra, the results are stored in `peaks_f#f#_#.bin`, where `f#f#` denotes the orientation of the plane being picked (e.g., `f1f3` or `f2f3`) and the last # denotes the number of the plane.

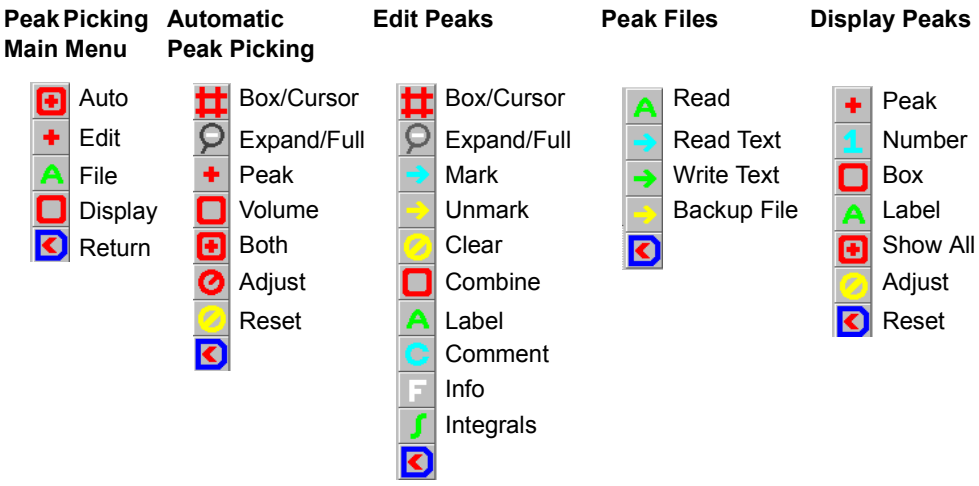
Binary peak files can be converted to text files for printing or for export to other programs.

For each peak in a peak file, the following information is stored:

- Peak number
- Interpolated peak frequency in both dimensions
- Interpolated peak amplitude
- Full width at half-height (FWHH) in both dimensions
- Bounds of the peak in both dimension
- Volume of the peak
- 15-character peak label
- 80-character comment

Interactive 2D Peak Picking Buttons

Most of the above options are accessible through a series of graphics control buttons. From the 2D Display, the Peak Picking button brings up the 2D Peak Picking buttons.



These buttons provide access to the following menus (the labels on some buttons change depending on what mode you are in):

2D Peak Picking Main Menu

This menu selects another 2D peak picking menu. The buttons function as follows:

Auto	Displays the buttons for automatically picking peaks.
Edit	Displays the buttons for interactively editing peaks.
File	Displays the buttons for manipulating peak files.
Display	Displays the buttons for controlling the display of peaks.
Return	Returns to the 2D Color Map Display buttons.

2D Peak Picking Automatic Menu

This menu provides automatic peak picking. The buttons functions as follows:

Box/ Cursor	Select cursor mode.
Expand/ Full	Expanded or full display.
Peak	Automatically finds peaks in the 2D spectrum. If one cursor is visible, all peaks above the current threshold in the currently displayed region of the spectrum are found and marked. A peak is defined as a data point that is higher than the eight points around it. Once such a point is found, the actual peak location is determined by interpolation in both dimensions.
Volume	Automatically finds the bounds of a peak and the integral of all points within these bounds. The bounds are found by descending down the sides of a peak until the point is reached where the amplitude of a data point is less than `th2d` times the current threshold. Thus, using a smaller value for `th2d` will cause `ll2d` to find and integrate a larger area for the bounds of the peaks. The peak volume is calculated by summation of all data points within these bounds. If the bounds of a peak already exist, the volume is recalculated.
Both	Pick peaks and calculate volumes. The Both button does both the peak and volume operations at once.
Adjust	Adjust peak bounds so that none overlap. The Adjust button adjusts all peak bounds in the displayed region of the spectrum so that none overlap and recalculates peak volumes with the new peak bounds.
Reset	Deletes all peaks that have been found in the current spectrum.
Return	Display the 2D Peak Picking buttons.

2D Peak Picking Edit Menu

This menu provides interactive peak editing. The buttons functions as follows:

Box/ Cursor	Select cursor mode.
Expand/ Full	Expanded or full display.
Mark	In `dconi` cursor mode, this button inserts a peak at the current cursor location. In `dconi` box mode, the cursors are taken as peak bounds and the area inside the cursors is integrated. These peak bounds are then assigned to all peaks within the cursors that do not already have their bounds defined. If a peak without bounds does not exist inside the area defined by the cursors, the highest point within that area is found, marked as a peak, and assigned the bounds defined by the cursors.
Unmark	In `dconi` cursor mode, this button deletes the peak nearest the cursor. In `dconi` box mode, this button deletes peak bounds from peaks whose bounds are entirely within the area defined by the cursors.
Clear	In `dconi` cursor mode, this button deletes all peaks in the area of the spectrum displayed in `dconi`. In `dconi` box mode, the Clear button deletes all peaks that are within the area defined by the cursors.

Combine	This button works only in `dconi` box mode. It combines all peaks within the area defined by the cursors into a single peak. This combination peak is located at the average frequencies of all of the original peaks and has bounds that encompass all of the original bounds of the peaks. The volume of the combination peak is calculated by summation of all data points within its bounds. You may wish to back up the peak file using the Backup File button in the 2D Peak Picking File Menu (see below) prior to using this button, because the original peaks are permanently deleted when the combination peak is created.
Label	Prompts for a 15-character label to be assigned to the peak nearest the cursor (`dconi` cursor mode) or to all peaks within the area defined by the cursors (`dconi` box mode). Based on the value of the parameter `ll2dmode`, this label can be displayed next to the peak in `dconi`.
Comment	Prompts for an 80-character string to be assigned to the peak nearest the cursor (cursor mode) or to all peaks within the area defined by the cursors (box mode).
Info	Prints the peak file information about the peak nearest the cursor to the text window.
Set Int	Set the value of the peak volume.
Return	Display the 2D Peak Picking Main Menu (see above).

2D Peak Picking File Menu

Read	Prompts for the filename of a binary peak file and reads that file into VnmrJ. When a file is read in, the current peak file (`peaks.bin` for 2D spectra) is overwritten by a copy of the peak file that was read in.
Read Text	Prompts for the file name of a text peak file and reads that file into VnmrJ. When a file is read in, the current peak file (`peaks.bin` for 2D spectra) is overwritten by a new binary copy of the peak file that was read in.
Write Text	Prompts for a filename to write a text version of the current `ll2d` peak file.
Backup File	Prompts for a file name to copy the current binary peak file. It is a good idea to do this occasionally when doing a significant amount of interactive peak editing, so that intermediate versions of the peak file can be recovered in the event of an error (such as inadvertently selecting the Clear or Reset button or making a mistake using the Combine button).
Return	Display the 2D Peak Picking Main Menu (see above).

2D Peak Picking Display Menu

Show Peak/ Hide Peak	Show Peak, the “+” is hidden and this button shows a “+” at the location of each peak. Hide Peak, the “+” is shown and this button hides the “+” at the location of each peak.
Show Num/ Hide Num	Show Num, the peak numbers are now hidden and this button shows a peak number next to each peak. Hide Num, the peak numbers are now shown and this button hides the peak numbers.

Show Box/ Hide Box	Show Box, the box is now hidden and this button shows a box with the area integrated to get the volume of the peak. Hide Box, the box is now shown and this button hides this box.
Show Label/ Hide Label	Show Label, the peak labels are now hidden and this button shows a peak label next to each peak. Hide Label, the peak labels are now shown and this button hides peak labels.
Show All/ Hide All	Show All displays a "+", the peak number, the peak bounds, and the peak label for each peak. Hide All hides all peak information.
Return	Display the 2D Peak Picking button.

10.11 3D NMR

VnmrJ includes full support for 3D NMR, including acquisition, processing, and display.

Many of the 3D-related macros and parameters—for example, `centersw2`, `cr2`, `crl2`, `delta2`, `dmg2`, `lp2`, `lsfid2`, `phfid2`, `rfl2`, `rfp2`, `rp2`, `sp2`, `wp2`—are normally used in the same manner as their 1D and 2D counterparts and are not described further in this section.

In a non-arrayed 3D experiment, there are two implicitly arrayed parameters: `d2` and `d3`. `d2` is associated with `ni` and `sw1`, `d3` with `ni2` and `sw2`. The order of these two arrayed parameters is such that `d2` is cycled the most rapidly.

In an arrayed 3D experiment, such as a single 3D with "superhypercomplex" data acquisition (States-Haberkorn method applied along both t_1 and t_2), there are, in reality, at least three arrayed elements. By convention, such an arrayed 3D experiment is implemented using four arrayed elements: `d3` (t_1 evolution time), `phase2`, `d2` (t_2 evolution time), and `phase`.

Assuming that `array='phase,phase2'` (see below), the order of arrays is such that the `phase2` array is cycled the most rapidly, followed by the `phase`, `d2`, and `d3` arrays.

3D Acquisition

3D data acquisition is accomplished with pulse sequences using the parameter `d3`, which is updated according to the parameters `ni2` and `sw2`. This is analogous to `d2`, which is incremented according to `ni` and `sw1` for 2D NMR (of course, `d2`, `ni`, and `sw1` are active in 3D as well). In addition, the parameter `phase2` is used to control the "mode" of acquisition (hypercomplex, TPPI, or absolute value) in the third frequency domain, just like `phase` in the second domain. All of these 3D parameters are created with the macro `addpar('3d')` along with other 3D parameters, including `fiddc3d` for 3D time-domain dc correction, `ptspec3d` for region-selective 3D processing, and `path3d` for the path to the currently displayed 2D planes extracted from a 3D data set. (The macro `par3d` is functionally equivalent to `addpar('3d')`.)

By convention, 3D sequences are described with the first evolution time being known as t_1, the second evolution time as t_2, and the time during which data are acquired as t_3. After transformation, these same dimensions are called the f_1, f_2, and f_3 dimensions.

3D Processing

Data processing includes the `ft3d` command for full 3D processing, governed by the usual parameters to control transform sizes, weighting, phasing, etc., with a "2" at the end of the parameter name signifying the third dimension. Unlike other commands, `ft3d` occurs in the background by default; that is, it is run as a separate task by UNIX, leaving VnmrJ free to continue with other tasks (including 1D and 2D processing of the same data set!). To increase the speed of 3D transforms further, the `wftt3` macro allows the software to process one dimension (the acquisition or t_3 dimension) as the data are being acquired. Also, the `ft3d` software can be configured to run on several computers simultaneously, for even greater speeds. The `killft3d` macro terminates any `ft3d` program that has been started in an experiment.

3D Display

Once the data are processed, the data can be displayed as two-dimensional planes of the 3D data set in any of the three orthogonal directions. Skew planes are not supported, nor are "full 3-dimensional" displays. One command, `getplane`, extracts the 2D planes from the 3D data set in one or more of the three orientations. After the planes are "extracted" in this manner, they are displayed with the `dplane` macro. The parameter `index2` keeps track of which plane is on display. The macro `nextpl` displays the next plane from the plane currently on view. Another macro, `prevpl`, shows the previous plane from the current plane.

The `dsplanes(start_plane,stop_plane)` macro produces a graphical 2D color or contour map for a subset of 3D planes specified by the arguments. The `dconi` program is used to display the planes. The `plplanes` macro is available to plot a series of 3D planes.

The new concept of time-domain frequency shifting can be employed to good use in 3D NMR, where spectra in the indirectly detected directions are often "folded" by accident or by choice. The parameters `lsfrq`, `lsfrq1`, and `lsfrq2` cause the frequency of the spectrum to be shifted as part of the Fourier transformation process.

3D Pulse Sequences

No standard and fully documented pulse sequences are provided for 3D NMR in the released software, although a number of sequences will be found in the user library. If you are writing your own sequences, you simply need to write a sequence that includes a `d2` and `d3` delay (these delays may also be `d2/2` or `d3/2`). If your sequence is to operate in the hypercomplex (or the hyper-hypercomplex) mode, you should use the parameters `phase` and `phase2` to select between the two orthogonal components of the hypercomplex experiment in the relevant domain. To ensure that your experiment is processed correctly using the default processing coefficients, you should write your pulse sequence so that the `phase=2` (and `phase2=2`) experiments leave the receiver unchanged (compared to `phase=1`) and either increment the phase of the pulse (or pulse sandwich) just prior to the relevant evolution, or decrement the phase of the pulse following evolution by 90 degrees (or for multiple-quantum experiments, by 90/*n*).

Experiment Setup

Setup is necessary in 3D experiments to position transmitters and decoupler, adjust pulse widths, etc. Just as the setup of 2D experiments can often be assisted by performing "first increment" experiments (i.e., a 1D experiment that represents the first increment of the 2D),

so 3D experiments can be assisted not only by 1D setup experiments, but also by “first plane” 2D experiments. To perform a 2D experiment in the sw1 dimension, set ni2=1 and phase2=1, with ni greater than 1 and phase=1,2 (or phase=3 for TPPI experiments). This combination of parameters will perform a “normal” 2D experiment, incrementing d2, and the data can be processed with the wft2da command (or its variants).

The “third dimension” 2D experiment is performed by setting ni=1 and phase=1, with ni2 greater than 1 and phase2=1,2 (or phase2=3, as desired). These parameters will produce a 2D experiment in which d3 is incremented, resulting in a spectral width sw2. The wft2d command must be given the special argument ni2 to process this data correctly, for example, wft2d('ni2',1,0,0,0,0,-1,0). You cannot use the wft2da('ni2') because the wft2da macro does not support this argument.

Notice that when you process a “first plane” 2D experiment, the axes are always labeled f_1 and f_2 because this is considered to be a 2D experiment, and hence the axis labeling corresponds to conventions used in 2D NMR.

When you are finished setting up the 3D experiment, reset ni, ni2, phase, and phase2 to their desired values. Check the value of the parameter array and make sure that array='phase,phase2' and not 'phase2,phase', which will acquire data in the incorrect order. To ensure the correct order, always enter phase before phase2, or simply enter array='phase,phase2'.

Data Processing

Just like processing 2D NMR, the proper processing of 3D NMR requires coefficients to select various components of the data to be combined to form the final data set. There are actually up to 40 coefficients required that are explained in more detail in the *Command and Parameter Reference*. In normal operation, the coefficients will be transparent to you, just as the 2D coefficients are. The set3dproc command can create a 3D coefficient file for processing 3D FID data under certain conditions.

The ft3d command determines from the values of phase and phase2 what the expected coefficients are, based on whether a hypercomplex (“States-Haberkorn”) or TPPI experiment has been performed in a particular dimension. This assumes that the pulse sequence has been written to perform “standard” phase cycling as described above. If your data are reflected along a particular dimension, it is possible (or probable) that different coefficients are required for data processing. In this case, the ft3d('nocoef') form is used to allow you to specify you own coefficients (which are found in a text file named coef in the 3D experiment directory, unlike in ft2d, where they are given as arguments to the command). By default, ft3d calls the make3dcoef macro to create a coefficient file using the f1coef and f2coef string parameter values.

The format for the 3D coefficient file is an extension of that used for 2D coefficients. The coefficient file contains four rows of eight coefficients used to construct the t_2 hypercomplex interferograms and a final row of eight coefficients used to construct the t_1 interferogram. The actual values of the coefficients depends on the order in which the States-Haberkorn components of the 3D FID data set were collected. This order depends in turn on the values of the parameters phase, phase2, and array.

If TPPI phase cycling is used to collect data along one or both of the indirectly detected dimensions, instead of four data sets per (ni,ni2) increment, there are only two or one data sets, respectively, per (ni,ni2) increment. If there are only two data sets per (ni,ni2) increment, the coef file contains four rows of four coefficients that are used to construct the t_2 hypercomplex interferograms, and a final row of eight coefficients that are used to

construct the t_1 interferogram. If there is one data set per (ni,ni2) increment, the coef file contains four rows of two coefficients that are used to construct the t_2 hypercomplex interferograms and a final row of eight coefficients that are used to construct the t_1 interferograms.

Phasing a 3D data set is best accomplished using 2D transforms. In general, the recommended method in writing 3D pulse sequences is to attempt to minimize frequency-dependent phase shifts in f_1 and f_2. Even so, there are generally small phase shifts that must be dealt with. The following steps are suggested:

1. Set **pmode='full'** to allow full phasing in both dimensions after a 2D transform.
2. Adjust **rp** and **lp** on a 1D spectrum (the first increment of the 3D), just as you would for 2D (e.g., by typing wft(1)).
3. Enter **wft2d('ni',1,1,0,0,0,0,0,0,0,0,0,0,0,-1,0,0,0)** to adjust f_1 phasing (there are 11 consecutive zeros in the middle of this argument).
4. You now have an f_1f_3 2D spectrum (with incorrectly labeled axes). Set **trace='f1'** to adjust the f_1 phase, then set **trace='f2'** to trim the f_3 phasing. You can now adjust **rp1** and **lp1** (as well as rp and lp).
5. Enter **wft2d('ni2',1,1,0,0,0,0,0,0,0,0,0,-1,0,0,0,0,0)** to adjust f_2 phasing (note that this argument has nine consecutive zeros in the middle and five zeros at the end).
6. You now have an f_2f_3 2D spectrum. Set **trace='f1'** to adjust the f_2 phasing (**rp2** and **lp2**), then set **trace='f2'** to trim the f_3 phasing if necessary.

One additional point on phasing. Some pulse sequences are written to result in a 180° phase shift across the spectrum. Remember that in VnmrJ, the "origin" for phasing is defined as the right edge of the spectrum; however, in "real" terms, the actual origin of phasing (i.e., the zero-frequency point) is at the center of the spectrum. Thus, if you expect a certain lp1 or lp2 value, such as –180°, you should simultaneously use a value of rp1 or rp2 equal to -lp1/2 or -lp2/2 (e.g., 90°).

If you want to adjust the weighting functions for the 3D transform by using the wti command and examine interferograms, you can do so along either the t_1 or t_2 axes. Use the same commands given above to adjust the phasing (the commands with the long series of zeros), but use wft1d instead of wft2d.

For the final transformation, the specdc3d parameter controls the dimensions in which a spectral drift correction is performed on the data. A three-letter value of 'ynn' gives drift correction along f_3 (the first letter) but not along f_1 (the second letter) or f_2 (the third letter); this value is probably a good starting point for your efforts.

The pmode parameter is ignored by the 3D transformation; no phasing is possible after the 3D transform.

The 3D transformation process needs to be followed by the process of extracting the 2D planes from the full 3D data set. This can be done separately, with the getplane command, but most often is combined with the ft3d command. In general, and especially for heteronuclear experiments, the f_1f_3 and f_2f_3 planes are the most interesting. The f_1f_2 plane is not only generally less useful, but also is considerably slower to extract from the data. The recommended command to use for 3D transformation, therefore, is ft3d('f1f3','f2f3'), which performs the 3D transform and extracts the two interesting planes in one step.

Solvent suppression works on t_3 FIDs of 3D spectra just like in the 1D and 2D cases.

Following the transform, set `plane='f1f3'` or `'f2f3'` and then use the `dproj` macro to display the projection of the data on that plane, or `dplane(n)` to display the nth plane. The `resetf3` macro will reset parameters after a partial 3D Fourier transform.

10.12 4D NMR Acquisition

The `addpar('4d')` macro creates the parameters `ni3`, `sw3`, `d4`, and `phase3` that can be used to acquire a 4D data set (the macro `par4d` functions the same as `addpar('4d')`).

The parameter `ni3` is the number of t_2 increments, `sw3` is the spectral width along the third indirectly detected dimension, `d4` is the incremented delay, and `phase3` is the phase selection for 4D acquisition. Processing and display in 4D is currently not available in VnmrJ.

Chapter 11. Indirect Detection Experiments

Sections in this chapter:

This chapter describes indirect detection experiments, also known as *heteronuclear multiple-quantum coherence* (HMQC) experiments. Indirect detection experiments show correlations between heteronuclei while detecting high-sensitivity protons. HMQC differs from the more traditional heteronuclear correlation techniques that detect the low-sensitivity heteronucleus (for example, ^{13}C or ^{15}N).

11.1 Probes and Filters

Indirect Detection Probes

The most commonly used probes for indirect detection experiments are the Varian, Inc. "indirect detection" NMR probes, such as the Triple Resonance, Penta, Tunable Triple, Indirect Detection, gHX Nano, Cold Probes, and others. Indirect detection probes have a ^{1}H coil and an X-nucleus coil with the ^{1}H coil positioned closer to the sample for the highest possible sensitivity of the observed nucleus. When connecting cables to the probe, ignore words like "observe" and "decouple" and think of ^{1}H (for observe) and X (for decouple) for connections.

Normal "broadband" probes similarly have a ^{1}H coil and an X-nucleus coil and can be used for indirect detection. But broadband probes have significantly lower proton sensitivity (about half that of indirect detection probes) and so are not optimum for indirect detection experiments. Nevertheless, broadband probes usually provide some sensitivity improvement over direct detection heteronuclear correlation experiments. Four-nucleus and "Switchable" probes also have a ^{1}H coil and an X-nucleus coil, with the X coil closer to the sample, and can satisfy the needs for indirect detection experiments.

For more information on Varian, Inc. NMR probes, go to the NMR probe product pages at `www.varianinc.com`. Refer the manual that shipped with the probe for installation and tuning instructions.

Filters

Additional filters are not needed for *MERCURYplus*/-Vx systems. For UNITY*INOVA* systems, filters might be needed on the transmitter, receiver, decoupler, and lock channels. Filters are part of the probe kit shipped with each indirect detection probe that Varian sells.

In the lock channel line, install a ^{2}H band-pass filter. When the filter is added, expect the lock phase to change. This filter can be left in the system at all times; it will, however, cause a small (about 3 dB) loss in lock sensitivity.

The following table lists part numbers for the bandpass filters supplied with indirect detection probes.

Filter	*300-MHz*	*400-MHz*	*500-MHz*	*600-MHz*	*750-MHz*
^{15}N	BE30.4-7.6-9BB	BE40-14-9BB	BE53-15-8BB	BE61-10-8BB	BE77-15-4BB
^{2}H	BE46-4-6BB	BE61-10-8BB	BE77-3.8-8BB	BE95-12-8BB	BE115-11-6BB
^{13}C	BE75-15-8BB	BE109-22-8BB	BE135-35-8BB	BE151-40-8BB	BE188-20-7BB
^{31}P	BE135-35-8BB	BE151-40-8BB	BE175-60-8BB	BE240-100-8BB	BE301-46-8BB

There is a “catch” with this configuration—the filters used for indirect detection tend to degrade specifications approximately 10% in terms of longer pulse widths and lower signal/noise. The user thus faces a classic trade-off of performance (manually insert filters only when needed but achieve better specs) versus convenience (leave filters in place continuously and achieve worse specs). The convenience factor, of course, is nonexistent if the instrument does anything other than ^{13}C and ^{1}H, because one cannot leave the ^{13}C bandpass filter in place on the X line while doing ^{31}P, ^{15}N, or anything else. All standard specifications are given with the indirect detection filters *not* in place.

11.2 The Basic HMQC Experiment

The essence of the HMQC experiment is the cancellation or elimination of the signals from protons attached to ^{12}C, leaving only signals from protons attached to ^{13}C, contributing to a ^{13}C–^{1}H chemical shift correlation spectrum. The three basic, independent mechanisms to generate this discrimination are:

- "Spin-Echo Difference Experiment," page 202
- "BIRD Nulling," page 204
- "Transmitter Presaturation for High-Dynamic Range Signals," page 205

Spin-Echo Difference Experiment

The heart of the HMQC sequence can be reduced to a heteronuclear spin-echo difference experiment that looks like Figure 38.

In Figure 39, *a*, *b*, and *c* represent the protons attached to carbons, where *a* are the protons attached to up-^{13}C, *b* are protons attached to ^{12}C, and *c* are protons attached to down-^{13}C. Assume that we are at the resonance frequency of the protons attached to the ^{12}Cs. In the rotating form, the following steps (shown in Figure 39) occur:

1. The first proton 90° pulse places all protons along the *y* axis.
2. After a time $\Delta = 1/(2J)$, the *b* protons are still along the *y* axis, but the *a* protons are along the $-x$ axis and the *c* protons are along the $+x$ axis.

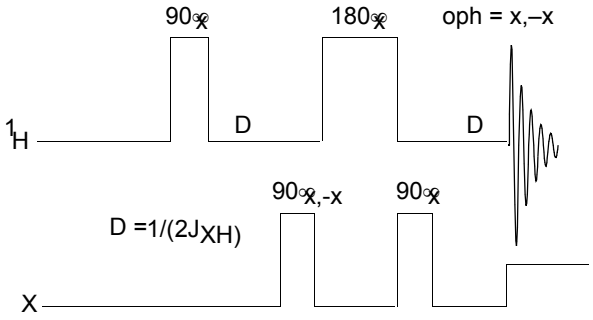


Figure 38. Heteronuclear Spin-Echo Difference Experiment

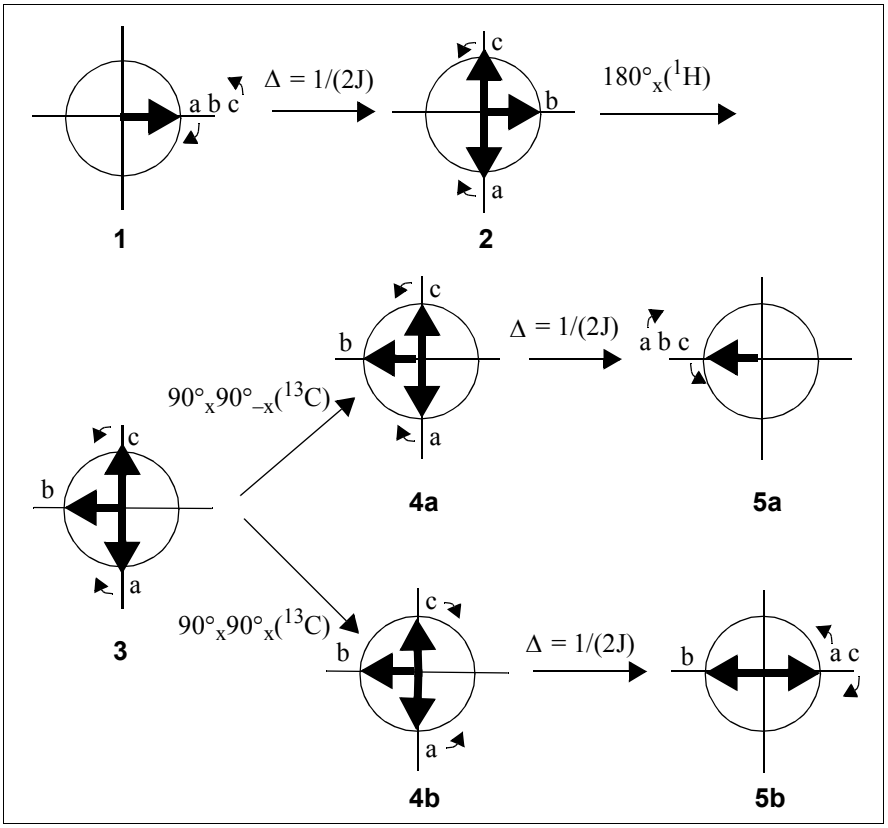


Figure 39. HMQC Pulse Sequence, Showing Movement of Attached Protons

3. Next, the $180°_X$ proton pulse places the *b* protons along the *–y* axis but does not affect the *a* and *c* protons.
4. The next pulse has the following effect:
   a. The $90°_X90°_{-X}$ carbon pulse is effectively a null pulse. All rotational directions are maintained.
   b. The $90°_X90°_X$ (= $180°_X$) carbon pulse reverses the ^{13}C, which makes the *a* protons attach to the down-^{13}C and the *c* protons attach to the up-^{13}C, essentially reversing their rotational direction.

5. After another period Δ = 1/(2J), the following occurs:
   a. The *a*, *b*, and *c* protons are refocused along the –*y* axis.
   b. The *b* protons are still along the –*y* axis, and the *a* and *c* protons are refocused along the +*y* axis.

Subtracting the signal resulting from step 5b and 5a, by changing the receiver phase `oph`, results in cancellation of the *b* protons, while the signal for the *a* and *c* protons doubles.

To create a 2D experiment with information about heteronuclear chemical shifts, we introduce an evolution time t_1 that occurs between the two X-nucleus 90° pulses, as shown in Figure 40.

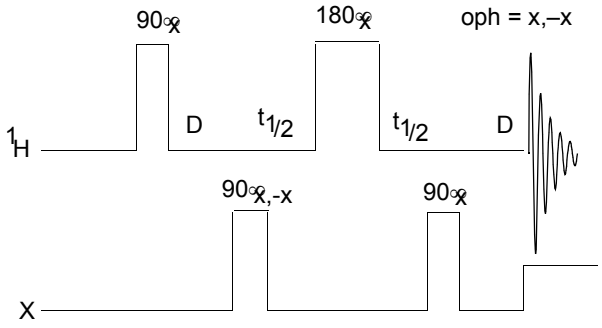


Figure 40. Evolution Time Added Between X-Nucleus Pulses

In this 2D experiment, which is now a full HMQC experiment, protons attached to ^{12}C show no different behavior and are still cancelled after two scans. For the ^{13}C nuclei, however, whether they experience a 180° pulse, a 0° pulse, or something in between, depends on the time between the two 90° pulses and their rate of precession during that time (i.e., their chemical shift). Therefore, this experiment produces a modulation of the intensity of the ^{13}C-bound protons, and the Fourier transform of that modulation yields the chemical shift of the ^{13}C bound to that proton.

In this way we detect ^{13}C chemical shifts with the intensity of protons, and simultaneously we obtain a correlation of the ^{13}C and ^{1}H chemical shifts. Appropriate variations of the experiment produce long-range coupling information.

BIRD Nulling

The second (optional) type of cancellation that can occur during an HMQC sequence is the so-called BIRD (Bilinear Rotation Decoupling) pulse nulling effect (Summers, Marzilli, and Bax, *JACS*, **1986**, *108*, 4285). A particular sequence of the BIRD pulse, three pulses on the ^{1}H channel and one on the X channel, inverts the z-magnetization of protons bound to ^{12}C and leaves the z-magnetization of protons bound to ^{13}C unaffected. The full sequence is illustrated in Figure 41, where $\Delta=1/2J_{XH}$.

After the BIRD pulse, a variable waiting period (τ in Figure 41) is inserted, allowing the ^{12}C-bound protons to relax back to equilibrium. If τ is adjusted so that the ^{12}C-bound protons are approximately at a null, then when the remainder of the pulse sequence (the normal HMQC sequence) is executed, cancellation of the ^{12}C-bound protons is enhanced (since those protons had very little magnetization at the start of the HMQC sequence). Obviously, not all protons will have the same relaxation time, so the choice of τ must be a compromise; generally, unless only one proton is involved, the additional suppression from the BIRD nulling will be a factor of two to five.

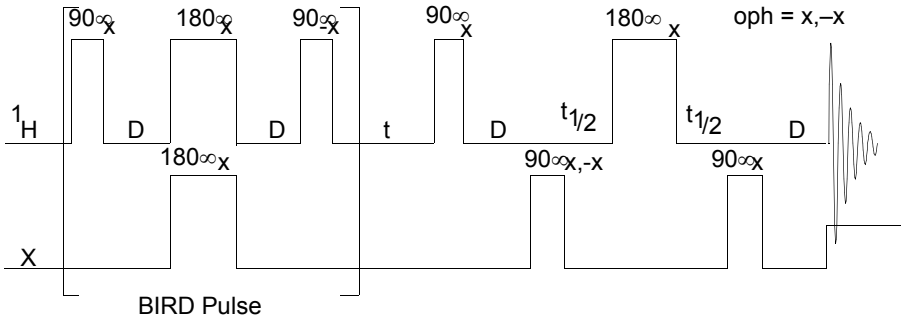


Figure 41. HMQC with BIRD Pulse Nulling Effects

For systems that exhibit a negative NOE, such as macromolecules, cross-relaxation between the inverted protons on ^{12}C and the noninverted protons on ^{13}C will decrease the intensity of the desired proton signal. The extent of this decrease can vary between 0% and 100%. For this reason, omission of the BIRD part of the sequence is advised for macromolecules.

BIRD pulse nulling is also not possible when long-range indirect detection experiments (Heteronuclear Multiple-Bond Coherence, or HMBC) are performed. In this case, protons that have long-range couplings to ^{13}C are directly bonded to ^{12}C (99% of them, anyway) so that BIRD pulse nulling would lose all intensity in the protons of interest.

Transmitter Presaturation for High-Dynamic Range Signals

When high-dynamic range situations, such as observing signals in H_2O, are involved, HMQC phase cycling and/or BIRD pulse nulling may be insufficient to produce cancellation of the large proton signals. For this reason a third mechanism, presaturation, may be necessary. Since one channel of the instrument is set to an X-nucleus like ^{13}C or ^{15}N, this presaturation must be accomplished with the other channel; that is, the same channel that will be applying observe pulses to the protons. During one or two different periods of the sequence (during the initial delay and during the τ delay), a change in power level and possibly frequency may be appropriate in order to perform the presaturation.

11.3 Typical Experimental Protocol for HMQC Experiments

A good “normal” sample to use for your first natural abundance sample is the 1% 3-heptanone in $CDCl_3$ sample (Part No. 00-968120-93). Throughout the following instructions, refer to Table 19 to understand which parameters control the features in your configuration.

1. Insert the sample and, after shimming, leave the spinner off. If you are going to run the experiment at a controlled temperature, regulate the temperature.
2. Set up to obtain a normal carbon spectrum and narrow the spectral width to the appropriate region. In some cases, the ^{13}C spectrum will be too weak to observe in a reasonable amount of time. To set the parameters controlling the ^{13}C frequency and spectral width if this is the case, you can take two approaches. First, if you have done similar experiments in the past on similar samples, just use the same parameters. Alternatively, set up a standard Carbon experiment and an appropriate

solvent. The spectrum obtained should be properly referenced. Now even if you can't see the peaks in the spectrum, you can apply the appropriate knowledge of the expected chemical shift range to place two cursors where you think the edges of that range will be, and narrow the spectral width.

3. Obtain a proton spectrum and narrow the spectral width. Check the calibration of the pulse width by entering **pw=4*pw ga**. Look only at the signals near the center of the spectrum and see if they produce a null signal. If they are negative, enter **pw=pw+0.8 ga**; if they are positive, enter **pw=pw-0.8 ga**; repeat until a good null is found, then enter **pw=pw/4**.

4. Switch to the HMQC experiment, and set the relevant parameters based on the results of steps 2 and 3.

5. Enter **phase=1 ni=1 dm='nnn' null=0 ai wexp='wft dssh'**. Set **j** to an appropriate value (normally 140 for C–H), and set **nt** to 4 or more transients, depending on the concentration of the sample (signal to noise needs to be sufficient to allow you to see the ^{13}C satellites). Now set **pwx** to an array of 0 and 90° and enter **au** to acquire two spectra. Proceed only if the two spectra are sufficiently different to give you confidence that the second spectrum is showing you satellite peaks only and not just residual uncanceled intensity of the protons attached to ^{12}C.

   If you are convinced that you are correctly connected but not happy with the quality of the spectra achieved at this step, skip ahead to step 8 and optimize the `null` parameter, then return here to check and optimize `pwx`. In either case, this is a good time to go over the checklist in the section "Cancellation Efficiency," page 209, making sure you have done everything possible to optimize cancellation.

   The spectrum in Figure 42 shows the result of this experiment on a sample of 1% 3-heptanone at 300 MHz, using `nt=64 null=2.0` and `d1=2`.

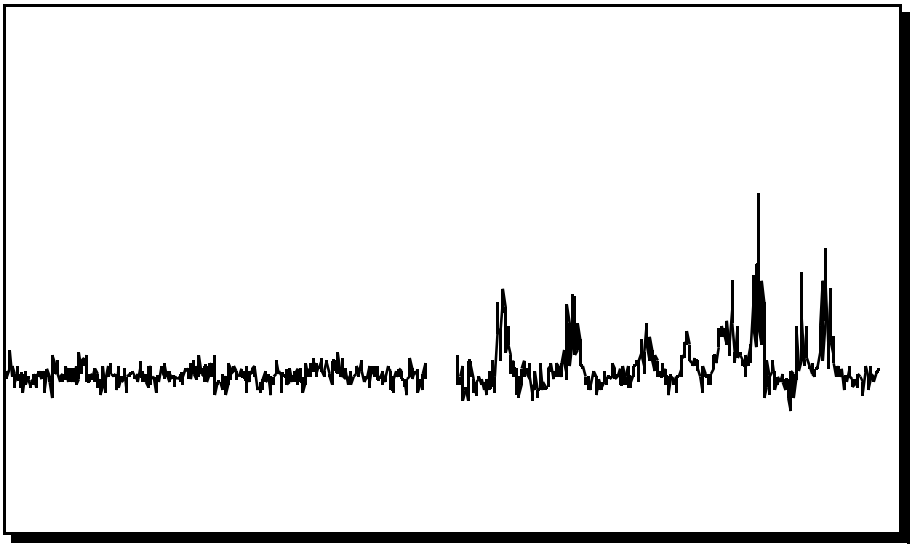


Figure 42. Verifying Cancellation with `pwx=0,90`

6. Now run an array of **pwx** around your expected 90° value, picking the one that gives you the largest satellite signals.

7. If you will be decoupling during acquisition, set **pwx** to its 90° value and enter an array of **dm='nnn','nny'**. These two experiments should show coupled and decoupled spectra, respectively.

8. Now, if appropriate, optimize the parameter `null`. Set **`nt=1 ss=4`** and enter an array of **`null`** values with at least one very short value (e.g., 0.001) and one very long value (e.g., 2.0). Because this experiment depends on the relaxation times of the spins involved, you'll also want to set **`at`** and **`d1`** to the same values you'll be using in the 2D experiment. Now run the array and select the value of **`null`** for which either most of the peaks, or the biggest peaks, or the peaks you are most interested in (the criterion is up to you), are approximately zero; remember, no one value of `null` will be correct for all peaks. Figure 43 shows this experiment run on a sample of 28 mg of gramicidin, with `null` arrayed over the range of values: 0.001, 0.05,0.1,0.2,0.3,0.4,0.5, and 2.0; examination of the spectra shows clearly how different values of `null` might be chosen.

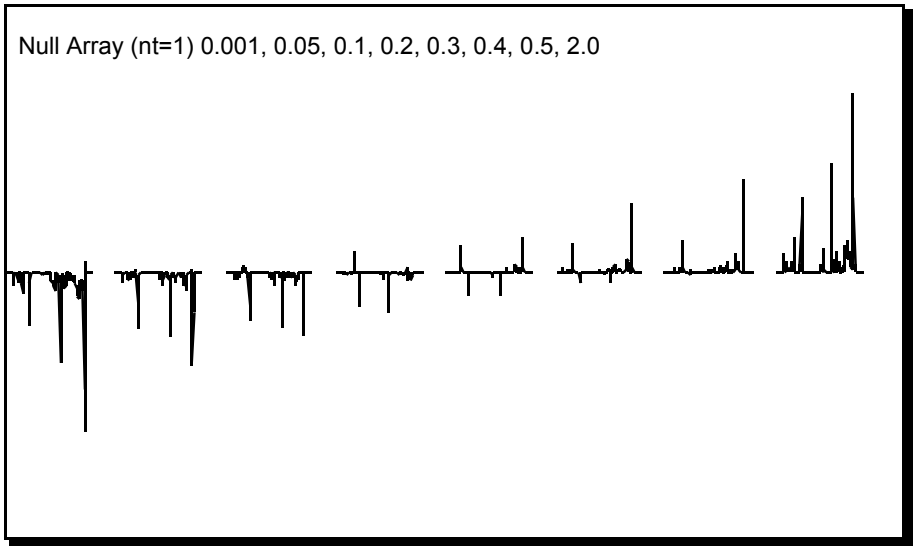


Figure 43. Optimizing the BIRD Nulling Time

9. If presaturation is desired, you can set it up in the following manner. We will need to observe the ^{12}C-bound protons, so set **`nt=1 dm='nnn'`**, and set **`null=0`** to omit the nulling period (for now at least).

   a. Enter **`ga`** and a proton spectrum will be observed.

   b. Move the FID to a different experiment, join the different experiment, and re-transform the data (e.g., **`mf(1,2) jexp2 wft`**).

   c. Set the cursor on a peak that is to be removed by presaturation, and enter **`nl movetof`**. Note the value of `tof` selected and then copy this value back to your original experiment into the parameter `satfrq` (e.g., **`jexp1 satfrq=x`**).

   d. Now set **`satflg='yn'`** and **`satdly`** equal to a time significant compared with T_1 of the peaks (e.g., `satdly=1`).

   e. Array **`satpwr`** to find the minimum value for which the peak will be removed (e.g., `satpwr=10,7,4,1 au`). When this is determined, if you wish to use presaturation, set **`satflg='yy'`**, reset **`null`**, and set **`satpwr`** to the value determined. If you do not wish to use presaturation, set **`satflg='nn'`**.

10. Set up the 2D experiment. Set **`ni`** between 128 and 256, **`phase=1,2`**, and **`nt`** to an appropriate number (comparable to what you were using in step 5).

11. Phasing in f_2 is accomplished by performing a 1D transform on the first increment with **`wft(1)`** and phasing the spectrum, paying attention only to the ^{13}C satellite peaks. In f_1, the combination of the usually large spectral width and the pulse in the center of the evolution time produces large negative values for `lp1`.

    Reasonably good starting points for the f_1 phase can be calculated according to the following formulas:

    $lp1 = -sw1 \times 360° \times ((4 \times pwx)/\pi)$

    $rp = -lp1/2$

Expect to see artifacts in these spectra. The residual uncanceled signals from protons attached to ^{12}C show up as stripes parallel to the f_1 axis at the frequency of each ^1H peak. This artifacts will be larger for peaks with long T_1, such as solvent peaks (e.g., residual protons on a deuterated solvent) or methyl groups. In Figure 44 they are seen at 2.4 ppm, 1.0 ppm, and 0.9 ppm.

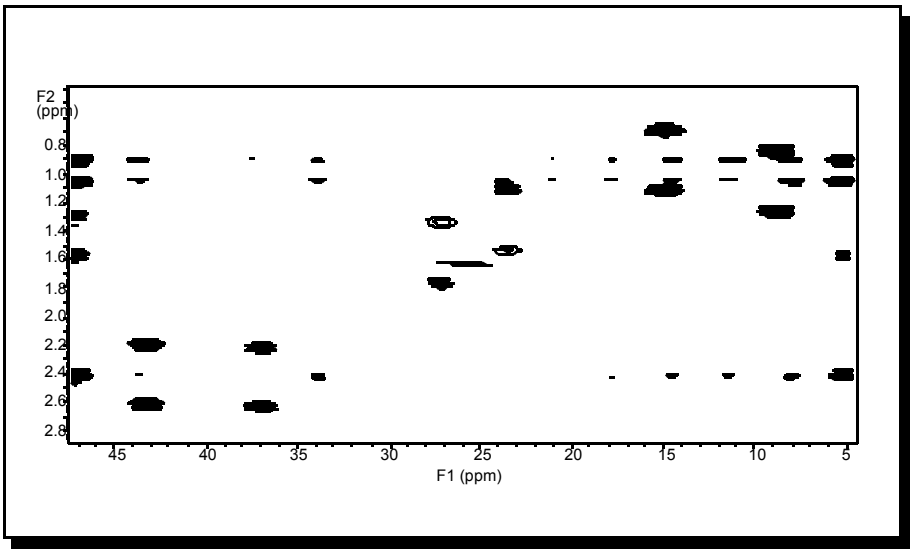


Figure 44. Coupled HMQC Spectrum of 3-Heptanone

Axial peak artifacts, which are common, will show up either at f_1=0 (the center of the spectrum in f_1) or, as in Figure 44, at the edges of the spectrum in f_1 (if FAD is used), again at f_2 frequencies corresponding to each ^1H peak and possibly through the entire spectrum. Another common artifact seen in Figure 44 is the "0,0" artifact in the exact center of the spectrum. Some peaks in the ^1H spectrum, of course, will not appear in the HMQC spectrum, because they represent protons that are not bound to ^{13}C (e.g., protons from water or NH groups). This is not the case with 3-heptanone, however.

If you are unfamiliar with HMQC spectra, you may be surprised to see multiplet structures. You should realize that during the detection period we are detecting a normal (i.e., with ^1H-^1H couplings) proton spectrum, albeit a spectrum of only those protons attached to ^{13}C. Thus, in Figure 45, we see that the proton attached to the carbon at 37.2 ppm is a quartet (it's adjacent to a CH_3 group), while the proton attached to the carbon at 43.4 ppm is a triplet (it's adjacent to a CH_2 group). In the ^1H spectrum itself, these two groups of protons are heavily overlapped (see Figure 46).

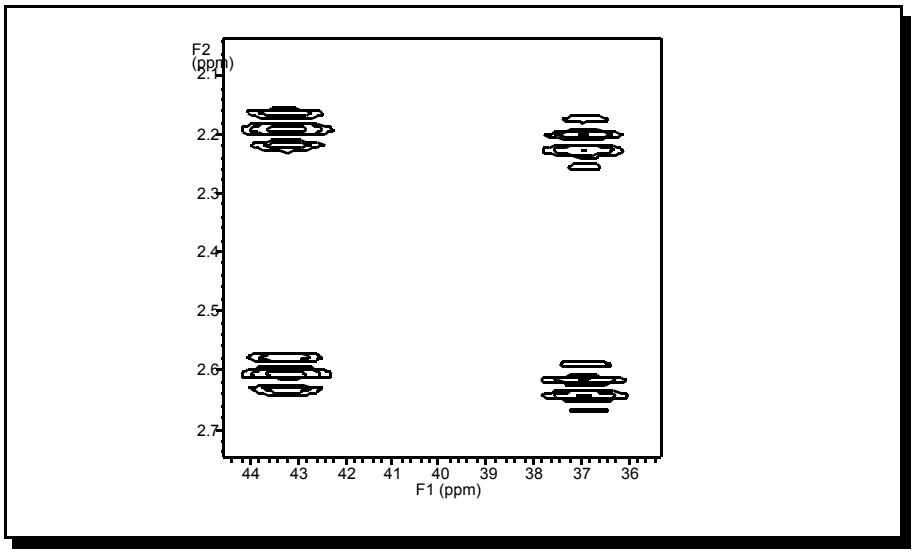


Figure 45. Expansion of Coupled 3-Heptanone HMQC Showing Multiplets

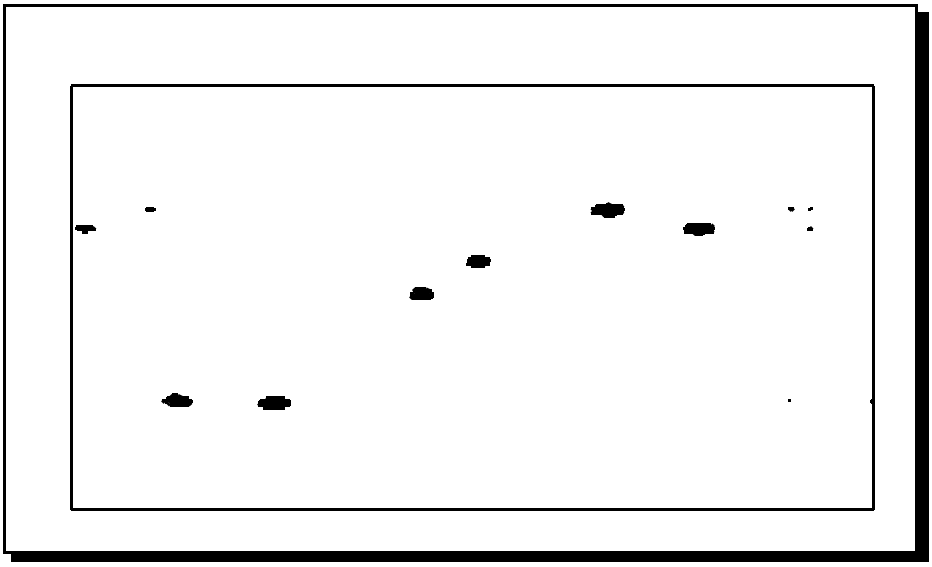


Figure 46. Decoupled HMQC Spectrum of 3-Heptanone

11.4 Cancellation Efficiency

Because indirect detection experiments involve cancellation of non-^{13}C-bound protons that are two orders of magnitude more intense (assuming unlabeled compounds), cancellation efficiency is critical. Cancellation efficiency, in turn, depends on the fundamental stability of the system rf and the reproducibility of anything else that can affect the signal. While stability is fixed by the instrumentation, you can control a number of operating conditions that can influence the quality of any cancellation experiment (NOE difference experiments are another good example). Some of these conditions are discussed here, roughly in order of importance:

- Run experiments non-spinning. This is a must.
- Use the highest lock power at which the lock is stable (be sure to shim with a non-saturating level, however) and keep the lock gain as low as possible, sufficient only to be sure that you don't lose lock during the experiment.
- Use a ^{2}H band-pass filter in the lock line. Interference between X-nucleus decoupling or even X-nucleus pulses can affect the lock and cause field instabilities, limiting the ability to perform cancellation experiments (if you don't have such a filter and want to prove this to yourself, try a short-term experiment in the unlocked mode).
- Use VT regulation, even at room temperature. Large changes in temperature of the environment can affect the VT gas stream. The frequency of peaks in the spectrum and of the lock resonance (which affects all peaks) is temperature-sensitive to some degree. Shimming may also change if the probe temperature varies, which can affect the lineshape.
- Be sure the system is in thermal equilibrium. If you are performing experiments with X-nucleus decoupling, you are applying large amounts of power to the system, which is almost certain to change the temperature of the probe, the sample, or both, even when you are performing VT regulation. The best way to ensure thermal equilibrium is to set up a "dummy" experiment with *identical* conditions (in terms of duty cycle) to your actual experiment, but which runs for perhaps several minutes (easily accomplished by setting *ni* to a small number). Now, if you queue your real experiment to follow the dummy one, the sample and probe are properly equilibrated.
- Be sure the system is in an NMR steady-state by using steady-state pulses.
- Use a large value of `nt`. Cancellation improves with larger `nt`, so the relevant cancellation is that which occurs at `nt` comparable to what you will be using in an indirect detection experiment (16 to 1024). Do not expect perfect results with `nt=2`.
- Minimize floor vibration. Where this cannot be fixed by spectrometer placement, an antivibration system should be installed.
- Use a moderate flow of body air through the probe to eliminate "rattling" from turbulent flow.
- Use lengthened pulses (attenuated rf) if you have a rise time or phase glitch problem.

Before beginning an HMQC experiment, you should assess the quality of your reproducibility by performing some simple difference experiments. The standard S2PUL pulse sequence is a good one to use for this purpose. The first pulse of S2PUL, controlled by `p1`, is held at a constant phase, while the receiver varies in phase. Thus, after four scans with `p1` set to the 90° value, `pw=0`, no signal should be seen. This can be compared to four scans with `pw` set to the 90° value, `p1=0`, which produces a full signal. Taking the ratio of these two spectra gives a concrete measurement of your cancellation efficiency, while repeating the null spectrum a number of times gives a measure of the reproducibility of the cancellation. Use this test to assess the value of the various steps and modifications described above, or of other differences (for example, the relative cancellation efficiency of experiments with and without X-nucleus decoupling).

11.5 Pros and Cons of Decoupling

While decoupling of X during acquisition seems advantageous—the spectrum is less crowded, with only half as many peaks, and each peak has twice the sensitivity—problems soon arise.

The disadvantage of X-nucleus decoupling stems from the need to use large (up to 8 kHz) decoupling fields. This high power can cause significant heating, particularly in lossy samples. As a consequence of sample heating, experiments with X-nucleus decoupling are generally limited to relatively short acquisition times, which in turn may produce less resolution in f_2 as well as less sensitivity for molecules with long T_2. Furthermore, the heating that does occur frequently produces worse cancellation efficiency. And finally, to prevent the buildup of heat in the sample, the duty cycle of the experiment may need to be limited to 10 to 20%, again possibly reducing sensitivity. For all these reasons, experiments performed without X-nucleus decoupling are perfectly reasonable, and may well be preferable.

If X-nucleus decoupling is desired, it is important to avoid sample heating. This form of sample heating can be non-uniform within the sample and can cause microconvection, producing poor cancellation. Keep the acquisition time short and the overall duty cycle less than 20%. To lower the decoupler power to tolerable levels, it may be necessary to add a fixed 6-dB attenuator to the X-nucleus channel on systems in which that power is not under computer control; this can have the unavoidable consequence of lengthening the pulse widths of the X-nucleus pulses.

Dual-broadband systems have no problem performing modulated decoupling, because the decoupling is being performed by the normal spectrometer decoupling channel. On single-channel broadband systems, however, the decoupling is being performed by the normal observe channel, and the standard modulation (WALTZ, for example) is not present.

In the sequences described here, broadband decoupling is achieved by using the acquisition computer to provide WALTZ-4 modulation of the X-nucleus channel through explicit software control. It also imposes some limitation on spectral widths and pulse widths, since the WALTZ-4 sequence (whose length is `6*pw90`) must fit in between successive data point samplings (which occur at time intervals of `1/sw`).

When WALTZ decoupling is used, the maximum power level for decoupling is the level that provides an rf field strength (in Hz) comparable to half the range of expected X shifts. The normal spread of protonated carbons is 150 ppm, which is 15 kHz on a 400-MHz system, and, consequently, a ^{13}C 90° pulse of 25 µs (corresponding to an rf field strength of 8 kHz) is adequate. The somewhat long proton pulses on broadband and switchable probes does not seem to present a problem because indirect detection experiments demand no more proton pulse power than DEPT or HETCOR.

11.6 ^{15}N Indirect Detection

Calibrations and operations for ^{15}N proceed largely along the lines outlined above for ^{13}C. In the standard sample, 2% ^{15}N-benzamide (Part No. 00-968120-97), the ^{15}N satellite lines are partially obscured by other resonances in the conventional 1D spectrum, and so the ^{15}N pulse width calibration must be done using multi-transient HMQC experiments as described in "Typical Experimental Protocol for HMQC Experiments," page 205. Be sure to use a J appropriate for NH couplings (90 Hz) in this case.

A step that can often be done in ^{15}N work of peptides is to make sure that the γB_2 is sufficient to decouple the relatively narrow range of ^{15}N chemical shifts expected in such samples but no more. This minimizes heating effects and improves cancellation. Typical acquisition times (`at`) are 0.075 to 0.1 seconds.

11.7 Pulse Sequences

Indirect detection experiments use the HMQC or HSQC pulse sequences.

- "HMQC Pulse Sequence," this page
- "HSQC Pulse Sequence," page 215

HMQC Pulse Sequence

The `hmqc<(isotope)>` macro sets up parameters for a HMQC (heteronuclear multiple-quantum coherence) pulse sequence. The optional `isotope` argument is the isotope number of the heteronucleus of interest, for example, `hmqc(1)` for ^{1}H (the default is ^{13}C). Figure 48 is a diagram of this sequence. The first `2*pwx` pulse on the X heteronucleus is a composite 180 consisting of 90(`v9`) – 180(`v1`) – 90(`v9`).

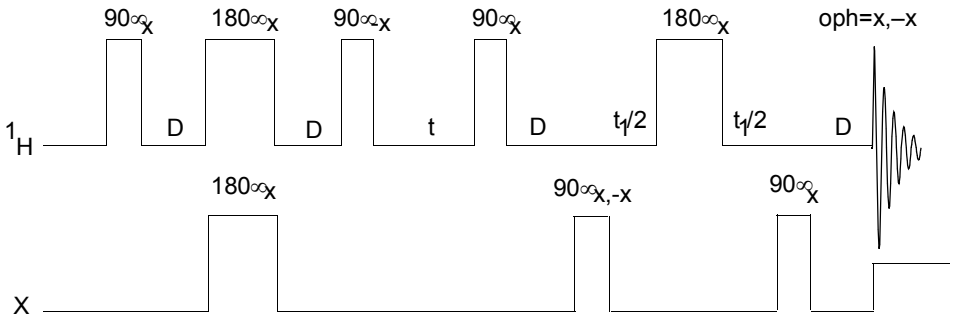


Figure 47. Basic HMQC Pulse Sequence

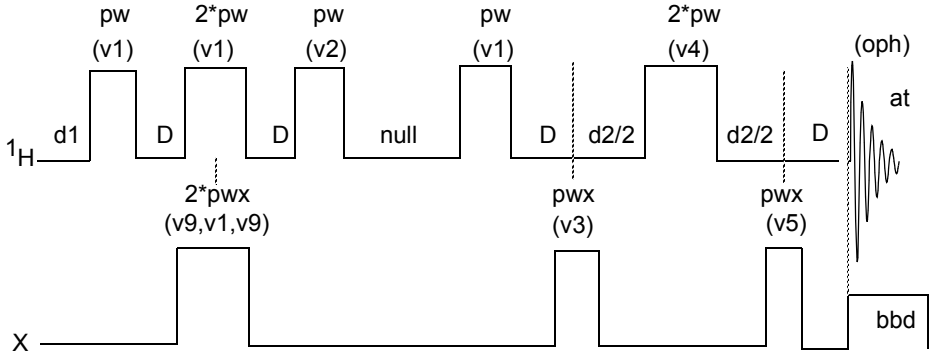


Figure 48. HMQC Pulse Sequence with `null`<>0 and `mbond='n'`

Phase-Sensitive Aspects of the Sequence

The parameter `phase`, as in other phase-sensitive 2D experiments, controls the f_1 phase detection. For 1D setup experiments or a 2D experiment without quadrature detection in f_1, use `phase=1`. For a normal 2D experiment using the States-Haberkorn-Ruben (hypercomplex) method, use `phase=1,2`. To acquire data with TPPI, use `phase=3`, and make sure `sw1` is twice the expected range.

The FAD, for "F1 Axial Displacement," technique (Marion, D.; Ikura, M.; Tschudin, R.; Bax, A. *J. Magn. Reson.* **1989**, *85*, 393) involves a change of phase cycling that shifts the axial artifacts in a hypercomplex experiment to the edge of the spectrum, giving the hypercomplex version the benefit of TPPI with none of the disadvantages. It is also referred to as "States-TPPI." The `hmqc` macros include FAD. Once implemented, use of the technique is totally transparent—just perform a standard hypercomplex experiment with `phase=1,2`.

Parameters

Table 19. Parameters for HMQC Pulse Sequences

Parameter	*HMQC*
^{1}H 90° pulse	`pw`
^{1}H 180° pulse	derived from `pw`
^{1}H amplifier power (if appropriate)	`tpwr`
^{1}H frequency	`tn, tof`
^{1}H spectral width	`sw`
X 90° pulse	`pwx`
X 90° pulse for WALTZ decoupling	`1/(4¥dmf)`
X 180° pulse	derived from `pwx`
X amplifier power for pulses (if appropriate)	`pwxlvl`
X amp power for decoupling (if appropriate)	`dpwr`
X frequency	`dn, dof`
X spectral width	`sw1`
Δ delay	`1/(2j) [if j=0, D=0]`
τ delay for BIRD nulling (if `null=0`, entire BIRD sequence is skipped)	`null`
Coupled experiment	`dm='nnn'`
X decoupling during acquisition.	`dm='nny'`
Setup experiments	`phase=1`
Hypercomplex experiment	`phase=1,2`
TPPI	`phase=3`
Minimum `nt` possible	`2`
Presaturation and/or multiple-bond correlation	see text
Axis parameter for proper ppm on both axes	`pd`

`pw` is a 90° pulse on the observed nucleus (protons) at power equal to `tpwr`.

`pwx` is a 90° pulse on the heteronucleus at power equal to `pwxlvl`.

`dpwr` is the decoupler power level for broadband X-decoupling.

`dmf` sets the modulation frequency ($4*\gamma B_1$) at decoupler power (`dpwr`).

`dmm` is decoupler modulation mode. `dmm='ccg'` is recommended.

`dm='nny'` activates heteronuclear broadband decoupling (recommended) during acquisition. Note that `dm` can be set to either `'nnn'` or `'nny'`, and that the duty cycle for the decoupler should be less than 20%.

`j` is the average scalar coupling constant between the protons and the heteronucleus (usually one-bond constants). `j` is 140 for ^{13}C or 90 for ^{15}N. The time Δ, shown in Figure 48, is calculated as `1/2j`.

`null` is a WEFT-like delay used to improve the suppression of the protons connected to ^{12}C (and not to ^{13}C) that have been inverted by the preceding BIRD pulse. Try a `null` value of 0.3 for ^{13}C, 1.0 for ^{15}N, and 0 for macromolecules. To optimize, set `ss=-8` and array `null` with `nt=1` and `phase=1`. This selects the value of `null` that best minimizes the sample's signals (typically 0.2 to 0.7 seconds). If `null` is set to 0, the BIRD element is omitted from the pulse sequence.

`at` is the acquisition time (t_2 period).

`ni` is the number of t_1 increments (set up with default values for either ^{13}C or ^{15}N).

`ss` is the number of complete executions of the pulse sequence not accompanied by data collection prior to the acquisition of the real data: if `ss` is positive, `ss` steady-state pulses are applied on the first increment only; if `ss` is negative, –`ss` transients are applied at the start of each increment.

`nt` is a multiple of 4 (minimum) or multiple of 8 (recommended).

`phase=1,2` (2D hypercomplex data with hypercomplex-TPPI method) or `phase=3` 2D TPPI data). `phase=1,2` is suggested. For `phase=3`, remember that `hmqc` sets `sw1` to *twice* the desired value for heteronuclear experiments.

`satflg='yn'` gives presaturation during `satdly`, and `satflg='yy'` gives presaturation during `satdly` and `null` (not on *MERCURYplus/-Vx*).

`satfrq=x` is the presaturation frequency (using the transmitter), `satdly` is the length of saturation time during the relaxation period (immediately after `d1`), `satpwr` is the power level for presaturation using the transmitter (not on *MERCURYplus/-Vx*).

`hs='yn'` gives a homospoil pulse at beginning of `d1` (length=`hst`). `hs='yy'` gives a homospoil pulse at beginning of both `d1` and `null`.

`taumb` is a fixed delay associated with the multiple-bond HMQC experiment (`taumb=0.055` is recommended).

`mbond='n'` is a normal HMQC experiment. `mbond='y'` is a multiple-bond HMQC experiment (HMBC).

To run HMBC (`mbond='y'`): (1) set `null=0`, otherwise, only protons that are both long-range and short-range (one-bond) coupled to a given heteronucleus (^{13}C, for example) will not be suppressed, (2) set `dm='nnn'`, (3) set `taumb`, and (4) run the single-bond (HMQC) and multiple-bond (HMBC) experiments with `phase=1,2` or `phase=3`.

Phase Cycling

The phase cycling is the following:

`v1`, `v2`, `v3`, `v4`, `v5`, `v9` are phases for pulses. `oph` is the phase for receiver.

```
v1   =  x  x  y  y
v2   = -x -x -y -y
v3   =  x -x  y -y
v4   =  x  x  y  y  y  y -x -x
v5   =  x  x  y  y  x  x  y  y
v9   =  y  y -x -x
oph  =  x -x  y -y
```

These phases are for `phase=1`. For `phase=2`, add 90° to `v3`. For `phase=3`, add 90*(`ix` - 1)° to `v3`, where `ix` is the increment counter.

Technique

The usual setup is to place a ^{1}H bandpass filter between the observe port on the probe and the $^{1}H/^{19}F$ preamplifier, and to place a 250-MHz lowpass LC filter and either a ^{13}C bandpass or a ^{15}N bandpass filter in the decoupler line just before the probe connection.

The experiment should be performed non-spinning and with VT regulation.

HSQC Pulse Sequence

The `hsqc` macro sets up parameters for the HSQC pulse sequence, a heteronuclear Overbodenhousen experiment using REVINEPT.

Parameters

`sspul='y'` selects for *trim(x)-trim(y)* sequence at the start of the pulse sequence; `sspul='n'` selects a normal experiment.

`satmode='yn'` gives presaturation during relaxation period (`satdly`) with the transmitter; `satmode='nn'` gives no presaturation during relaxation period (`satdly`); `satmode='ny'` gives presaturation during only the null period.

`satfrq` sets the presaturation frequency.

`satdly` sets the saturation time during the relaxation period.

`satpwr` sets the saturation power for all periods of presaturation with xmtr.

`hs='yn'` sets a homospoil pulse (`hst`) during the d1 relaxation delay.

`null` is the delay associated with the BIRD nulling.

`tpwr` is the power level for ^{1}H transmitter pulses.

`pw` is a 90° transmitter pulse length for protons (the observed nucleus).

`pwxlvl` is the power level for X decoupler pulses.

`pwx` is a 90° decoupler pulse length for X.

`jxh` is a one-bond heteronuclear coupling constant to X (in Hz).

`phase=1,2` for hypercomplex experiment with F1 quadrature (complex F1-FT).

Chapter 12. Pulse Analysis

Sections in this chapter:

12.1 Pandora’s Box

Pandora’s Box (Pbox) software creates shape pattern files for experiments involving shaped rf pulses, composite pulses, decoupling and mixing patterns, adiabatic rf sweep waveforms, and pulsed field gradient shapes. The goal of Pbox is to simplify generation and use of different waveforms in NMR experiments to a level where the user does not need to be an expert in selective excitation. Pbox makes the use of complex waveforms as simple as using ordinary rectangular pulses. Indeed, not only does Pbox provide all the necessary parameters (pulse width, power, `dmf`, `dres`, etc.) when the shape files are created, but this information can be extracted at any time from Pbox shape files by macros or directly within pulse sequences. More than 160 different shapes are available from the Pbox library.

Pbox Window

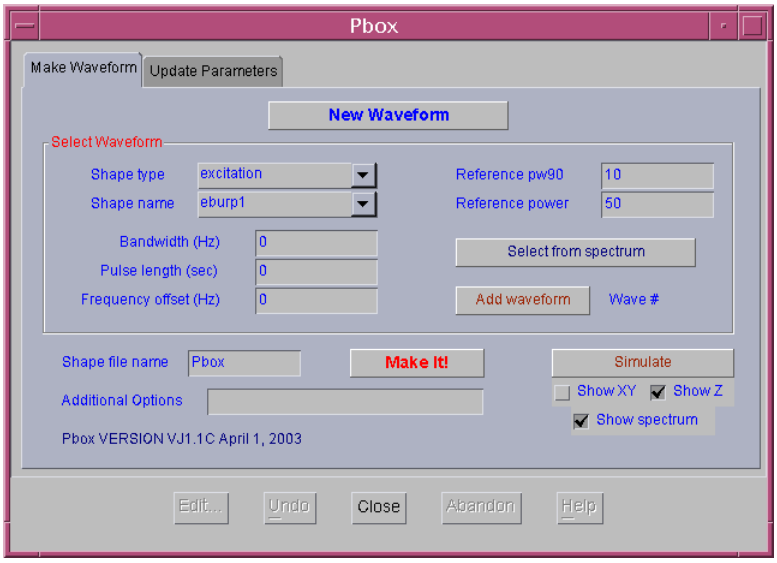
The Pbox window provides tools for creating waveforms.

Click **Process->Pbox** from the menu bar. The Pbox window opens.

Create a New Waveform

1. Click the **Make Waveform** tab.

2. Click the **New Waveform** button.



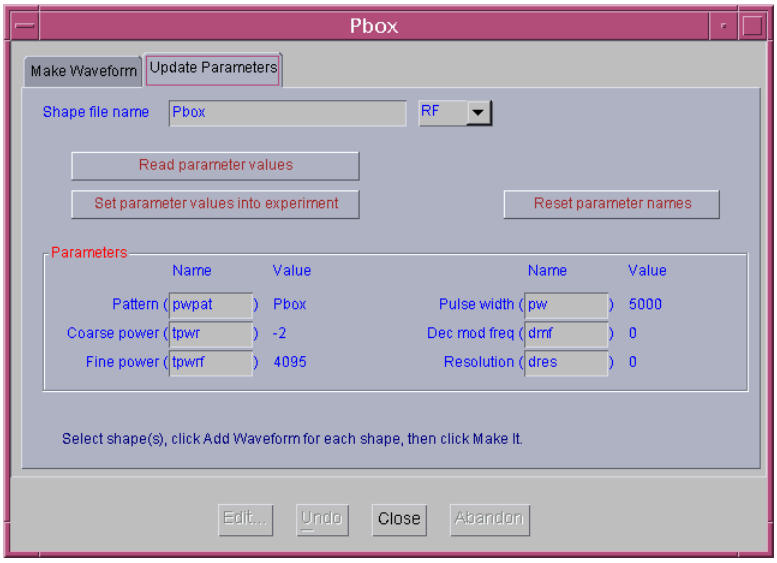
3. Select the desired shape from the **Shape type** menu.
4. Select the desired shape name from the **Shape name** menu. The Shape name choices depend on the selected Shape type.
5. Set the **Reference pw90** and **Reference power** from the 90-degree pulse length and power.
6. Set the waveform selection region as follows:
   a. Display a spectrum in the graphics window.
   b. Select a selective excitation band using two cursors.
   c. Click the **Select from spectrum** button. This sets the **Bandwidth**, **Pulse length**, and **Frequency offset** from the cursors.

   You can also explicitly enter Bandwidth, Pulse length, and Frequency offset.
7. Click **Add Waveform**. The Wave # field updates.
8. To set multiple waveform shapes into a waveform, repeat steps 3-7 for each desired waveform. The Wave # field updates for each selective shape added to the waveform.
9. Enter the **Shape file name**.
10. Enter **Additional Options** as desired (refer to Table 20 for Pbox commands and parameters).
11. Click **Make It!** to generate the shape file.
12. To simulate the effect of the waveform, do the following:
    a. Display a spectrum.
    b. Select **Show XY** or **Show Z**.
    c. Select **Show spectrum if** you want to show the spectrum.
    d. Click **Simulate**. The simulation displays in the graphics window.

Update Parameter Values

Update the parameter values used by the waveform into experimental parameters as follows:

1. Click the **Update Parameters** tab.



2. Enter the desired **shape file name** in the entry box.
3. Click **Read parameter values** to read parameter values from the Pbox shape file.
4. In the Parameters area, set the names of the parameters you wish to use for the waveform: Pattern, Coarse and Fine power, Pulse width, Decoupler modulation frequency, and Decoupler resolution.
5. Click **Reset parameter names** to reset the parameter names to the default values, if desired.
6. Click **Set parameter values into experiment** to set the parameter values into the experimental parameters. If a parameter does not exist, it is not set in the experiment. Parameter limits are not checked (especially for pulse width)--you will need to check them.

Calibrating the RF Field

To obtain the pulse calibration numbers in the Pbox output, provide the rf field calibration data `ref_pwr` and `ref_pw90` in the input. Therefore, before waveform creation, make sure the rf field has been calibrated and you know the length of the 90° pulse at a given power level.

If the spectrometer amplifiers are linear, which is standard on Varian NMR spectrometer systems, it does not matter at what power level the rf field is being calibrated. However, for maximum accuracy, do the calibration close to the field used in the experiment. An estimate of the rf field is obtained by providing approximate calibration data and using `cal` as an output file name. No waveform is created in this case, and only the calibration results appear in the output.

Creating Waveforms from Macros

Pbox macros provide useful tools for customizing NMR experiments. The simplest way to create a shape is using the pxshape macro. For example, a single band excitation pulse using the E-BURP-1 shape, covering 400 Hz, and shifted off-resonance by –880 Hz from the carrier frequency (middle of the spectrum) can be created and stored in the `alpha.RF` file as follows:

```
pxshape('eburp1 400.0 -880','alpha.RF')
```

The following steps are necessary to create multiply-selective pulses. If the spectrum of interest is on the screen, use the cursors.

1. Enter **`opx('hadamard.RF')`** to open the `Pbox.inp` file and write the file header.
2. Select an excitation band using cursors.
3. Enter **`selex('rsnob')`**.
4. Select the second excitation band using cursors.
5. Enter **`selex('rsnob')`**.
6. Repeat steps 2 to 5 as many times as needed.
7. Enter **`cpx`** to close the `Pbox.inp` file.
8. Enter **`dshape`** to display the last created shape.

If an experimental spectrum is not available, the following slightly different set of macros are used:

1. Enter **`opx('myshape')`** to open Pbox and provide a file name.
2. Enter **`setwave('sech 400.0 -880.0')`** to select first band at –880 Hz.
3. Enter **`setwave('sech 400.0 1240.0')`** to select second band at 1240 Hz.
4. Enter **`cpx(ref_pw90, ref_pwr)`** to close Pbox.
5. Enter **`dshape`** to display the shape file.

The `pbox_pw` and `pbox_pwr` macros are used to load the parameters of the last created shape file into the current experiment:

```
pbox_pw:selpw
pbox_pwr:selpwr
```

Alternatively, the calibration data is directly retrieved from the shape file provided as an argument to the `pbox_dmf` and `pbox_dres` macros:

```
pbox_dmf('ccdec.DEC'):dmf
pbox_dres('ccdec.DEC'):dres
```

where `ccdec.DEC` is the name of the decoupling shape file.

The excitation profile of shaped pulses is conveniently verified using the Pbox Bloch simulator:

1. Enter **`opx`** to open Pbox.
2. Enter **`setwave('iburp2 400.0 -880')`** to select first band at –880 Hz.
3. Enter **`setwave('iburp2 400.0 1240.0')`** to select second band at 1240 Hz.
4. Enter **`pbox_rst`** to reset par-s and write comments.
5. Enter **`pboxpar('name','test.DEC')`** to define the output file name.

6. Enter **pboxpar('bsim','y')** to activate the Bloch simulator.
7. Enter **cpx** to close Pbox.
8. Enter **dshape** to display the shape file.
9. Enter **dprofile('z')** to display inversion (Mx) profile.

In the vast majority of cases, you don't have to tell Pbox whether you are creating a 90° excitation pulse or 180° inversion pulse, or even whether an .RF, .DEC, or .GRD type of waveform is needed, because this information is stored with the corresponding wave file in the `wavelib` directory. Pbox can be forced to change the waveform type by you simply providing the required extension to the output shape file name. Wave files are modified, if necessary, by copying it into your `wavelib` and editing the text file as required. See "Pbox Macro Reference," page 228, for a more complete description of macros.

Creating Waveforms from UNIX

It is sometimes more convenient to create waveforms from the UNIX shell:

```
> Pbox
```

The name of the output shape file is passed as an argument:

```
> Pbox filename
```

The input data are typically stored in the `Pbox.inp` file in your `vnmrsys/shapelib` directory and are modified using standard text editors. Alternatively, most of the necessary data can be provided as arguments to the `Pbox` command. For example,

```
> Pbox myfile -w 'eburp1 480 -1200' -p 40 -l 104
```

generates an E-BURP-1 excitation pulse covering 480-Hz-wide band and shifted –1200 Hz off-resonance using for calibration 104 µs long pw90 at 40 dB power level and stored in `myfile.RF`. Note that the name of the output shape file is always passed as the first argument.

Several other options are accepted by `Pbox`; for example, `-b` activates the Bloch simulator, `-c` calibrates the waveform without creating a shape file, and `-o` print out the available options. (see "Pbox UNIX Commands," page 229, for further information).

Pbox File System

All the information about the waveform to be created (e.g. calibration data, output file name, excitation band definition) is stored in the `Pbox.inp` text (ASCII) file in the user directory `vnmrsys/shapelib`. This file is generated whenever Pbox menus or macros are used. You can also create it by using one of the standard text editors.

Any shape file can consist of one or several shaped pulses that are combined into a single waveform. Each excitation band is defined by a wave definition string (a string of wave variables enclosed between delimiters {and}). The number of wave definition strings in a single `Pbox.inp` file is unlimited. In order to simplify the input file format, the wave variables are entered without names in a strongly predefined order:

```
sh bw(/pw) ofs st ph fla trev d1 d2 d0 wrap
```

The following list describes each of the variables.

`sh`	Shape name as stored in `wavelib`
`bw(/pw)`	Bandwidth in Hz, or pulsewidth in sec, or both
`ofs`	Offset from transmitter offset or carrier in Hz

st	Spin status (0 for Mz or 1 for Mxy)
ph	Phase or phase cycle
fla	Flip angle
trev	Time reversal flag
d1	Prepulse delay
d2	Postpulse delay
d0	Delay before the pulse
wrap	Wraparound parameter

The order of parameters has been chosen such that the importance of parameters is decreasing and rarely used parameters can be omitted or defaulted by assigning a value of n (not used). The following examples are valid wave definition strings.

{qsneeze}	q-SNEEZE pulse applied on resonance, the pulse length will be internally defaulted to 5 ms.
{G3 800}	G3 pulse covering bandwidth of 800 Hz and applied on-resonance.
{sech 400/0.05 -1200}	50 ms long hyperbolic secant pulse covering 400Hz and shifted off-resonance by –1200 Hz.
{WURST2 2k/5m 12k n t5}	5 ms long WURST-2 decoupling pulse covering 2 kHz and shifted off-resonance by 12 kHz uses t5 phase cycle.
{eburp1 450 0.0 n 180} {eburp1 450 820 1 0.0}	Two E-BURP-1 pulses mixed in a single waveform, both covering 450 Hz wide band. The first pulse is applied on-resonance with a phase of 180 deg. The second pulse is shifted to 820 Hz of-resonance, has zero phase and is a de-excitation pulse (status 1). By default such a pulse is time reversed.

A set of Pbox parameters can be used to define the waveform to be generated. The syntax of the Pbox.inp file is straightforward, parameter=value, for instance, name=myshape.RF, or simply name=myshape. The following list describes Pbox parameters and their default values (see "Pbox Parameters," page 225, for more details):

name=Pbox	Shape file name, the extension is optional.
type=r	Shape type, r - RF, d - DEC, g - GRD.
dres=9.0	The default value is stored in wavefile.
steps=200	Minimum number of steps (< 64k). The default value is stored in a wave file.
maxincr=30	Max phase incr, deg (<<180).
attn=i	Attenuation, i (internal), e (external) or d (nearest dB step)
sfrq=0	Spectrometer frequency, MHz.
refofs=0	Reference offset, Hz (/ppm).
sucyc=d	Super Cycle, d (default), n (no), name as in wavelib/supercycles. The default value is stored in wavefile.
reps=2	Amount of reports (0-4).
stepsize=n	Size of a single step (ms).
wrap=0	Wraparound parameter (0-1).
header=y	Shape header, y (yes) n (no) i (imaging).
bsim=n	Bloch simulation, y (yes), n (no), a (add), s (subtract), 200 (time in sec).
T1=n	Relaxation time T1 (sec).

T2=n	Relaxation time T2 (sec).
dcyc=1	Duty cycle (0 - 1).
sw=0	Spectral width (Hz)
ptype=selective	pulse type (for imaging only).

The number and order of input parameters is optional and not important.

You can redefine the internally defaulted Pbox parameters by entering the default values in the .Pbox_globals file.

Parameters describing software and hardware limitations are also pre-defined internally and can be redefined by the user in the .Pbox_globals file that is stored in user's home directory. The following list describes global parameters and their default values.

shdir=$HOME/vnmrsys/shapelib/	default shape directory
wvdir=/vnmr/wavelib	default wave directory
maxst=65500	maximum number of steps in waveform
defnp=100	default number of steps
minpw=0.2	minimum step length, in µs
minpwg=2.0	minimum gradient step length, in µs
drmin=1.0	minimum dres
maxamp=1024.0	maximum amplitude
maxgr=32767.0	maximum gradient amplitude
amres=1.0	amplitude resolution
phres=0.1	phase resolution, in degrees
tmres=0.05	time resolution, in µs
dres=9.0	default dres
maxpwr=63	maximum power level, in dB
minpwr=-16	minimum power level, in dB
maxitr=5	maximum number of iterations
maxdev=2.0	maximum deviation, in percent
cmpr=y	waveform compression
minstps=64	minimum steps in Bloch simulation
pw=0.005	default .RF and .DEC pulse length, in sec
pwg=0.001	default .GRD pulse length, in sec

The parameters of individual shapes—Gaussian, E-BURP-1, or hyperbolic secant pulse, etc.—are stored in the wavelib directory, which has several subdirectories, such as excitation, inversion, refocusing. Every individual shape is defined by a set of parameters that can be grouped in several categories.

Wave definition parameters are the following:

amf	amplitude modulation function
fmf	frequency modulation function
su	default supercycle
fla	default flip angle on resonance
pwbw	pulsewidth to bandwidth product
pwb1	pulsewidth to B1max product
pwsw	pulsewidth to sweepwidth product
adb	adiabaticity on resonance
ofs	offset of excitation bandwidth

dres	default tipangle resolution, in degrees
dash	dash variable
wf	window function
st	default status
dutyc	duty cycle
c1	constant
c2	constant
c3	constant
steps	default number of steps

Wave truncation parameters are the following:

min	minimum truncation threshold (0 to 1)
max	maximum truncation threshold (0 to 1)
left	truncation from left (0 to 1)
right	truncation from right (0 to 1)
cmplx	flag, retain real (1), imag (–1) or complex(0) part of wave
wrap	wraparound factor (0 to 1)
trev	time reversal flag (yes = 1, no = 0)
srev	frequency sweep reversal flag (0 to 1)
stretch	stretching factor (>= 0)
dcflag	dc correction, y/n

Additional parameters are usually data matrices, such as Fourier coefficients or square wave parameters e.g. length, phase, amplitude, etc. These matrices are listed without parameter names. The size of the data matrix given is defined by:

cols	number of columns
rows	number of rows

Pbox incorporates the following amplitude modulation (AM) functions:

sq	square (constant amplitude)
sqa	square wave amplitude modulation (used for “composite” pulses)
gs	Gaussian
lz	Lorentzian
sch	sech (hyperbolic secant)
hta	tanh (hyperbolic tangent)
tra	triangular amplitude (ramp)
sc	sinc function
csp	cosine power
wr	wurst (wideband uniform rate smooth truncation)
sed	seduce-1, mixture of sech and sin
qp	quadrupolar
ata	amplitude mod for CA atan frequency sweep pulse
exa	exponential amplitude
tna	tangential amplitude
fs	Fourier Series
ft	inverse Fourier Transform

Pbox incorporates the following frequency modulation (FM) functions:

ls	linear sweep (chirp)
tns	tangential sweep (tan)
ht	hyperbolic tangent sweep (tanh)
lzs	constant adiabaticity Lorentzian sweep
ca	constant adiabaticity (CA) sweep (frequency modulated frame)
cas	constant adiabaticity sweep (phase modulated frame)
cs	cosine / sine pulse frequency sweep
cs2	CA cosine square frequency sweep
ccs	CA cosine frequency sweep
sqw	squarewave phase modulation
fsw	frequency switch (step function)
fslg	frequency switched as per Lee-Goldburg

Pbox Parameters

The following list describes Pbox parameters.

name	Name and extension of the output shape file. If the extension is not given, the shape type is set according to the type parameter. The default name is internally set as Pbox. This can be changed in the .Pbox_globals file.
type	Shape type. Allowed values are r (.RF type), d (.DEC) or g (.GRD). If the shape type is not defined and the shape file is given without an extension, the shape file type is determined from the wave file according to the following criteria: • type is set to r if pwbw > 0.0. • type is set to d if dres > 0.0. • type is set to g otherwise.
dres	Corresponds to dres parameter in VMNR. Active only with .DEC files.
steps	Defines the required number of steps in the waveform. The default number of steps is stored with each individual shape in the corresponding wave file. This number can be overridden by Pbox if it is smaller than the internally calculated minimum number of steps, which is necessary to maintain the functionality of the waveform. This number is defined according to the following criteria: • By the minimum number of steps necessary for adequate representation of the waveform (as in wave file). • If the waveform is shifted off-resonance, by the Nyquist condition (see maxincr).
maxincr	Maximum phase increment. By default, set to 30°. This number is active only if the waveform is shifted off-resonance or the shape itself is frequency modulated (e.g., adiabatic sweeps). In order to satisfy the Nyquist condition, maxincr should not exceed 180°, otherwise the waveform gets folded back. In fact, the degradation of performance and interference with sidebands can be observed even with a maxincr of greater than 90°, but a maxincr of less than 90° is recommended.

attn	Fine attenuation mode, which uses the following allowed values: i (Internal), default.Fine attenuation is implemented by internally rescaling the waveform within the amplitude range set by maxamp (0 to 1023). e (External) Fine attenuation is implemented by externally rescaling the waveform using linear modulators. The internal amplitude is set to maxamp (1023.0) and the required fine attenuator setting is produced in the output. d Attenuate to the nearest dB step by changing the pulse width, which will affect the excitation bandwidth typically within 5%, which is tolerable in most applications. The internal amplitude is set to maxamp (1023.0) 4.5i Internally attenuate to a given (4.5 kHz) B1 field strength by adjusting the pulse length. 4.5e Externally attenuate to a given (4.5 kHz) B1 field strength by adjusting the pulse length. 4.5I Internally attenuate, keeping course power level at a given (45 dB) power level. 4.5E Externally attenuate (with fine power), keeping course power level at a given (45 dB) power level. 45d Attenuate to a given (45 dB) power level by changing the pulse width. The internal amplitude is set to maxamp (1023.0).
sfrq	Spectrometer frequency in MHz.
refofs	Reference offset, usually 0.0. Can be specified if the excitation bands are shifted by or referenced to some frequency. Units: Hz, kHz, or ppm (if sfrq is defined).
sucyc	Super cycle. Allowed values are n (no), d (default) or any name of a super cycle stored in the wavelib/supercycles directory. By default, it is internally set to d. Super cycles can be nested by separating the names with a comma, for example, t5,m4 represents 5 step TPG super cycle nested in four step MLEV-4 super cycle.
reps	Defines level of reporting. Allowed values are 0-4: 0=silent, 1=single line, 2=minimum, 3=medium, 4=maximum. The default is 2.
stepsize	The length of a single step in a waveform. The default units are µs. Note that stepsize disables the maxincr parameter.
bscor	Initiates correction for Bloch-Siegert effect in multiple band excitation, inversion or refocusing pulses. Allowed values are y (yes) or n (no, default). Active only if the number of bands is two or more. Reduces the rf interference effects (see M. Steffen, L.M.K. Vanderseypen and I.L.Chuang, Abstracts of the 41st ENC, p. 268, Asilomar 2000).
wrap	Wraparound parameter. It allows wrapping around the waveform. The allowed values are between 0 and 1.0.
header	Shape file header. Allowed values are y (yes, default), n (no shape file header) and i (imaging). Information required for imaging systems is stored in the shape file header.
bsim	Bloch simulator. Performs Bloch simulation for the given waveform at the moment of waveform generation. Allowed values are y (yes), n (no, default), a (add to the previous simulation), s (subtract from the previous simulation) and any positive integer limiting the simulation time in seconds. The default maximum length of simulation is internally set to 60 seconds and can be redefined in the .Pbox_globals file. Note, that Bloch simulator can also be externally activated, e.g., from menus or using the dprofile macro.
T1	Longitudinal relaxation time, T1 in seconds. Can be required by some waveforms (e.g. SLURP pulses). Optional for the Bloch simulation.
T2	Transversal relaxation time T2, in seconds. Can be required by some waveforms (e.g. SLURP pulses). Optional for the Bloch simulation.

dcyc	Duty cycle. Usually required for homonuclear decoupling applications. Only values between 0.0 and 1.0 are active. Outside these boundaries dcyc is reset to 1.0 (default).
sw	Spectral width. If given, the step size of waveform is set equal to the dwell time (1/sw). Recommended for H-H homo-decoupling applications. It also helps to make sure that excitation sidebands are kept outside the spectral window. Also required for Bloch simulation.
ref_pw90	Reference 90° pulse width (in µs) at ref_pwr. Required for calibration of waveforms. If set to 0.0, the maximum B1 field intensity (in kHz) is reported instead of the power setting.
ref_pwr	Reference power level (in dB steps). See ref_pw90.
ptype	Pulse type. Only necessary with imaging header. By default, set to selective.

Wave String Variables

A reminder is given in Pbox.inp files generated by menus and macros because these parameters appear without names. The wave string variables are listed as they appear in the reminder.

sh	Shape name as in wavelib.
bw/pw	Bandwidth and/or pulsewidth. For most waveforms, only one of the two parameters is required. Pbox distinguishes between bw (in Hz), which is always greater than 1.0, and pw (in sec), which is always less than 1.0. It is up to you which of the two parameters to provide for input, because they are mutually related via the pw*bw product, which is stored with each individual shape in wavelib. Some waveforms (e.g., adiabatic sweep pulses) can require both bw and pw. In such cases, both variables can be provided in a single string using the "/" separator. For example, {WURST2 200.0/0.05} denotes a 50-ms long WURST-2 pulse covering 200 -Hz-wide band. Alternatively, units can be used for clarity, e.g., {WURST2 0.2k/50m}. If the sfrq parameter is defined, bandwidth can also be specified in ppm, e.g., {WURST 20p/5m}.
ofs	Offset of the center of the excitation band in Hz with respect to the carrier frequency (middle of the spectrum). Note that if the sfrq spectrometer frequency, (in MHz) is defined, ofs can also be specified in ppm. In order to specify ofs in terms of absolute frequency, the reference offset refofs (i.e., chemical shift value of carrier frequency) must be defined. For instance, {WURST2 20p/5m 170p} sfrq=225.0 refofs=55p.
st	Spin status. Defines whether the waveform is used for excitation (st=0), refocusing (st=0.5) or de-excitation (st=1), which, in turn, defines whether the wave starts with phase defined by ph (st=1), the ph occurs in the middle of the pulse (st=0.5), or the pulse ends with phase ph (status 0). In addition, the waveforms are time reversed if status is 1, as required for proper de-excitation. Undesired time reversal can be undone using the trev parameter. Furthermore, if several waves of different width are generated, they are bound to the beginning (st=1), middle (st=0.5), or end (st=0) of the waveform. The spin status of the first wave is also used by Bloch simulator as the starting magnetization.
ph	Phase in degrees or phase cycle (super cycle). Usually phase is externally set in the pulse program and this parameter is not required. You can also apply any phasecycle (super cycle) from wavelib/supercycles. The difference between this phase cycle and the sucyc parameter is that phase cycling is carried out before waveform mixing and is therefore independent of other Super cycles, whereas sucyc is applied to the final (mixed) waveform. In this way, several waves of different width can be independently phase cycled and use different super cycles.

fla	Flip angle, in degrees. Usually, fla is defined in the wave file and there are very few applications where intermediate flip angles are required.
trev	Time reversal flag (see st). Allowed values are y (yes) and n (no, default).
d1	Prepulse delay, in seconds. Normally not required. If defined, it disables the internal wave shifting according to the spin status.
d2	Postpulse delay, in seconds. Normally not required. If defined, it disables the internal wave shifting according to the spin status.
d0	Pre-d1 delay, in seconds. Essentially repeats d1. It is used only for convenience, e.g., if internal duty cycle is defined in shape parameters in wavelib. If set to 'a', the wave is appended to the previous wave.
wrap	Wraparound parameter. Can take values between 0 and 1.0.

Pbox Macro Reference

Although most of needs for generating selective pulses can be satisfied by using the Pbox window, a set of macros is provided for those who them. The following table lists the macros in the order of decreasing importance. For additional information on Pbox macros, refer to the manual *Command and Parameter Reference*.

pboxvnmrj	Opens the Pbox dialog window.
opx	Opens Pbox, writes the Pbox.inp file header, and resets parameters r1-r7 and n1-n3.
selex	Defines the excitation band from the position of cursors in the graphics window and reports them to the user. It also sets r1 to excitation bandwidth and r2 to offset. selex uses the pbox_bw and putwave macros.
cpx	Calls the Pbox command, which generates the specified waveform as defined by the Pbox.inp file. cpx also checks if parameters ref_pwr and ref_pw90 exist in the current experiment and puts their values into the Pbox.inp file. If the parameters do not exist, cpx creates them and asks the user for parameter magnitudes.
setwave	Sets up a single excitation band in the Pbox.inp file. An unlimited number of waves can be combined by reapplying setwave.
putwave	Sets up a single excitation band in the Pbox.inp file. An unlimited number of waves can be combined by reapplying putwave.
pxshape	Generates a single-band waveform based on wave definition provided as a single string of wave parameters.
pboxpar	Adds a parameter definition to the Pbox.inp file.
pboxget	Extracts calibration data from the file shapefile.ext generated by Pbox or, if the file name is not provided, from the pbox.cal file containing parameters of the last created Pbox shape file. Note that the parameter is not changed by this macro if it was set to 'n' (not used)!
pbox_pw	Extracts pulse length from the file shapefile.RF generated by Pbox or, if the file name is not provided, from pbox.cal file containing parameters of the last created Pbox shape file.
pbox_pwr	Extracts the power lever from the file shapefile.ext generated by Pbox or, if the file name is not provided, from the pbox.cal file containing parameters of the last created Pbox shape file. Note that the parameter will not be changed by this macro if previously set to 'n' (not used).

pbox_pwrf	Extracts the fine power lever from the file shapefile.ext generated by Pbox or, if the file name is not provided, from the pbox.cal file containing parameters of the last created Pbox shape file. Note that the parameter will not be changed by this macro if it was previously set to 'n' (not used).
pbox_dmf	Extracts the dmf value from the file shapefile.DEC created by Pbox or, if the file name is not provided, from the pbox.cal file containing parameters of the last created Pbox shape file.
pbox_dres	Extracts the dres value from the file m shapefile.DEC created by Pbox or, if the file name is not provided, from the pbox.cal file containing parameters of the last created Pbox shape file.
pbox_name	Extracts name of the last shape file generated by Pbox and stored in the pbox.cal file. Note, that the file name extension is not stored explicitly and is not provided by this macro.
dshape	Displays real (X) and imaginary (Y) components of a shaped pulse. Any type of waveform (.RF, .DEC or .GRD) can be displayed.
pshape	Generates a single-band waveform based on wave definition provided as a single string of wave parameters.
dshapef	Displays the real (X) and imaginary (Y) components of last generated shaped pulse, stored in pbox.fid file.
dshapei	Interactively displays the real (X) and imaginary (Y) components of last generated shaped pulse, stored in pbox.fid file
dprofile	Displays the X, Y, and Z excitation (inversion) profile for a pulse shape generated by the Pbox software.
pprofile	Plots the X, Y, and Z excitation (inversion) profile for a pulse shape that has been generated with the Pbox software. If a shape name is not provided, the last simulation data stored in shapelib/Pbox.sim are plotted.
pph	Prints out the shape file header (i.e., all lines starting with #).
pbox_bw	Defines the excitation band from the position of cursors in the graphics window and reports them to the user. It also sets r1 to excitation bandwidth and r2 to offset. This macro is used mainly in Pbox menus and macros.
pbox_bws	Defines the excitation band from the position of cursors in the graphics window and reports them to the user. It also sets r1 to excitation bandwidth and r2 to offset. Note, the left cursor should be placed on the left side of the excitation band and the right cursor on resonance of the solvent signal. This macro is mainly used in Pbox menus and macros.
pbox_rst	Resets r1=0, r2=0, r3=0, r4=0, n2='n', n3='', and adds some standard comment lines to the Pbox.inp file. This macro is used in menus and other Pbox macros.
pbox_files	This macro is used only in conjunction with Pbox file menus.

Pbox UNIX Commands

The Pbox program is always executed when a shaped pulse is created. Any of the Pbox parameters can be used as an argument followed by the parameter value. The arguments

and shortcuts listed in Table 20 are available. Note that the output filename is optional and is always the first argument.

Table 20. Pbox Commands and Parameters (continued)

Command	*Parameter*	*Action*
`Pbox*`	`-b time`	Activate Bloch simulator, `opt=a` (add), `s` (subtract), or time in sec.
	`-c`	Calibrate only, do not create a shape file.
	`-f file`	Set name of the output file.
	`-h wave`	Print wave file header.
	`-i wave`	Print wave file parameters.
	`-l ref_pw90`	Length (in µs) of reference pw90 pulse.
	`-o`	List options.
	`-p ref_pwr`	Reference power level (dB).
	`-r file`	Reshape Pbox pulse.
	`-s stepsize`	Define the length (in µs) of a single step in the waveform.
	`-t wave`	Print shape title from wave file.
	`-u userdir`	Set user home directory.
	`-w wavestr`	Set wave definition string.
	`-v`	Run in verbose mode. Also print Pbox version.
	`-x`	Prints all Pbox parameters.
	`-value`	Sets reps to value.
`Pxsim`		Used in Pbox menus and macros for simulation of excitation profiles of shaped pulses.
`Pxfid`		Used by `dshape` and `dshapei` to format shape file into a FID-format text file.
`Pxspy`		Converts alien shapes (.RF, .DEC and .GRD) into Pbox compatible file format. Essentially converts a time-domain shape file into (frequency-domain) Fourier coefficients, which can be used to create a wave file in the `wavelib` directory.

Examples:

```
Pbox -i eburp2
Pbox newshape -wc 'eburp1 450 -1280.0' -1
Pbox sel.RF -w 'eburp1 420 -800' 'eburp1 420 1200'
Pbox -w 'eburp1 200 -1200' -attn e -pl 45 54.2 -b
Pbox tst.RF -w 'esnob 20p 170p' -sfrq 150.02 -refofs 55p -refpwr 45  \
-ref_pw90 54.2
```

12.2 Pulse Shape Analysis

The `pulsetool` program is designed to display and examine shaped rf pulses. The standard pulse template file format is the same as for shaped pulses in `/vnmr/shapelib`. Data points are listed as `phase amplitude time-count`, where `phase` is in degrees, `amplitude` is a value between 0 and 1023, and `time-count` is an integer

which describes the relative time duration of the step. The program is started by entering the command `pulsetool` in a UNIX window. Table 21 summarizes the command and parameters associated with pulse shape analysis.

Table 21. Pulse Shape Analysis Commands and Parameters

Command	
`pulsetool <-shape filepath>`	RF pulse shape analysis (UNIX)
Parameters	
`phi`	Amount of rotation about the Z-axis
`theta`	Declination relative to XY-plane

The `amplitude` and `phase` are displayed in the small windows at the top of the display, along with the effective frequency of the pulse, the quadrature components of the pulse, and its Fourier transform. You can select the contents of any of the smaller windows for display in the large graphics window in the center of the screen.

Between the small windows at the top of the display and the large central, graphics window is the control panel, home to a number of buttons that perform various operations or activate the routines described below.

Below the main graphics window is a panel that contains miscellaneous information about the current pulse and display status. The directory file name, pulse name, vertical scale, and vertical reference fields display current information that can be altered by the user.

The Steps, Fourier size, Power factor, and Integral fields are advisory only, and may not be entered or changed by the user. Power factor is calculated when a pulse is loaded, and is the mean square amplitude of the pulse. A square pulse has a power factor of 1. The integral of the pulse is an attempt to calculate the tip-angle per unit time and B_1 field strength. This number is strictly valid only for pulses that are modulated in amplitude only and can be used to determine the B_1 field required to obtain, for example, a 90° tip with a sinc pulse. To do this, divide the desired tip angle (in revolutions) by the product of the integral value and the pulse length (in milliseconds). The result is the required B_1 field strength, in kHz.

The directory system may be viewed, and pulse files selected for loading through the use of the Files button.

Simulation of the actual response to a pulse, based on Bloch equation calculations, is available by selecting the Simulation button.

A number of standard pulses can be created, with attributes tailored through the Create utility. The data currently displayed in the main graphics window can be saved with the Save button.

Directory and File Operations

In selecting files, both the working directory and the pulse template file name can be specified by direct entry into the Directory and Pulse name fields found in the panel at the bottom of the display (use the Delete key to erase characters, if necessary, and type in the desired name, followed by Return to indicate completion). When the Pulse name field is selected, pressing Return causes the named file to be loaded and displayed.

Alternatively, the Files button causes a popup window to be displayed, listing the contents of the current directory. A trailing slash "/" following a member of the list indicates a subdirectory, and an asterisk "*" an executable file. The Load, Chdir, and Edit buttons operate on an item selected from this listing with the left mouse button:

The Load button causes the selected file to be read, and displayed in the graphics windows. If the file does not correspond to the proper format for pulse template files, an error message is displayed. Comment lines beginning with the pound character "#" are ignored. Descriptive information about the pulse is displayed in the bottom panel—the name of the file, the number of steps in the pulse, the Fourier size required to do the FFT of the pulse, and a "power factor" calculated for the pulse. The power factor is based on the mean square amplitude of the pulse.

The Chdir button changes to and then lists the selected directory.

The Parent button changes to and then lists the parent of the current directory.

The Save button located in the main control panel can be used to save data currently displayed in the main graphics window to a file. When this button is selected, a second button labeled Done appears, along with a type-in field that holds the name of the file that will be created. First, enter an appropriate name, then select the Save button once again to write the file. Once you have entered the Save mode, you can repeat this as many times as you like— display a different attribute in the main window, enter a new file name, and select Save. To exit from this mode, select Done.

The Print button located in the main control panel can create a file that can be used to print the main graphics window on a PostScript printer. When this button is selected, a second button labeled Done appears, along with a type-in field that holds the name of the file that will be created. Selecting the Print button with a appropriate file name in the type-in field writes the file. The file can subsequently be sent to a PostScript printer with the UNIX `lp` command.

Attribute Selection

The six small graphics windows at the top of the tool initially display the different attributes of the current pulse:

- Amplitude
- Phase
- Effective off-resonance frequency
- Real and imaginary quadrature components
- Fourier transform

Any of these six windows can be displayed in the large graphics window by clicking in the appropriate small window with either the left or middle mouse buttons:

- The left mouse button causes the large window to be cleared before drawing and sets the clear mode to on.
- The middle mouse button turns off the clear mode and displays the selected attribute, overlaying any current display in the large graphics window.

Repeated selection of the small Fourier transform window will result in the large window cycling through the magnitude of the Fourier transform, the real component, and the imaginary component.

Scale and Reference

The vertical scale can be adjusted either by clicking the middle button inside the boundary of the large graphics window or by manually entering a value in the Vertical scale field of the bottom panel, ending by pressing Return. Using the middle mouse button causes the scale to be adjusted interactively so that the active curve passes through the mouse arrow.

Note that no rescaling occurs if the y-value specified with the middle button does not have the same sign as the actual attribute value at that point on the x-axis. A negative value can, however, be entered as a vertical scale if so desired.

The vertical reference controls the vertical position of the active curve on the large graphics window, representing the offset from zero measured in y-axis units. A positive value moves the curve up, and a negative value moves it down. Like the vertical scale, the vertical reference can be adjusted in one of two ways—a value may be entered manually into the Vertical reference field in the bottom panel, or the middle mouse button can be used interactively anywhere in the large graphics window, while simultaneously holding down the Control key. In the second case, the vertical reference is set so that the curve passes through the mouse arrow.

The vertical scale and reference are reset whenever an attribute is selected from any of the small graphics windows. If things get out of hand, use this by reselecting the current small window with the left mouse button.

Cursors

Interactive left, right, and horizontal cursors are available, and display a readout of position at the bottom of the large window when active. The left cursor is activated by clicking the left mouse button inside the large window. When the left cursor is present, the right cursor can be activated by clicking the right mouse button anywhere to the right of the left cursor. At this point, the right mouse button controls the position of the right cursor independently, while the left mouse button moves both cursors in tandem.

When both cursors are active, the control panel button normally marked Full will read Expand, and can be used to display an expanded view of the region selected between the two cursors. (Note that the clear mode will always be set to on after an Expand or Full operation.) The left and right cursors are turned off by clicking the appropriate mouse button in the large window while simultaneously pressing the Control key.

The horizontal cursor is activated with the Thresh button located in the control panel. When this cursor is active, it is controlled interactively with the middle mouse button. The interactive scale and reference functions normally controlled with the middle mouse are not available when the horizontal cursor is present. Select the Scale button in the control panel to turn off the horizontal cursor and reactivate the scale and reference functions (vertical scale and reference can be adjusted even with the horizontal cursor active by direct entry in the appropriate fields in the bottom panel).

Simulation Overview

The simulation routine simulates the effects of an rf pulse by use of the classical model of nuclear spin evolution described by the Bloch equations. T_1 and T_2 relaxation effects are ignored, in which case the evolution of a magnetization vector in the presence of an applied rf magnetic field can be evaluated by multiplication with a 3 by 3 rotation matrix. The simulation consists of repeated multiplication by such a matrix, whose elements are determined at each step by the values of amplitude and phase found in the pulse template file, and by user input values of initial magnetization, B_1 field strength, pulse length, and resonance offset. The simulation is performed over one of three possible independent variables— resonance offset, B_1 field strength, or time, and is determined by the Sweep cycle in the small button panel.

Simulation Parameters

Select the Simulation button in the control panel to activate the Bloch Simulation subwindow. This window consists of a panel containing all required parameters (the pulse length is taken from the value in the bottom panel of the main window) and a small button panel at the bottom of the window. To change the value of any parameter, select it with the left mouse button, then delete the appropriate characters and enter the desired value from the keyboard. Parameters are updated each time the Go button is selected or when the Steps button is selected with Index equal to zero.

The first three parameters in the left hand column describe the starting values for the magnetization components Mx, My, and Mz, whose vector sum must be less than or equal to 1.

The next three fields change to reflect the state of the Sweep cycle, which can be toggled between B_1, Freq, and Time. When Freq is selected, the first of these fields will read B_{1max}, the value of B_1 at the maximum pulse amplitude. The second and third fields determine the lower and upper off-resonance frequency boundaries. When the Sweep cycle is set to B_1, these three fields are reversed so that the first determines a constant off-resonance frequency and the remaining two determine the lower and upper boundaries of the maximum B_1 amplitude. Selecting Time will yield a display of the magnetization as a function of progression through the pulse, at the frequency and B1 field strength specified by the parameter values displayed. In this last case, the number of steps in the simulation is taken from the number of points in the pulse template and may not be altered externally. To get finer resolution in the simulation, use a pulse template with a greater number of steps.

The Initialize cycle determines if the magnetization is reinitialized to the values of Mx, My, and Mz, or if the simulation uses the values at each point that were the result of the previous simulation. In this way, the effect of a series of pulses can be evaluated by loading the pulse and performing the simulation with Initialize set to Yes, then loading the next pulse, setting Initialize to No, and selecting Go. Any number of pulses can be concatenated in this fashion. This feature works only for Frequency and B_1 sweep, but not Time.

When Time sweep is selected, the results can also be displayed in the form of a projected three-dimensional coordinate system, showing the path of the magnetization over the course of the pulse. This display is obtained by selecting the 3D button after first selecting the Go button. When the 3D display is active, the left mouse button controls the viewing angle from within the canvas region delineated by the blue corner markers. This viewing angle is described by the two parameters `phi` (the amount of rotation about the Z-axis) and `theta` (the declination relative to the XY-plane). A “family” of trajectories can be displayed by first selecting any of the small canvases with the middle mouse button, then selecting the 3D button. Changing either the B_1 field strength or the resonance offset followed by the Go button will result in display of the result without clearing the display. To reactivate the automatic clearing feature, select any of the small canvases with the left mouse button. To see the 3D display drawn in real-time, enter a nonzero integer value in the Time field. The appropriate value depends on the number of steps in the pulse and the type of computer you have. Try a value like 100 for a SPARCstation.

The last parameter in this column determines the number of points at which the simulation will be performed along the y-axis. A larger number will give more detail in the result, but will require proportionally more time to complete.

The Index parameter is a counter that updates the status of the simulation, and cannot be set externally. The value displayed is the number of steps in the pulse template that have been completed.

The Step Inc. parameter is used by the Step button, described in the section, “Performing a Simulation,” to control the number of intermediate steps to be calculated.

Performing a Simulation

When you have adjusted the parameters to your liking, you will probably want to select the Go button. This does simulation calculations and then displays the results in the first five small graphics windows, replacing (but not destroying!) the pulse information that was displayed there. The Fourier transform information remains unaffected, so that comparisons can be made between this and the exact simulation results.

All of the display functions described elsewhere are active as well, with the simulation data. Additionally, the original pulse data is still present in the background and can be swapped into view with the Display cycle found in the main control panel.

The Step button offers the ability to view the course of the magnetization at intermediate stages through the pulse. When this function is selected, the next Steps Inc. steps of the pulse are simulated, starting at the current value of Index. The intermediate result is then displayed in the normal fashion.

During a Go simulation, a small panel containing a Cancel button will pop into view. Use this to stop the simulation if necessary (there may be some delay between selecting the button and the end of the process; it won't do any good to click on Cancel more than once).

Creating a Pulse

The pulse creation routine currently offers the following pulse types:

Square	Hermite 90	Tan swept inversion
Sinc	Hermite 180	Sin/cos 90
Gaussian	Hyperbolic secant inversion	

A file containing the pulse template for any of these pulses can be created from scratch with this utility. Alternatively, pulses can be created for examination only, using the display capabilities of `pulsetool`. Each pulse is generated with user-definable parameters appropriate for the pulse in question.

When the Create button is selected, a menu of pulse types appears. Hold the right mouse button down on the Create button, select one of the pulses in the resulting menu, and release the mouse button. If you decide you don't like any of the possibilities, move the mouse arrow out of the menu area and release the button. When a pulse type is selected, a small window appears with a brief description of the characteristics of the pulse and a set of changeable attributes whose values you may alter if so desired. The number of steps in the pulse is limited to powers of 2 and can be set by clicking the left mouse button, or by holding the right mouse button down and selecting the desired value from the resulting menu. All other attributes, which vary depending on the pulse type, can be altered from their default values by first selecting the appropriate field with the left mouse button, deleting with the Delete key, and typing in the desired value (pressing Return is not required).

At this point, you may select one of the three buttons at the bottom of the window: Preview, Execute, or Done:

- Preview uses the attributes as they appear on the screen to create a pulse that is loaded internally into `pulsetool`. All `pulsetool` features can then be used to examine and evaluate the new pulse. Any previous pulse information is deleted.

- Execute uses the attributes as they appear on the screen to create a pulse, which is written to a standard UNIX file. The name of the file is taken from the file name field in the Create window and written into the current directory, listed in the Directory field in the bottom panel. If a file of the same name already exists, you are asked to confirm your request. If, for any reason, the program is unable to write to the named file, an error message appears. This is generally symptomatic of not having write permission in the current directory.

Currently, there is no convenient way for a user to add new pulse types to those listed above. Suggestions for those pulse types that should be included in the future are welcomed. However, any user-created shaped pulse may be examined using the Files button.

Appendix A. VnmrJ Liquids NMR Interface

The VnmrJ liquids NMR interface is shown in Figure 49. This chapter describes major areas of the interface.

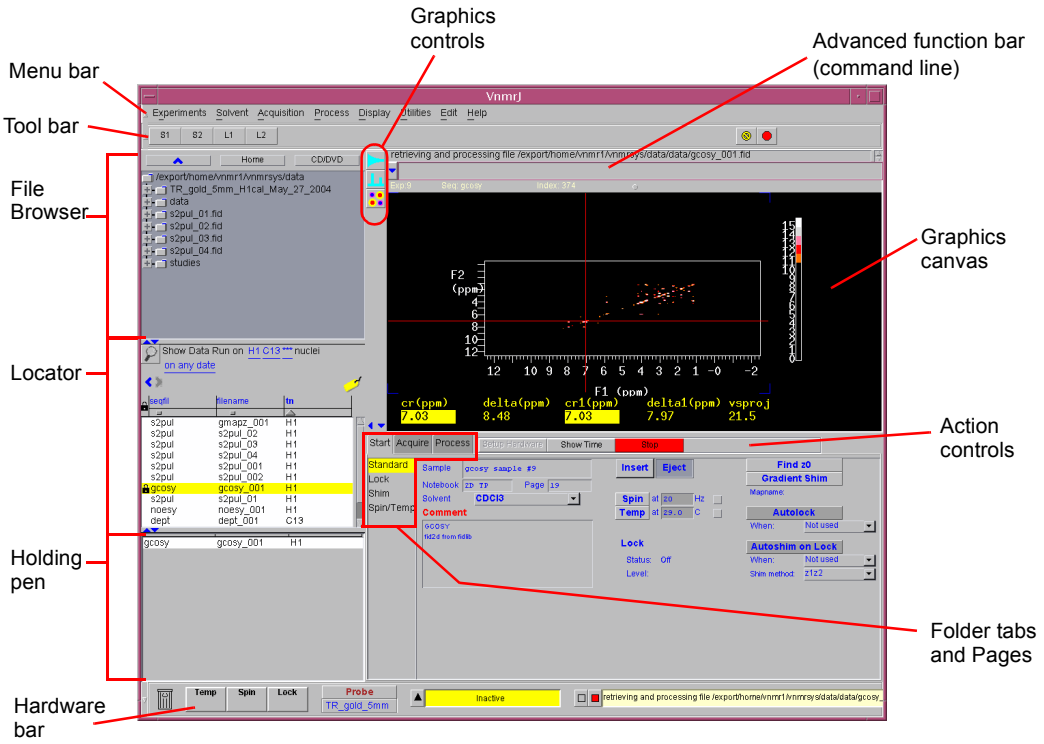


Figure 49. VnmrJ Interface

A.1 Menu Bar

The menu bar sits along the top of the interface. Its function is twofold: First, the menu bar provides access to operations needed to acquire, process, display, and plot a spectrum. Second, the menu bar provides access to little-used features, settings, and preferences.

The menu bar contains the following menus:

- "Experiments Menu," page 238
- "Solvent Menu," page 238
- "Acquisition Menu," page 238
- "Process Menu," page 239
- "Display Menu," page 240
- "Utilities Menu," page 241
- "Edit Menu," page 243
- "Help Menu," page 244

Hiding and Showing the Menu Bar

To hide the menu bar, click the left mouse button on the arrowhead in the menu bar. The menu bar closes and the arrowhead changes to a rectangular box. To show the menu bar, click the left mouse button on the box.

Experiments Menu

The Experiments menu enables you to select Varian-provided experiments.

Solvent Menu

The Solvent menu provides a list of commonly used solvents. A solvent can also be selected from the Setup panel.

Acquisition Menu

This menu enables you to perform acquisition-related operations. These operations are generally for acquiring data while you are not running a study, unless otherwise noted.

Acquisition Menu Items	*Descriptions*
Parameter arrays...	Use this function to open the Array Parameter window set arrayed parameters. Refer to 5.4 "Parameter Arrays," page 75.
Show Pulse Sequence	Use this function to show a pulse sequence. This function is similar to `dps`. When the sequence is displayed, click on the mouse icon on the left side of the Advanced function bar to get more detailed information about a sequence element.
Experiment Time	Use this function to show the total acquisition time for a pulse sequence. This function is similar to `time`. The total acquisition time in minutes and seconds is displayed in the message window.

Acquisition Menu Items	*Descriptions*
Setup Hardware	Use this function to initialize the hardware with values present in the current parameter set. This function is similar to `su`.
Set Shims Into Hardware	Use this function to load the current shim values into the hardware. This function is similar to `load='y' su`.
Acquire Data	Use this function to acquire data without performing any data processing.
Acquire and WFT	Use this function to acquire data and process it with a 1D weighted Fourier transform.
Acquire and Process	Use this function to acquire data and perform user-defined processing.
Abort Acquisition	Use this function to abort acquisition.
Pause Acquisition	Use this function to pause an acquisition at the next block size.
Resume Acquisition	Use this function to resume a paused acquisition.
Do Gradient Shimming	Use this function to perform gradient shimming. Gradient shimming must be set up on your system before it can be performed. See Utilities -> Set Up Gradient Shimming.

Process Menu

This menu has the following choices to perform processing functions.

Process Menu Items	*Descriptions*
Process and Display 1D	performs a 1D Fourier transform and shows a spectrum.
Full Process	performs automatic processing for 1D and 2D data and displays the results.
1D FDM	If the FDM option is installed, this menu item opens the 1D FDM window.
Drift Correct Spectrum	performs a 1D drift correction on a spectrum.
Automatically Set Integrals	automatically defines the integral regions for a 1D spectrum.
Baseline Correct	performs a 1D baseline correction on a spectrum.
Set Spectral width between Cursors	moves the spectral window to the window defined by the cursors.
Set Transmitter at Cursor	moves the carrier frequency to the current cursor position.
Pbox	opens the Pbox tool shown below. Use Pbox to generate shape files for rf and gradients. For more information about Pbox, see 12.1 “Pandora’s Box,” page 217.
Full Process 2D	performs automatic processing of 2D data and displays the results. Refer to 10.5 “Processing Phase-Sensitive 2D and 3D Data,” page 179.
Process 2D (Individual Steps)	submenu that becomes available for 2D datasets, which contains the following: **Phase and Set Weighting F2** **Do first FT (t2 Domain)** **Adjust Weighting in F1 (must do first FT)** **Baseline Correct F2** **Full 2D-FT (t1, t2 domains)** **Baseline Correct F1**
Add and Subtract 1D Data	submenu that contains the following:

Process Menu Items	*Descriptions*
	Clear Buffer and Add Current Spectrum clears the buffer and adds the current spectrum. **Add Second Spectrum Into Buffer** adds a second spectrum into the buffer for adding and subtracting.

Display Menu

This menu enables you to control how data is displayed in the graphics canvas.

Display Menu Items	***Descriptions***
Display Multiple Spectra Horizontally	presents a horizontal view of multiple spectra.
Display Multiple Spectra w/ Labels	presents a view of multiple spectra with identifying LABELS.
Display Multiple Spectra Vertically	presents a vertical view of multiple spectra.
Increase Vertical Separation by 20%	increases the vertical separation between multiple spectra by 20%.
Decrease Vertical Separation by 20%	decreases the vertical separation between multiple spectra by 20%.
Plot result	automatically plots the spectrum.
Create a Plot Design	opens the Plot Designer tool shown below. Plot Designer enables you to see and design a plot before you print it. For more information about Plot Designer, see 8.2 “Plot Designer,” page 117.
Create an Inset of the Current Display	opens a menu for making an inset display, with the following submenus: **Save Current Display Parameters** saves the current display parameters. **Plot Current Display before Making Inset** plots the current display before making an inset. **Make Inset** creates an inset spectrum. **Plot Inset and return to Original Display** plots an inset spectrum and restores the original display.

Utilities Menu

This menu has the following choices to perform system utilities functions. They are useful for setting the system configuration and interface settings.

Utilities Menu Items	*Descriptions*
System Settings...	opens the System Settings window, which enables you to set system parameters. System tab -- provides system-level settings. Display/Plot tab -- provides display and plot settings. System config button -- opens the System Configuration window (config). **System tab**: Application mode -- Standard (liquids) or imaging. Gradient amplifier -- On/Off selection for each gradient axis that is installed. Type of digital signal processing -- None for no DSP, Inline for software DSP, or Real time for hardware DSP. Frequency-shifted quadrature detection -- check to enable. Hardware Z1 shimming -- None for no Z1 shimming, Delay for Z1 shimming enabled during delay time, or Presat for Z1 shimming enabled during delay time preceding presat. Probe protection -- check to enable. Solids VT System -- check to enable VT cutoff (0-50) -- specify VT cutoff temperature; 25^{o}C recommended. Process data after acquisition -- check to enable. Autosave data after acquisition -- check to enable. Set processing when loading experiment-- check to enable. Set plotting when loading experiment-- check to enable. **Display/Plot tab** Process data on drag-and drop from locator -- check to enable. Set display from plotter aspect ratio (wysisyg) -- check to enable. Spectrum updating during phasing (0-100) -- set the percentage of the display that is updated during interactive phasing. 100 is recommended. Max # of pens -- number of plotter pens to use. Show Tooltips -- check to enable. Display only matching items in locator -- check to enable. Trash study node preferences -- set whether custom or completed study nodes are deleted, skipped, or not allowed.
Make a New Protocol...	opens a window, shown below, for saving the current parameter set as a new protocol.

Utilities Menu Items	*Descriptions*
Configure EXEC Parameters	opens the window shown below, which provides access to the parameters used to control the generic acquire, prep, process, plot, and prescan functions. The execpars concept provides setup, preparation, prescan, processing, and plotting customization based on the type of NMR data. This selection is based on the value of the `apptype` parameter. This panel allows you to view and/or to set the "exec" parameters. The check box is a convenient way to temporarily disable the parameters. The **Select local exec parameter** menu will read in current values for the selected local set. This menu is present only if local protocols exist. The **Select global exec parameters** will read in the current values for the selected global set. The parameters can be saved using the current pulse sequence. Alternatively, a name can be specified in the entry box next to the **Save apptype parameters**. The actual save process occurs when one of the "Locally" buttons is clicked. If you have permission, you can also save the parameters globally. In this case, buttons labeled "Globally" will appear next to the "Locally" buttons.
Create SpinCAD Seq Parameters	makes pulse sequence parameters. SpinCAD sequences have been provided for high-resolution liquids spectroscopy. ***Creating a Pulse Sequence Parameter in VnmrJ*** To create pulse sequence parameters, change the pulse sequence name in the current parameter set, then click on the **Utilities** menu. From **Utilities**, choose **Create SpinCAD Seq Parameters**, which also compiles the pulse sequence if necessary. ***The Parameter Creation Process*** After you begin the parameter creation process from either SpinCAD or VnmrJ, SpinCAD parses the pulse sequence and looks for new parameters that are not in the current parameter set. It forms a list of new parameters that do not exist and presents the list with other information in a pop-up panel in VnmrJ titled SpinCAD Create New Parameters. ***Correcting Parameters*** SpinCAD attempts to ascertain the type of the parameter from the usage but might not always be accurate. Various details about the new parameters are shown: name, parameter type (such as pulse, delay, string etc.), maximum limit, minimum limit, increment size, and initial value. If the type of the parameter is incorrect, you can easily correct it by using the **Type** pull-down menu. In such a case, you must also change the other field for limits and initial values. If you do not want to create a parameter, set **Type** to **IGNORE**. ***Constructing VnmrJ Parameter Panels*** Before you click **OK** to start the parameter creation process, set the panel template to be used as a starting point for making the Acquire panel for the new pulse sequence. VnmrJ parameter panels are constructed by adding a panel tab (SpinCAD PS) to the Acquire panel group. You can model the panels after any other template pulse sequence that has a layout directory.

Utilities Menu Items	*Descriptions*
	To construct a panel, enter the name of the starting template pulse sequence in the entry field for **Start Acquire Panel from template:** If you do not specify another template pulse sequence, SpinCAD uses the default template. If you want to cancel the entire process so that no parameters or panels are created, click **Cancel**. Otherwise, click **OK** at the bottom of the pop-up panel.
Create a Workspace	makes a new workspace of the next available workspace number.
Set Up Gradient Shimming	loads the pulse sequence and panels for making a shim map for gradient shimming.
Standard Calibration Experiments	a submenu containing the following **Calibrate Probe...** opens a window for running a series of experiments to calibrate the probe. **Start Autotest...** opens a window for starting the system autotest. **Autotest settings...** opens a window for showing or setting parameters used in system Autotest.
Save data	a submenu containing the following choices: **Save current fid** saves the current FID with a filename constructed from the **`sqdir`**, **`sqname`**, and `svfname` parameters if you are in a study. It uses the **`svfdir`** and **`svfname`** parameters if you are not in a study. **Save process as** saves reprocessing to given record.
Display options...	opens a window for setting symbolic colors and fonts in the interface.
Printers...	opens a window for setting printers and plotters from previously defined printers and plotters. For information about connecting printers, see the manual *Host Computer Setup for VnmrJ*.
Update locator	opens a submenu that provides choices for updating the different parts of the Locator.
Import files to locator	opens a window for importing files to the locator.
Save custom locator statement	opens a window for saving the current locator view.
Delete custom locator statement	opens a window for deleting custom locator statements.
Exit VnmrJ	exits VnmrJ.

Edit Menu

This menu enables you to edit objects.

Edit Menu Items	*Descriptions*
Tool Bar...	opens the Tool Editor, for editing the Tool Bar. To create a new tool bar button, drag the New Tool button onto the tool bar. To edit a button that already exists, click on the button in the tool bar

Edit Menu Items	*Descriptions*
	To select an icon for a button, click on an icon in the scrolling list. The **Command** entry contains the command or macro executed when the button is clicked. The **Set Command** entry contains the command or macro executed when the button is held down for 3 seconds. The **Tool Label** entry contains the text label for the button if an icon is not used. The **Tool Tip** entry contains the text for the tool tip. Click on the **Edit** button to perform the following actions: Return to initial state restores the initial state of the toolbar. Make a snapshot creates a snapshot of the current state of the toolbar. Return to snapshot restores the snapshot state of the toolbar. Return to system defaults restores the toolbar to the system defaults. To exit the Tool Editor and save your changes, click on the **Close** button. To exit the Tool Editor without saving your changes, click on **Abandon**.
Pulse Sequence...	opens the SpinCAD program, for editing SpinCAD pulse sequences. Click on **Sequence**, then **Exit, abandoning unsaved changes** to exit the program without saving a new pulse sequence. For more information about this program, see the *SpinCAD* manual.
Parameter Pages...	opens the Edit panel, for editing the VnmrJ parameter panels. In the window you can see the commands performed by the various widgets in the template, and the parameters they depend upon. Double-click on a widget to view its properties. Click on the **Close** button at the bottom of the window to exit the editor.

Help Menu

This menu contains a demonstration on running a study and the VnmrJ online help manuals.

Help Menu Item	*Description*
VnmrJ Help	opens the HTML help files in an Internet Browser.

The VnmrJ help uses a Web browser (Netscape or Mozilla) to display help files. Netscape 4.78 or newer, or Mozilla 1.2 or newer should be installed (Netscape can be downloaded from `www.sun.com/netscape`). Solaris 7, 8, and 9 shipped with Netscape 4.78. If a newer version of Netscape was installed, or if Mozilla was installed, be sure it was added to the `PATH` variable. If the current Netscape is not found in the `PATH` variable, the VnmrJ help files are displayed in Netscape 4.78.

A.2 Tool Bar

The tool bar is directly below the menu bar. These buttons provide quick access to common functions. The following tools are the default available in this tool bar:

- S1 S2 save the current locator data sort display. To save the display, click the button for three seconds. To return to the saved display, click again on the button.
- L1 L2 save the current screen layout (graphics, a parameter panel, locator sizes). To save the layout, click the button for three seconds. To return to the saved layout, click again on the button.
- cancels commands.
- stops acquisition.

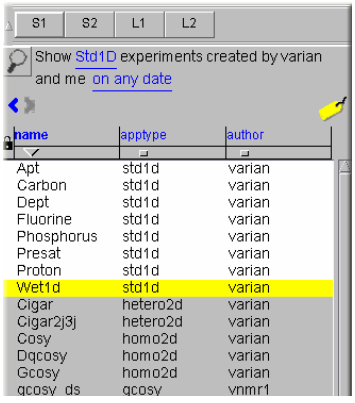
Hiding and Showing the Tool Bar

To hide the tool bar, click the left mouse button on the arrowhead in the menu bar. The tool bar closes and the arrowhead changes to a rectangular box. To show the tool bar, click the left mouse button on the box.

A.3 File Browser

The VnmrJ File Browser offers a convenient way to look for files and data and to limit the scope of what appears in the Locator. For more information about the File Browser, refer to B.4 “File Browser,” page 258.

A.4 Locator



The Locator provides access to data sets, experiments, shim sets, and commands. For more information about the Locator, see Appendix B, “Locator”.

Clicking the magnifying glass with the left mouse button opens a menu of searches. Selecting one changes the *search sentence* (next to the magnifying glass). The results of the search are displayed in the list. Those items in the white part of the list satisfy the search sentence. Those in the gray part do not. For each item that is found by the search, three attributes are displayed. These correspond to the three columns in the list. Clicking on the attribute name at the top of the list with the left mouse button opens a menu of attribute choices.

Clicking on an item in the Locator list selects that item. That item can then be dragged to the graphic area or the parameter panel area to cause the appropriate action. For example, dragging a data set to the graphic canvas retrieves that data set into the current workspace (experiment) and displays the spectrum. Dragging a workspace to the graphic canvas

causes that workspace (experiment) to be *joined* with the graphic area. Double-clicking on an item performs the same action as dragging the item to the graphics canvas.

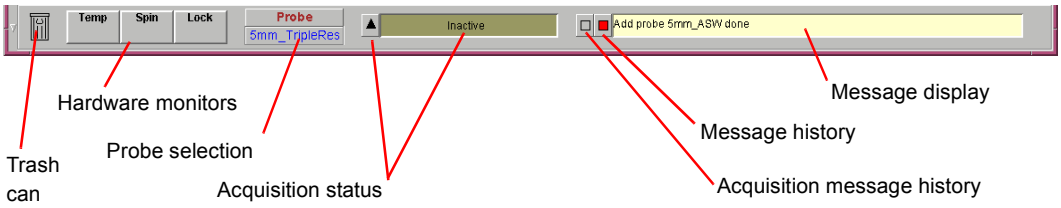
A.5 Holding Pen

Use the holding pen to store commonly accessed items from the Locator. To add an item from the Locator to the holding pen, select (click on) the item and drag it into the holding pen. The item remains in the holding pen even if the Locator view changes.

Selecting an item or dragging it from the holding pen performs the same actions as selecting an item or dragging it from the Locator.

You can remove an item from the holding pen by selecting it and dragging it to the trash can.

A.6 Hardware Bar

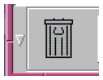


The hardware bar contains the following:

- “Trash Can,” page 247
- “Hardware Monitors,” page 247
- “Probe Selection,” page 247
- “Acquisition Status Details,” page 248
- “Acquisition Status Display,” page 248
- “History of Acquisition Messages,” page 248
- “History of All Messages,” page 248
- “Message Display,” page 248

The right portion displays the current state of the acquisition system and system messages:

Hiding and Showing the Hardware Bar

To hide or show the hardware bar, click on the arrow icon  to the left of the trash can with the left mouse button.

Trash Can

Dragging an item to the trash can from the Locator or other area generally removes the item and adds it to the trash can.

Double-clicking on the trash can enables you to view items in the trash can area. In this mode, you can restore objects from the trash can by selecting them and then clicking the **Restore items** button. In Figure 50, the trash can is empty. To exit this mode, double-click on the trash can.

CAUTION: **Emptying the trash can deletes data from the disk.**

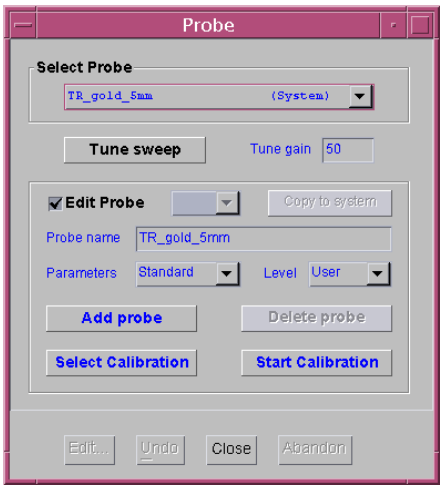
Figure 50. Trash Can Mode

Hardware Monitors

When you click Temp, Spin, or Lock, a window appears with a line graph showing the history of the relative hardware function. To close the window, click the icon again.

Button	*Description*
Temp	Shows a history of the sample temperature
Spin	Shows a history of the sample spin rate
Lock	Shows of a history of the sample lock level

Probe Selection

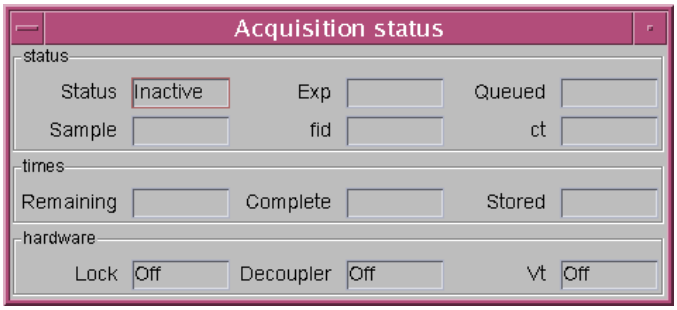


Click on this button to open the Probe window. Use it to perform the following actions:

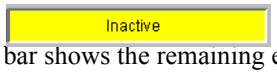
- To select a probe from the list of available probes, click on the **Select Probe** menu.
- To open the probe tuning (Q tune) window, click on **Tune sweep**. Do not use this for normal probe tuning.
- To edit a probe entry, click on **Edit Probe**.
- To open a window to edit probe attributes for a particular nucleus, click on the nucleus menu.
- To select initial parameter attributes for adding a probe, click the **Parameters** menu.
- To add a new probe, enter a new probe name and then click **Add Probe**.
- To remove a probe name from the list of available probes, enter the probe name and then click **Delete probe**.
- To open a window for running a series of probe calibration experiments, click **Edit Probe**, **Select Calibration**, and **Start Calibration**. Refer to the *System Administration* manual for detailed calibration procedures.

Acquisition Status Details

To open a window showing acquisition status details, click on the  icon. To close the window, click on the icon again.



Acquisition Status Display

The acquisition status bar 

is always visible in the hardware bar. During an acquisition, the bar shows the remaining experiment time as a thermometer display. Click the right mouse button inside the bar to change the displayed text.

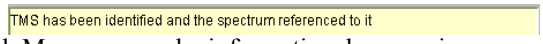
History of Acquisition Messages

To see a history of all acquisition messages, click on the  icon. To close the window, click on the icon again. Click the right mouse button within the scrolling message window to change the text view options.

History of All Messages

To see a history of all spectrometer messages, click on the  icon. To close the window, click on the icon again. Click the right mouse button within the scrolling message window to change the text view options.

Message Display

The message display 

shows the last message that occurred. Messages can be informational, a warning, or an error message.

A.7 Folder Tabs and Pages

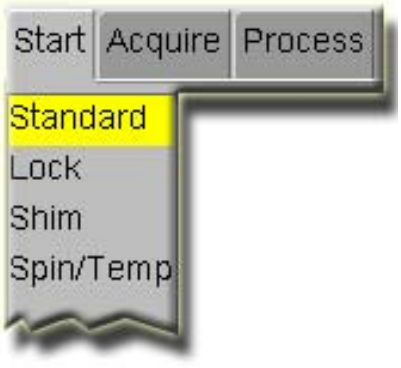
Select folders by clicking on the tabs on the Action Controls bar.

Use the **Start** panel to perform basic functions for setting up a new sample and preparing to run experiments.

Use the **Acquire** panel to set acquisition parameters.

Use the **Process** panel to adjust processing parameters and process data.

Each tab contains a list of relevant pages. Here is shown the Start tab with its typical pages.



Editing Parameter Pages

To edit a page, select the **Edit** menu, then select **Parameter Pages**. This template editor is also useful for viewing commands and parameters that are used in the panels.

A.8 Action Controls

To the right of the parameter panel selection tabs (Setup, Acquire, Process), shown below, are series of buttons that change, depending on the currently displayed panel. For example, the buttons shown below appear when you select the **Setup** panel.

A.9 Graphics Canvas

This portion of VnmrJ, shown in Figure 51, is used to display and interact with graphic and text information.

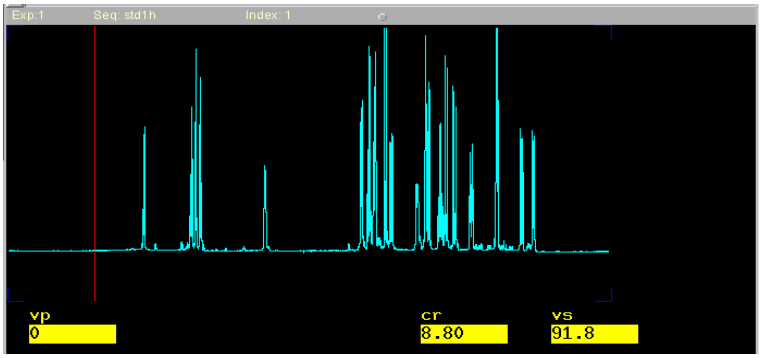
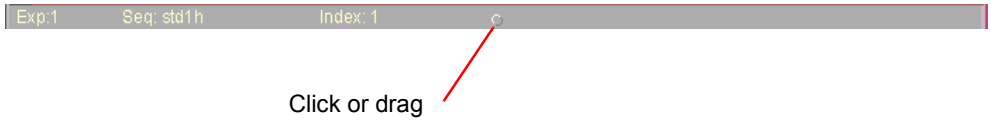


Figure 51. Graphics Canvas

- Resize the graphics canvas by clicking on the canvas boundary line with the left mouse (the cursor changes form) and dragging the line (e.g., between graphics and parameter templates or between graphics and Locator).
- Toggle the graphics canvas to show or hide the parameter templates area by clicking the small arrow between the graphics canvas and templates with the mouse button.
- Flip the parameter templates behind the graphics canvas without resizing the graphics canvas by clicking the small arrow with the middle mouse button.
- Toggle the parameter area to maximum height required to remove the parameter template scroll bar by clicking on the small arrow with the right mouse button.
- Toggle the graphics canvas to show or hide the Locator area by clicking the small arrow between the graphics canvas and Locator with the left mouse button.
- Flip the Locator area behind the graphics canvas without resizing the graphics canvas by clicking on the small arrow with the middle mouse button.
- Toggle the Locator area to maximum by clicking on the small arrow with the right mouse button.

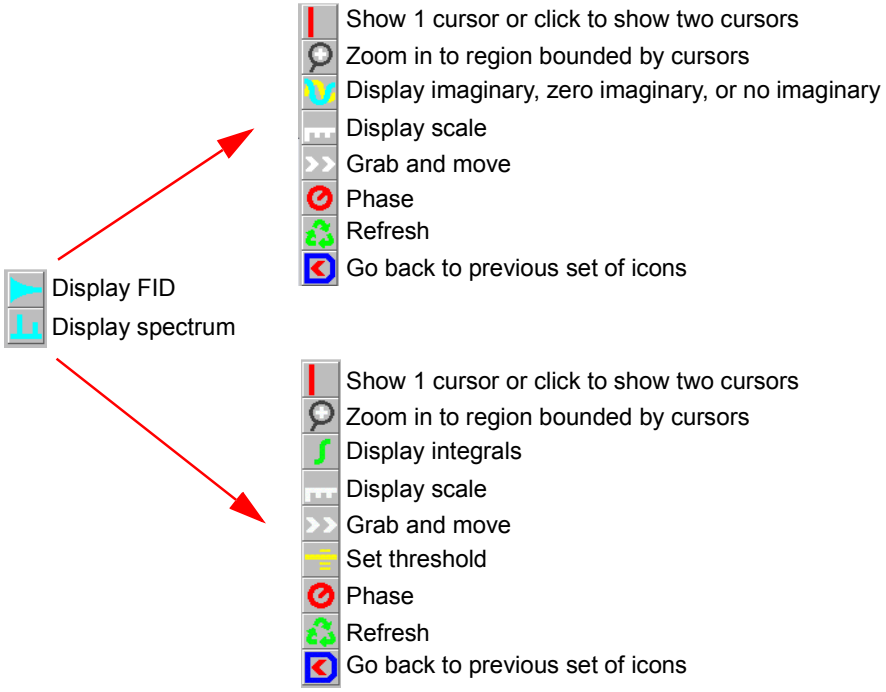
A.10 Advanced Function Bar



Dragging down the button in the advanced function bar opens a command entry field and a text output field. Open or close the field by clicking once on the button, which restores it to its most recent view. Any command or macro can be entered into this command window. Error and information messages are displayed in the scrolling text window above the command line in addition to the hardware bar. Command history can be shown if you click on the arrow with the left mouse button. Select a command from the command history by highlighting it and pressing **Return** to execute it.

A.11 Graphics Control Buttons

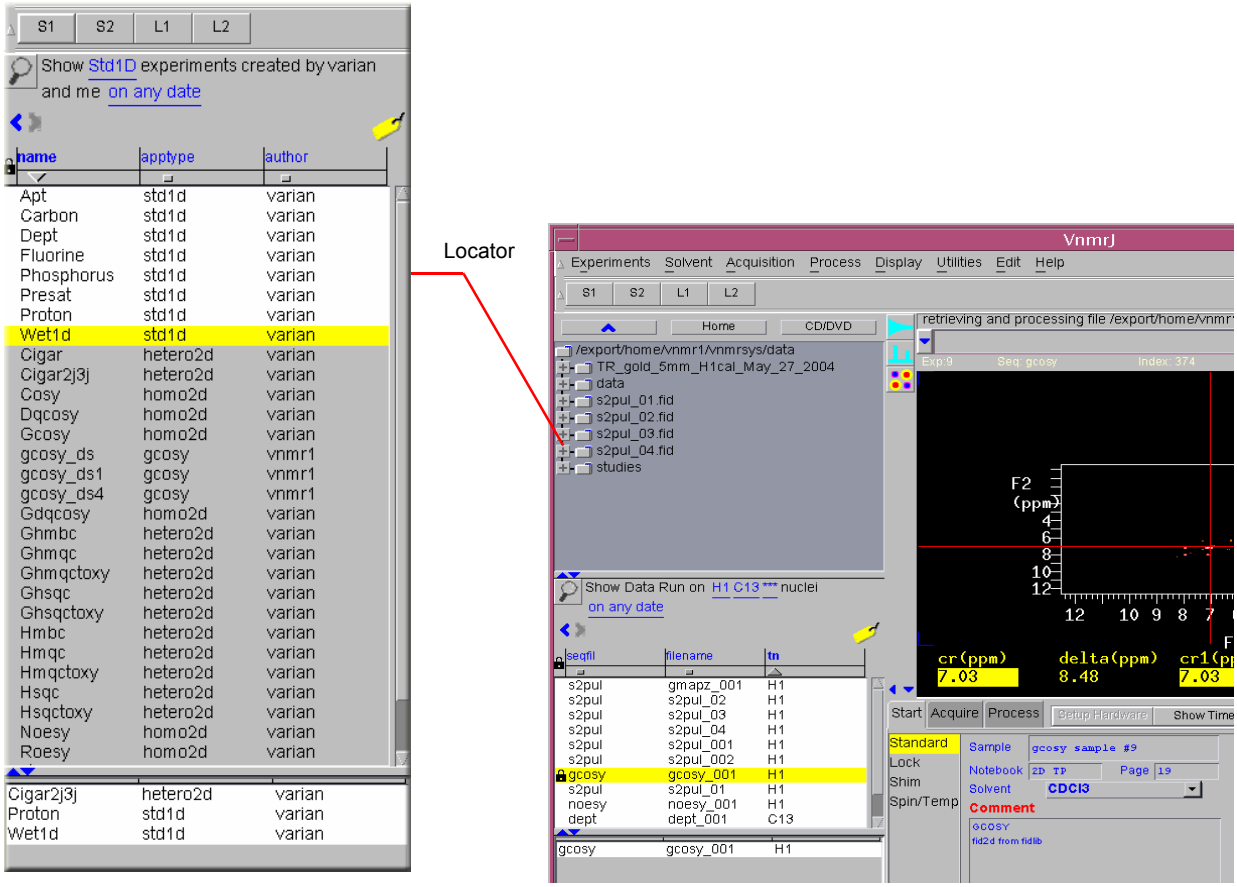
The graphics control bar is to the left of the graphics canvas. Use the buttons in the bar to control the interactive display in the graphics canvas. When a spectrum or FID is displayed, these buttons enable you to perform interactive actions on the spectrum or FID. These actions include integral displays, phasing, threshold adjustments, and other actions. These buttons change depending on the type of data that is displayed in the graphics canvas.



Appendix B. Locator

Sections in this chapter:

The Locator, shown below, is a database browser that provides access to data sets, experiments, shim sets, commands, and other things.



The Locator is designed to enable fast access to information on all or part of your disk environment. The scope of the Locator’s actions is determined by your administrator who will globally and locally set the scope of the Locator.

The Locator works similar to a directory or file manager, but uses minimal filtering of the information. Rather than only showing the files that satisfy the requirements, it shows two

or three lists of information. With two lists, one list shows the items (or objects) that satisfy all terms of the search while the other list shows the objects that do not. Where some terms have a boolean relationship, the Locator shows three lists:

- Objects that meet all criteria
- Some of the boolean terms met
- Remaining objects

The determination of which of these lists it are shown is determined by the construction of the underlying Locator statement.

Within each list, the Locator displays three attributes for each object. The displayed attributes need not necessarily be those in the Locator statement. Any one of the attributes can be designated as the sort attribute, in which case the objects in each list are sorted by the value each has for this attribute.

B.1 Locator Interface Elements

The Locator interface elements are described in the following sections:

Locator Statements and Menu

At the top of the Locator is a magnifying glass and the current Locator statement, as shown in Figure 52.

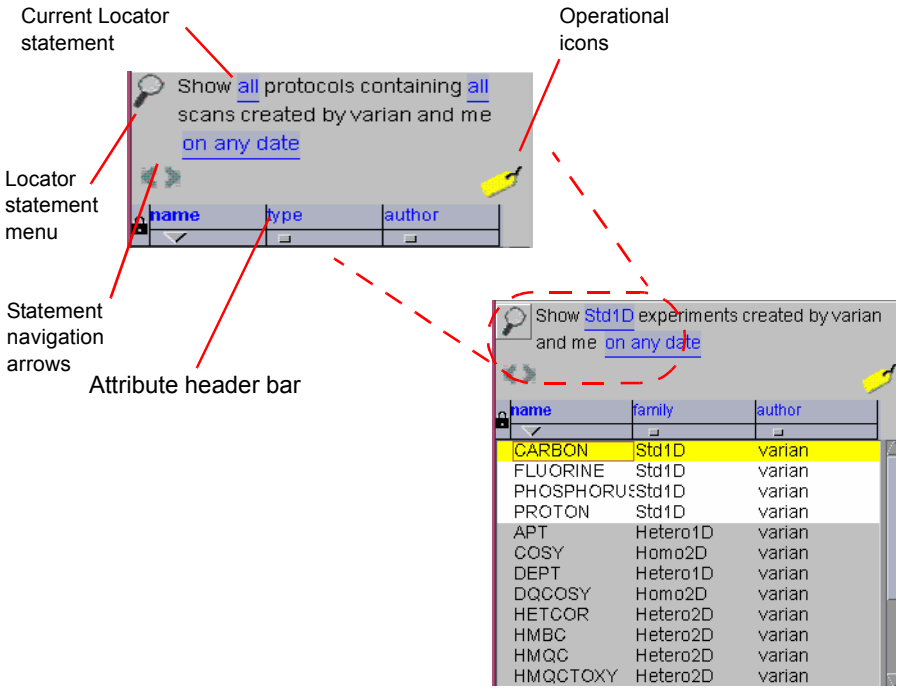


Figure 52. Locator Statement

Clicking on the magnifying glass opens a menu of currently available Locator statements. This menu includes both statements provided by Varian, Inc. and those customized and saved by the user.

Statements are Locator sentences in which a number of words or phrases are colored and underlined in a manner reminiscent of links in a web page. Each link hides a menu of choices, of which the currently displayed phrase is one. The choices available vary with the types of data currently known to the Locator.

Navigation in the Locator

Below the Locator statement is a pair of arrows (statement navigation arrows), which enable you to move forward and back through past Locator operations, applying each to the current Locator environment. Thus a set of Locator statements can be rapidly applied in a changing environment.

Locator Groups

To the right of the arrows is the tag icon for adding and removing Locator groups. Use this tool to define (tag) new user groups and apply those to existing

or new objects, for example, to tag FIDs with a given project name. Clicking on this icon opens a pop-up menu, shown in Figure 53.

Figure 53. Locator Groups Menu

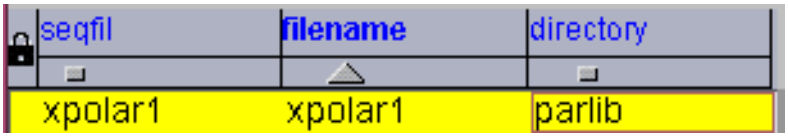
Using Locator Groups

To use Locator groups, do the following steps:

1. Select a statement from the Locator statement list (described in section B.3 "Locator Statements," page 256) that sorts on groups, such as **Sort NMR Data by group**.
2. Select the group that you want to see from the pull-down menu in the statement.

Attributes

Below the icons is the **attribute header bar**. This bar enables you to select the attributes displayed and to arrange the objects in each list in a number of ways.



The **padlock** enables you to lock an object against archival (this feature is not yet implemented).

The next three fields are the currently displayed attributes. You can change any column simply by clicking on the attribute label and selecting a different attribute from the drop-down list. Below each label is its "Hot zone". The hot zone can be either a **block** or a **triangle**. Only one attribute has a triangle and it is the attribute that serves as the sort term for each list.

- Click on a **block** to change the sort attributes.
- Click on the **triangle** to reverse the sort order.

The boundaries between the attribute labels are adjustable. Simply place the mouse cursor on the boundary you wish to adjust. When you get the adjust cursor, click and drag the boundary to its new position, and release.

Objects in the Locator are available for a number of actions. Currently, a single click selects an object. The selected object can then be dragged to another part of VnmrJ in which case the action taken will depend on the type of object and where the object is dropped. Alternatively a double click on an object will cause the most likely action to occur. These actions will be discussed shortly.

The value of an attribute might be longer than the width of the column in the Locator. When the mouse cursor rests on an attribute value, a tool tip appears for a period of time. The tool tip contains the full value of the attribute.

B.2 Using the Locator

Use the mouse to select or drag-and-drop items in the Locator interface.

- "Searches," page 255
- "Dragging and Dropping Items from the Locator," page 255
- "Editing File Names from the Locator," page 256
- "Configuration Files," page 256

Searches

Clicking the magnifying glass with the left mouse button brings up a menu of searches. Selecting one changes the *search sentence* displayed at the top of the Locator. The results of the search is displayed in the list. Those items in the white part of the list satisfy the search sentence. Those in the gray part do not. For each item that is found by the search, three attributes are displayed. These correspond to the three columns in the list. Clicking on the attribute name at the top of the list with the left mouse button brings up a menu of attribute choices.

Dragging and Dropping Items from the Locator

Clicking on an item in the Locator list selects that item. That item can then be dragged to the graphic area or the parameter panel area to cause the appropriate action. For example, dragging a data set to the graphic area retrieves that data set into the current workspace (experiment) and (optionally) displays the spectrum. Dragging a workspace to the graphic area selects that workspace (experiment). Dragging on an object causes the most likely action to occur.

An item can be dragged from the Locator and dropped into the holding pen. The item is then available for further selection no matter what Locator statements are active. One such example might be when you use the Locator to inspect the available shim sets. You can then select the current best set and put this into the holding pen. This set of shims is then immediately available.

Dragging and dropping an item has an action appropriate to the context. In many cases the same effect can be obtained by double-clicking on an object. Some examples are:

- Dragging a protocol experiment into the graphics canvas loads the experiment.
- Dragging a FID from NMR data retrieves the FID. The process macro can also be invoked so that the FID is transformed.
- Double-clicking a workspace joins that workspace. Dragging and dropping a workspace into the graphics area also joins the workspace (`jexp`).
- Double-clicking a parameter set loads that set in the current workspace, as will a dragging and dropping a parameter set.
- Double-clicking a shim set loads the shims. Dragging and dropping a shim set to the current shim buttons also loads the shims into acquisition.

- Dragging either data or shims and dropping them in the trash can (in the lower left portion of the hardware bar) moves the item to the trash can. You can retrieve an object from the trash can by double-clicking on the trash can, selecting it, and then clicking the **Restore items** button.

Editing File Names from the Locator

When a new file is added to the locator from within VnmrJ, the new item appears in its appropriate spot in the Locator, and it appears in green at the top of the locator window. If one of the columns in the Locator is "filename," you click on the green file name and changed it.

After changing the file name, press Return or click on another line to remove the old name from the Locator and add the new one. The Locator redisplays to show the new name.

Configuration Files

Configuration files for the locator are contained in the following directories for the different appmode types:

Interface	*Directory*
Standard (experimental)	`/vnmr/shuffler`
Imaging	`/vnmr/imaging/shuffler`
Walkup	`/vnmr/walkup/shuffler`
Individual users	`$vnmruser/shuffler`

B.3 Locator Statements

We supply a number of Locator statements with VnmrJ. You can add to or edit these statements in the following ways:

- Save the current Locator statement by clicking on **Utilities** in the main menu, then **Save Custom Locator Statement**. In the Custom Locator Statement pop-up window, enter a name for the statement.
- To delete a Locator statement, click on **Utilities**, then **Delete Custom Locator Statement**. A Custom Locator Statement Removal window appears. Select the statement from the list in the window, then click on **Delete** to remove it or **Cancel** to exit the window without removing the statement.
- Sort **Protocols Entries** show the know protocol experiments. Double click on the protocol to execute the associated macro.

Locator statements are defined in a file named:
`locator_statements_default.xml`.
This file can reside in the system `appmode` directories (see "Configuration Files," page 256), but not in users individual directories.

You can sort Locator items as follows:

- "Sort Workspaces," page 257
- "Sort NMR Data," page 257
- "Sort NMR Parameter Files," page 257

Sort Workspaces

You can sort all workspaces in numeric order or selectively sort workspaces by groups. Click on the tag icon  to create locator groups. Double-click on a workspace to join the workspace.

Sort NMR Data

Entries show the known NMR data sets, but differ in the actual format of the statement as well as the initial set of attributes shown. The most comprehensive statement is the last one, **by user defined attributes and date** (this is also the one that is least likely to be used, but it is discussed here to explore the scope of the data statements).

The generic statement is shown in Figure 54.

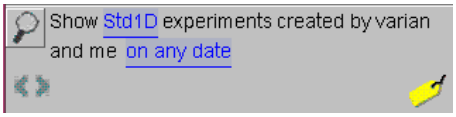


Figure 54. Generic Locator Statement

There are two separate underlined choices in this statement: `Std1D` and `on any date`.

Clicking on either of the underlined phrases produces a drop-down menu of the choices in this position. The menus are environment sensitive so they will not display choices that do not exist.

The logic of this statement is of the form:

Show attributes A and B of type C with additional limitations.

First, the additional limitations phrases enable you to select the owner of the data. Currently this selection is determined by the administrator at the time a directory is made available to the Locator. In in the future, it will become a more rich criterion.

Second, the additional limitations enable you to reorder by date. There are various dates associated with data, for example, the time started or the time saved. You can specify these date fields several ways, for example, since a certain date, by changing **on any date** to **since**.

You can alter the date by using the left or right arrows to decrease or increase the date by one day each click.

All other statements supplied by us are simpler than the generic one. If there are certain statements that you use frequently, these can be promoted to the top of the menu simply by saving them again as your local variants.

Sort NMR Parameter Files

The statements in this category show the list of NMR parameter sets. One major category of parameter set is **My Param Files**. You can also select the statements **Test Files** and **by user defined attributes** to do other selective searches. After you select a category, the Locator statement changes, e.g., **Test Files**.

Sort Shimsets

The statements in this category enable you to access the shim sets that you have saved. Note that the shim sets can be saved with a descriptive shim name provided by you using the Save Shims button in the Shim panel.

Sort Command Macros

The generic statement in this category enables you to find a VnmrJ command or macro based on its attributes. The Locator enables you to reorder commands and macros by a number of attributes. Once you find the command that you wish to use, a double click will execute it.

Attribute Lists

The list of attributes in the drop-down lists are controlled by configuration files. There are three file names, for three different types of items in the locator. These are:

- shuffler_param_list for 'vnmr data' and 'vnmr parameter' files
- study_param_list for 'study' items
- data protocol_param_list for 'protocol' items

Each of these can exist for each of the appmode types and for individual users. That is, appmode types of 'imaging', 'standard' (experimental liquids & solids) and 'walkup'. The attributes visible in the drop down menu for each appmode type, will be controlled by files in the appropriate directories. If a user does not have an individual file, the file in the appropriate appmode directories will be used. If there is no file in the imaging or walkup directories listed above, then the file in `/vnmr/shuffler` will be used. If users have their own individual files, the attributes listed in it must also be in the appmode directory file. That is, a user's files can limit attributes shown, but cannot add to the list of attributes shown beyond the attributes in the system files.

B.4 File Browser

The File Browser operates in conjunction with the Locator.

- "Activating the File Browser," page 258
- "Using the File Browser," page 259
- "Working with the Review Queue," page 259
- "Entering a Directory Path Directly," page 259

Activating the File Browser

If the File Browser does not appear above the locator, you can activate it by adding a line to the file `ToolPanel.xml`. Each VnmrJ interface contains its own instance of this file.

- Experimental interface:
  `/vnmr/templates/vnmrj/interface/ToolPanel.xml`
- Walkup interface:
  `/vnmr/walkup/templates/vnmrj/interface/ToolPanel.xml`

Add the following line:

<tool name="Browser"/>

between <toolDoc> and </toolDoc>, just above the line
<tool name="Locator"/>

Example for Experimental interface

```
<toolDoc>
    <tool name="Browser"/>
    <tool name="Locator"/>
    <tool name="Holding"/>
<toolDoc>
```

Example for Walkup interface

```
<toolDoc>
    <tool name="Browser"/>
    <tool name="ExperimentPanel"/>
    <tool name="Sq"/>
<toolDoc>
```

For the imaging interface, refer to the *VnmrJ Imaging* or the *VnmrJ Installation and Administration* manual.

Using the File Browser

When the file browser panel is open, the Locator limits its scope to only files and directories at the current file browser level or below. Closing the file browser by clicking the arrows or dragging the border, causes the Locator to go back to an unlimited display of items.

Drag & drop from Locator to the File Browser sets the File Browser to the director where the dragged item resides.

File browser buttons and controls:

^	click to go up one level
Home	click to go to the current users home data directory
CD/DVD	click to go to the currently open CD or DVD directory
+	(or double clicking a directory)— to open that directory and make it the top level currently in the file browser and also the Locator

Double clicking on a file or directory— operate on that file or directory as the Locator does. That is, double clicking on an NMR data file loads it just as double clicking on the same data file in the Locator does.

Drag a files and directories from file browser — operate on that file or directory as the Locator does when dragging to other windows including the graphics canvas and the study queue.

Working with the Review Queue

- ***.img** directories and ***.fdf** files — drag to the review viewport or the Review Queue panel for display.
- Double clicking on an ***.img** directory — browse the directory.

Entering a Directory Path Directly

The top level of the file browser is editable.

1. Click on the top item to select it.
2. Click again to get the editing cursor.
3. After editing, hit the **Return** key to display the new directory.

Appendix C. Variable Temperature System

Sections in this chapter:

This chapter describes startup and operation of the optional variable temperature (VT) unit. A VT unit is available for Varian, Inc. NMR spectrometers to vary the sample temperature. A thermocouple senses the temperature, which the VT controller continuously displays on the front panel. The controller compares the user-requested value with the current probe temperature and changes the heater current accordingly. The VT controller then reports the temperature of the gas flow and status to the spectrometer through a serial port at the rear of the console.

C.1 VT Setup

Use the System settings window to configure the spectrometer for the VT accessory and to enter a variable temperature cutoff value. The variable temperature cutoff (Utilities->System settings: VT cutoff) determines the temperature below which the gas is cooled.

1. If the VT controller is off and you cannot turn it on, open the **System Configuration** window (Utilities->System settings->System config).
2. Check that the VT Controller label is set to Present.
3. Open the **System settings** window (Utilities->System settings) and enter an appropriate value for **VT cutoff** and click OK.

   Set the VT cutoff to a temperature near the ambient VT gas temperature (normally VT cutoff is correct and need not be changed). Based upon the value of VT cutoff compared to the entered temperature, the system routes the VT gas flow for either heating or cooling.

C.2 VT Startup

The VT hardware must be installed and calibrated as described in the *VT Accessory Installation* manual. Starting up the VT unit takes the following steps:

1. If the VT unit is off, turn in on with the unit power switch:

- Highland VT units -- the power switch is located on the back panel. The heater on/off switch is located on the front panel.
- Oxford VT units -- the power switch is located on the front panel.

2. If the system power has been off or the VT unit has been disconnected from the probe, reset the VT controller by pressing the POWER switch to turn the unit off, then pressing POWER again to turn it on. The VT controller also can be reset with the **Reset VT** button on the Spin/Temp page in the Start folder.

CAUTION: **For VT and probe operation, use either dry nitrogen gas or air. A mixture of nitrogen gas and air can cause spikes in the baseline adjacent the large peaks in the spectra. For temperatures above 100°C, the use of air as the VT gas is not recommended. Such use will destructively oxidize the heater element and the thermocouple.**

3. Use dry nitrogen gas if the requested temperature is over 100°C or below the dew point or 0° C, whichever is higher. Otherwise, air may be used as the VT gas. If the requested temperature is below –40°C, dry nitrogen gas is recommended for cooling the bearing, spinner, and decoupler. This prevents moisture condensation in the probe and spinner housing.

   The source of heating or cooling gas is not automatically selected. To use nitrogen, you must attach a nitrogen gas source to the VT system. The same is true when using air. The VT system only selects the routing of the gas flow.

4. Use the flow control meter on the magnet leg to adjust the flow to about 10 LPM (as shown on the flow gauge).

5. A sample that can be handled at ambient temperature can now be placed in the probe, NMR lock obtained, and field homogeneity adjusted. Samples that cannot be handled at ambient temperature should wait until the system reaches the requested temperature.

C.3 Temperature Array

If the temperature is an array, set a preacquisition delay that allows sufficient time for the sample to equilibrate after a temperature change. The system will then wait specified delay in between each temperature before starting data acquisition. Delays of several minutes are optimum because the sample will take longer to equilibrate than it takes the VT controller to stabilize the heating/cooling gas at the set point.

1. Open the **Acquire** folder, select the **Acquire** page, and click the **Arrays** button. The **Array Parameter** window opens.
2. Click New Array and enter temp in the Param Name column.
3. Specify Array Size, First Value, Increment, and Last Value.
4. Click Close.
5. Set the preacquisition delay in the Flags page under the Acquire folder.:
   `Delay __ sec before starting (for VT etc.)`

C.4 Operating Considerations

The following recommendations should help achieve better VT performance.

- The spectrometer system was designed and tested with a VT gas flow rate of about 10 LPM. Sizable deviation from this rate can result in significant inaccuracy in temperature calibration and reduce the attainable temperature limits.
- Initial cool-down of the exchanger and transfer tubing after the coolant is added increases the initial time required to reach regulation (about 5 to 10 minutes for –40°C with liquid nitrogen). Because this may be longer than the `vtwait` parameter, an `su` command is the best way to start up.
- Below –40°, using dry nitrogen gas for the spinner and bearing air supply avoids moisture and frost buildup on the spinner housing and turbine. Should this happen, the sample spinning can become erratic or stop altogether.
- Every sample has some vertical temperature gradient. Minimize the gradient by *not* filling the sample tube more than about 25 to 32 mm (1 to 1.25 in.), by inserting a vortex plug or glass wool plug in the tube just above the sample solution, and by entering the liquid column to the probe coil center lines. The plug reduces refluxing of the solvent in the upper portion of the tube. Any mass movement, such as refluxing or convection, can seriously degrade resolution and lock stability.
- Above 100°C, use dry nitrogen gas to reduce heater and thermocouple oxidation.
- High-power decoupling adds heat to the sample. The increase in temperature depends on the dielectric of the solution and the power level. Under these conditions, the temperature accuracy under VT control is significantly affected. If necessary, reduce decoupler power and use a more efficient decoupling mode, such as WALTZ-16 or GARP.
- Overnight or long-term unattended VT operation at low temperatures is hampered by the fact that the usual coolant, liquid nitrogen, provides only about 1 to 2 hours of operation on a single fill of the coolant bucket. Some other coolant that lasts longer can be used if the operating temperature does not require the low temperature of liquid nitrogen. A common alternative is a mixture of dry ice and acetone. Another option is a fluid such as isopropyl alcohol or ethylene glycol cooled indirectly by a refrigerating device. Do not use aromatic, ketone, and chlorinated solvents (including acetone) in the coolant bucket. Such coolant media attack the standard polystyrene bucket.
- The ability of the VT unit to achieve temperature stability is directly affected by the stability of the room temperature. The VT unit compensates for about 80% of external changes (leaving 20% uncompensated for). Thus if the temperature of the room changes by 1°, the sample temperature will change by about 0.2°, which will not be reflected in a change in the numerically displayed temperature. For best results, the room temperature should be made as stable as possible. Any cycling of the temperature due to air conditioning or heating should be accomplished with the shortest possible cycle time and the minimum possible temperature variation.
- High-stability and independence from room temperature can be achieved if the VT controller is equipped with an optional cold junction (CJ) compensator. With the high-stability feature, the VT controller is no longer compensated for room temperature changes, but instead receives its reference from the cold junction devices mounted in the magnet leg. As the CJ compensator reduces the room temperature influences on the system, the influences of the VT gas supply become more apparent. For optimum performance of the CJ compensator, the flow and temperature of the VT gas supply must be as stable as possible.
- A possible setup to help stabilize the VT gas supply is to run the VT gas through a heat-exchanger coil in a water bath at a regulated temperature. For best results, use an ice bath to cool down the VT gas to between 5°C and 10°C, and keep the flow as stable as possible for experiments below 40°C. Generally, for best performance of the VT

controller and heater in the probe, the VT gas supply temperature should be a minimum of 10°C below the set temperature.

- For exact determination of sample temperature, a temperature calibration curve must be made for each probe used. All data, such as gas flow, must be noted. Samples of ethylene glycol are used for high-temperature calibration, and samples of methanol are used for low-temperature calibration.

    a. After bringing the sample to the desired temperature and allowing sufficient time for equilibration, obtain a spectrum.

    b. Display two cursors and align them on the two resonances in the spectrum.

    c. If the sample is ethylene glycol, enter `tempcal('e')`; if the sample is methanol, enter `tempcal('m')`.

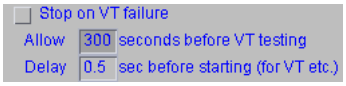
    d. The temperature is calculated and displayed based on the difference frequency between the cursors.

C.5 VT Error Handling

Select how VT errors are handled in the Spin/Temp page under the Start folder.

Interlock selections
(Start folder, Spin/Temp page)

Wait setting
(Acquire folder, Flags page)



Abort after temperature error	the VT regulation light is monitored during the course of the experiment, and if it starts to flash (regulation lost), the current data acquisition is stopped. The acquisition does not resume automatically if regulation is regained. With Abort selected, a maximum limit is imposed on the time that the system waits for regulation to be established. This limit is determined by the wait before testing setting and is independent of the delay setting. If regulation is not established after the wait time (normally set to 180 seconds), the system displays the message `VT FAILURE` and does not proceed with the experiment. If the regulation problem is later corrected, the experiment can be resumed.
Warn after temperature error	the VT regulation light is monitored during the course of the experiment, and if it starts to flash (regulation lost), a warning is generated but acquisition is not stopped.
Ignore temperature error	the temperature interlock feature is turned off

The temperature interlock selections (Abort, Warn, or Ignore) and VT wait time (Acquire folder, Flag page) check VT operation and stop the experiment if temperature regulation is lost.

For both the Abort and Warn selections, the lost regulation causes error processing to occur, thus providing a user-selectable mechanism to respond to VT failure.

The interlock operation does not apply to the cases when VT regulation is temporarily lost as a result of a programmed temperature change in an experiment where temperature is an array. The VT gas flow has no sensor or interlock. If gas flow stops, the heater is protected

by an internal temperature limit sensor that turns off the heater current before the element overheats. Because a loss of gas flow will result in a loss of regulation, any experiment in progress is stopped if Abort is selected. Only the sample is left unprotected if VT gas stops.

CAUTION: **Do not run unattended a sealed sample of highly volatile materials that must be kept cold to avoid excessive pressure buildup. The undetected loss of VT gas or exchanger coolant could result in the rupture of the sample tube and damage to the probe components.**

C.6 VT Controller Safety Circuits

The VT controller includes safety circuits to avoid damage to the heating element and probe. The following error conditions produce an error code:

- Open circuit in the thermocouple circuit.
- Open circuit, short circuit, or over-temperature at safety sensor.
- Short circuit or software/microprocessor failure at the output transistor.

Over-temperature at the safety sensor initially turns off the heater. If this method fails to correct the condition within 5 seconds, either the gas flow has been interrupted or an output transistor failure has occurred, whereupon a protective relay operates, isolating the heater from the control electronics. Failure of any of the sensors also results in this relay operating.

Once the protective relay has operated, the output will remain off. A power-down and power-up cycle of the VT controller is required to release the relay.

The over-temperature circuit can be inadvertently tripped if the VT is started at a below ambient temperature and the temperature is increased greater than 70°C. If the circuit is tripped, reset it by turning the VT off and on, then change to the desired temperature in 50°C steps.

Excessive heat requirements that cause the current to remain near the maximum can also trip the second circuit. Therefore, when using liquid nitrogen for cooling and when operating from 0°C through +25°C, reduce the gas flow rate to between 8 and 9 LPM. Reset will also occur if the VT cable is removed from the probe while the VT is on.

Refer to the *VT Accessory Installation* manual for system failure analysis.

Appendix D. Probe Tuning with qtune

This section describes how to use the graphical probe tuning program, `qtune`, for swept-tune-type NMR probe tuning.

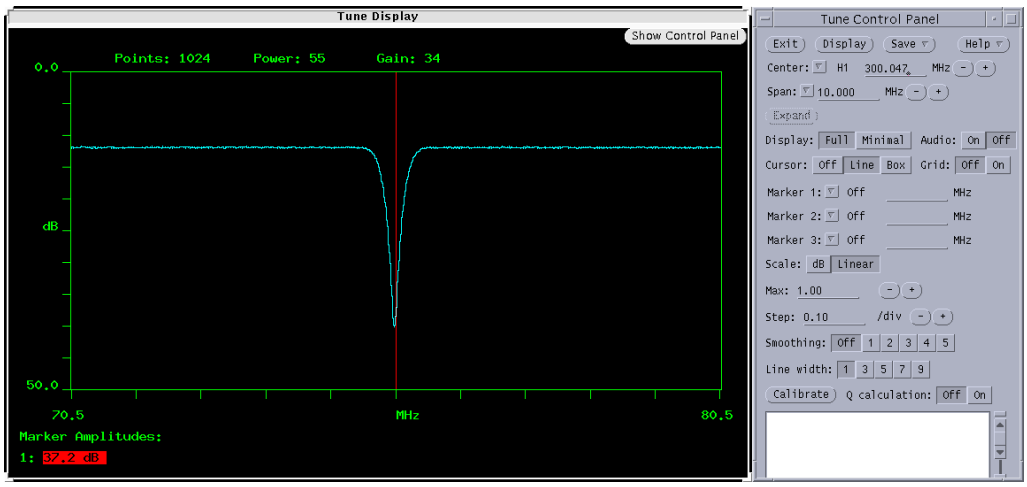


Figure 55. Probe Tuning Window (`qtune` Program)

`qtune` runs on the host computer and offers you an interactive tuning method that provides separate information for matching and resonant frequency. This program is especially useful for tuning probes with complicated coil configurations, such as imaging probes.

After the system is put into tune mode, the reflected power from the probe passes through the directional coupler and is detected and digitized by the receiver circuitry: Any power that the receiver detects is reflected power. Taking one (or more) complex pair of data points at each frequency gives a data set that shows reflected power versus frequency. A coil tuned to a specific frequency (usually the frequency of the nucleus the user wants to observe) reflects little power at that frequency. The acquisition system then sweeps through the desired frequency range and gathers data on reflected power interactively. The user can adjust certain parameters interactively during the experiment.

D.1 Tuning a Probe with qtune

This procedure describes how to use the `qtune` program to tune an NMR probe.

1. Set up the system for tuning:

   UNITY*INOVA* – leave the thumbwheel switch on the TUNE INTERFACE set to observe mode, or 0.

   MERCURYplus/-Vx – connect the following cables:

   - Connect the appropriate cable from the probe (J5102 or J5302) to the TUNE.
   - Connect the appropriate cable from the transmitter (J5602 or J5603) to the TUNE J5604 connector.
   - Connect the appropriate cable from the receiver (J5303 or J5103) to the Q TUNE J5403 connector.

   MERCURY Magnet Interface Box – connect the following cables:

   - Connect the appropriate cable from probe (J5102, Hi Bnd PreAmp, or from J6001, BB Probe,) to J5402, Tune.
   - Connect the appropriate cable from the transmitter (J5602, Hi Bnd Tx, or J5603, Lo Bnd TX) to J5604, Tune.
   - Connect the appropriate cable from the receiver (J5103, Hi Bnd Out, or J5303, Lo Bnd Out) to J5403, Q-tune Tx Hi/Lo Bnd Obs Rx.

2. Enter **`tn='`*`n`*`' su`**, where *n* is the nucleus to be tuned (e.g. `tn='H1'`).

3. Click the Probe button on the VnmrJ hardware bar. Enter an appropriate value for Tune gain and click Tune sweep.

System	*Tune gain*
INOVA	50
MERCURYplus/-Vx	15

   The Tune Display and Tune Control windows open, similar to Figure 55. The Tune Display is centered on the resonant frequency of the current experiment (`sfrq`).

   To change the gain and power values, click Exit in the Tune Control Panel and reenter the `qtune` command with more appropriate values.

4. In the **Display** field, select **Full** or **Minimal**.

   - Full display shows the network-analyzer-like graph, as shown in Figure 55.
   - Minimal display simplifies the tune display by showing numerical values instead of the graph, as shown in Figure 56. When sweeping over a range of frequencies, the minimal display shows minimum reflected power and center frequency. When in CW mode, the minimal display shows the average reflected power and the current frequency. Several display controls on the Tune Control Panel are disabled in minimal display mode, including cursors, grid, and markers.

5. In the **Audio** field, select **On** or **Off**.

   If set to On, a volume slider becomes available and a sound is generated that indicates how close the minimum reflection is to the center of the sweep window—the lower the pitch of the sound, the closer the minimum reflection is to the center.

Figure 56. Minimal Display Mode for the Tune Display

The sound immediately stops if the response lacks a discernible minimum reflection or if CW mode is set.

To use the audio capability requires:

- An audio speaker on the Sun computer.
- The probe tuned well enough so that a clearly discernible minimum reflection exists in the signal.
- A range of frequencies sent to the probe (i.e, audio requires sweep mode rather than CW mode).

6. Place a marker on the resonant frequency to which you want to tune the probe as follows (this option is not available in minimal display):

   a. In the Tune Control Panel (see Figure 57), click on the triangle next to one of the markers to open a pull-down menu (Figure 61 shows the Marker 1 menu).

      For more detail on using markers, see the "Using Cursors, Grid, and Markers," page 272.

   b. In the menu, select the resonant frequency to which you are tuning the probe.

A marker corresponding to the selected frequency appears in the Tune Display window.

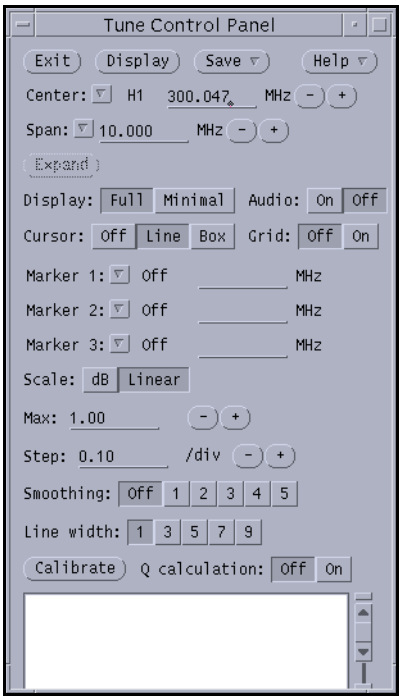


Figure 57. Tune Control Panel (`qtune` Program)

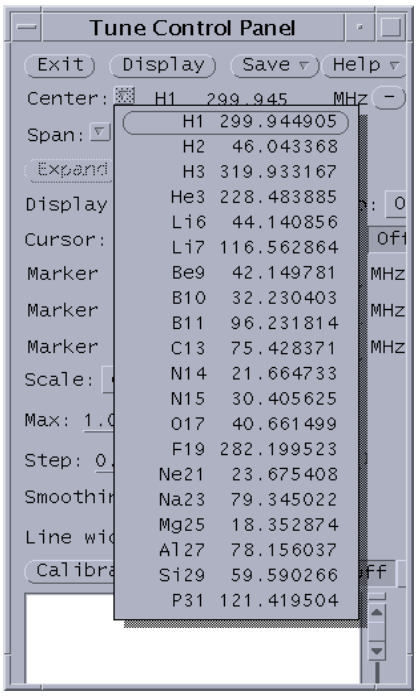


Figure 58. Pull-Down Menu for Center Frequencies

7. In the Tune Control Panel, type values as appropriate in the **Span**, **Scale**, and other fields on the Tune Control Panel.

   Using the dB scale usually facilitates probe tuning. See the "Adjusting the Span," page 271.

8. Adjust the tune and match capacitors while watching the Tune Display window. Use the match capacitor to increase the depth of the dip as much as possible. Use the tune capacitor to center the dip on the marker created in step 6.

   The dip displayed in the Tune Display window shows where little power is reflected at the frequency being observed. The depth of the peak shows the accuracy of the impedance matching of the probe coil to the transmitter and receiver. The horizontal location of the dip shows the frequency at which little power is reflected. The goal of probe tuning is to increase the depth of the peak (matching) while centering the dip at the desired frequency.

9. After the probe is tuned, click **Exit**. You are ready to begin the experiment.

D.2 Selecting a Center Frequency

The center frequency is the resonant frequency to which the probe is to be tuned. A list of center frequencies appears in the Center pull-down menu in the Tune Control Panel (see Figure 58).

Note that any frequency you type into the field will not be read until you press Return. This applies to all text entry fields.

1. In the **Center** field, click on the triangle to open the pull-down menu, then select a frequency that equals or is close to the frequency you want.
2. Adjust the frequency by typing a new value in the Center field or by clicking the – and + buttons.

The – or + buttons decrease or increase the value by the width of one span.

If the center frequency is either typed in or changed by the – or + button, user is displayed next to the frequency. If the typed value happens to correspond to a nuclear frequency in the pull down menu, that nucleus is displayed.

If the new center is too close to either of the system frequency limits, the span is decreased to allow the new center to be accepted, the message window will beep, and an error message will appear. If the specified center is past either frequency limit, the message window will beep, and an error message will appear.

D.3 Adjusting the Span

The span is the sweep width, in MHz, used in the Tune Display.

1. To decrease or increase the span to the next in the series 1,2,5 10,20, etc., in the **Span** field, enter a value or click the – or + button.

   Setting the span to less than 1000 Hz causes the message window to beep and an error message to appear.

   Setting the span to more than maximum span causes the message window to beep and an error message to appear.

   Setting the span to a value beyond the maximum or minimum frequencies causes the span to decrease, the message window to beep, and user warnings to appear.
2. In the **Span** field, click on the triangle to open the Span pull-down menu (see Figure 59).
   - Select **Last Span** to return to the previous span value
   - Select **Maximum Span** to make the spectrometer sweep from the minimum to maximum allowable frequencies.
   - Select **CW** to temporarily stop frequency sweeping and to make the transmitter put out a CW signal. This sets the frequency to the currently selected center frequency. This mode is useful for checking the reflected power on the tune meter or for making other tests that require a fixed frequency.

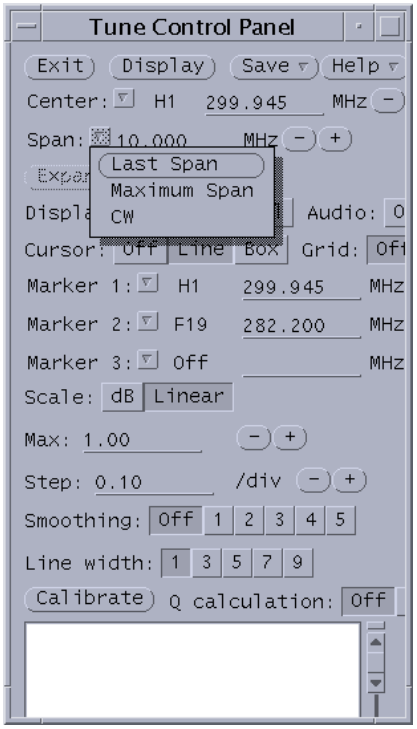


Figure 59. Pull-Down Menu For

D.4 Using Cursors, Grid, and Markers

Cursors and markers appear on the Tune Display and are used the same way they are used in VnmrJ. Cursors and markers are color coded and the frequency positions of each are displayed on the bottom of the Tune Display. Cursor, grid, and marker controls are not available in minimal display.

Cursors – In the **Cursor** field, the Tune Control Panel provides three cursor modes (see Figure 60)

Figure 60. Cursor and Grid Controls

- Select **Off** to turn cursors off so that no cursors are displayed.
- Select **Line** to display a single cursor that specifies one frequency. The frequency position of the cursor and the signal amplitude at that frequency are displayed on the bottom of the Tune Display window.
- Select **Box** to display two cursors for use with the Expand button (for setting the span to a narrower range). After a region is expanded, the cursor mode switches back to the Off mode. The frequency positions of the cursors, as well as the delta and the signal amplitudes, are displayed on the bottom of the Tune Display window.

Grid – In the **Grid** field, select **Off** or **On** (see Figure 60) to control a grid display in the Tune Display window. The grid helps in reading the reflected power levels off of the graph. The grid does not slow down the drawing time of the graph.

Markers – In the **Marker 1**, **Marker 2**, and **Marker 3** fields, the Tune Control Panel provides three markers for marking fixed frequencies. Each marker has a pull-down menu (see Figure 61) that lists the same nuclear frequencies and the Center pull-down menu. Each marker also has an entry field for entering a frequency.

You can use markers for observing fixed frequencies, for example the two nuclear frequencies of a double tuned probe. You can then vary the span, and the markers will appear and disappear, depending on whether their frequencies are being scanned in the current experiment.

Attempting to set the markers to values beyond the system frequency limits causes the message window to beep and produces an error message.

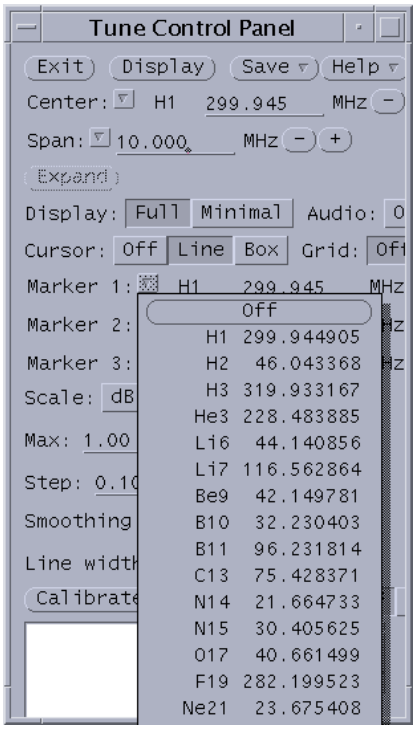


Figure 61. Pull-Down Menu for Marker 1

To Change the Vertical Scale

You can change the vertical scale of the Tune Display window by selecting whether the data is scaled in a linear or logarithmic manner (see Figure 62).

1. In the **Scale** field, the Tune Control panel provides two scaling modes:
   - Select **dB** to provide a logarithmic scale in units of dB.
   - Select **Linear** to a linear scale with arbitrary units.

   The dB scale shows a deeper dip for more accurate match adjustment.

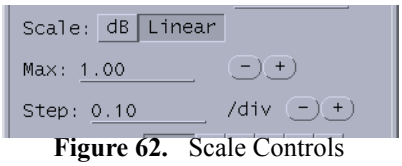


Figure 62. Scale Controls

2. In the **Max** field, type a value you want for the top line in the Tune Display.

   Setting a Max value beyond system limits generates a beep and an error message.

3. In the **Step** field, type a value you want for the number of units per division mark of the current scale (dB or linear) on the *y* axis of the Tune Display.

   Step helps by making the height of a probe resonance appear larger or smaller.

   Setting a Scale value beyond system limits generates a beep and an error message.

D.5 Q Calculation

Probe Q factor determines sensitivity. Q is defined as the frequency of the resonant circuit divided by the half power bandwidth.

1. Near the bottom of the Tune Control Panel, select Q calculation: On.

   The current Q value and resonant frequency appears at the top of the Tune Display window, and a horizontal cursor appears on the plot.

2. In the Tune Display window, use the middle mouse button to place the horizontal cursor on the base line (reflected power level outside of the resonance line). The Q calculation appears at the top of the Tune Display window.

The Q that is shown is determined as follows:

- The software finds the lowest point on the display and designates this as the resonance. The frequency displayed is, at best, only as accurate as the frequency difference between points. You must take this into account when quoting Q measurements.
- This lowest point is used as V_{min}. The software takes the level of the horizontal cursor as the baseline, or V_{max}.

  V_{max} is assumed to be a frequency where all the rf energy is reflected by the probe.
- The two frequencies that have the signal level of Equation 3 are ω_1 and ω_2.

$$\frac{V_{max}}{\sqrt{5}} \qquad \text{[Eq. 3]}$$

- Q is calculated from Equation 4, where ω_r is the resonant frequency

$$Q = \frac{\omega_r}{|\omega_1 - \omega_2|} \qquad \text{[Eq. 4]}$$

- The software checks that the low point (the bottom of the dip) is at least 15 dB below the baseline. If this is not true, the calculated Q value is not accurate, and is, therefore, not reported (the string "- - -" appears in the Q value field). The resonance frequency, however, is still given.

D.6 Calibrating the Tune System

A Tune Calibration window is available for calibrating the tune system using a shorting load, an open load (no device attached), and a 50-ohm load.

To calibrate the tune system.

1. Set up the system for tuning:

   UNITY*INOVA* – Leave the switch set to observe mode.

   MERCURY-Vx and *MERCURYplus* – Connect the following:

   - Connect the appropriate cable from the probe (J5102 or J5302) to the TUNE.
   - Connect the appropriate cable from the transmitter (J5602 or J5603) to the TUNE J5604 connector.
   - Connect the appropriate cable from the receiver (J5303 or J5103) to the Q TUNE J5403 connector.

   Mercury Magnet Interface Box – connect the following cables:

   - Connect the appropriate cable from probe (J5102, Hi Bnd PreAmp, or from J6001, BB Probe,) to J5402, Tune.
   - Connect the appropriate cable from the transmitter (J5602, Hi Bnd Tx, or J5603, Lo Bnd TX) to J5604, Tune.
   - Connect the appropriate cable from the receiver (J5103, Hi Bnd Out, or J5303, Lo Bnd Out) to J5403, Q-tune Tq Hi/Lo Bnd Obs Rx.
   - Enter **`tn='n' su`**, where *n* is the nucleus to be tuned (e.g., `tn='H1'`).

2. Click the **Calibrate** button on the Tune Control Panel. The Tune Calibration window opens (see Figure 63).

   You must run the calibration tests in the following order:

   - Short Test
   - Open Test
   - 50 Ohm Test

   The test buttons are only available in this order.

   The program stores only correction coefficients in files. Therefore, you will not be able to view old calibration sweeps after the experiments are finished.

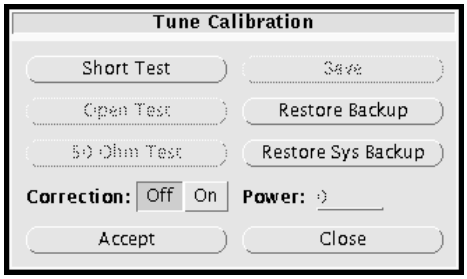


Figure 63. Tune Calibration Window

3. In the **Correction** field, select **Off** or **On** to disable or enable calibration corrections, respectively. The default is enabled.

   If valid calibration files are not present, this button is not available.

4. Connect a shorting load:

   - UNITY*INOVA* – Connect a shorting load to the tune port.
   - *MERCURY-Vx* and *MERCURYplus*– Connect a shorting load to TUNE, J5402 on the inside (probe side) of the magnet leg or for the Magnet Interface box connect the shorting load to J5402, Tune.

5. Click the **Short Test** button.
6. Remove the shorting device from the tune port.
7. With no load on the tune port, click the **Open Test** button.

   The program divides the full range of the spectrometer into 32 frequency bands and runs 32 experiments. This data is merged into one high-resolution data set. After enough data are collected, the Accept button becomes available. If ADC overflow occurs, a warning message appears.
8. After you see a strong signal (the signal will be stronger at low frequencies than at high frequencies) that does not cause ADC overflow, click the **Accept** button.
9. Attach a 50-ohm load:
   - UNITY*INOVA* – Connect the 50-ohm load to the tune port.
   - *MERCURY-VX* and *MERCURYplus* – Connect the 50-ohm load to TUNE, J5402 on the inside (probe side) of the magnet leg or for the Magnet Interface box connect the 50-ohm load to J5402, Tune.
10. Click the **50 Ohm Test** button.

    The program divides the full range of the spectrometer into 32 frequency bands and runs 32 experiments. The data is merged into one high-resolution data set. After enough data are collected, the Accept button becomes available. If ADC overflow occurs, a warning message appears.
11. Click the **Accept** button and remove the 50-ohm load from the tune port.
12. After all three tests are finished, click the **Save** button.

    The old calibration files are moved to `.bak` files, and new calibration files `std_ed.cal`, `std_es.cal`, and `std_er.cal` are calculated. These files are stored in `$vnmrsystem/tune/tunecal`. If the files cannot be saved, the program produces a beep and an error message.
13. Click the **Close** button.

To Restore the Previous Calibration File

If you have an incorrect calibration file and do not want to re-run the calibration tests, restore the previous calibration files:

1. Click the **Calibrate** button on the Tune Control Panel. The Tune Calibration window opens (see Figure 63).
2. Click the **Restore Backup** button.

The program replaces the new calibration files with the `.bak` files. If the program does not find all of the backup files, an error message appears.

D.7 Restoring the Original Calibration Files

System backup files can be installed by copying the following `.cal` files on top of the corresponding `.sys` files:

- `std_ed.cal` to `std_ed.sys`
- `std_es.cal` to `std_es.sys`
- `std_er.cal` to `std_er.sys`

To restore these system calibration files, click the Restore Sys Backup button. The program replaces the new calibration files with the `.sys` calibration files created at system installation.

Appendix E. Shimming Using the Ultra•nmr Shim System

The Ultra•nmr Shims unit is a matrix shim system designed to achieve a high level of sample homogeneity over large sample volumes (e.g., 10-mm diameter), to generate more orthogonal shim gradients that are easier to use, and to provide this in a more reproducible and stable manner.

Ultra•nmr Shims are integrated with the UNITY*INOVA* system and all shim functions are available. Select the Ultra•nmr Shims from the CONFIG window or use the Ultra•nmr Console Data button in the window. The Ultra•nmr Shims unit can be connected to either port 2 of the acquisition CPU or PJ1 of the Magnet Sample Regulation (MSR) board. The MSR connection is preferred. If the interface box is present, the shims set by the console should be visible in its window, but the box should not be used.

CAUTION: **If an Ultra•nmr shim system can be controlled from the Acquisition window, do not set the shim value beyond ±32000 DAC counts.**

E.1 Hardware Overview

The system uses a matrix of 48 channels to provide current into 49 coils within the room-temperature shim tube from which 38 or 39 shim gradients are generated. Each of the axial shim gradients is produced by applying currents simultaneously in up to 12 different channels, and each of the radial gradients, in up to 6 different channels with the 48-channel matrix system. The distribution of current within the allowed channels for each gradient is under computer control and is calibrated to produce the purest (most orthogonal) possible single shim gradient. For example, the distribution of current for the Z2 gradient is adjusted so that the Z2 gradient produced has minimal amounts of other gradients, such as Z (=Z1) or Z0.

An interface box provides five user-selectable, optically encoded knobs to control the shim gradients. The interface box also provides the user with a means of storing, viewing, and recalling up to 63 shim sets on a floppy disk. Each of the shim gradients has both coarse

and fine controls, with each control having a range of ±32767 DAC counts. The following gradients are controlled:

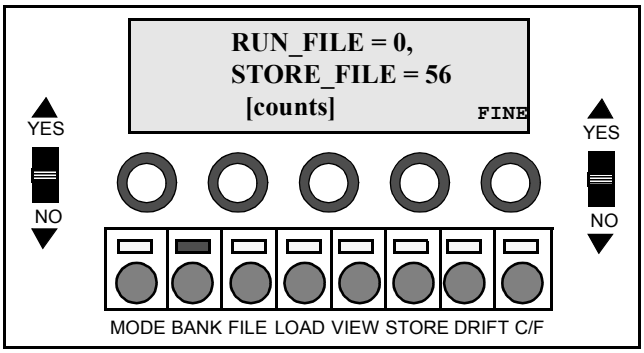
Axial:	Z0, Z, Z2, Z3, Z4, Z5, Z6, Z7, Z8
Radial:	X, ZX, Z2X, Z3X, Z4X, Z5X
	Y, ZY, Z2Y, Z3Y, Z4Y, Z5Y
	C2, ZC2, Z2C2, Z3C2, Z4C2 (where C2 = XY)
	S2, ZS2, Z2S2, Z3S2, Z4S2 (where S2 = X2Y2)
	C3, ZC3, Z2C3, Z3C3 (where C3 = X3)
	S3, ZS3, Z2S3, Z3S3 (where S3 = Y3)

E.2 Interface Box

The interface box has a small display screen and the following controls:

- Five knobs whose gradient assignments are selected by toggle switches.
- Two identical toggle switches, each labeled YES/NO.
- Eight buttons: MODE, BANK, FILE, LOAD, VIEW, STORE, DRIFT, and C/F. A small indicator above each button turns green if the button is active.

Ultra•nmr Shim Interface Controls and Display



The MODE, BANK, and FILE buttons select the system mode:

- MODE mode allows the user to reconfigure the system and the display screen by using the toggle switches to select configuration options.
- BANK mode makes shim gradient control accessible through the knobs. In this mode, the toggle switches select banks of knob-to-gradient assignments, which are displayed on the display screen directly above the knobs.
- FILE mode lets the user load from, view, or store to whichever STORE_FILE file number is selected by the toggle switches. These file actions are initiated by the LOAD, VIEW, or STORE buttons, respectively. Immediately after any LOAD or STORE operation, the RUN_FILE and STORE_FILE file numbers are identical. If any changes are subsequently made to the existing gradient DAC values, RUN_FILE is reset to 0.

The last button on the right, the coarse/fine gradient control (C/F) button, is active in both the BANK and the FILE modes.

Files 2 through 63 are available for user storage. Users can *not* write to file 0, which contains all zeros, or write to file 1, which contains converged shim values. Files are stored

on a DOS-formatted 1.4-MB floppy disk in a drive accessible from the front of the main unit. The small cover panel at the right edge of the upper card cage must be removed to gain access to the system floppy disk. New disks must be formatted using DOS version 5.0 or higher. Numerous calibration files will be copied to that disk. Copies of existing floppy disks can be made using the DOS `diskcopy` command on any machine running DOS version 3.3 or higher; for example, enter `diskcopy a: a:` if the particular floppy drive is configured to be drive A.

One of the selectable options in the MODE mode allows system lock out for security reasons. If this feature is activated, all knob, button, and toggle control is disengaged until the MODE, BANK, and VIEW buttons are simultaneously pressed. It is also possible to disable (turn off) the lock from the interface box in the MODE mode. This feature is contained in the Z0 enable/Z0 disable option.

After Ultra•nmr Shims are installed, Z0 is not controlled via the Acquisition window but only from the Ultra•nmr Shims interface box. All other lock parameters—`lockpower`, `lockgain`, and `lockphase`—remain controlled through the Acquisition window. The value of Z0 changes about +3500 coarse DAC units on a 500-MHz magnet (about +4200 on a 600-MHz magnet) when the lock solvent changes from $CDCl_3$ to acetone-d_6. The other shim gradients are also no longer controlled through the Acquisition window.

Homospoil (Z gradient) produces approximately a 0.6 G/cm field (typically 99% of transverse magnetization is gone within 1.5 ms and signal recovery is 90% within 40 ms) and is activated in the same manner as a spectrometer linked to Oxford room-temperature shims.

E.3 Shimming

In the normal “counts” display mode, each shim gradient has a coarse and a fine control. The coarse control is 50 times more sensitive than the corresponding fine control. All gradient DACs have a range of ±32767; a knob twisted beyond this range will continue to turn but have no effect. Each of the gradient DACs has an identical and a logical polarity. For example, a clockwise adjustment of Z2 moves an asymmetry to the right, with Z2 coarse (Z2C) and Z2 fine moving asymmetries in the same direction. Furthermore, Z4, Z6, and Z8 also move asymmetries in the same direction as Z2. The even-order axial shims (Z2, Z4, Z6, and Z8) may jerk the lock if large, sudden changes are made; the severity of the jerk decreases typically in the order Z4, Z8, Z2, and Z6.

Within gradient families, you also observe sensitivity differences, with higher order gradients causing less effect per DAC unit. The following suggestions are offered:

- Z0: use coarse only (fine is too fine).
- Z: use fine only (coarse is *far* too coarse).
- Z2, Z3, Z4: use coarse or fine, whichever seems best.
- Z5, Z6, Z7, Z8: use coarse only.
- Radial shims: use coarse, with the possible exception of X and Y.

Remember that the gradients produced are purer (more orthogonal) so shimming methods subjectively generated on other shim systems may not behave the same. If you adjust any axial shim, you should probably touch up Z before touching any other gradients (for example, if you change Z4, then optimize Z before touching up Z2). In addition, be forewarned that 80% of what might be corrected with Z4 using an Oxford shim system is probably corrected with Z2 when using Ultra•nmr Shims.

E.4 Installing the First Probe

Starting from the final converged results obtained with field mapping, it is possible to obtain a lineshape of less than 7/12 Hz (non-spin) in under an hour or two on 1% $CHCl_3$ in acetone-d_6 (ASTM ^1H lineshape sample). Z and Z2 typically change the most (Z fine needs about –70 units, Z2 about –400 units), but Z3, Z4, Z5, X, and Y will need adjustment. The higher-order radial shims are typically converged to a better value than a user can determine spectroscopically and should therefore not be adjusted manually (only X, Y, ZX, ZY, and possibly C2 and S2 should need adjustment). A simple maximization of the lock level has proven to be a sufficient criterion.

E.5 Floppy Disk Use

Every Ultra•nmr Shims floppy should contain 83 system files and up to 62 user files. The system files include 39 strength files (`*.str`), 39 divider files (`*.div`), a gradient-channel configuration file (`*.him`), a shim-coil resistance file (`*.res`), a shim configuration file (`*.cfg`), and two shim files (`file0.dac` and `file1.dac`). `file1.dac` is to contain the shim values for the $CHCl_3$ lineshape sample in one particular probe. The 62 user files are the shim files 2 to 63 (`file#.dac`).

A version of each of the system files is stored in the system PROM (programmable read-only memory). If any system file is missing on the floppy disk, the PROM version is used instead. It is acceptable to use the PROM versions for the `*.him`, `*.res`, and `*.cfg` system files. It is *not* acceptable to use the PROM versions for the `*.div` and `*.str` files because these files are set at the time of system installation and are magnet dependent.

The `readultra` macro reads shim set files for the Ultra•nmr shim system from a floppy disk on a Sun workstation (e.g., entering `readultra(6)` reads shim set file 6). Entering `readultra` with no argument reads all of the shim set files. Before using `readultra`, the floppy disk is expected to be mounted as `/pcfs` on the Sun workstation. For details, refer to the description of `readultra` in the *Command and Parameter Reference*.

In a multiuser environment, it is probably best if each user has a separate floppy disk on which that user's particular shim files can be written. In any environment, backup floppy disks are always a good idea. Two backup system disks are made at the time Ultra•nmr Shims is installed: one is to be kept by the customer; the other is sent to Varian. Backup disks can be easily made on any DOS-based computer using the DOS `diskcopy` command.

CAUTION: **Keep floppy disks away from the magnet dewar. Data on the disk is susceptible to damage from intense magnetic fields.**

When changing floppy disks, observe the following procedure:

1. Save the current shim values into one of the 62 available shim files.
2. Load in file 0.
3. Remove the current floppy from the drive and immediately insert the new system floppy into the drive. *Do not operate the controls on the interface box when there is no floppy in the drive*.

E.6 Turning the System Off and On

The system is typically left on continuously except for maintenance.

To Turn Off the System

Use the following procedure to power down the Ultra•nmr Shims system:

1. Save the current shim values into one of the 62 available shim files.
2. Load in file 0.
3. Turn off the orange-lighted rocker switch on the power strip in the lower back of the shim power supply.

To Turn On the System

Do the following procedure to turn the shim system back on:

1. Press the rocker switch on the power strip.

   The orange light in the switch should turn on and the interlock board in the bottom left front of the shim power supply should display two red lights and one green light. The bootup of the computer in the shim power supply takes approximately one minute. During the bootup, all eight indicators, situated above the eight buttons on the interface box, are usually lighted green. Note that the system will not boot unless a readable floppy disk is in the system floppy drive.

   When the bootup is complete, the interface box displays several status messages indicating how many PROM-based default files were used during bootup.

2. When the final question on the interface box, answer “yes” to disengage the high-power interlock.

When the high-power interlock disengages, one of the two red lights previously displayed at the front of the interlock board turns off. It is *perfectly normal* for the other red light to stay on all the time.

Symbols

Numerics

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W

X

Z

Index