Contents

Chapter 1

Introduction

XWIN-NMR is the standard NMR software package provided by Bruker for spectrometer control and for visualising, processing, and printing the acquired data. The program features the following concepts:

- Data acquisition using Bruker AVANCE and AMX/ARX/ASX spectrometers.

  XWIN-NMR supports the power user by allowing for flexible control of all acquisition hardware and software parameters, including the design and visualisation of pulse programs, real time observation of fid, lock signal, and transmitter output, and monitoring temperature and magic angle spinning rate.

  The routine user is supported by the simple user interface of ICON-NMR which allows for painless execution of complicated measurements using Bruker's standard library of experiments, or experiments designed by the laboratory manager.
- Acquisition of n-dimensional data sets, and processing of 1D, 2D, and 3D data sets. Bruker's advanced analysis package AURELIA/AMIX also treats 4D data and the complicated spectra of mixtures.
- The data analysis part of XWIN-NMR provides line deconvolution based on Lorentzian and Gaussian shapes, the calculation of T1 and T2 relaxation times, and the fitting of multi-exponential decays. Baysian analysis and an interface to

the MaxEnt maximum entropy package make advanced methods available which are complementary to the Fouier transform.

- XWIN-NMR is capable of processing data acquired with Bruker spectrometers based on Aspect 2000/3000 and X32 computers, and with Jeol and Varian spectrometers using JNMR or VNMR software, respectively. Furthermore, XWIN-NMR is able to import and export data in the JCAMP/DX ASCII format. XWIN-NMR data can be further processed and analyzed by other Bruker programs such as AURELIA/AMIX, DAISY, WIN-NMR, and the results can be re-imported. Please check the release CD of XWIN-NMR for brochures of these programs.
- Different users may start multiple copies of XWIN-NMR on the same computer in a network environment, provided a suitable X Windows server is installed on a user's PC or workstation. Licensing is discussed further below in this chapter.
- Bruker provides printed and online manuals for software, hardware, and applications. XWIN-NMR is delivered with a large set of such documentation. Please check the *Help* menu of XWIN-NMR, or the subdirectories

  XWINNMRHOME/prog/docu/english/

  containing the documentation files. Use the command acroread (at Unix level) to open the documentation viewer.

  Throughout this manual, *XWINNMRHOME* will denote an environment variable representing the directory where XWIN-NMR was installed, e.g. */u*.
- Data output to plotters and printers can be performed in two ways: Using parameter driven plotting software including a plot pre-viewer (commands cfpp, edo, edg, plot, ...), or the interactive What-You-See-Is-What-You-Get plot editor XWIN-PLOT (commands xwinplot, autoplot). XWIN-PLOT is part of the XWIN-NMR distribution media.

The XWIN-NMR manual comes in 2 parts: This is part 1 and covers XWIN-NMR's general features as well as data processing, analysis, and plotting using the parameter driven plot commands. Part 2 covers data acquisition and pulse programming. Please note that there are separate manuals for ICON-NMR and XWIN-PLOT.

1.1 Hardware/software requirements

Please refer to the *NMR Software Release Letters* manual (part number Z30742).

1.2 Notation

Throughout this manual, XWIN-NMR commands appear underscored, e.g. ft or zg. Executing a XWIN-NMR command often requires parameters to be set before. Parameters are printed in capital letters, e.g. LB or SWH. All parameters may be modified by means of a parameter editor, or by typing them on the keyboard using the parameter names as commands. For example, type lb, followed by Return, to change LB. The current parameter value will be printed, and you may change or confirm it with Return. You may skip the dialog by entering the desired new value behind the command name, separated from it by a space character, e.g. lb 0.5.

1.3 Versions

This manual corresponds to XWIN-NMR version 2.0. XWIN-NMR contains the manual as online documentation accessible from the *Help* menu. In addition, this menu provides some additional documentation.

1.4 Support

Software support is available via e-mail from the following address:

nmr-software-support@bruker.de

The Bruker ftp servers *ftp.bruker.de* and *ftp.bruker.com* provide additional information such as known bugs and their workarounds in the directory

/pub/nmr/xwinnmr.

1.5 Installation of XWIN-NMR

Installation may depend on the particular program version. It is therefore described in it own manual that is distributed together with the release CD. Please refer to the *NMR Software Release Letters* manual (part number Z30742).

1.6 XWIN-NMR screen layout

The XWIN-NMR window (Figure 1.1), from top to bottom, consists of

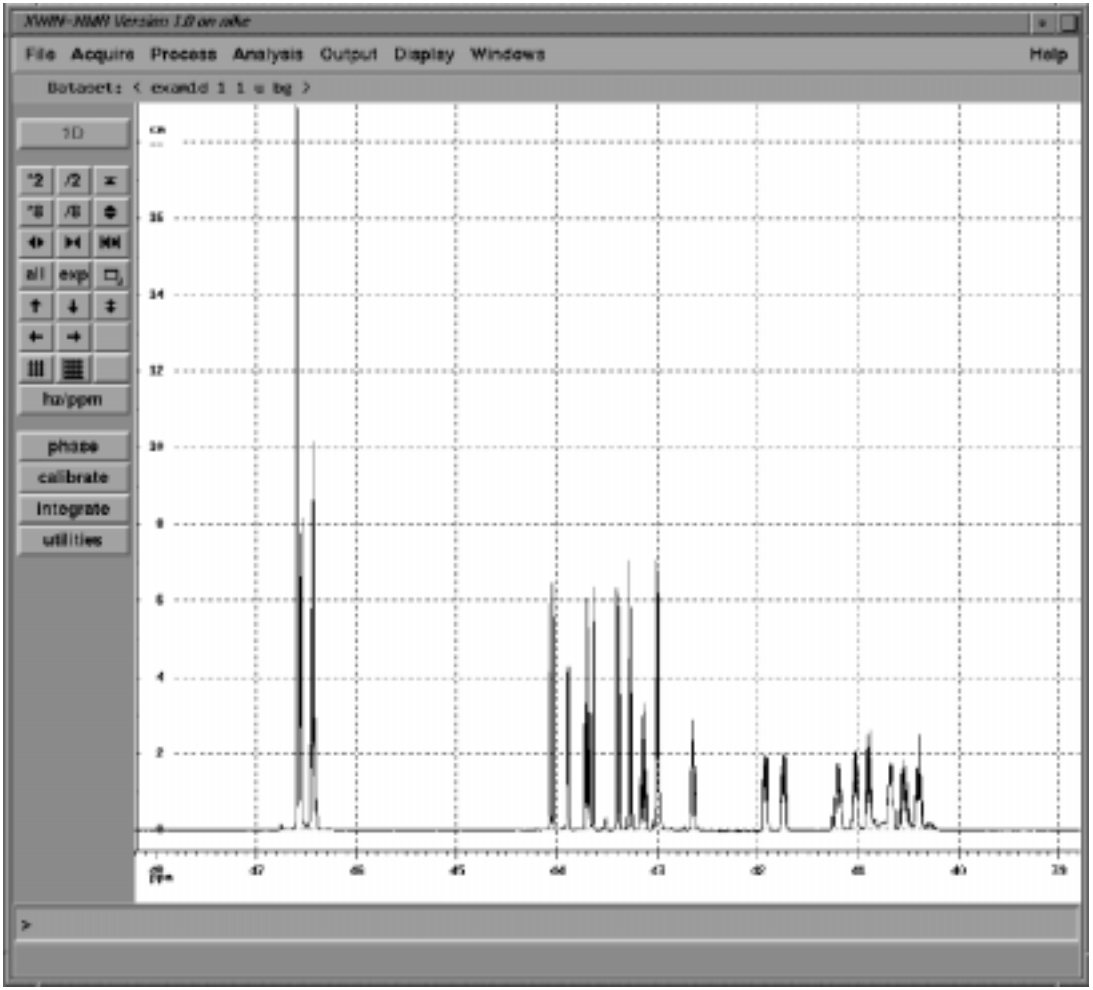


Figure 1.1 XWIN-NMR screen layout

1. **The window manager bar with program version and computer host name.**

The host name lets you know if you have started the program remotely from another workstation, an X terminal, or a PC equipped with an X server.

2. **The main menu bar with pull down menus for XWIN-NMR operations**. While the menu bar remains the same in all program modes, the contents of the pulled down menus may change depending on the context. For example, the *Process* pull down menu will contain commands to process 1D data sets if a 1D data set is visible on the screen, and 2D commands if the current data set is a 2D spectrum or 2D free induction decay. Under certain circumstances it might not be meaningful to execute a command from a menu. The menu contents then appears to be *grayed*, i.e. the intensity of the characters is reduced.
3. **The data set info**, a line showing the name of the current data set.
4. **The data display area, with an array of data manipulation buttons,** which allow quick access to frequently used operations. These buttons depend on the context, and will, for example, be different for 1D and 2D data sets. They are displayed at the left side of the window. An extended mode is provided showing even more buttons for the experienced user. In 1D mode, you may optionally display the buttons at the top of the window from left to right, below the menu bar. Enter the command setres to switch between these modes. When you exit from XWIN-NMR, the current setting is saved.
5. **The command line for keyboard input.** A large number of commands usually executed from the pull down menus may also be activated by typing them in. The command names are displayed within the menus, enclosed in brackets. For example, the Process menu contains an entry *fourier transform [ft]*. In order to execute this command from the keyboard, type *ft*, followed by the *Enter* key. The command line can be edited using the mouse (place the cursor at the desired position and use the backspace or delete keys to erase a character; mark a number of characters by positioning the cursor, then hold the left mouse button depressed while moving the mouse; type new text to replace the marked text, or use the delete key to remove it). Make sure that the mouse pointer is located within the XWIN-NMR window before you start entering commands.
6. **The status line for the display of information during command execution.** For example, when a 2D transform is in progess, the currently processed row or column numbers are shown. This line is also used to display on line help messages for the quick access buttons at the left side of the XWIN-NMR window. A one line description of a button will appear as soon as the mouse points to the corresponding button.

1.7 Manipulation of the XWIN-NMR window

Iconify
Click on the small dot at right side of the window manager bar in order to iconify XWIN-NMR.

De-Iconify
Double-click on the XWIN-NMR icon.

Resize
Move the mouse pointer to one of the window corners, where the cursor shape changes. Drag the window to the desired size while the left mouse button is depressed. Release the button. Please note that the window cannot be sized smaller than a certain limit in order to keep the quick access buttons readable. Resizing will (under the Motif window manager) always be carried out such that the height-to-width ratio remains constant.

Maximize
Click on the large dot at right side of the window manager bar. The window will automatically be resized so as to utilize the whole screen. Another click on the same button will restore the last size before maximizing.

Restore default size
When you exit from XWIN-NMR, and restart the program, it will come up with the window size of the last session. In order to obtain the default size as proposed by Bruker, start the program with the command *xwinnmr -d*.

Move to another position
While the mouse pointer is located in the window manager bar of the XWIN-NMR window, depress the left mouse button and move the mouse. Release the button when the desired position is reached.

Shuffle into foreground The XWIN-NMR window may be invisible because it is hidden behind other windows. If at least part of its border is still accessible, click on the border to get the entire window into foreground. Otherwise start iconifying other windows until XWIN-NMR appears.

Open a pull down menu; Command execution
Click onto the name of the desired menu in the top menu bar. The menu remains open until you click on a command, which is then executed. Clicking outside the

menu on an insensitve window area will close the menu without action. You can also open a menu by pressing the left mouse button when the cursor points to a menu name. Now keeping the button depressed, you can move the mouse to point to a command, which will be executed on button release. If you release the button when the cursor is outside the menu, no action is performed, and the menu is closed.

Tearing off a pull down menu

One or several pull down menus may be *torn off* the top menu bar and placed at an arbitrary position on the screen, where it remains open until closed explicitely. This feature gives you quick access to frequently used commands. Move the cursor to the dashed line at the top of each menu. Press the *middle* mouse button, and drag the menu while holding down the button. After releasing it, the menu will be redrawn with a window frame that includes the menu's name. You can now reposition the menu like any other window. Tear off menus are not supported on AspectStations, and may cause problems with some window managers.

Changing the window color

The color of the main XWIN-NMR window is controlled by the contents of the textfile

/u/prog/<XWIN-NMR version>/app-defaults/XWinNmr.

This file has 4 entries,

```
XwinNmr*foreground              black
XwinNmr*background              #a3a3a3
XwinNmr*XmText.background       #b98e8e
XwinNmr*XmTextField.background  #b98e8e
```

describing the colors for the menu text, the menus, and the backround colors for the data set bar and the command line, respectively. A 6-digit color code starting with a # character is given in RGB format. For example, *a3a3a3* gives the same intensity *a3* to the RGB components, resulting in some sort of grey. In order to find out which color names are legal, start XWIN-NMR and enter the command setres. A list of colors will be shown. Double-click on a color to see how it looks and feels. You may also mix your own color in setres using the sliders. The resulting color is displayed with a leading # character with 4 digits per Red, Green, or Blue component. Leave the last 2 digits of each component off, thus creating a 6-digit color code which you may use in the reource file.

If you change the colors in the *XWinNmr* file, it will change for any user starting XWIN-NMR. If you rename the file, let's say to *XWinNmr.save*, the window color will change to the standard Motif blue. If you want to give each user the permission to set her (his) own colors, proceed as follows: Insert the 4 lines in the file *.Xresources* in the user's home directory (create this file if it does not exist), and change the color there. The entries in the *.Xresources* file will get priority over those in the *XWinNmr* file.

The *XWinNmr* file only influences the main XWIN-NMR window, not the data objects such as spectra, axes, integrals, etc. Their color may be adjusted from the setres window while XWIN-NMR is running. Furthermore, some XWIN-NMR commands such as pulsdisp, acbdisp, view, search, etc. create own windows whose colors may be adjusted independently. All these commands have own resource files in the *app-defaults* directory.

Chapter 2

1D Data Processing Tutorial

This chapter presents 1D data processing based on the *exam1d* stored in the directory *XWINNMRHOME/data/guest/nmr/*. Please note that *XWINNMRHOME=/u* after a standard installation of XWIN-NMR. In order to have full access permissions to it, you should have a user *guest* installed on your system, and be logged in as *guest*. Start XWIN-NMR by typing xwinnmr -r. The *-r* option ensures that everything is cleaned up before starting the program, even if the last session was terminated by some problem. The program will start without displaying a data set. Instead, the XWIN-NMR logo will be shown in the data area. In later sessions, you can start XWIN-NMR without specifying the *-r* option, and you will immediately get the last used data set displayed.

2.1 Getting the data set

The first step is to get the 1D spectrum *exam1d* on screen. Open the *File* menu, select the *Open* entry, and from there the command dir. You should get a dialog box where the data sets *exam1d*, *exam2d*, and *exam3d* are listed. If there are other data sets in the directory */u/data/guest/nmr/*, they will also appear in the box. Click on *exam1d* in oder to instruct the program to make this data set the *current data set*. XWIN-NMR will automatically show the *1D layout*, because it identifies *exam1d* as a 1D data set. The identification is not performed via the name of the data set, but through the contents of the parameter file *meta* stored in the directory

/u/data/guest/nmr/exam1d/1/pdata/1/. The data are now displayed in the data area of the XWIN-NMR window. If transformed data are already present you will see the spectrum, otherwise the free induction decay. If no data at all were present, for instance after executing a delete command, the message 'No Data Available' is shown.

You will have noticed that the *File->Open* menu contains a number of different dir commands which all serve the purpose of locating and loading a data set. You can try them out, or find detailed information in the chapter *The File Menu*. Another important tool for getting an overview of available data sets is the command search in the *File Menu*.

Please note that most XWIN-NMR commands can be executed directly from the keyboard. The command names are displayed in brackets [] in the pull down menus, e.g. [dir]. They can also be used to build user defined *macros* and *automation (AU) programs*.

When a data set is visible on the screen, the top line of the data display area shows the data set name, in our example *<exam1d 1 1 u guest>*. This information reveals the directory in which data and parameter files are stored on disk, namely:

/u/data/guest/nmr/exam1d/1/pdata/1/.

The items *data*, *nmr*, and *pdata* are prescribed by XWIN-NMR and cannot be modified by the user. */u* is the initial path name of the data set's directory and therefore specifies, in which disk partition or on which disk the data set resides. The item *guest* is the login name for which the data were created, and *exam1d* is the data set name. You can learn more about data and parameter files in the description of the command New in the *File* menu.

2.2 Manipulating fid or spectrum on the screen

You should now have Figure 2.1 on the monitor, representing the standard 1D layout.

The data area shows the fid. Perhaps the data scaling on your display isn't exactly what you're looking for. The following pararagraphs describe how you can adjust the data according to your needs. If you want to look first at some acquisition parameters which show the *status* of the fid, for example its time domain size TD, or the number of scans NS which accumulated the fid, you can use the command dpa (*display acquisition status parameters*) in the menu *Output->Display status*

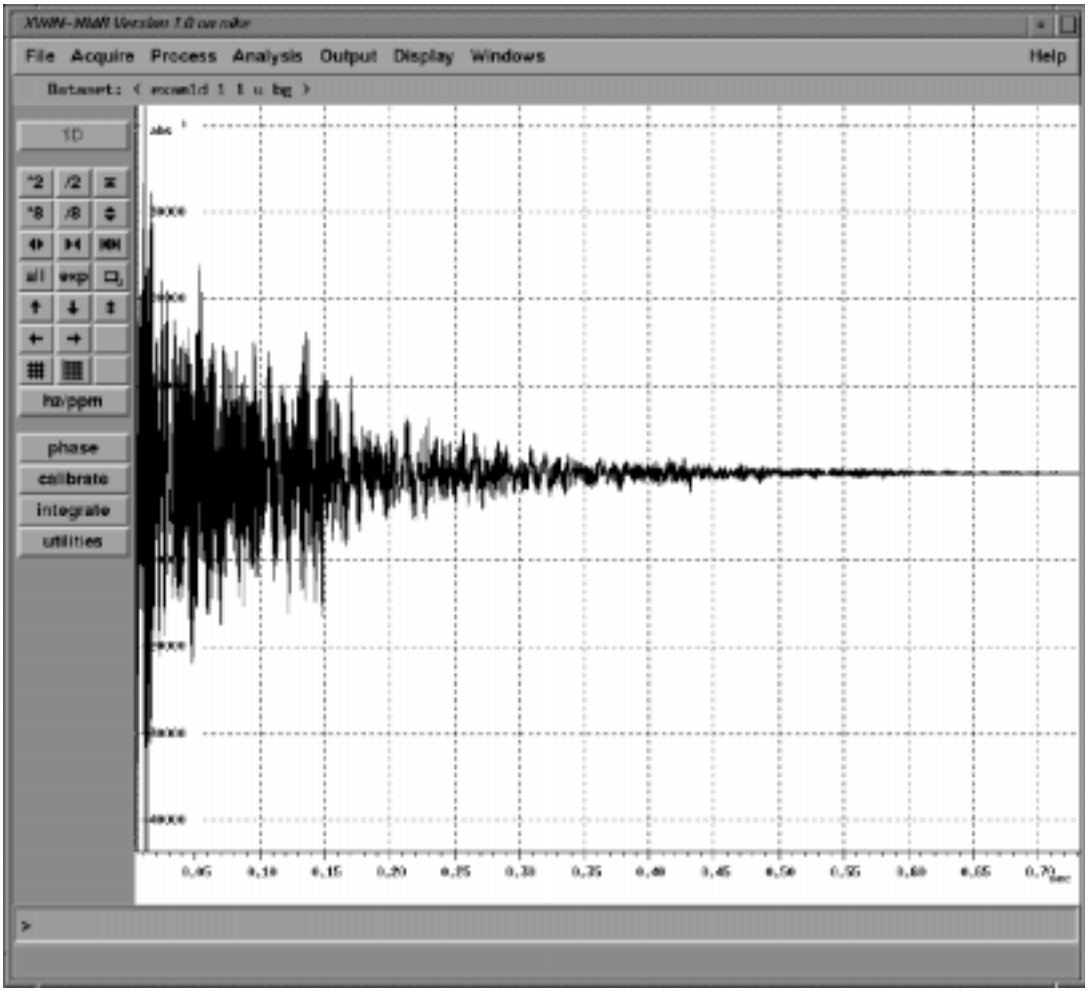


Figure 2.1 1D layout with fid. No transformed data available yet.

pars.

The button panel at the left side of the XWIN-NMR window allows you to interactively scale, shift, and expand the data. These operations are *screen operations* only, and do not modify the data file itself. You must click on a button to execute

the function. Some buttons require that you keep the left mouse button pressed while the cursor is located in the command button field, and then move the mouse. When you move the cursor over the command buttons (without clicking), their function is displayed in the status line at the bottom of the XWIN-NMR window.

1. Multiply (magnify) data by 2
2. Divide data by 2
3. Reset vertical scaling of the data so that the largest peak fits in the data area.

1. Multiply data by 8
2. Divide data by 8
3. Vertically scale data arbitrarily. Here you must keep the left mouse button pressed, and then move the mouse until the desired scaling is reached.

1. Expand (zoom) data by 2 from the center of the data area without changing the vertical scaling.
The following describes a method to expand an arbitrary area leaving the y scaling the same: Click the left mouse button while the cursor is located in the data area of the XWIN-NMR window. Now by moving the mouse, the cursor will move along the spectrum trace (and can be released from there by clicking the left button again). Clicking the middle button will mark the current position (the mark can be removed using the right button). Clicking the middle button a second time at a different cursor position will expand the area between the marked and the current cursor position.
2. Compress (un-zoom) data by 2
3. Reset zoom region to „full“ spectrum. The vertical scaling will remain unchanged.

1. Display complete spectrum, but do not change the zoom region
2. Display last zoomed (expanded) region
3. The buttons decribed so far allow you to independently scale the data on screen either horizontally or vertically. The purpose of this button is to draw a box around the spectrum part you want to zoom.
Click on the button. Position the cursor to the desired screen position. Now keep the left mouse button pressed while moving the mouse. A rectangle will be drawn until you release the mouse button. Then the rectangle will remain on screen, with small squares at the edges and in the center. The center square allows you to move the rectangle to another screen position, the other squares serve to resize the area of the rectangle. Again, in order to activate such a function, move the cursor into a square and move the mouse while its left button is pressed. Click the right button to zoom the region define by the rectangle.

1. Shift the zero line of the spectrum or fid to the center of the screen
2. Shift the zero line of the spectrum or fid to the bottom of the screen
3. Shift the zero line of the spectrum to an arbitrary screen position. Move the mouse while the left button is pressed, and the cursor is located within the command button.

1. Shift data left by 1/2 screen width
2. Shift data right by 1/2 screen width

1. Display a grid which generates a fixed screen partitioning
2. Display a grid bound to the coordinate axes. The grid lines change depending on the selected zoom area.

Toggle the x axis units between Hertz and ppm. If a fid is displayed, the axis unit are seconds from 0 to the acquisition time of the fid (given by the acquisition parameter AQ).

You can find additional display options in the menu *Display->Options.*

2.3 Setting the size of the real spectrum

The main processing steps to calculate a spectrum from the fid are window multiplication and fourier transformation. The most important parameter common to all processing commands is the size of the real spectrum, SI. You must set it before any processing begins.

1. Open the *Process* menu and select the command Real spectrum size [si]. Enter the desired value, followed by Return. This is the number of real data points your spectrum should finally contain. More convenient is to type (*for exam1d*) 32k, followed by Return. The unit *k* amounts to 1024 points. In most cases SI is set equal to TD, the time domain size or number of points of the fid. If TD doesn't happen to be a power of 2, SI should still be a power of 2, normally the next power of 2 larger than TD. Otherwise the fast fourier transform algorithm cannot be applied, resulting in a significantly increased processing time. Whenever SI is larger than TD, zero filling is applied: the fid is supplemented with SI minus TD zeroes.

2. Alternately, you can enter the command si on keyboard, followed by Return. It is also possible to type, e.g., si 32k (followed by Return). The SI parameter value is here specified as an argument.

3. A third method of setting SI is to call the processing parameter editor by typing in the command edp, or by opening the *Process* menu and selecting General parameter setup [edp]. A dialog box is displayed containing all processing parameters. You can locate SI, enter the desired value, and exit from the dialog

window. More details can be found in the description of edp. It should be mentioned here that the contents and layout of the edp dialog box can be tailored by the user by editing the so-called *format file*
/u/exp/stan/nmr/form/proc.e .

2.4 Applying a window function to the fid

The widest used window function is the multiplication of the fid with an exponential weighting function, which forces the end of the fid toward zero. It leads to some line broadening as well as to an improvement in the signal-to-noise ratio. There are several ways in XWIN-NMR to apply such a function.

1. Open the *Process* menu and activate the command Enter line broadening factor [lb]. Type in the desired value in Hertz (2.0 in our *exam1d* data set). Open the *Process* menu again and select the command Exponential multiply [em]. The program will apply the function to the fid and then display the result.
Please note: The program does *not* overwrite the measured fid with the exponentially multiplied values. Instead, it creates new files containing the result. This behaviour is actually true for all processing commands of XWIN-NMR, which will never destroy your acquisition data. The fid is stored in the file
/u/data/guest/nmr/exam1d/1/fid,
while the processing result ist stored in the files
/u/data/guest/nmr/exam1d/1/pdata/1/1r,
and
/u/data/guest/nmr/exam1d/1/pdata/1/1i.
pdata is the *processed data subdirectory,* the files *1r* and *1i* contain the even and odd data points, after the fourier transform the real and imaginary parts of the data.

2. Instead of calling the commands from the *Process* menu, you can enter them directly on the keyboard: type lb, followed by Return, enter the line broadening value, followed by Return, and then type em to execute the function. You can also specify the LB value on the command line: type lb 0.3, followed by Return.

3. The line broadening factor is a processing parameter called LB. You can enter all XWIN-NMR parameters directly from the keyboard by typing their name in lower case characters. You can also view (and change) all processing parameters at once by calling the processing parameter editor edp.

4. When you examine the *Process* menu you will find that XWIN-NMR provides a number of additional window functions such as *Gaussian*, but you can also specify your own *user defined* window functions.
5. XWIN-NMR allows you to adjust the parameters of window functions interactively with the mouse, and view the effect on the fid and the transformed spectrum simultaneously. Open the *Process* menu, and activate the command Manual window adjust. The screen layout will change to that of Figure 2.2:

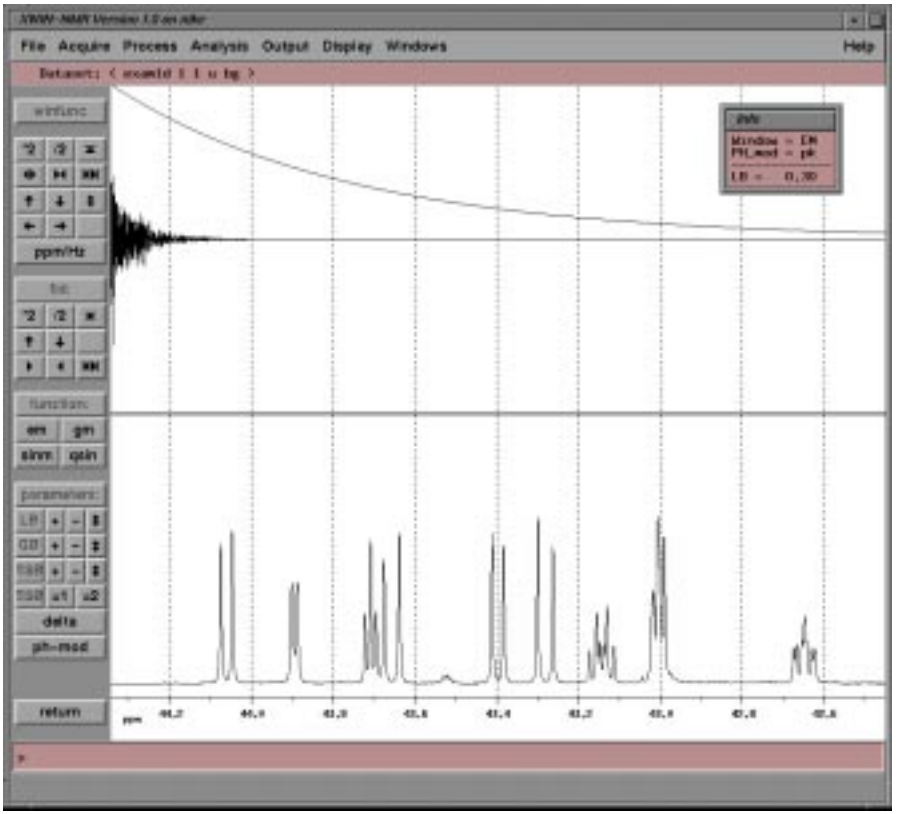


Figure 2.2 Interactive adjustment of window function

The upper part shows the fid and the window function, the lower part the transformed spectrum. The latter can be the unphased (*no*) or phased (*pk*) spectrum, the

magnitude (*mc*) or the power spectrum (*ps*). Select your preferred mode from the phase mode button *ph-mod* at the lower left. *pk* mode uses the processing parameters PHC0 and PHC1 for phasing, which requires that a phase correction has already been performed. In the *exam1d* data set, correct phase parameters are already present, so you can set *pk* mode.

The command button area at the left side is divided into several sections. The upper section provides the usual buttons manipulating the spectrum. The section *fid:* provides similar functions for the fid and the window function curve. The section *function:* allows you to select the desired window function. When you activate one of those buttons, a small *Info* window at the upper right will show the chosen function, and the parameter(s) required by this function: exponential multiply (*em*) requires LB to be adjusted, Gaussian multiply (*gm*) requires LB and GB, sine multiplication (*sin*) and squared sine multiplication (*qsin*) require SSB (the phase of the sine).

The last section *parameters:* allows you to adjust these parameters, thereby executing the window multiplication and the fourier transform in the chosen phase mode. Clicking the '+' button will increment the corresponding parameter by the value selected via the button *delta*, and redisplay the new window function curve, the fid, and the spectrum. It is recommended to zoom into the spectrum and to observe a particular peak or peak group to see the effect of line broadening or signal-to-noise improvement in detail. Clicking the '-' button will decrement the corresponding parameter analogously. You can also continuously increase or decrease a parameter by moving the mouse forwards or backwards. Set the cursor onto the up-down arrow command button rightmost to the '-' button, and move the mouse while the left mouse button is depressed. You will see window function, fid, and spectrum changing on line if the time domain and transform sizes are not too big.

Two special settings for SSB (=1 and =2) are provided. They result in half a period of a sine wave or a quarter of a period of a cosine wave.

In order to terminate the window function display mode, click the *return* button at the bottom. There are four possible options: *Cancel* leaves you in the window function mode. *Store & return* will force the program to exit from window function mode, and store the adjusted parameter in the processing parameter file. This means you can the examine it by typing the parameter name on the keyboard, or via the parameter editor edp. Also note that the next execution of the window function command will use this parameter value. *Store2D & return* works similarly. However, it does not store the parameter with the data set on screen, but in the

parameter file of the last used 2D data set. This operation is only useful if the current fid is a row or a column of a 2D fid. Finally, *return* will terminate window function mode without saving the adjusted parameter. The original value in the parameter file remains unmodified.

2.5 Applying the Fourier Transform

The next processing step after window multiplication is the fourier transform. The result of the fourier transform is the real spectrum, used by peak picking, integration etc., and the imaginary spectrum which is only needed for phase correction and can be deleted afterwards if disk space is to be saved (see deli command).

1. Open the *Process* menu and select the command Fourier tranform [ft]. The calculation is executed and the resulting unphased real spectrum is displayed (you can also view the imaginary part, or again the fid. Open the menu *Display->Options* for this purpose). What happened to the data files? We started off with the fid, stored in the file
   /u/data/guest/nmr/exam1d/1/fid.
   Window multiplication left this file unchanged, and stored the result in the files
   /u/data/guest/nmr/exam1d/1/pdata/1/1r,
   and
   /u/data/guest/nmr/exam1d/1/pdata/1/1i.
   The Fourier tranform operates on these files and replaces their contents with the transform result. *1r* now contains the real spectrum, *1i* the imaginary part.
2. Like most XWIN-NMR commands, you can initiate the transform also via keyboard: type ft, followed by Return.
3. Exponential multiply (em) and Fourier transform (ft) can be executed with a single keyboard command ef. Be sure that SI and LB have been set before.

2.6 Phase correction

After the Fourier transform the spectrum is not normally a pure absorption spectrum. XWIN-NMR provides an automatic and a manual way to correct zero (nonfrequeny dependent) and first order (proportional to the frequeny) phase distortions. Manual phase correction must be applied in cases where the automatic correction is not sufficient.

Automatic

Open the *Process* menu and execute the command Auto Phase correction [apk], or type apk on the keyboard. The corrected spectrum is displayed when apk terminates. The computed phase parameters PHC0 and PHC1 (in degrees) can be inspected (or modified) by typing the commands phc0 and phc1, or by calling the parameter editor edp. When an fid is transformed again, or acquired again under similar conditions and then transformed, the same phase parameters can be applied. In this case the command pk can be used, which uses the currently valid PHC0 and PHC1 values, but does not recalculate them like apk. The command apk0 only calculates PHC0, and uses the current setting of PHC1 to correct the first order angle. The command apks also performes an automatic phase correction, but uses a an algorithm different from apk.

Manual

Manual phase correction can also be invoked from the *Process* menu. It is more convenient, however, to click the command button *phase* at the left side of the XWIN-NMR window.

The upper part of the command button panel at the left side is identical to the standard mode. Further below, there are a few additional buttons required by manual phase correction.

Manual phase correction is executed in two steps: First phase the biggest peak or a peak of your choice first, the so-called reference peak for zero order correction. This defines the zero order phase PHC0. Position the cursor on the command button *PH0*, and keep the left mouse button depressed while moving the mouse. The current PHC0 value is displayed in the small *info* window at the upper right of the screen. When the zero order correction is done, use the *PH1* command button in the same way to phase another peak distant from the zero order reference peak. The first order phase is also shown in the *info* window.

When entering *phase* mode, the zero order reference peak is automatically set to the biggest peak in the spectrum. If you click on the button *biggest*, a vertical line indicates the peak position. If you prefer to use a different one, click on the button *cursor*. When you now move the cursor into the data area of the XWIN-NMR window, it will run along the spectrum curve. Position it on top of the desired peak, then press the middle mouse button to mark the position.

The *change phase increment* buttons **2, /2, DEF* allow you to modify the sensitivity of the mouse. The phase change during correction is doubled or cut in half,

thereby reducing or increasing the sensitivity. The *DEF* button resets the increment to its default value. The current phase increment value is shown in the *info* window.

In order to terminate phase mode, click the *return* button at the bottom. There are four possible options: *Cancel* leaves you in the phase mode. *Store & return* will force the program to exit from phase mode, store the adjusted parameters PHC0 and PHC1 in the processing parameter file, and overwrite the unphased spectrum with the phased one. *Store2D & return* works similarly. However, the parameters are also stored in the parameter file of the last displayed 2D spectrum, assuming that the current spectrum is a row or a column thereof. The 2D spectrum is now ready for execution of a 2D phase correction in the corresponding dimension. Finally, *return* terminates phase mode without storing the phase parameters, and without writing the phased data back to the spectrum file.

2.7 Spectrum calibration

The x axis of the spectrum is displayed in either Hertz or ppm units. If the complete spectrum is visible on screen, the length of the axis corresponds to the sweep width in Hertz or in ppm, given by the acquisition parameters SWH or SW, respectively. You can examine them by entering the command dpa on the keyboard, which gives a display of the acquisition status parameters. Defining the origin of the axis is a procedure called *spectrum calibration*. In XWIN-NMR, there a two possible methods: manual calibration and automatic calibration with the command sref. Both can be called from the *Analysis* menu. sref can also be typed on the keyboard. Its major application is the inclusion in automation (AU) programs for autocalibration of data measured with a sample changer. sref will only work if the files *2Hlock* or *19Flock* are installed in the directory */u/conf/instr/<instrument>/* for the spectrometer on which your data were measured. You can find more details in the chapter *The Analysis Menu* under sref.

In order to manually calibrate the spectrum, click the *calibrate* button located below the *phase* button. When you move the cursor into the XWIN-NMR data area, it will automatically move along the spectrum trace. Put it on the desired reference peak, and click the middle mouse button. A dialog will invite you to enter the frequency you want to assign to this data point (usually 0 for TMS). The units of the number to be entered are the same as the current x axis units (Hz or ppm). You can release the cursor from the spectrum trace by clicking the left (instead of the mid-

dle) button.

2.8 Baseline Correction

Certain spectra may show a baseline distortion, i.e. the noise in the signal free portions of the spectrum is not scattered around the zero line, but around a polynomial or some other curve. If this effect is large enough to prevent accurate integration of the spectrum, the baseline correction routines of XWIN-NMR should be applied.

XWIN-NMR provides automatic and user interactive baseline correction routines, and a semi-automatic mode using spline functions.

2.8.1 Automatic baseline correction [abs, absf, absd]

These commands assume that the baseline distortion can be removed by subtracting a polynomial from the spectrum. They try to find the suitable polynomial automatically. The only parameter you have to submit is the degree of the polynomial, ABSG. Type in the command absg, or set this parameter via the processing parameter editor edp. 5 is usually a reasonable value, the allowed maximum is 6. If you want to correct the entire spectrum, enter the command abs on the keyboard, or select it from the menu *Process->Special processing->Baseline correction*. There are cases where the baseline distortion cannot be be described by a polynomial over the whole spectrum, and it is more appropriate to apply the correction only to a particular region. The region can be specified by setting the parameter ABSF1 to its left limit (in ppm), and ABSF2 to its right limit. The command absf will execute the correction between the specified limits. absd is an alternate command to abs. It uses a slightly different algorithm to detect the baseline, and may be of advantage in special cases.

Please note that abs does not only perform automatic baseline correction, but also automatic detection of the spectral regions containing signals as required by automatic integration. Before terminating, abs stores the detected regions in a file *intrng*. For this reason, abs can also be called from the *Analysis* menu. You will find more details in the section *Spectrum integration.*

2.8.2 Manual baseline correction and bcm

Manual baseline correction is invoked from the menu *Process->Special processing->Baseline correction.*. The XWIN-NMR button layout will change as shown in

Figure 2.3.

Figure 2.3 Interactive baseline correction buttons

This window lets you select a polynomial, sine, or exponential curve whose coefficients A, B, C, D, E you can adjust with the mouse by putting the cursor on the corresponding command button and moving the mouse while its left button is pressed.

Curve	**equation**
polynomial	$A+Bx+Cx^2+Dx^3+Ex^4$
sine	$A+B\sin(Cx+D)$
exponential	$A+Be^{Cx}$

When you enter the baseline correction mode, the default function type is the polynomial. The mathematical equation is displayed in the *Info* window. The *sine* or *expon* buttons will bring up the other functions. The function trace is displayed simultaneously with the spectrum, and will initially be a zero line as all coeffi-

cients are zero.

Adjust the coefficients such that the displayed curve fits best to the baseline of the spectrum. Then click on the *diff* button to subtract the baseline curve from the spectrum. The program now displays the result. You may further adjust the coefficients, and the effect on the spectrum will be seen in real time. The button *undo* lets you return from the difference display, and the original spectrum with the calculated baseline reappear. The *reset* button resets all coefficient to zero, and lets you restart baseline adjustment.

You exit from baseline correction mode by clicking the *return* button or by openening the *File* menu. The program lets you select a normal *return*, leaving the spectrum file on disk unchanged, or a *save & return*, which applies the baseline correction to the spectrum file. If you have expanded the spectrum on the screen to display only a selected region, *save & return* will apply the correction only to that region. This procedure can generate discontinuities at the region edges, but the effect is not important if only that region is to be plotted or integrated. Correcting special regions in a spectrum is often more successful than correcting the entire spectrum at once.

XWIN-NMR stores the function type and the coefficients in the text file *base_info* whose structure is described in the chapter *The File Menu* under the command edmisc. You can copy this file from the current data directory to a save directory using the command wmisc, and read it back from there to the data directory of another data set with rmisc. Both commands can be called from the *File-> Copy* menu, and are described in greater detail in the chapter *The File Menu*. The processing command bcm then allows you to apply exactly the same correction to the other spectrum. You can type in bcm, or call it from the *Process->Special processing->Baseline correction* menu.

2.8.3 Baseline correction using splines [sab]

The automatic and manual baseline flattening methods described so far try to fit a poynomial or other curve to the overall baseline of the entire spectrum or a selected region. The success of this method depends on whether the shape of the baseline can be approximated by such a function. In contrast, the cubic spline interpolation method provided by the command sab fits the overall baseline piecewise by determining a different set of coefficients between each pair of selected points. The necessary condition to achieve a fit requires that the overall curve be smooth and continuous. Thus, weirdly shaped baselines can be approximated quite

accurately. The name (cubic spline) is derived from a thin, flexible device used by draftsmen to draw curves.

Before you can execute the command sab, you must define the baseline in the following way.

Select the display region on which you want to execute the spline baseline correction (any desired region or the whole spectrum). Enter *manual* baseline correction mode from the *Process->Special processing->Baseline correction* menu. Click on the button *def-pts*. Answer the questions allowing you to append new points to an already existing file of baseline points, or overwrite this file. The cursor will then be bound to the spectrum trace until you click the left mouse button. Move the cursor along the spectrum, starting at the left edge (although, if you want, you can select the points in any order). Select carefully points on the spectrum you consider as true baseline points. Click the middle button to store the point corresponding to the current cursor position in the file *baslpnts*. Selected points will be marked by an arrow. Proceed to the right until the file *baslpnts* contains enough points to describe the baseline adequately. Up to 200 points are legal. When you click the left mouse button, point selection is terminated and the *baslpnts* file is closed. You can later append more points by clicking *def-pts* again.

Now you can execute the command sab (type it in, or select it from the *Process->Special processing->Baseline correction* menu). It calculates the spline from the stored baseline points and subtracts it from the spectrum. Only the region between the leftmost and rightmost baseline point stored in the *baslpnts* file is processed.

The *baslpnts* file is stored in the current data directory of the processed data, where the spectrum file is also located. Please refer to the command edmisc for its structure. You can copy it using the commands wmisc and rmisc from the *File->Copy* menu, and use it for similar spectra.

2.9 Spectrum integration

This section presents the standard integration procedures. In the chapter *The Analysis Menu* we will explain the command nmrquant, which was designed for the quantitative analysis of spectra comprising complex overlapping signals.

XWIN-NMR provides two modes of integration: automatic and manual.

2.9.1 Automatic integration [abs]

Select this command from the *Analysis* menu, or enter abs on the keyboard. abs first executes a baseline correction of the spectrum by fitting a polynomial (see previous section *Automatic baseline correction*). Then it searches for spectral regions containing signals (integration regions) and stores them in the file *intrng* in the same directory where the spectrum file is located (see command edmisc in the *File->Open* menu for the description of the structure of this file). If you want to get the integral traces and values displayed, you must enter *Manual integration mode* (see page 26), open the *File* menu, and execute the command Read 'intrng'. In order to print the integral values, refer to the command li below.

abs always applies the polynomial baseline correction to your spectrum. If you are only interested in the found integral regions, but you want to keep the original spectrum, you must re-process the fid by typing efp (equivalent to the command sequence em, ft, pk) after execution of abs. This will restore the non-baseline corrected spectrum, but keep the *intrng* file generated by abs. A variant of abs is absf, which is identical to abs, but does not operate on the entire spectrum. Instead, it only looks at the spectral region defined by the processing parameters ABSF1 (low field limit) and ABSF2 (high field limit).

There now follows a description of how you can influence the way abs defines the integration limits. You can modify the parameters involved by entering their name on the keyboard (in lower case characters), or by calling the processing parameter editor edp.

Controlling the separation of integrals. Assume your spectrum contains multiplets, and you want to have a single integral drawn over the entire multiplet, not integrating the individual lines separately. The automatic routine would however recognize individual regions when the peaks in the multiplet come down to the baseline. In order to force the program to join the individual regions to a single one, you can set the processing parameter AZFW to a desired ppm value. If two integrals are farther apart from each other than this value, they will stay individual integrals. Otherwise, they will be combined to a single one.

Extending integral regions on either side. An integral region begins where a signal emerges from the noise level, and ends where the signal disappears in noise. These limits are stored in the *intrng* file as one region. If you want to let the region start a little earlier at its left side, and to let it end a little later at its right, you must specify the desired amount via the processing parameter AZFE (in ppm, default

value = 0.1ppm). There is, of course, the danger that due to the extension the region will now overlap with the previous or next one. In such a situation the center point between the overlapping regions will become the region limit for both.

Changing the signal detection threshold. An integral region begins where a signal emerges from the noise level, and ends where the signal disappears in noise. The noise level is calculated as the product (ABSL*standard deviation). ABSL is a processing parameter which you can increase to make integration less sensitive. The standard deviation is calculated by the program from the data. The default value of ABSL is 3.

Discarding small integrals. Automatic integration does not store those integral regions in the file *intrng* with integral values too small to be of interest. An integral is rejected, if it is ISEN times smaller than the largest integral in the whole spectrum. ISEN is a processing parameter. Its default value is 128.

2.9.2 Manual integration

This command allows you to define the integration regions manually. Select the command Manual integration from the *Analysis* menu, or simply click on the *integrate* button of the button panel at the left side of the XWIN-NMR window. The screen will change to the layout presented in Figure 2.4.

The upper part of the button panel is identical to the standard layout, and allows you to shift and scale the data on screen. In addition, there are three special sections headed by *current:*, *all:*, and *mouse:*. The command buttons in these sections work on the *current* integral marked by the user, on *all* integrals on the screen, and on the *mouse* sensitivity, respectively. Before discussing them, we'll explain how to define integration regions.

Defining integration regions

Move the cursor into the data area of the XWIN-NMR window. Click the left mouse button. The cursor is now bound to the spectrum, and moves along the spectrum trace when you move the mouse. It can be released from there by clicking the left button again. Clicking the middle button will mark the current position (the mark can be removed using the right button). Clicking the middle button a second time at a different cursor position will define the area between the mark and the current cursor position as the integration region, and the corresponding integral trace is displayed along with the value of the area under the integral. This procedure can be continued for all desired regions. Click the left button to release the cursor from

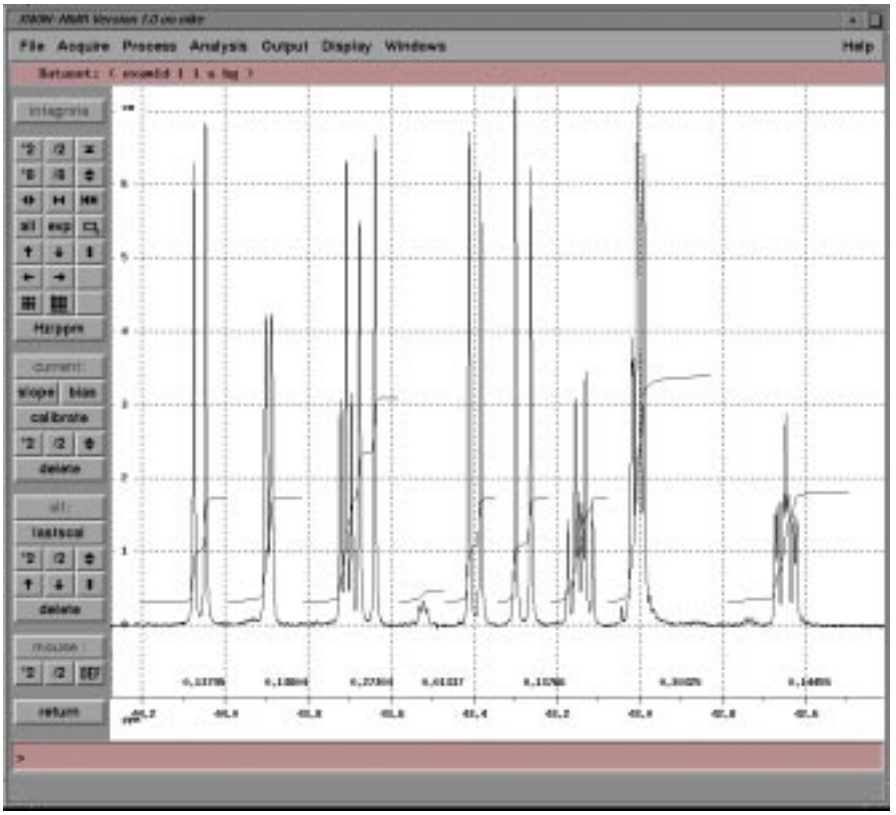


Figure 2.4 Manual integration

the spectrum when you are finished.

Storing integration regions on disk

When you have defined the desired integrals as described above, you can save the regions in the disk file *intrng*, which is stored with the current data set, in our example in the directory */u/data/guest/nmr/exam1d/1/pdata/1/*. Open the File menu and execute the command Save as 'intrng' or Save as 'intrng' & return. The latter one will terminate integration mode. The screen manipulations with integrals to be presented in the following paragraphs will not change the *intrng* file. You can easily restore the integrals on screen by executing the command Read 'intrng'

from the *File* menu. You can also save different sets of integration regions with the copy command wmisc described in the chapter *The File Menu.*, and retrieve them with rmisc.
A special storing option in the File menu is the command Save as 'reg'. The defined regions are now stored in a file called *reg* rather than *intrng*. It is only used by the command plotx, which plots the spectral regions contained in *reg*, thereby applying an automatic y scaling for each region to fit into the available space on the paper.

Defining and manipulating the current integral
In order to mark one of the defined integrals as *current* integral, move the cursor into the data area of the XWIN-NMR window. Click the left mouse button. The cursor is now bound to the spectrum, and moves along the spectrum trace when you move the mouse. Select the integral you want to make the current integral by moving the cursor under it, and release the cursor by clicking the left button again. The integral will be marked with an asterisk. All button panel commands in the section *current:* can now be applied to this (and only this) integral. The commands slope and bias perform a baseline correction of the region before integration. bias adds a constant to the spectrum while the cursor is located on this button, the left mouse button is kept depressed, and the mouse is moved. The result on the integral is a linear correction. You should use this command to correct the initial part of the integral. If, after that, the end part of the integral is still not horizontal, use the command slope analogously. It adds a linear function to the spectrum before integration. The calibrate command allows you to enter a number, which is assigned to the current integral. All other integrals are rescaled accordingly. The delete command removes the current integral from the screen (but not from the *intrng* file if the regions were already stored on disk), and you can redefine it if required. The *2 /2 etc. buttons allow you to expand the current integral on screen.

Manipulating and scaling all integrals
All button panel commands in the section *all:* operate on all defined integrals. delete removes all integrals from the screen if you want to start integration from scratch (but not from the *intrng* file if the regions were already stored on disk). The *2 /2 etc. buttons allow you to expand the integrals simultaneously, leaving the spectrum unchanged. The lastscal button enables you to scale the integral values relative to the previously integrated spectrum. Assume you want to compare the integrals of two spectra. Select the first (reference) spectrum via the *File->Open* menu, or use the search comand. Then proceed as follows:
- enter manual integration mode

- define and manipulate the regions as desired
- store them in the *intrng* file and exit from integration mode
- select the second spectrum
- enter integration mode
- read in the *intrng* file via the File-> Read 'intrng' command (or define new limits)
- click the *lastscal* button

This command puts the integral values of the current spectrum onto the same scale as the reference spectrum. Please note that comparing the integrals of different spectra is only meaningful if the data were acquired under the exactly same experimental conditions.

Changing the mouse sensitivity
The command buttons *2, /2, and DEF in the *mouse:* section of the button panel allow you to adjust the mouse sensitivity for the integral baseline correction (slope and bias) according to your needs. *2 doubles the change applied to slope or bias when moving the mouse, /2 decreases the change by the factor 2, and DEF restores the default setting of the increment value.

Terminating integration mode
Click on the *return* button. A window appears from where you can select an immediate return, leaving the *intrng* file unchanged. The integrals on screen are not stored in this case, and are discarded. You can also take this return if you have stored the integrals earlier from the *File* menu. The other return option stores the defined integral regions automatically in the *intrng* file. The return commands can also be accessed via the *File* menu.

2.9.3 List integrals [li]

The command li (type it in or call it from the *Analysis* menu) will give you a printout of the integration limits and the integral values.The output is directed to the currently defined printer or to the screen, depending on the setting of the parameter CURPRIN. Call the *output device editor* by typing edo, and set CURPRIN to *$screen* if you want to see the integrals on the monitor. Otherwise click on the down-arrow button rightmost to CURPRIN and select the printer on which you want to get the listing. If no printer name is displayed, you or your system administrator must install one with the command cfpp (see menu *Output->Printer/plotter installation*). Printer names, when inserted in the parameter CURPRIN, must always be preceeded by a $ sign. If omitted, the name is interpreted as a file name,

and the list is stored under this name in the current data directory (where the spectrum is).

li reads the *intrng* file of the current data set, calculates the integrals of the regions contained, and prints the result. XWIN-NMR allows you to influence integral calculation by changing the processing parameters INTBC and INTSCL, and by modifying the *intrng* file itself. INTBC is only taken into account if the *intrng* file was generated by abs, and not by manual integration. If set to *yes*, each integral will be linearly baseline corrected before it is printed. If set to *no*, this step is skipped. *intrng* files created by manual intgration contain slope and bias baseline correction values adjusted by the user (see command edmisc for the exact file structure of intrng). The printed integral values are presented in arbitrary units. In manual integration, you can calibrate a selected integral, the other ones are scaled in relation to this reference. In automatic integration you can insert scaling factors in the *intrng* file using a text editor (see command edmisc). The parameter INTSCL plays a role when you want to compare the integrals of different data sets. If set to -1, li scales the calculated integrals relative to the last data set integrated with INTSCL not equal to -1. As usual, you can enter INTBC and INTSCL at the keyboard (lower case) or from the edp editor.

All integration details described so far for li are also valid for the commands lipp, lippf (which list peak positions and integrals together, see *Analysis->Peak picking*), and the plotting commands when integral plotting is enabled.

2.10 Peak picking

Invoke the peak picking commands from the *Analysis->Peak picking* menu. Most of the commands can also be typed in. Please use the command names shown in brackets. In order to generate a peak list, the following steps are required.

Define spectral region

Peak picking can work on the entire spectrum or on a selected region. Expand the desired region on screen, or reset the horizontal scaling to display the entire spectrum. Open the *Analysis->Peak picking* menu and activate the command Define region. Since this command is also used to define the region for a plot, some questions will be asked. Just answer them with Return. The program stores the region limits in the plot parameters F1, F2 (in Hertz), and F1P, F2P (in ppm), and you can always examine (or modify) them by typing the parameter names in lower case characters. When you change the spectrum expansion or scaling on the screen,

these parameters will not be affected. You can always re-display the region via the command Show region in the *Analysis->Peak picking* menu.

Define intensity scale, thresholds, and sensitivity

The peak picking routine searches for signals larger than the so-called *minimum intensity* MI, and smaller than the *maximum intensity* MAXI. MI and MAXI are processing parameters that can be entered at the keyboard, or from the parameter editor edp. They can also be adjusted interactively with the mouse, as we shall see in a moment. Their units are centimeters (cm) for the following reason. XWIN-NMR facilitates the comparison of plotted spectrum and printed peak list by outputting the peak heights in centimeter units. The scaling reference peak (usually the highest peak in the spectrum when an overview spectrum is plotted) will be plotted with the height CY centimeters, and will also appear with this height in the peak list. You can define CY by typing in the comand in lower case characters or interactively.

In order to adjust CY, MI, and MAXI interactively, open the *Display->Options* menu and toggle the y axis and cm units on, or click the rightmost *grid* button on the panel. Then call the command Adjust min./maximum intensity from the *Analysis->Peak picking* menu, or alternately click the button *utilities* at the button panel. The button panel layout will change and show the command buttons CY, MI, and MAXI. Before you continue, examine the processing parameter PSCAL (enter its name in lower case characters) and set it to the value *global* if it does not have this value. *global* means that the largest peak of the entire spectrum will be assigned the height CY centimeters. Other values of PSCAL are not considered here, because they cannot be used to set CY interactively. Now click on the CY button. A horizontal line appears which you can move up and down the screen. Put it on top of the peak to which you want to assign a certain peak height. Then click the left mouse button and enter the desired peak height in cm units (the right mouse button allows you to exit without change). If you had selected the biggest peak and you examine CY afterwards, it will contain the value you typed in. If the peak was let's say 1/2 of the largest peak, CY will be set to twice the value you entered, etc. This routine is useful for adjusting CY so that your selected peak gets a well defined height in the peak listing and on the plot. Similarly, when you click on MI or MAXI, a horizontal line appears. Shift it to the desired position of the spectrum and click the left mouse button to set the parameter, or click the right button to quit.

All spectra are superimposed with noise, not only the signal free regions, but also

the peaks themselves. So even if you have chosen a minimum intensity threshold to discard very small signals, the peak picking algorithm will (depending on the data) detect signals in the remaining area which a user would reject, as he would consider them as fluctuations due to noise. An experienced user, however, could possibly suppose a peak under such a fluctuation. XWIN-NMR uses the following approach to influence the peacking sensitivity with respect to noise: The processing parameter PC (with default value 4.0) can be increased to make the peak search less sensitive, or decreased to make it more sensitive. For example, if a peak shoulder is not detected with the default value, you should enter a smaller number. The best sensitivity is achieved with PC=1.0, but of course many noise peaks could then be contained in the peak list. We recommend you work with the default value, unless particular data require a change.

One more parameter must be set before peak picking can be started. PSIGN=*pos.* will force the peak picking routine to search for positive peaks only, PSIGN=*neg.* for negative peaks only, and PSIGN=*both* for peaks with any sign. Type in PSIGN, or set it from edp.

Start peak picking

All peak picking commands can be called from from the *Analysis->Peak picking* menu, or typed in. pps displays the peak list on the screen, pp, pph, lipp, and lippf on the current printer CURPRIN (to be set with edo, refer to the command li earlier in this chapter, where we already described CURPRIN. Remember if CURPRIN=*$screen* the output will go to the screen). pp outputs the same list on the printer which pps displays in a window. pph works like pp, but for every peak it prints a number of asterisks at the rightmost column, derived from the intensity. With a single glance at this histogram you can find the biggest (or smallest) peaks without having to scan the intensity values. This is particularly useful for long lists. lipp combines the output of pp and li (integrals) in one listing, lippf always works on the full spectrum (remember that all other commands consider only the defined region).

As you can see from the following examples, signals are listed by peak number, address, frequency in Hertz, ppm and intensity. The address is the data point number in the spectrum file at which the peak is located. It is not an integer because the true maximum usually lies between two data points, and is found by the peak picking algorithm via parabolic interpolation.

Example of a list generated by pp:

```
DU=u, USER=guest, NAME=exam1d, EXPNO=1, PROCNO=1
F1=8.221ppm, F2=4.101ppm, MI=1.82cm, MAXI=31.86cm, PC=4.000
# ADDRESS            FREQUENCY           INTENSITY
                  [Hz]        [PPM]

 1 2799.1     3446.439       6.8946        24.83
 2 2851.1     3431.485       6.8647         8.57
 3 2873.6     3424.995       6.8517        10.58
 4 3027.9     3380.557       6.7628         2.29
 5 3036.3     3378.163       6.7580         3.24
 6 3057.8     3371.963       6.7456        11.22
 7 3083.2     3364.655       6.7310        13.21
 8 3110.4     3356.826       6.7153         3.68
 9 7174.5     2186.748       4.3746         7.74
10 7223.4     2172.687       4.3465         8.38
11 7475.8     2100.023       4.2011         5.47
12 7498.4     2093.505       4.1881         5.49
```

Example of a list generated by pph:

```
DU=u, USER=guest, NAME=exam1d, EXPNO=1, PROCNO=1
F1=8.221ppm, F2=4.101ppm, MI=1.82cm, MAXI=31.86cm, PC=4.000
# ADDRESS            FREQUENCY     INTENSITY HISTOGRAM
                               [Hz]        [PPM]

 1 2799.1     3446.439       6.8946 16.83  ************
 2 2851.1     3431.485       6.8647  8.57  *******
 3 2873.6     3424.995       6.8517 10.58  *********
 4 3027.9     3380.557       6.7628  2.29  **
 5 3036.3     3378.163       6.7580  3.24  ***
 6 3057.8     3371.963       6.7456 11.22  *********
 7 3083.2     3364.655       6.7310 13.21  ************
 8 3110.4     3356.826       6.7153  3.68  ***
 9 7174.5     2186.748       4.3746  7.74  ******
10 7223.4     2172.687       4.3465  8.38  *******
11 7475.8     2100.023       4.2011  5.47  ****
12 7498.4     2093.505       4.1881  5.49  ****
```

Example of a list generated by lipp (with 4 integral regions defined)**:**

```
Current data set:
NAME = exam1d EXPNO = 1   PROCNO = 1
USER = guest DU = u

MI=1.82 cm, MAXI=31.86 cm, PC=4.000
F1=8.221 ppm,  F2=4.101 ppm

Peak-Nr         Frequency      Intensity    Integral
[Hz]          [ppm]    [cm]

1            3446.439 6.8946   24.83
2            3431.485 6.8647    8.57
3            3424.995 6.8517   10.58
Int-Reg:   1 6.9955   6.8020               0.567275
-----------------------------------------------------
4                          3380.557 6.7628    2.29
5                          3378.163 6.7580    3.24
6                          3371.963 6.7456   11.22
7                          3364.655 6.7310   13.21
8                          3356.826 6.7153    3.68
Int-Reg:   2               6.7876                    5.3370
              0.362723
-----------------------------------------------------
9                          2186.748 4.3746    7.74
10                         2172.687 4.3465    8.38
Int-Reg:   3               4.4487                    4.3104
0.110188
-----------------------------------------------------
11                         2100.023 4.2011    5.47
12                         2093.505 4.1881    5.49
Int-Reg:   4               4.2448                    4.1094
0.111234
-----------------------------------------------------
```

2.11 Plotting

For historical reasons, XWIN-NMR provides two different plot systems:

2.11.1 Xwinplot

Plotting is performed using the interactive What-You-See-Is-What-You-Get plot editor xwinplot. Enter this command, or invoke it from the *Windows* menu. xwinplot allows you to define the plot layout interactively on the screen and output it to almost any kind of printer, or into a file for inclusion in documents. For details please refer to the XWINPLOT manual. The command autoplot allows you to plot the current spectrum using an existing plot layout constructed earlier. The name of the layout must be defined via the parameter LAYOUT. Enter the command edo to set it. A number of example layouts is provided.

2.11.2 Parameter driven plot

This is the older XWIN-NMR plot system where a plot layout is not defined interactively like in xwinplot, but via a set of parameters (to be defined using the command edg).

Before you can output your spectrum, integrals, peak lists, and parameters on a laser printer, inkjet/thermo printer or pen plotter, you or your system administrator must have made known your plotting device to XWIN-NMR using the command cfpp (type it in or call it from the *Output* menu). This work has usually already been done after XWIN-NMR installation. cfpp also allows you to check which printers/plotters are currently known to XWIN-NMR. A detailed description of cfpp can be found in the Chapter *The Output Menu*. This menu in fact contains all details on plotting. The following is a recipee how to generate plots quickly.

XWIN-NMR lets you configure several plotters: More than one plotter may be connected to the computer, and different plots can be executed at the same time on them. This requires that for each data set to be plotted the device on which the output is to occur must be defined. XWIN-NMR provides the parameter CURPLOT for this purpose, which can be set from the output parameter editor edo (type in this command or call edo from the *Output->Setup* menu). When the edo dialog box is open, click on the downarrow rightmost of CURPLOT. You will see all plotters configured with cfpp. Select the desired one and exit from edo.

XWIN-NMR allows you to define the layout of a plot from the plot parameter editor edg (type in this command or call it from the *Output->Setup* menu). Details about the various layout parameters are described in the chapter *The Output Menu*. When you exit from the edg editor, the parameters are stored in the file *meta* in the current data directory (where also the spectrum file is located).

Instead of setting up all layout parameters using edg, it is easier to start with a predefined layout. The sample 1D data set *exam1d* contains a *meta* file representing a layout suitable for an HP Deskjet 550C (or another A4 size) printer. In order to be able to use this layout, you should have configured an *hpdj550c* using cfpp, even if your own printer or plotter is of different type (remember that cfpp allows you to have several plotters configured). Enter the edo command, set the parameter CURPLOT to this printer and exit from edo. Open the *Windows->Plot Preview* menu and click on the command view (you can also type view on the keyboard). A window will be opened which shows the current plot exactly the way it would be output by the printer if the command plot would have been given rather than view. Depending on the plot type, you will have to wait a few seconds until the plot is drawn. The previewer is a window which may remain open while you execute other XWIN-NMR commands. You should now close the previewer by clicking the *Quit* button, set CURPLOT to your own plotter and restart the previewer. You may now change any layout parameter using the commands edg or setti (set plot title), or by entering a plot parameter name such as cy directly on the keyboard. Clicking on the *Restart* button of the viewer window will redraw the spectrum, taking into account all parameter changes. As soon as the plot looks as you intended, you can print it by typing plot.

In edg, you may among many other parameters specify the spectral region to be plotted in Hertz or ppm (parameters F1, F2 or F1P, F2P). In addition, you may set these limits interactively according the current display limit from the menu *Output->Define/show plot region*.

XWIN-NMR is delivered with a number of parameter sets defining various NMR experiments, comprising acquisition, processing and plot parameters. These sets are available to the user after execution of the *expinstall* command with the option enabled to install the standard experiments. They contain suitable plot layouts which can also serve as a basis for any similar plot. In order to make such a layout available for your current data set, enter the command rpar (or call it from the *File ->Copy* menu). Select the desired experiment from the displayed table, then select the *plot* entry from the next table, and click the *Copy* button to copy the *meta* file

of the chosen experiment on top of the current *meta* file in your data directory. You can now use edg to view the layout parameters, and, after having defined the desired spectral region to be plotted, enter view to preview the plot result.

Chapter 3

2D Data Processing Tutorial

This chapter presents 2D data processing based on the data set *exam2d.* The data set is stored in the directory */u/data/guest/nmr/*. In order to have full access permissions to it, you should have a user *guest* installed on your system, and be logged in as *guest*. Start XWIN-NMR by typing xwinnmr -r. The *-r* option ensures that everything is cleaned up before starting the program, even if the last session was terminated by some problem. The program will start without displaying a data set. Instead, the XWIN-NMR logo will be shown in the data area. In later sessions, you can start XWIN-NMR without specifying the *-r* option, and you will immediately get the last used data set displayed.

3.1 Getting the data set

The first step is to get the 2D spectrum *exam2d* on screen. Open the *File* menu, select the *Open* entry, and from there the command dir. You should get a dialog box where the data sets *exam1d*, *exam2d*, and *exam3d* are listed. If there are other data sets in the directory */u/data/guest/nmr/*, they will also appear in the box. Click on *exam2d* in order to instruct the program to make this data set the *current data set*. XWIN-NMR will automatically show the *2D layout*, because it identifies *exam2d* as a 2D data set. The identification of the layout is not performed via the name of the data set, but through the contents of the parameter file *meta* stored in the directory */u/data/guest/nmr/exam2d/1/pdata/1/*. For space reasons only the

acquisition data of *exam2d* are delivered on the release tape. You must enter the command xfb to execute a 2D transform. After xfb is finished, the transformed spectrum will be visible on the screen.

3.2 Manipulating the spectrum on the screen

You should now have Figure 3.1 on the monitor, representing the standard 2D lay-

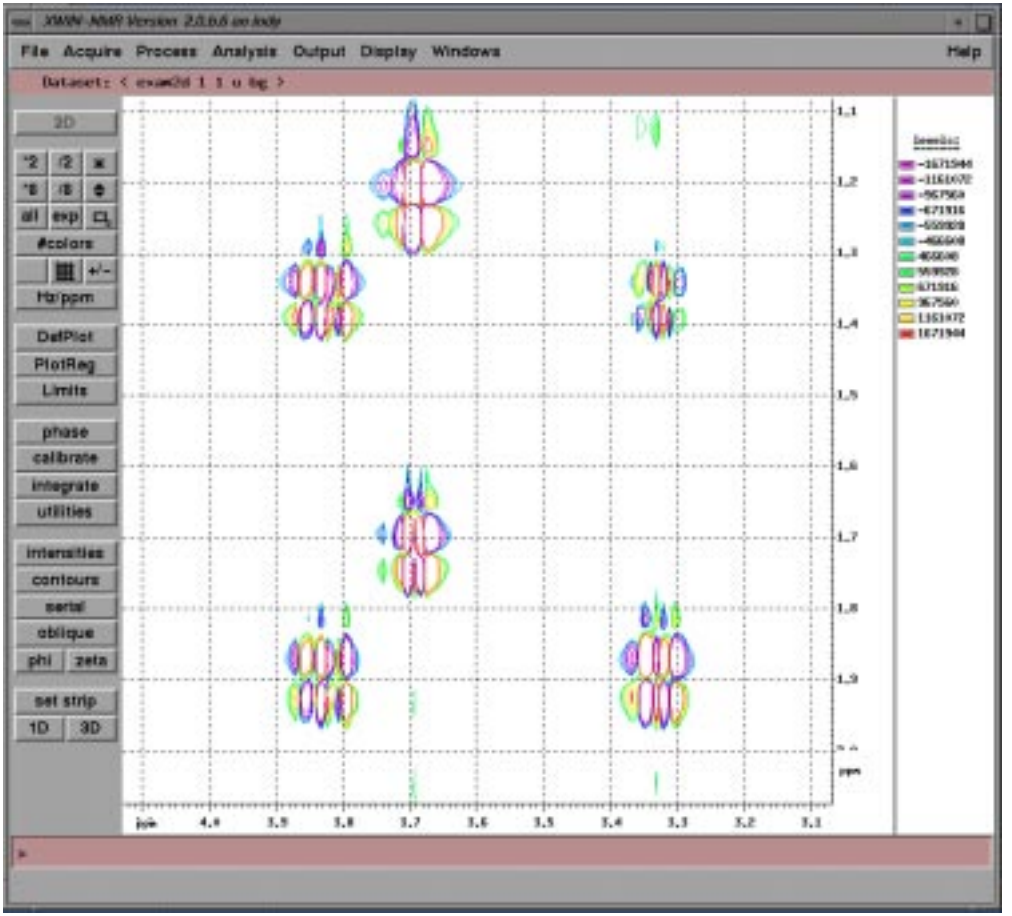


Figure 3.1 The 2D screen layout

out.The data area shows the 2D spectrum in intensity display mode. The right side indicates the numerical intensity values of the contour levels.

The button panel at the left side of the XWIN-NMR window allows you to interactively scale and expand the data, to enter various subroutines and to toggle the 2D spectrum display between contour mode, intensity display mode and oblique view mode. These operations are *screen operations* only, and do not modify the data file. You must click on a button to execute a function. Some operations require that you keep the left mouse button pressed while the cursor is located in the command button field, and then move the mouse. When you move the cursor over the command buttons (without clicking), their function is displayed in the status line at the bottom of the XWIN-NMR window.

1. Multiply (magnify) data by 2 (lower the cutoff level)
2. Divide data by 2 (raise the cutoff level)
3. Reset the vertical scaling of the data to the default value

1. Multiply data by 8 (lower the cutoff level)
2. Divide data by 8 (raise the cutoff level)
3. Vertically scale data arbitrarily (lower or raise the cutoff level depending on the direction of the mouse movement). You must keep the left mouse button pressed, and then move the mouse until the desired scaling is reached. This button is only active in *intensity* display mode, not in *contour* or *oblique view* mode. Scaling is achieved by changing the color lookup table. In contour or oblique view mode, the contours or the oblique projection must be recalculated. These processes are time consuming and cannot be performed in real time.

1. Display complete spectrum

2. Display last zoomed (expanded) region
3. The purpose of this button is to draw a box around the part of the spectrum that you want to zoom in on. Click on the button. Position the cursor to the desired screen position. Now keep the left mouse button pressed while moving the mouse. A rectangle will be drawn until you release the mouse button. Then the rectangle will remain on screen, with small squares at the edges and in the center. The center square allows you to move the rectangle to another screen position and the other squares serve to resize the area of the rectangle. In order to resize the rectangle function, move the cursor into a square and move the mouse while its left button is pressed. Click the right button to zoom the region defined by the rectangle.

The following describes a second method to expand an arbitrary area: Click the left mouse button while the cursor is located in the data area of the XWIN-NMR window. Position the cross hair to the upper left corner of the area to be expanded. Fix the position by clicking the middle mouse button. Position the second cross hair to the lower right corner of the region and click the middle button to fix it. Release the cursor from the spectral window by clicking the left button. Click on the rightmost icon as shown above to zoom the area.

1. No function assigned
2. Display a grid
3. Normally, when the spectrum contains negative peaks, both negative and positive intensity values (levels) are displayed. The standard colors assigned are *red-yellow-green* shadings for positive data, and *light blue-dark blue-violet* shadings for negative data. This button allows you to toggle through display modes where the negative or the positive data are turned off, or where the whole color range from red to violet is assigned to the positive data.

Toggle the x and y axis units between Hertz and ppm. If an axis unit is displayed in seconds (sec) rather than Hertz or ppm, the acquisition data were not fourier transformed in this dimension. For example, T1 experiments are acquired as a sequence of fids into a *ser* file, like a 2D experiment. However, they are only transformed

along one dimension using the command xf2. Use the *calibrate* button to define the origin of the axes (see section *Spectrum calibration*).

3.3 2D spectrum display modes

The command buttons *intensities, contours, serial,* and *oblique* provide different representations of the current 2D spectrum.

intensities
The intensities of the individual spectrum points are (logarithmically) mapped to the available color range.

contours
A contour map is calculated, and the diffrent contour levels are displayed using diffrent colors.

serial
Enters a special display mode where you can examine the 2D spectrum row by row or column by column, or where you can examine the 2D acquisition data fid by fid. This is essentially a 1D display mode.

oblique
An oblique view of the 2D spectrum is generated. You may change the viewing perspective by manipulating the azimuth and zenith angles phi and zeta: Put the cursor onto one of the respective buttons, depress the left mouse button and move the mouse until you get the desired angle. When you release the left mouse button, the new view will be calculated and displayed.

3.4 Setting up the parameters for a 2D transform

The main processing steps to calculate a spectrum from a 2-dimensional fid (*ser* file) are window multiplication and fourier transformation. In 2D (unlike 1D), window multiplication and fourier transform (and optionally linear prediction, phase correction or magnitude calculation) are part of the transform commands (xfb, xf2, xf1). The reason for this is the size of 2D data sets. Execution speed is considerably improved if the data need only be read once from disk and stored back after processing. However, all required parameters must be set up before the transform.

Call the processing parameter editor edp, either by typing in this command, or by

opening the *Process* menu and selecting General parameter setup [edp]. A dialog box is displayed containing all processing parameters in two colums for the two dimensions to be processed. Some parameters exist for either dimension, others only for one dimension. It should be mentioned here that the contents and layout of the edp dialog box can be tailored by the user by editing the so-called *format file* */u/exp/stan/nmr/form/proc.e* .

The most important parameters you must set in the edp window are SI, WDW, PH_mod (for *F1* and *F2*), and MC2 (for *F1*). *F2* is the dimension defined by the acquisition dimension of the 2D experiment and *F1* is the orthogonal dimension defined by number of fids acquired in a 2D experiment. The parameter SI specifies the number of real points the spectrum should have after the transform in the corresponding dimension. The parameter WDW allows you to select an appropriate window function. With PH_mod you can select whether to apply phase correction or not. For phase sensitive spectra, an extra step is usually required to find the correct phases. Phasing is then performed using extra commands after the transform. PH_mod also allows you to select magnitude or power spectrum calculation.

XWIN-NMR also allows you to modify the parameters by typing the parameter names (in lower case letters) on the keyboard. For example, the command si allows you to enter the SI parameter for the *F2* dimension and the command 1 si is used for the *F1* dimension. The command wdw allows you to enter the WDW parameter for the *F2* dimension, and the command 1 wdw is used for the *F1* dimension. The command lb allows you to enter the LB parameter for the *F2* dimension and the command 1 lb is used for the *F1* dimension, etc.

Please note: 2D processing commands do *not* overwrite the measured 2D fid with the processed values. Instead, they create new files containing the processed data. The 2D fid (in our example) is stored in the first file of Table 3.1, while the

/u/data/guest/nmr/exam2d/1/ser
/u/data/guest/nmr/exam2d/1/pdata/1/2rr
/u/data/guest/nmr/exam2d/1/pdata/1/2ri
/u/data/guest/nmr/exam2d/1/pdata/1/2ir
/u/data/guest/nmr/exam2d/1/pdata/1/2ii

Table 3.1 Acquisition and processed data files of *exam2d*

processing results are stored in the *pdata directory* which is the *processed data*

subdirectory. The file *2rr* contains the real data points after the fourier transform and the other files contain the imaginary data points. The files *2ri* and *2ir* are only present in the case of phase sensitive spectra.

3.5 Applying the 2D Fourier Tranform

Type xfb on the keyboard, or open the *Process* menu and click on 2D transform [xfb]. The tranform starts with the *F2* dimension by transforming the individual fids in the *ser* file of the acquisition data, and then continues with the *F1* dimension. The progress is reported on the status line at the bottom of the XWIN-NMR window. At the end of the transform, the software calculates a compressed spectrum corresponding to the display window size from the real part *2rr*, and stores it in the file *dsp*. This file is loaded to the display whenever this data set is selected as current data set, or when the display shows only an expansion of the spectrum and you click on the *all* button to display the full spectrum. When you expand a 2D spectrum on the screen as described earlier, the expansion is stored in the file *dsp.exp* in the current data directory. Both files, *dsp* and *dsp.exp*, are only needed for data display and, if missing, are recalculated automatically from the spectrum file *2rr* .

After completion of the xfb transformation, the 2D spectrum is corrected for a quad spike, i.e. the center of the *2rr* file is replaced by the median of its two neighboring points (in *F1*).

You can apply the transformations in F2 and F1 separately using the commands xf2 and xf1 (in this sequence). This corresponds to xfb, but *without* quad spike correction.

Some 1D experiments, such as T1 measurements, are acquired like 2D experiments. The result is a sequence of fids stored in a *ser* file, but need only be transformed in the *F2* dimension using the command xf2.

You will find all details about 2D transforms in the chapter *The Process Menu for 2D Data*. Particularly, options not yet mentioned such as strip transforms (whose result is not the entire spectrum, but only a desired frequency range), linear prediction (to get rid of artefacts introduced by non-decayed fids), inverse transforms etc. are discussed.

3.6 Strip and other transforms, and linear prediction

XWIN-NMR provides a number of 2D transform options such as linear prediction, strip transform, inverse transform, user defined transform, and Hilbert transform.

Linear prediction is an attractive option to perfect initial data points of the acquisition data before the transform in order to obtain better baselines. Furthermore, completing truncated Fids with linear prediction gives better estimates of the data and therefore allows for faster acquisition of multidimensional experiments. Linear prediction is executed as a part of the 2D transform, and is enabled by setting the processing parameter ME_mod to *LPfr* (forward) or *LPbr* (backward) prediction. The default value of ME_mod is *no*.

A *strip transform* is useful if you are only intereseted in a particular spectral region. It allows you to fourier transform only a selected portion of the spectrum defined by the processing parameters STSR (strip start, measured in points from the beginning of the spectrum) and STSI (size of region in points). The result is a file of significantly smaller size, which can be handled faster on the display and consumes less disk space. You are also able to set the transform size for the region (and therefore its resolution) to a larger value than it would be possible when tranforming the whole spectrum. You may also combine the strip transform with linear prediction. The region limits for the strip transform may be selected interactively. For this purpose, click on the *utilities* button. Move the cursor into the data area and click the left mouse button. Position the cross hair to the upper left corner of the desired region. Fix the position by clicking the middle mouse button. Position the second cross hair to the lower right corner of the region and click the middle button to fix it. Now move the cursor out of the data area and click on the *set strip* button. XWIN-NMR will set the parameters STSR, STSI to the selected region and ask you to enter the desired transform sizes.

The *Hilbert transform* allows you to generate the imaginary part of a spectrum from the real part. This enables you to delete the imaginary part of a 2D spectrum using the command deli to save disk space, and recreate it if desired (e.g. for an additional phase correction). You may even force the 2D transform not to store the imaginary parts at all using the command xfb n instead of xfb.

More details of these transform options are described in the chapter *The Process Menu for 2D Data*.

3.7 Phase correction

2D Manual phase correction can be invoked from the *Process* menu. It is more convenient, however, to click the command button *phase* at the left side of the XWIN-NMR window. A screen according to Figure 3.2 will appear. The upper part of the command button panel at the left side is identical to the standard mode. It applies to the 2D spectrum window. The lower part refers to the selected and moved rows or columns.

The idea behind manual phase correction of a 2D spectrum is as follows:
For the correction in the *F2* dimension, select a row of the 2D spectrum and move it to the first 1D window at the right part of the screen. Select one or two additional rows, if you want to look at several rows simultaneously while phasing, and move them to the 1D windows 2 and 3. Phase the rows just as with 1D spectra, and apply these phase parameters to the whole 2D spectrum. For the F1 dimension, use colums instead of rows.

We now present these steps in more detail:

1. Select a suitable region of the 2D spectrum. When you enter phase correction mode, the 2D spectrum displayed at the upper left part of the screen is shown in compressed form, since it is impossible to have a fully resolved spectrum in a small display area. As soon as you move the row to a 1D window, it is loaded from the data file and may therefore appear different from the compressed row in the 2D window. If you expand the 2D spectrum sufficiently before you select a row, then they should apper the same. The program informs you if the expansion is still too small: The Info box below the 2D spectrum says *Attention: 2D is compressed* in this case.

2. Select a row. Click on the row button. Position the horizontal line as desired and click the middle mouse button. The row at this position will be displayed. Click the left mouse button the leave row selection mode. You can scale the row using the **2* and */2* buttons below the row button, and you can increment or decrement the row number with the + and - buttons.

3. Move the row to a 1D window. Click on the *mov:1* button. The program will extract the selected row from the data file (which may take a few seconds) and

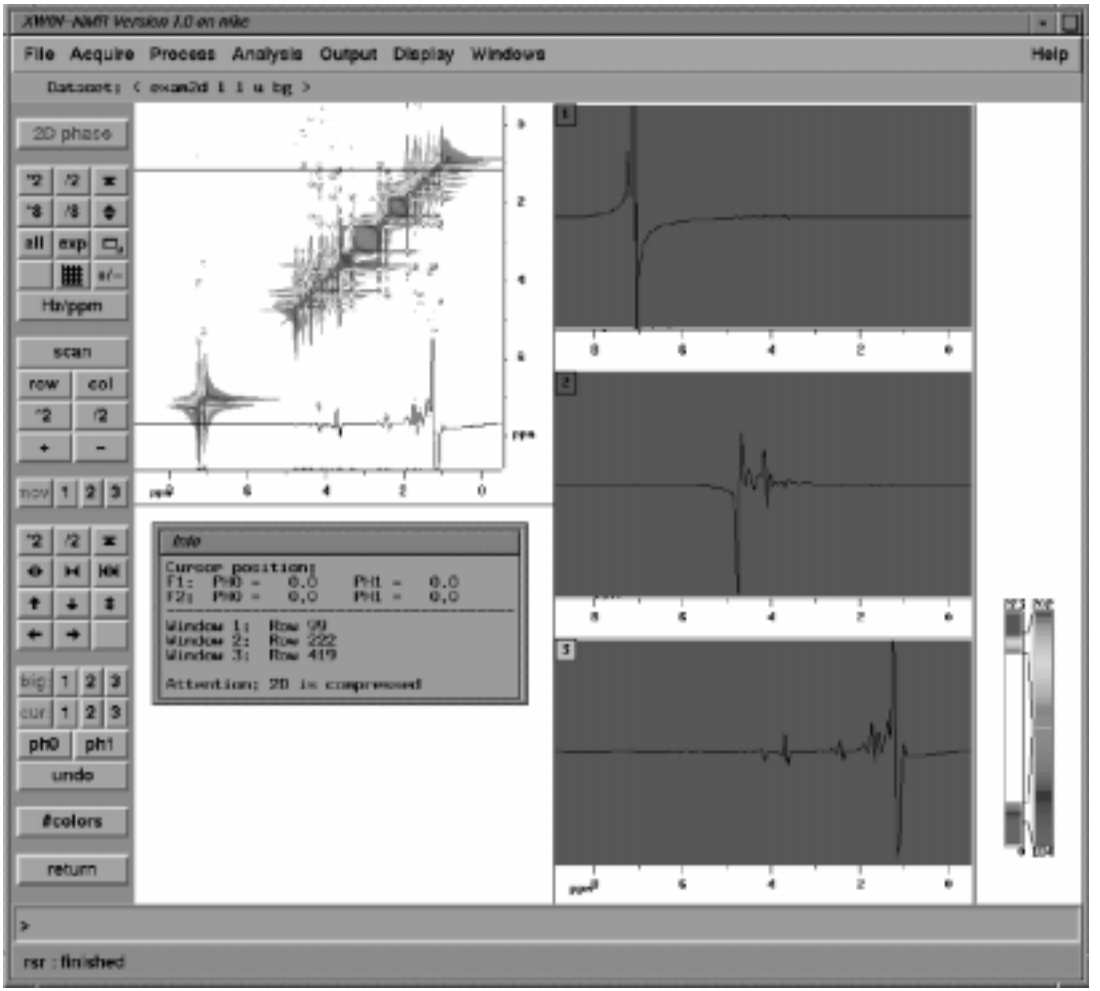


Figure 3.2 Interactive 2D phase correction

place it into the first 1D window.

4. Select one or two other rows. Put them into the second or third 1D window by clicking on the *mov: 2* or *mov: 3* buttons.
5. You may scale and expand the spectra in the 1D windows using the buttons

below the *mov:* button panel. They work analogously to the corresponding 1D spectrum operations described in the last chapter (1D tutorial), and influence the active 1D window. You can make one of the three 1D windows the active window by clicking on it. The window number of the currently active window is highlighted.

6. The *big* and *cur* button panels allow you to set the zero order phase reference point of the 1D window 1, 2, or 3 to the *biggest peak* or the *cursor* position.
7. The buttons *ph0* and *ph1* allow you to adjust the zero and first order phase of the spectrum contained in the active window. Position the cursor on the desired button, press the left mouse button and move the mouse. You can reset the values with the *undo* button.
8. Exit from 2D phasing using the *return* button. A pop up window allows you to return to the main 2D screen without storing the adjusted phases (choose *return*), or to store the phase values as parameters PHC0 and PHC1 in the processing parameter file of the current data set (you can then view them using the command edp). The program then asks you whether to apply the phase correction to the entire 2D spectrum. To phase all rows, the command xf2p is invoked. The corresponding command to phase all columns is xf1p. xfbp phases both, rows and columns and is identical to xf2p, followed by xf1p. You need not execute these commands right now. You can start them from the *Process->Phase* menu outside the manual phase correction screen at a later time using the stored constants.
9. In order to phase the spectrum in the F1 dimension, proceed in the same way, but select columns instead of rows using the *col* button.

3.8 Spectrum calibration

XWIN-NMR provides manual calibration and automatic calibration with the command sref. Both can be called from the *Analysis* menu. sref can also be typed on the keyboard. Its major application is the inclusion in automation (AU) programs for auto-calibration of data measured with a sample changer or with QUICKNMR. sref will only work if the files *2Hlock* or *19Flock* are installed in the directory

/u/conf/instr/<instrument>/

for the spectrometer on which your data were measured. The edlock menu must be set up properly for frequency locking. You can find more details in the chapter *The Analysis Menu,* under sref.

In order to manually calibrate the spectrum, click the *calibrate* button located below the *phase* button. A cross hair will be displayed. Move its center to the desired reference point, and click the middle mouse button. A dialog will invite you to enter the *F2* and *F1* dimension frequencies that you want to assign to this data point. The units of the numbers to be entered are the same as the current x and y axis units (Hz or ppm). You can release the cursor from the spectrum by clicking the left (instead of the middle) button.

Calibrating the F1 dimension can fail if certain parameters are not set correctly:

1. If only the nucleus is wrong, use the following commands to set it correctly:

2s NUC2 (Avance)
2s DECNUC (AMX etc.)

2. If only ND0 is wrong (which is typically the case if SW(F1) differs by a factor 0.5, 2, or 4 from the correct value), use the command 1s nd0 to correct it. Then, reprocess the spectrum with xfb. If new processing is not possible, correct the sweep width status parameter (by doubling it or cutting it in half) using the command *1s SW_p*. In most cases, re-calibration of the spectrum is necessary afterwards since these changes affect the parameter OFFSET.

3. If the frequencies are completely wrong, use the following commands to check the respective parameters and to correct them:

a) 1s SFO1
(if correct, note the value, otherwise type Return).

b) 1s BF1
(set the correct value if required).

c) 1s O1
(set the correct value. If SFO1 was correct, BF1+O1=SFO1 must be valid).

d) 1s ND0
(check and correct if necessary)

e) 1s SWH
(check and correct if necessary)

f) 1s SR
(if a 1D preparation experiment is available, take over the value from there, otherwise set to 0 for now).

3.9 Baseline Correction

3.9.1 Automatic baseline correction [abs1,abs2]

In the 1D tutorial chapter, we discussed the baseline correction commands abs and absf for 1D spectra which assume that a baseline distortion can be removed by subtracting a polynomial from the spectrum. The command abs1 uses the same algorithm to correct each column of a 2D spectrum, while abs2 operates on the rows. The commands can be typed in, or invoked from the *Process->Baseline correction* menu. As in 1D, the parameter ABSG determines the degree of the polynomial, and can be set independently for the two dimensions in edp. The processing parameters ABSF1 and ABSF2 determine the left and right limits of the region to be corrected and they can be entered from the parameter editor edp. For the F2 dimension, the limits can also be entered from the keyboard by typing absf1 or absf2. In contrast to the 1D command absf, the 2D commands do not generate integral range files.

3.9.2 Additional methods [bcm1,bcm2; sub1,sub2, abst1,abst2]

XWIN-NMR provides a number of other methods to correct the baseline of rows or columns in a 2D spectrum. bcm2 and bcm1 are generalizations of the 1D command bcm for all rows or columns. You can extract a row or column from a 2D spectrum, baseline-correct it manually as we have seen in the 1D tutorial, and then apply the correction function to all rows or columns. See also the description of bcm in the 1D tutorial.

The commands sub1, sub2, sub1d1, sub1d2, abst1,abst2 are described in the chapter *The 2D Process Menu*.

3.10 Symmetrization, Tilting

XWIN-NMR provides commands for these operations. If your data set needs to be processed with these routines, please refer to the chapter *The Process Menu for 2D Data*.

3.11 Spectrum integration

For 1D spectra, XWIN-NMR supports manual and automatic integration. 2D spectra can only be integrated manually. You must interactively define the integration regions, which can be stored in a text file of your choice. The 2D integration commands int2d or int2dref will compute the integrals and store the results in the file *int2d* in the current data directory (where the spectrum file *2rr* is located). It may be output to screen, printer, or a desired destination file with the li command.

Defining integration regions

Enter integration mode by clicking on the *integrate* button located at the left hand button panel, or call Manual integration from the *Analysis* menu. The program asks you to enter the name of a file which will later contain the frequency limits of the defined integration regions. The file will be stored in the directory

/u/exp/stan/nmr/lists/roi/.

The screen layout will change to the 2D integration mode. In order to define a region proceed as follows:

1) Move the cursor into the data window and click the left mouse button
2) Position the cross hair to the upper left corner of the region to be integrated
3) Click the middle button to fix the upper left corner, or the left button to quit
4) Move the second cross hair to the lower right corner of the region
5) Click the middle button to fix the lower right, or the right button to delete

The defined region is shown as a rectangle. Now click the left mouse button. A popup window will offer you the following choices:

a, +, -, a+-, a+, a-, +-

These codes control how integration of this region is to be performed by the int2d command. If 'a' is selected, all intensities in the area are added to give the integral. If '+' is selected, only the positive intensities added. If '-' is selected, only the negative intensities added. If 'a+' is selected, you will get two results from the int2d command: the first corresponds to option 'a', the second to option '+'. Similarly 'a-', '+-' and 'a+-' provide additional combinations. The a+ - option delivers three results. The chosen option is written aside the rectangle of a region and is stored in the region file along with the region boundaries.

The *clear* button

This button allows you to remove all integration regions from the screen. You will be asked whether to override the region file. If you answer *yes*, the file contents will also be cleared, and you will lose the regions defined so far. If you answer *no*,

the regions are only deleted from the screen, the file contents are retained. You can now define new regions, which will be appended to the file.

The *read* button
When you click on this button, the programs invites you to enter the name of a region file. It will read the region stored there, and display it. You can now define new regions, which will be appended to the file.

Computing the integrals [int2d, int2dref]
These commands may be typed in or called from the *Analysis* menu. Either command first invites you to enter the name of the file where the integration regions are stored. int2d then calculates the integrals immediately and stores the result in the text file *int2d* in the current data directory. int2dref, before proceeding, asks you to enter an integral region number and an integral value you want to assign to this region. It also stores the result in *int2d*. Both commands perform the same calculation. int2dref, in addition, generates an output column in which the reference region has the specified value assigned. The integrals of all other regions are scaled relative to this reference value.

You can use the same integral region file to integrate different 2D spectra. In a region file, integration regions are defined by the limiting row and column numbers as well as the corresponding ppm values. When integrating spectra of the same size (having the same SI parameters), the row and column numbers will be used as integration limits, otherwise the ppm values are taken.

Outputting integral values [li]
This is the same command name also used to print the integrals of 1D spectra. XWIN-NMR knows that you are currently working with a 2D data set and adapts itself accordingly. li (type it in or call it from the *Analysis* menu) will give you a printout of the integration limits and the integral values as stored by the int2d or int2dref command in the *int2d* file in the current data directory. The output is directed to the current printer or to the screen, depending on the setting of the parameter CURPRIN. Call the *output device editor* by typing edo, and set CURPRIN to *$screen* if you want to see the integrals on the monitor. Otherwise click on the down-arrow button rightmost to CURPRIN and select the printer on which you want to get the listing. If no printer name is displayed, you or your system administrator must install one with the command cfpp (see menu *Output->Printer/plotter installation*). Printer names, when inserted in the parameter CURPRIN, must always be preceeded by a $ sign. If omitted, the name is interpreted as a file name, and the list is stored under this name in the current data directory (where the spec-

trum is).

Terminating integration mode
Click on the *return* button or invoke the command Return from the *File* menu.

Example of a region file
The first line of the file contains two numbers, n and m. These are currently set to zero and reserved for future extensions. Each spectral region is given by two lines containing the rows and column parameters defining the region. Specifically, these are the spectrum size in this dimension, the row or column numbers and the corresponding ppm values. The codes 'a', '+', '-' indicate whether only positive, only negative or all data points should be used by int2d or int2dref to calculate the integral of this region.

```
0 0
a 1024 918 948 0.433316 0.156820
  1024 453 479 4.719352 4.480560

+ 1024 918 948 0.433316 0.156820
  1024 453 479 4.719352 4.480560

- 1024 918 948 0.433316 0.156820
  1024 453 479 4.719352 4.480560
```

Example of an *int2d* results file
The format of the results file is similar to the region file, except for the additional entries for the integral values. The following example shows the format of an *int2d* file generated by the command int2dref. The output format created by the command int2d would be same except that the column *norm. Int.* (normalized integrals) is missing.

```
Current data set:
Name =          exam2d ExpNo =    2   ProcNo =    1
User =          bg   Disk  = u
Baseline = 0    Noise = 0

#  SI_F1  row1  row2  row1(ppm)  row2(ppm)  Integral  norm. Int.  Mode
   SI_F2  col1  col2   col1(ppm)  col2(ppm)

1  1024   918   948   0.43332    0.15682        24.976      13.089      a
   1024   453   479   4.71935    4.48056
```

2	1024	918	948	0.43332	0.15682	38.163	20	+
	1024	453	479	4.71935	4.48056			
3	1024	918	948	0.43332	0.15682	-13.188	-6.9112	-
	1024	453	479	4.71935	4.48056			

3.12 Peak picking

The following setup is required:

1. Select a spectrum region and define it as plot region. Alternatively, use edg to define the desired region via the parameters F1PLO, F1PHI, F2PLO, F2PHI.
2. Set the parameters MI, MAXI, PSIGN, PC for the F2 dimension (cf. 1D peak picking for a detailed discussion).
3. Enter edo and define the output device by setting the parameter CURPRIN to *$screen*, to a printer name, or to a file name just as for 1D peak picking.
4. Enter the command pp2d or xau pp2d to start peak picking.

In AU programs, peak picking can be called with:

xcmd(“xau pp2d“)

pp2d first calculates the F1 projection of the specified region and executes a 1D peak picking on the projection. For each peak found, the corresponding F2 row is extracted from the spectrum, and 1D peak picking is applied.

The AU program pp2dmi is a variant of pp2d which calculates the parameter MI automatically. It computes the noise of the 2D spectrum and multiplies it with a number in order to obtain a meaningful intensity threshold. This number can be adapted to your needs by setting the constant *noise_fac* in the AU program to the desired value.

Additional 2D peak picking algorithms are available in Bruker’s AURELIA program.

3.13 Plotting

Basic plotting was already handled in the 1D tutorial. Like 1D plots, 2D plots may be obtained either using xwinplot or parameter driven plot. Please refer to the

XWIN-PLOT manual if you prefer the interactive plot editor.

This introduction to 2D plotting using the older parameter driven plot system assumes that you are familiar with 1D plotting. Like in 1D, you can set up all plot parameters from the edg plot parameter editor, which will now show the settings for 2D spectra. You can also preview 2D plots with view. The spectral region to be plotted and the contour levels are often set up interactively from the screen. We shall describe this in the following paragraph.

3.13.1 Defining the spectrum region and the contour levels to be plotted

1) Zoom the part of the spectrum you want to plot.
2) Adjust the contour levels as desired using the *2, /2,*8, /8 buttons. In intensity display mode (not in contour mode) you may also use the online adjust button.
2) Open the *Output->Define/show region/levels* menu
3) Select the command According to current screen limits

Before the program stores the expanded region as plot region, it asks you whether to change the contour levels. If your answer is *yes*, the levels to be plotted will be those currently visible on the display. You can also enter the number of levels to be plotted at this time. Switch to contour mode to see the intensity values of the levels at the right side of the screen. If your answer is *no*, only the spectrum region is stored, but the contour levels remain as defined earlier (i.e. the *level* file will not be changed).

In the same menu, you can alternatively specify the exact frequency limits of the region you want to plot by calling the command Enter region limits numerically.

The region limits are stored in the *meta* file. At any time you can inspect the region limits by looking at the parameters F1* and F2* using the plot parameter editor edg->EDCONTP. You can force the program at any time to display the plot region by invoking the command Show plot region from the menu *Output->Define/show region/levels.*

The contour levels to be plotted are stored in the file *level* in the current data directory. It is stored in a binary format with the internal structure of Table 3.2. Instead of defining the plot levels interactively as described above, you can define their numeric intensity values using the command edlev (type it in or call it from the menu *Output->Define/show region/levels*). A dialog window according will be opened, allowing the following operations:

Total number of contours to be plotted Number of negative levels Intensity of level 1 Intensity of level 2 ..... etc. ....

Table 3.2 Structure of *level* file

Changing a level
Move the cursor on to a value of a level, and click the left mouse button. Enter a new number.

Deleting a level
Click on a level number (left column of the table). The corresponding level will be deleted. You can delete all levels from a certain number onwards, by giving this number the value zero (right column).

Inserting a new level
Enter a value in the emtpy field at the end of the table (right column). The number is automatically sorted in, depending on its size.

Generate table of equidistant levels
Click on the INCR: command button. A number (the increment) can be entered. New values for all levels except for the smallest one are calculated. XWIN-NMR generates the new level values by successively adding the entered number to the previous level, starting with the smallest positive one. If the table also contains negative levels, subtraction is applied instead. The result is a table of equidistant levels.

Generate a table of levels with constant ratio
Click on the FACT: command button.When this command button is activated, a number can be entered that is used to generate new values for all levels except for the smallest one. XWIN-NMR generates the new levels values by successively multiplying the entered number with the previous level, starting with the smallest positive one. The result is a table of levels with constant ratio.

Accept all changes
Click on the SAVE command button. All changes in the table are stored in the *level* file, and edlev is terminated.

Discard all changes
Click on the QUIT command button. All changes in the table are discarded (i.e. the *level* file is not modified), and edlev is terminated.

3.13.2 Defining Projections

Often contour maps are plotted together with projections or 1D spectra. Enter edg and select EDPROJ1 or EDPROJ2 to open the parameter box for the projection to be drawn along the F1 or F2 axis. Click on PF1EXT (PF2EXT) to get the table of supported projection types. Choose *external* if you want to plot an arbitrary spectrum along this axis. In this case use the parameters P1(2)DU, P1(2)USER, etc. to specify the name and location of this spectrum.

3.13.3 Simulating a large size plotter

In XWIN-NMR, you can generate plots of arbitrary size and resolution even on a small size printer or plotter. Use the parameters CX1 and CX2 to define the desired plot size (in cm) in the F1 and F2 dimensions. Set the parameter CLIP to *no* (you can type in cx1, cx2, clip as commands, or set the parameters from edg). The plot command will generate as many pages as necessary. Marks will be drawn at the edges so that the different sheets may easily be joined together. Even the previewer command view can handle this mode. Click on the *Next Page* button in the previewer window to toggle through the pages.

3.13.4 Using the previewer

For 2D spectra, the view command may take some time (depending on the size of the plot region) before the spectrum is drawn in the previewer window. During this time, where contour calculation from the data file on disk takes place, the cursor shape changes to a watch.

3.14 Projections and Cross Sections

XWIN-NMR allows you to display rows and columns of a 2D spectrum and projections of the entire spectrum or parts of it *(partial projections)* onto the F1 or F2 axis. Furthermore, this data may be stored as 1D spectra and processed or plotted accordingly. All these routines are accessible from the *utilities* screen. If you click on the *utilities* button of the main 2D display, the command buttons will change

according to Figure 3.3.

Figure 3.3 The *utilities* button panel

You can enter the *utilities* screen either with the 2D spectrum shown in colored intensity mode or in contour display mode. Please select the desired mode before entering *utilities*.

Fast row/column scanning mode

Click on the *scan* button to enter this mode. The program will display a cross hair cursor you can move around with the mouse. Clicking the left mouse button will display the horizontal cross section (row) defined by the position of the horizontal line of the cross hairs. Clicking a second time will toggle to the vertical cross section (column) defined by the position of the vertical line of the cross hairs. Clicking a third time will show both, the row and the column. Clicking a fourth and then a fifth time will once again only display the row, etc. In any mode, you can move the mouse to change the position of the cross hairs, and the displayed rows or columns will follow in real time. You may also use the middle or right mouse button to magnify or reduce the size of the current trace. Exit from fast scanning mode by clicking the middle and right mouse button at the same time.

Please note that fast scanning mode operates on the displayed 2D data. The rows and columns are not read in from the original 2D spectrum file. For this reason the resolution of the displayed rows and columns might not be fine enough to show all details. You will have to expand the 2D spectrum before entering fast scanning mode to overcome this limitation.

Extracting rows or columns

Click on the *row* or *col* button. The program will display a cross hair cursor you can move around with the mouse. Clicking the middle mouse button will display the horizontal or vertical cross section (*row* or *col*) defined by the position of the horizontal or vertical line of the cross hairs. In this mode, you may move the cross hairs to a different place and click the middle button again to get the cross section of this position. Click the left mouse button to release the cross hair cursor. The following buttons are useful in this situation, where the 2D spectrum and a cross section are displayed together:

- Use the buttons **2* and */2* located below the *row* and *col* buttons to scale the cross section.
- Click on the *blank* toggle button to switch off the 2D spectrum temporarily, to make the cross section more visible. Another click on *blank* will redisplay the 2D spectrum. Another way of improving the visibility of a cross section is to increase the intensity threshold for the 2D spectrum.
- Use the buttons +and - located below the **2* and */2* buttons to display the next or previous cross section. These buttons allow you to adjust any particular row or column number.

- If you want to save the row or column as a 1D spectrum, click on the *return* button or open the *File* menu. Select one of the commands Save row/col as ~TEMP or Save row/col as PROCNO=... .

Projections
Click on the button labelled *p* to display the positive projection onto the corresponding axis (projection of positive intensity values only). Click on *n* to view the negative projection. Projections are always calculated over the entire spectrum, even if the current display only shows a region of it. The calculation takes place during the Fourier transformation and are stored in the current data directory. The files names are given in Table 3.3.

Axis	Projection type	File
F1	positive	*p2r1*
F1	negative	*n2r1*
F2	positive	*p2r2*
F2	negative	*n2r2*

Table 3.3 Files containing projections

You may vertically scale projections using the **2* and */2* buttons, located below the *row/col* buttons. Projections may be stored as 1D spectra like rows and columns: click on the *return* button or open the *File* menu. Select one of the commands Save row/col as ~TEMP or Save row/col as PROCNO=... .

The calculation of projections can also be initiated with the command proj (type it in or call it from the *Process->Calculate projections* menu).

External spectra
Click on ext to view an arbitrary (but suitable) external spectrum to be drawn along this axis. You must define this spectrum using the edg plot parameter editor. Enter edg and select EDPROJ1 or EDPROJ2 to open the parameter box for the spectrum to be drawn along the F1 or F2 axis. Click on PF1EXT (PF2EXT) to see the table of supported projection types. Choose *external*. Use the parameters P1(2)DU, P1(2)USER, etc. to specify the name and location of this spectrum. This is also the correct procedure to set up a 2D plot with external spectra as already described in the section about plotting. You may vertically scale the displayed spectrum using the **2* and */2* buttons, located below the *row/col* buttons.

Partial projections

A partial projection is not calculated over the whole spectrum, but only over a specified region. You must define the region and start the calculation of the partial projection before it can be displayed. Click on *calc* at the right side of the *part* button for the desired dimension. The program will display a cross hair cursor you can move around with the mouse. Clicking the middle mouse button will display and set the first region limit (this is a horizontal or vertical line depending on the chosen dimension). You may now move the cross hairs to a different place and click the middle button again to fix the second region limit. The program will then invite you to enter a PROCNO under which the partial projection is to be stored as a 1D spectrum. As soon as you have entered the number, the calculation will start, and then the result will be displayed. If you want to redisplay the partial projection at a later time, you need not recalculate it (provided the PROCNO was not overridden by other data in the mean time). Just click on *part* to get it on screen. You may vertically scale partial projections using the **2* and */2* buttons, located below the *row/col* buttons.

The calculation of partial projections can also be initiated with the commands specified in Table 3.4 (type them in or call them from the *Process->Calculate projections* menu).

Axis	Projection typ	Command
F1	positive	f1projp
F1	negative	f1projn
F2	positive	f2projp
F2	negative	f2projn

Table 3.4 Commands to calulate projections

As you can see from this table, the program allows you to obtain positive and negative partial projections. After startup, these commands invite you to enter the row or column numbers defining the spectral range for the calcuation. Please note that the buttons *calc* for *F1* or *F2* dimensions, calls the commands f1projp and f2projp respectively and will display the positive projections.

Sums

Partial projections are projections in the mathematical sense: along each projection trace the largest positive or the smallest negative intensity is picked. Alternatively, XWIN-NMR allows you to calculate the sum of the intensities using the *sum/calc* button combination. The handling is identical to partial projections, the associated commands are f1sum and f2sum.

Disco projections

Similar to sums as described in the previous section, the sum of a number of columns or rows is calculated. However, unlike *F1 Sum* and *F2 Sum*, *F1 Disco* adds a column to the sum if the point at which it intersects a reference row is positive, otherwise it is subtracted. Similarly, *F2 Disco* adds a row to the sum if the point at which it intersects the reference column is positive, otherwise it is subtracted. Only those data points whose value is greater than the plot parameter MI, are used in building the sum.

The procedure for using the *disco* buttons is similar to the *sum* buttons, except that once the column or row region has been defined it is necessary to position the cross hair a third time to define the reference row or column, respectively, and press the middle mouse button once more.

The calculation of disco projections can also be initiated with the commands f1disco and f2disco (type them in or call them from the *Process->Calculate projections* menu).

If the user wishes to calculate a Disco projection from several regions, but from the same reference, he may, once the first calculation is finished, activate the appropriate Disco command once more with the middle button and redefine the new region. Since the last reference used is still visible on the screen it is easy to position the cross hair once more in order to have the same reference for the new calculation.

Available literature about Disco-projections:

Kessler, Müller, Oschkinat: Mag. Res. in Chemistry 23 (1985), 844

Chapter 4

3D Data Processing Tutorial

This chapter presents 3D data processing based on the data set *exam3d.* The data set is stored in the directory */u/data/guest/nmr/*. In order to have full access permissions to it, you should have a user *guest* installed on your system, and be logged in as *guest*. Start XWIN-NMR by typing xwinnmr -r. The *-r* option ensures that everything is cleaned up before starting the program, even if the last session was terminated by some problem. The program will start without displaying a data set. Instead, the XWIN-NMR logo will be shown in the data area. In later sessions, you can start XWIN-NMR without specifying the *-r* option, and you will immediately get the last used data set displayed.

4.1 Getting the data set

The first step is to get the 3D spectrum *exam3d* on screen. Open the *File* menu, select the *Open* entry, and from there the command dir. You should get a dialog box where the data sets *exam1d*, *exam2d*, and *exam3d* are listed. If there are other data sets in the directory */u/data/guest/nmr/*, they will also appear in the box. Click on *exam3d* in oder to instruct the program to make this data set the *current data set*. XWIN-NMR will automatically show the *3D layout*, because it identifies *exam3d* as a 3D data set. The identification is not performed via the name of the data set, but through the contents of the parameter file *meta* stored in the directory */u/data/guest/nmr/exam3d/1/pdata/1/*. For space reasons only the acquisition data

of *exam3d* are delivered on the release tape. You must enter the transform commands tf3, tf2, tf1 in this sequence to execute a 3D transform (type in the commands, followed by Return, or invoke them from the *Process->Fourier transform* menu). The commands will ask you whether to store the imaginary parts. You may answer *n* (for *no*) for now to save space on disk. An imaginary part is needed if a phase correction is to be applied in the corresponding dimension. You may retransform the data later for this purpose and answer *y* to the question, or alternately start a *Hilbert* transform to obtain imaginary parts from the already existing real part. After tf3/tf2/tf1 are finished, the transformed spectrum will not yet be visible on the screen. You must click on the *display* button on the left hand button panel in order to initiate the calculation of peak contours. Similar to the contour representation of 2D spectra, 3D spectra are displayed by means of contour lines. While in the 2D case, contours are always calculated in the F1-F2 plane, in the 3D case the two additional planes F2-F3 and F1-F3 are present. The contour calculation routine will ask you whether to compute the contour for a particular plane. The best impression of the 3D spectrum on screen is usually achieved by having the program calculate all contours. However, this takes the longest, too.

4.2 Manipulating the spectrum on the screen

You should now have Figure 4.1 on the monitor, representing the standard 3D layout.The data area shows the 3D spectrum in contour display mode. The right side indicates the colors assigned to the different intensity values.

The button panel at the left side of the XWIN-NMR window allows you to interactively scale and expand the data, and to enter a subroutine. These operations are *screen operations* only, and do not modify the data file. You must click on a button to execute a function. Some buttons require that you keep the left mouse button pressed while the cursor is located in the command button field, and then move the mouse. When you move the cursor over the command buttons (without clicking), their function is displayed in the status line at the bottom of the XWIN-NMR window.

1. Enlarge spectrum by a factor of 1.1

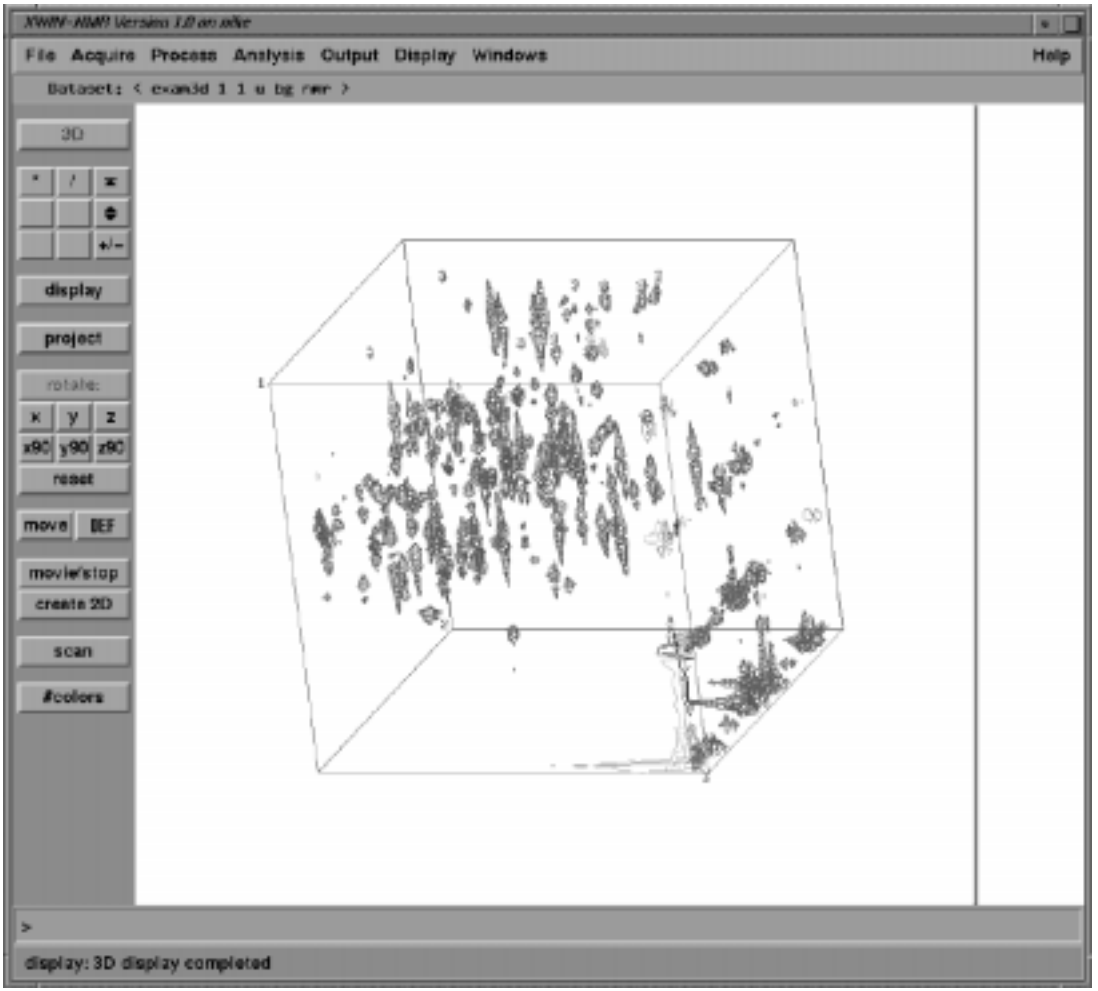


Figure 4.1 The 3D screen layout

2. Shrink spectrum by a factor of 1.1
3. Reset the spectrum size to the default value

1. No function assigned
2. No function assigned
3. Change the assignment of colors to intensities by keeping the number of colors used constant. The mapping can be observed by looking at the color bar at the right side of the XWIN-NMR window. Keep the left mouse button depressed while the cursor is located on this button, and move the mouse.

t

1. No function assigned
2. No function assigned
3. Normally, when the spectrum contains negative peaks, both negative and positive intensity values (levels) are displayed. The standard colors assigned are *red-yellow-green* shadings for positive data, and *light blue-dark blue-violet* shadings for negative data. This button allows you to toggle through display modes where the negative or the positive data are turned off, or where the whole color range from red to violet is assigned to the positive data.

Calculate contours and display the spectrum as already described above. If the spectrum does not fit in memory, a file *dsp3d* is created. It is stored in the current processed data directory, i.e. the directory where the transformed spectrum file is located. The name of the spectrum file (real data only) is *3rrr*. You can delete the file *dsp3d* before archiving, if you want to save space on the archive, or speed up archiving.

Display the projection of the spectrum.

Rotate the spectrum around the horizontal (x) or vertical (y) axis, or around the axis perpendicular to the screen surface (z). Click on the desired button to enter rotation mode, and move the mouse to rotate the spectrum in real time. Press the right mouse button to leave rotation mode.

Rotate the spectrum around the axes described above by 90 degrees.

Reset rotation angles to their default. This button also exits from *project* mode.

1. Enter *move* mode. You may shift the spectrum in real time on the screen. Press the right mouse button to quit.
2. Reset the position of the spectrum on the screen to its default value.

Clicking on this button invokes a demonstration loop, where the spectrum is automatically enlarges, rotated and moved on the screen. Another click on the same button terminates the movie.

Click on this button if you intend to execute a 2D transform on 3D acquisition data. The result is a 2D spectrum which is a slice of the 3D spectrum. The effect of this button is that a new data set is created differing from the current one only in the PROCNO processed data number, which you will be asked for. The 2D slice will later be stored under this PROCNO. In addition to creating the directory for this PROCNO, its processing parameters are initialized such that the slice corresponds to a F3-F2 section of the 3D spectrum. XWIN-NMR then switches from the current 3D data set to this new 2D data set, where you may execute e.g. a xfb command.

Enter scan mode (Figure 4.2). In this mode, a number of possibilities are offered to view the 2D cross sections of the 3D spectrum parallel to the three main axes.

Real time display of the F2-F3, F1-F3, or F1-F2 cross section, respectively. Click on the desired button to enter scan mode. Move the mouse to position to the desired plane. Press the right mouse button to quit this mode.

Display the next or previous F2-F3, F1-F3, or F1-F2 plane.

Click on the desired button to start auto-scanning the F2-F3, F1-F3, or F1-F2

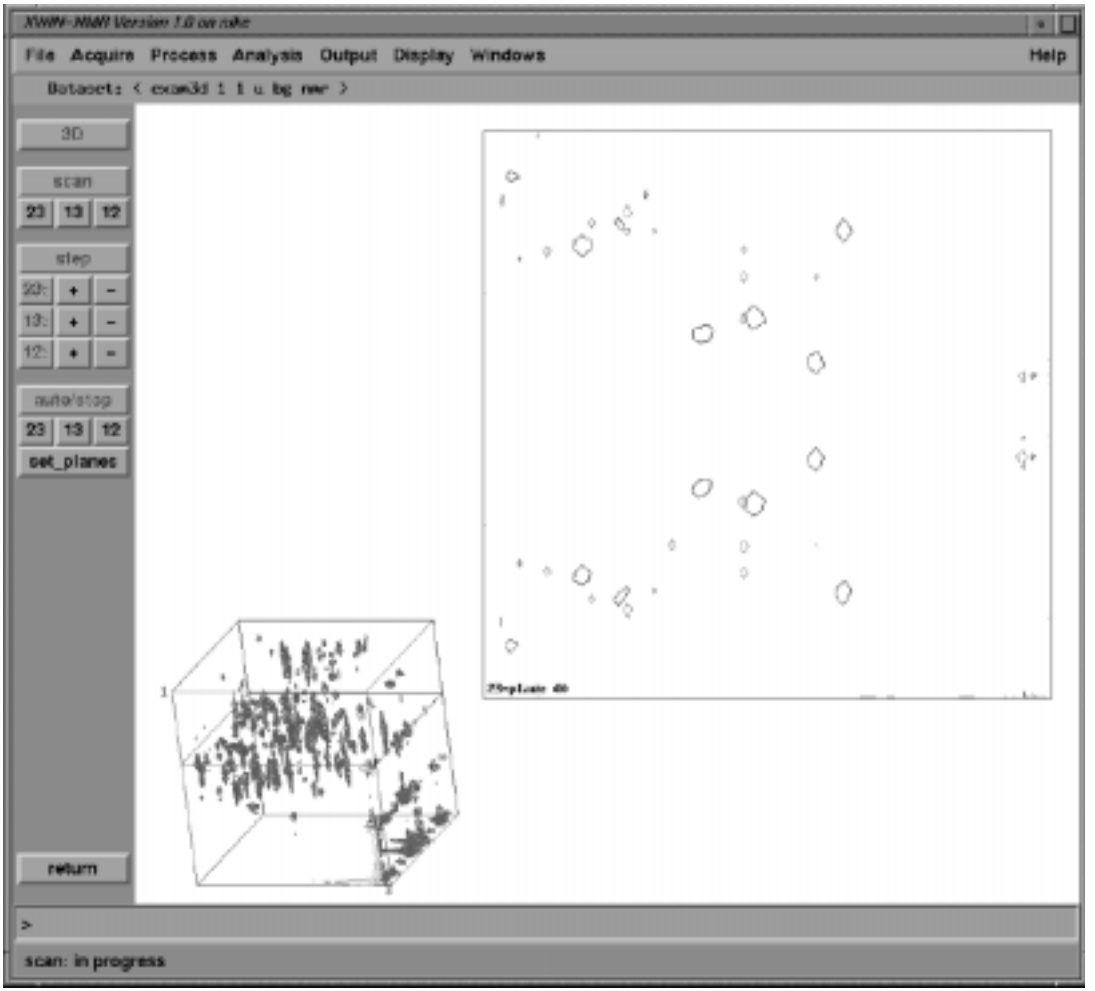


Figure 4.2 Viewing 2D sections of a 3D spectrum

planes. Another click on the same button will stop auto-scanning.

This command button allows you to select planes by their plane number.

4.3 Setting up the parameters for a 3D transform

The main processing steps to calculate a spectrum from a 3-dimensional fid are window multiplication and fourier transformation. In 3D (like in 2D), window multiplication and fourier transform (and optionally linear prediction, phase correction or magnitude calculation) are part of the transform commands (tf3, tf2, tf1). Unlike in 2D, no combined transform commands exist.

Call the processing parameter editor edp, either by typing in this command, or by opening the *Process* menu and selecting General parameter setup [edp]. A dialog box is displayed containing all processing parameters in three colums for the three dimensions to be processed. Some parameters exist for all dimensions, others only for one dimension. It should be mentioned here that the contents and layout of the edp dialog box can be tailored by the user by editing the so-called *format file*

/u/exp/stan/nmr/form/proc.e .

The most important parameters you must set in the edp window are SI, WDW, PH_mod (for *F3, F2* and *F1*), AQSEQ (for *F3*), and MC2 (for *F1,F2*). *F3* is the dimension defined by the acquisition of one fid of the 3D experiment, *F1* and *F2* are the orthogonal dimensions defined by the acquisition status parameters TD for these dimensions of the 3D experiment. SI specifies the number of real points the spectrum should have after the transform in the corresponding dimension. WDW allows you to select an appropriate window function. With PH_mod you can select whether to apply phase correction or not. For phase sensitive spectra, an extra step is usually required to find the correct phases. Phasing is then performed using extra commands after the transform. PH_mod also allows you to select magnitude or power spectrum calculation. AQSEQ is a parameter describing the order in which the fids comprising the 3D acquisition data are stored in the *ser* file. There are two possible orders depending on the pulse program used, called 321 and 312 (cf. the chapter on 3D processing for details). MC2 determines the phase type of the transform and also depends on the performed experiment. You will find a detailed description of these parameters in the chapter *The 3D Process Menu*.

XWIN-NMR also allows you to modify the parameters by typing the parameter names (in lower case letters) on the keyboard. For example, the command si allows you to enter the SI parameter for the *F3* dimension, the command 1 si for the *F1* dimension, and 2 si for *F2*. The command wdw allows you to enter the

WDW parameter for the *F2* dimension, the command 1 wdw for the *F1* dimension. The command lb allows you to enter the LB parameter for the *F2* dimension, the command 1 lb for the *F1* dimension, etc.

Please note: 3D processing commands do *not* overwrite the measured 3D fid with the processed values. Instead, they create new files containing the result. The 3D fid (in our example) is stored in the first file of Table 4.1, while the processing

/u/data/guest/nmr/exam3d/1/ser
/u/data/guest/nmr/exam3d/1/pdata/1/3rrr

Table 4.1 Acquisition and processed data files of *exam3d*

result is stored in the other files. *pdata* is the *processed data subdirectory.* The file *3rrr* contains the real data points after the fourier transform. Files with imaginary data points (e.g. *3iii*) are only present if the transform was executed with the option to store the imaginary data in the case of phase sensitive spectra.

4.4 Applying the 3D Fourier Tranform

Type tf3 on the keyboard, or open the *Process->Fourier transform* menu and click on tf3 to transform the acquisition dimension. Continue with tf2 and tf1 for the other dimensions. The progress is reported on the status line at the bottom of the XWIN-NMR window. At the end of the transform, the software calculates a compressed spectrum corresponding to the display window size from the real part *3rrr*, and stores it in the file *dsp3d* if the spectrum is larger than the available memory. *3rrr* or *dsp3d* are used to calculate the contours when clicking on the *display* button of the left hand button panel. *dsp3d* is only needed for data display and is recalculated automatically from the spectrum file *3rrr* if missing.

4.5 Strip transform, linear prediction, other transform options

XWIN-NMR provides a number of 3D transform options such as linear prediction, strip transform, inverse transform, user defined transform, and Hilbert transform.

Linear prediction is an attractive option to perfect initial data points of the acquisi-

tion data before the transform in order to obtain better baselines. Furthermore, completing truncated Fids with linear prediction gives better estimates of the data and therefore allows for faster acquisition of multi-dimensional experiments. Linear prediction is executed as a part of the 3D transform, and is enabled by setting the processing parameter ME_mod to *LPfr* (forward) or *LPbr* (backward) prediction. The default value of ME_mod is *no*.

The *strip transform* allows you to execute the fourier transform in such a way that the result is not the entire 3D spectrum, but only a selected portion of it, defined by the processing parameters STSR (strip start, measured in points from the beginning of the spectrum) and STSI (size of region in points). A strip transform is useful if you are only interested in a particular spectral region. The result is a file of significantly smaller size, which can be handled faster on the display and consumes less disk space. You are also able to set the transform size for the region (and therefore its resolution) to a larger value than would be possible when tranforming the whole spectrum. You may also combine the strip transform with linear prediction.

The *Hilbert transform* allows you to generate the imaginary part of a spectrum from the real part.

More details of these transform options are described in the chapter *The 3D Process Menu*.

4.6 Phase correction

The 3D phase correction can either be performed during the 3D Fourier transformation or at a later time. The former method saves computation time since the entire 3D data set only needs to be processed completely once. The two methods differ somewhat in the way that the phase angles PHC0 and PHC1 for each dimension are determined.

4.6.1 Correction during the transformation

Set the parameter PH_mod before issuing the transformation commands for the respective dimension so that the phase angles will be taken into account during the calculation. The phase angles are determined as follows:

F3 dimension

When applying the 2D transform command xf2 to a 3D data set, you will be asked for the slice number to be transformed, the rest of the *ser* file is ignored. (Please note that xf1, xf2, xfb may all be executed on a 3D *ser* file. The desired slice number can be specified as an argument behind the commands. In order to display the resulting 2D data set, the parameter PARMODE should be changed to *2D*. The parameter AQORDER is taken into account). The phase angles can now be determined from the resulting 2D spectrum and may then be used in tf3. These phase angles can also be determined by extracting a single fid from the *ser* file with the command rser, performing a 1D Fourier transformation and then phase correcting the spectrum in 1D mode.

F2 dimension
After the tf3 transformation, extract a F3-F2 slice with the command r23, transform this slice with xf1 and determine the phase angles from the resulting 2D spectrum. Then perform tf2 with these phase angles.

F1 dimension
After the tf2 transformation, extract a F3-F1 slice with the command r13, transform this slice with xf1 and determine the phase angles from the resulting 2D spectrum. Then perform tf1 with these phase angles.

4.6.2 Correction after the transformation

The phase correction for a certain dimension must follow the transformation in that dimension immediately, i.e. before the next dimension is transformed. The reason for this lies in the incomplete storage of imaginary parts after the transformations (in order to save disk space, details are described in the chapter *The 3D Process Menu*).

The phase correction commands are tf3p, tf2p and tf1p for the three dimensions F3, F2, and F1, respectively. The spectrum is phase corrected with the corresponding phase angles PHC0 and PHC1. They are to be determined as follows:

1) transformation tf3 (with PH_mod=*no*)
2) extract a F3-F2 or a F3-F1 slice with r23 or r13
3) determine PHC0 and PHC1 from this slice in 2D mode
4) execute tf3p

The two other dimensions are treated analogously. The phase correction commands read or write the files in Table 4.2. In order to save computation time and

	read	write
tf3p	3rrr, 3irr	3rrr, 3irr
tf2p	3rrr, 3rir	3rrr, 3rir
tf1p	3rrr, 3rri	3rrr, 3rri

Table 4.2 Files read and written by the phase correction commands

disk storage the commands also offer the possibility of dropping imaginary parts after execution. The user will be asked if this should be done. The question may be suppressed by giving an argument with the command, e.g. tf3p y to store the imaginary part, or tf3p n not to store it.

In AU programs the phase correction commands must always be given with one (single) argument which specifies whether the imaginary parts are to be stored, e.g. tf3p("y").

4.7 Baseline Correction

In the chapter *1D Data Processing Tutorial*, we discussed the baseline correction command absf for 1D spectra which assumes that a baseline distortion can be removed by subtracting a polynomial from the spectrum. The commands tabs3, tabs2 and tabs1 use the same algorithm to correct a row of a 3D spectrum in the corresponding dimension. The commands can be typed in, or invoked from the *Process->Baseline correction* menu. As in 1D, the parameter ABSG determines the degree of the polynomial, and can be set independently for the three dimensions from edp. The parameters ABSF1 and ABSF2 determine the left and right limits of the region to be corrected. In contrast to the 1D command absf, the 3D commands do not generate integral range files.

4.8 More on 3D

Please check the chapter *The 3D Process Menu* for more details on 3D data processing. It should be mentioned here that XWIN-NMR only offers a basic set of 3D tools. A broad range of 3D visualisation and analysis routines are available in Bruker's AURELIA software package.

Chapter 5

The *File* Menu

Like the majority of window based programs today, XWIN-NMR collects file handling commands in the *File* menu (Figure 5.1)

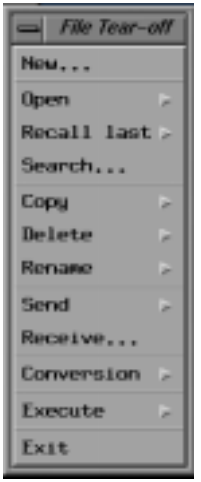


Figure 5.1 The *File* menu

5.1 New

New creates a new data set. This command (which is identical to edc) is required if the next data acquisition (command zg) is not to be started from an existing data set, but from a new one now to be defined by the user. A dialog box according to Figure 5.2 is opened where you must specify the location and name of the new data

Figure 5.2 The *File->New* dialog box

set.

From the parameters NAME, EXPNO, PROCNO, DU and USER, a directory is created to contain the new data after acquisition:

/u/data/guest/nmr/sucrose/1/,

or, in general,

/DU/data/USER/nmr/NAME/EXPNO/.

The parameters in capital letters may be chosen by the user, the items *data* and *nmr* are inserted by the program automatically. DU determines the location of the data set in the file system of the operating system. Examples: *DU=u*, or *DU=usr/people/guest*. Under UNIX it implicitely specifies the physical disk drive used, which can be a second drive, or a drive mounted from a network. USER is the login name of the current user. EXPNO (the *experiment number*) must be a number, and is used to count different experiments with the same NAME. The result of an acquisition is a file *fid* (for 1D) or *ser* (for 2D, 3D, ...) containing the acquired data, which will be stored in this directory. When the directory is created, it is initialized with default parameter files, which are copied from the current data set visible on

the screen. If there is none, the parameter files are copied from the standard parameter sets delivered with the XWIN-NMR release media:

XWINNMRHOME/exp/stan/nmr/par/standard1D/

where *XWINNMRHOME=/u* in an XWIN-NMR standard installation. The parameter PROCNO (*processed data number)* is used to create a subdirectory

/u/data/guest/nmr/sucrose/1/pdata/1/,

or, in general,

/DU/data/USER/nmr/NAME/EXPNO/pdata/PROCNO/

which will contain the processed data (usually the transformed spectrum) of the corresponding acquisition data. A 1D real spectrum, for example is stored in the file *1r* in this directory, the corresponding imaginary part in the file *1i*. The real part of a 2D or 3D spectrum is stored in the files *2rr* or *3rrr*, respectively.

5.1.1 Data access permissions

When a new data set is created, it will get the Unix read / write permissions of the login user who started XWIN-NMR, and the group read / write permissions of the Unix group she/he belongs to. Other users will not be able to overwrite or delete these data sets using XWIN-NMR commands.

In certain cases it can happen that the Unix permissions of a data set stored under the parameter USER (see previous section) are wrong and prevent the user from processing the data. Examples for such cases:

1. Assume that the login user who started XWIN-NMR is 'nmr1'. Assume further that the data set was created during a run of a sample changer equipped with a bar code reader, and the bar code label for an experiment contained user 'nmr2'. Then, the data set created by this experiment would be stored under USER=nmr2, but get the permissions of 'nmr1'.
2. A data set is retrieved from a tape and stored under USER=nmr1 (e.g. using the Unix command tar xv). The data set on tape, however, originates from a different laboratory, where it belonged to user 'nmr2'. The result is a data set on disk stored under USER=nmr1, but with the permissions of 'nmr2' (depending on the tar options used).

The XWIN-NMR command touser solves this problem: Make the respective data set

the current data set in XWIN-NMR and enter touser on the keyboard. The permissions will be set correctly for user 'nmr1', and the data set can be further processed.

5.2 Open: Data Files

The *Open* menu (Figure 5.3) contains commands which display a list of existing

Figure 5.3 The *Files->Open* submenu

data sets. If you select one from the list, it will be made the current data set and displayed on the screen.

The command names in brackets [] may be used to quickly access the function from the keyboard. The commands are described in Table 5.1.

The dir commands
Please note that all dir type commands can be used with *wildcard* arguments, such as dir *, dir suc*, dir ??abc*, etc, which allow you to choose data sets according to a desired mask. The asterisk serves as a placeholder for a group of unknown characters, the question mark for a single character.

The re/rep command
re is usually used for quick operation at the keyboard, and may be applied as

dir	display a list of *data set names* (NAME parameter) of the current USER, stored in the user's data directory (*/DU/data/USER/nmr/*)
dirdat	like *dir*, but data sets are ordered according to their acquisition dates
dirp	processed data, any dimension, e.g. *1r,1i*- and *2rr, 2ii, ...* -files
dira	acquisition data, any dimension (*fid*- and *ser*-files)
dirs	processed one-dimensional data (*1r, 1i*-files)
dir2d	processed two-dimensional data (*2rr, 2ii, ...*-files)
dirf	one-dimensional fids (*fid*-files)
dirser	multi-dimensional fids (*ser*-files)
diro	users owning data sets in the current data path (default: */u/data/*)

Table 5.1 The *dir* commands

shown by the examples in Table 5.2.

re	XWIN-NMR asks you to enter the data set specifications
re sucrose 2 4 u guest	directly specify NAME, EXPNO, PROCNO DU, USER
re sucrose 2 4	specify NAME, EXPNO, PROCNO, leave DU, USER
re sucrose	change NAME only
re 2	change EXPNO only
re 2 4	change EXPNO and PROCNO
rep 2	change PROCNO only

Table 5.2 Examples of commands re and rep

The search command

search, also member of the *File* menu, is an alternate way to find and display existing data sets. It is described later in this chapter.

The browse command

browse, yet another way to find and display existing data sets. Browse may be

specified with a numerical argument, e.g. browse 3, defining the entry level into the data set hierarchy.

5.3 Open: Other Files

Please note:
In this and other sections of this manual the names of various directories are introduced containing important files such as lists, pulse programs, etc. These directories usually start with the string /u. This applies for the case of an XWIN-NMR standard installation. If your XWIN-NMR was installed somewhere else, /u must be replaced by the correct installation directory. In order to find it, examine the environment variable XWINNMRHOME: type 'env' to your operating system command shell to display the setting of this variable. XWIN-NMR must be active while you execute the 'env' command.

Other than data files, XWIN-NMR requires special purpose files for a number of operations. These files can be accessed via the *other files* box (Figure 5.4).

Figure 5.4 The *Files->Open->Other files* submenu

5.3.1 Parameter sets [dirpar]

A parameter set is a collection of parameter files, stored in the directory
/u/exp/stan/nmr/par/PARSETNAME/.
This command displays all currently available parameter sets. XWIN-NMR is

always delivered with a large number of parameter sets, which include predefined acquisition, processing, and plot parameters, and cover the most important NMR experiments. After installing XWIN-NMR from the release media, these parameter sets are contained in the directory

/u/exp/stan/nmr/par.300/,

and are prepared for a spectrometer frequency of 300 MHz. For example, one of the standard parameters sets is called C13DEPT135, and is prepared for such an experiment. The command expinstall (see menu *Acquire -> Spectrometer setup*), which must be executed after installation of XWIN-NMR, will convert the parameter sets to your spectrometer frequency, and place them in the working directory

/u/exp/stan/nmr/par/.

A parameter set directory, e.g. */u/exp/stan/nmr/par/C13DEPT135/*, usually contains the files of Table 5.3.

acqu	acquisition parameter file
acqu2, 3	acquisition parameter file for 2nd (3rd) dimension (if 2D parameter set)
proc	processing parameter file
proc2, 3	processing parameter file for 2nd (3rd) dimension (if 2D parameter set)
meta	plot parameter file
meta.ext	plot parameter file for automatic plot expansions
outd	parameter file with output device names for plotting and printing

Table 5.3 Parameter file names

Parameter files are stored in ASCII format, conforming to the JCAMP-DX standard for NMR files. They are set up by the user via the parameter editing commands eda, edp, edg, edo for acquisition, processing, plotting, and output device parameters, respectively, and can be found in the menus *Acquire, Process, Output*. They are described in more detail in the corresponding manual chapters. Copies of parameter files from and to parameter set directories can be made via the commands rpar and wpar (see *File->Copy* menu).

5.3.2 Pulse programs [edpul]

Pulse programs are text files containing statements in the Bruker pulse program language (described elsewhere in this manual). This command allows you to create a new pulse program, and to edit or view an existing one.

XWIN-NMR locates pulse programs in the directory

/u/exp/stan/nmr/lists/pp/.

The command edpul displays all pulse programs in this directory in a dialog box. On XWIN-NMR release media, a large number of sample pulse programs are delivered. After installation of XWIN-NMR, they are stored in the directories of Table 5.4.

/u/exp/stan/nmr/lists/pp.exam	AMX high resolution
/u/exp/stan/nmr/lists/pp.rexam	ARX high resolution
/u/exp/stan/nmr/lists/pp.dexam	AVANCE
/u/exp/stan/nmr/lists/pp.solids	AMX/ASX solids
/u/exp/stan/nmr/lists/pp.imag	micro-imaging
/u/exp/stan/nmr/lists/pp.tomo	tomography

Table 5.4 Sample pulse program directories

The command expinstall (see menu *Acquire -> Spectrometer setup*), which must be executed after installation of XWIN-NMR, will copy the selected pulse program library to the working directory */u/exp/stan/nmr/lists/pp/*.

edpul opens a dialog window with two colums. The right column shows the programs provided by Bruker, after expinstall was done. The left column lists those pulse programs that were created by the user. A Bruker standard pulse program that was modified by the user is also listed in the left column. Modification of a Bruker standard pulse program, however, requires input of the superuser password if one is defined in the system. When selecting a pulse program from the right column the text is displayed in a dialog window. In order to modify the program the command buton '*->Edit*' at the lower edge must be activated and then the program be selected. This brings the program into the text editor. For user pulse programs (left column), the text is always loaded into the text editor. If a new pulse program is to be created, the command field 'New Name' at the lower edge of the dialog window must be activated. After entering a name for the pulse program the text editor is called and the program may be written.

Text editor
A text editor for your system is called. You can specify your own editor by invoking User Interface... from the *Display->Options* menu, or by typing in the command setres.

Arguments and Wildcards
The command edpul, when entered on the keyboard, can optionally take the name of a pulse program as an argument. If it exists it is shown in a dialog window in the case of a Bruker pulse program, or otherwise loaded into the text editor for modification. If it does not exist, the text editor is called and a new one can be set up. If the argument is not a complete name but contains the wildcard characters * or ?, the same dialog window as for edpul without argument appears. However, this dialog window then only lists those pulse programs which match the wildcard pattern.

Examples:

edpul cos*	(list all pulse program beginning with *cos*)
edpul [m-z]*	(list all pulse program beginning with m,n,..,z

5.3.3 Current pulse program [edcpul]

When a data acquisition is started, XWIN-NMR enforces the execution of the pulse program whose name is stored in the acquisition parameter PULPROG of the current data set. The command edcpul does the same as edpul <PULPROG>, i.e. it uses the name of the pulse program specified in the acquisition parameter PULPROG as argument.

The command edcpul <pulse program name> does the same as edpul <pulse program name>, but additionally sets the acquisition parameter PULPROG to the specified pulse program name.

5.3.4 Composite pulse decoupling programs [edcpd]

Composite pulse decoupling (CPD) programs are text files containing statements in the Bruker CPD program language (described elsewhere in this manual). This command allows you to create a new CPD program, and to edit or view an existing one.

XWIN-NMR locates CPD programs in the directory

/u/exp/stan/nmr/lists/cpd/.

The command edcpd displays all CPD programs in this directory in a dialog box from where the desired one can be selected.

On XWIN-NMR release media, a number of sample CPD programs are delivered. After installation of XWIN-NMR, they are stored in the directory

/u/exp/stan/nmr/lists/cpd.rexam/ (for A*X systems)
/u/exp/stan/nmr/lists/cpd.dexam/ (for Avance type systems)

The command expinstall (see menu *Acquire*), which must be executed after installation of XWIN-NMR, will copy the contents of the respective directory to the working directory

/u/exp/stan/nmr/lists/cpd/.

The command edcpd may also be entered on the keyboard, followed by the name of the desired CPD program, or followed by an argument containing wildcard characters (see edpul and how to change the default text editor for details).

5.3.5 Gradient programs [edgp]

Gradient programs are text files containing statements in the Bruker gradient program language (described elsewhere in this manual). This command allows you to create a new gradient program, and to edit or view an existing one.

XWIN-NMR locates gradient programs in the directory

/u/exp/stan/nmr/lists/gp/.

The command edgp displays all gradient programs in this directory in a dialog box from where the desired one can be selected. The phase, read, and slice gradients are contained in three different files with the extensions *.p*, *.r*, and *.s*, respectively.

On XWIN-NMR release media, a number of sample gradient programs are delivered. After installation of XWIN-NMR, they are stored in the directory

/u/exp/stan/nmr/lists/gp.exam/
/u/exp/stan/nmr/lists/gp.solids/
/u/exp/stan/nmr/lists/gp.imag/

The command expinstall (see menu *Acquire -> Spectrometer setup*), which must be executed after installation of XWIN-NMR, will copy the contents of the selected directory to the working directory

/u/exp/stan/nmr/lists/gp/.

The command edgp may also be entered on the keyboard, followed by the name of

the desired gradient program, or followed by an argument containing wildcard characters (see edpul and how to change the default text editor for details).

5.3.6 Automation programs [edau]

Automation (AU) programs are text files written in the language C, with the extension that XWIN-NMR commands may be included. edau allows you to create a new AU program, and to edit or view an existing one. When exiting from the text editor, the AU program must be compiled before it can be executed. XWIN-NMR asks you whether you want to compile the current AU program. If yes, the computer's C compiler is invoked. Details about writing AU programs are described in a special chapter of this manual. The purpose of AU programs is to implement user defined commands in XWIN-NMR, either based on XWIN-NMR internal commands, or entirely self-written. The new command is just the name of the AU program, and can be entered on the keyboard or called in other AU programs (see menu *File->Execute*).

XWIN-NMR locates AU programs in the directories

/u/exp/stan/nmr/au/src/ (C sources)
/u/prog/<XWIN-NMR version>/au/bin/ (executables)

The command edau displays all AU programs in the *src* directory in a dialog box from where the desired one may be selected. The box consists of two halves. The left side shows those AU programs which were written by the user himself, or which represent modifications of Bruker AU programs. Bruker AU programs are write protected and are displayed at the right side of the box. When you select one from there, you can only scroll through the text, but not modify it. The superuser of the system, however, may also modify Bruker AU programs. He must activate the command button *'->edit'* at the bottom of the edau dialog box, and then select the desired Bruker AU program. The superuser password is then requested. After a Bruker AU program was modified, it will no longer appear on the right side, but will be moved to the left side of the edau window. You can always restore the original Bruker AU program from the directory *src.exam* to be described below.

On XWIN-NMR release media, a number of AU programs designed by Bruker are delivered. After installation of XWIN-NMR, they are stored in the directory

/u/prog/<XWIN-NMR version>/au/src.exam/

The command expinstall (see menu *Acquire -> Spectrometer setup*), which must be executed after installation of XWIN-NMR, will compile them if user requested,

and install their C sources in the working directory

/u/exp/stan/nmr/au/src/,

and their executables in

/u/prog/<XWIN-NMR version>/au/bin/.

The command edau may also be entered on the keyboard, followed by the name of the desired AU program, or followed by an argument containing wildcard characters (see edpul and how to change the default text editor for details).

edau is almost identical to the command edaumod . The only difference is that when edau is called, AU programs are displayed, while edaumod shows the so-called AU modules. You can switch to AU module display by clicking on the *>Modules* command button at the bottom of the dialog box.

5.3.7 Macros [edmac]

Macros are text files containing XWIN-NMR commands. This command allows you to create a new macro, and to edit or view an existing one. The purpose of macros is to implement user defined commands in XWIN-NMR, based on XWIN-NMR internal commands. In contrast to AU programs, macros do not allow for control structures such as loops and branches, nor arithmetic expressions. The new command is just the name of the macro, and can be entered on the keyboard or called in other macros (see menu *File->Execute*). Macros can also be called in executable AU program.

XWIN-NMR locates macros in the directory

/u/exp/stan/nmr/lists/mac/

The command edmac displays all macros in this directory in a dialog box from where the desired one may be selected.

The command edmac may also be entered on the keyboard, followed by the name of the desired macro, or followed by an argument containing wildcard characters (see edpul and how to change the default text editor for details). Table 5.5 shows an example. The charcacter '#' signals the beginning of a comment. Otherwise, each line corresponds to a XWIN-NMR command, which is specified exactly the way it would be entered on the command line.

5.3.8 Parameter lists [edlist]

Parameter lists are text files containing a series of values for parameters which

```
dir         #select data set
lb 0.5      #set line broadening parameter LB
si 16K      #set transform size SI
ef          #perform exponential multiply
            # and transform
```

Table 5.5 Example of a macro file

may vary during an experiment. This command allows you to create a new list, and to edit or view an existing one.

XWIN-NMR locates parameter lists in the directory

/u/exp/stan/nmr/lists/<list type>/

The command edlist displays all lists in this directory in a dialog box from where the desired one may be selected. See Table 5.6 for the available list types.

list type	contains
vd	delay lists
vp	pulse lists
f1	frequency lists (AVANCE)
f1, f2, f3	frequency lists (A*X)
vt	temperature lists
vc	loop counter lists
ds	data set lists
masr	MASR rotation values

Table 5.6 Types of parameter lists

Since pulse, CPD, gradient programs and macros are stored in the same directory, the corresponding list types *pp, cpd, gp,* and *mac* will also be displayed when edlist is called. It is therefore possible to edit pulse programs etc. via the dedicated commands edpul, etc., but also via edlist by specifying the corresponding list type.

The command edlist may also be entered on the keyboard, followed by the type of the desired list, or followed by the list type and the desired list name, which may contain wildcard characters (see edpul and how to change the default text editor

for details).

Format of parameter lists
The length of a parameter list is unlimited.

A *delay* or *pulse list* (Table 5.7, left column) contains one duration value per line,

10s	O 500.13
20m	3000
1.5u	3150

Table 5.7 Example of a delay (left) and frequency list (right)

followed by a time unit (s=seconds, m=milliseconds, u=microseconds). The values in a delay list are interpreted by the variable delay statement vd in pulse programs. The values in a pulse list are interpreted by the variable pulse statement vp in pulse programs. These commands use the list whose name is defined in the acquisition parameter VDLIST or VPLIST, respectively.

The first line of a *frequency list* (Table 5.7, right column) contains the absolute frequency in MHz for the corresponding nucleus, preceded by an O, and a space character. The following lines contain offsets in Hz. The transmitter frequency is the sum of absolute and offset frequencies. The list is evaluated by the variable frequency statements fq1,...,fq8 in pulse programs. The command fq1 uses the current value in the list whose name is defined in the acquisition parameter FQ1LIST. Likewise, fq2 gets the list name from the parameter FQ2LIST, etc. These commands are valid for AVANCE spectrometers. For AMX/ARX/ASX instruments, the frequency setting statements are called o1,o2,o3. Instead of 8, only 3 list name parameters F1LIST, F2LIST, F3LIST are available.

A *temperature list* (Table 5.8, left column) contains one temperature value in Kel-

300	4
320	7
340	20

Table 5.8 Example of a tempearture (left) and loop counter list (right)

vin per line. They are evaluated by the variable temperature statement vt in pulse programs. This command uses the list whose name is defined in the acquisition parameter VTLIST.

A *loop counter list* (Table 5.8, right column) contains one positive integer number per line. The numbers are evaluated by the variable loop counter statement lo to n times c in pulse programs. This command uses the list whose name is defined in the acquisition parameter VCLIST.

A *data set list* (Table 5.9) contains one data set specification per line in the form

```
sucrose  1 1 u guest new
sucrose  2 1 u guest new
fructose 1 1 v guest old
```

Table 5.9 Example of a data set list

<NAME> <EXPNO> <PROCNO> <DU> <USER> <flag>

The data sets in a list are evaluated by the variable data set statement wr #n in pulse programs. This command uses the list whose name is defined in the acquisition parameter DSLIST. The *new* flag indicates that the data set is to be created newly by possibly deleting an existing one. The *old* flag will retain an existing data set.

5.3.9 Miscellaneous files [edmisc]

Miscellaneous files are text files containing information produced by various XWIN-NMR operations, such as integral ranges, peaks for deconvolution etc. which are stored on disk for later use. This command allows you to create a new miscellaneous file, and to edit or view an existing one.

XWIN-NMR locates miscellaneous files in the directory

/u/exp/stan/nmr/lists/<file type>/

The command edmisc displays all files in this directory in a dialog box from where the desired one can be selected. Table 5.10 shows the available file types.

The command edmisc may also be entered on the keyboard, followed by the type of the desired file, or followed by the file type and the desired list name, which

file type	contains
intrng	integration regions, generated by manual integration or automatic integration (command abs). Used in plotting and integral printout (commands li, lipp)
base_info	baseline points from manual baseline correction, to be used by the baseline correction command bcm
baslpnts	baseline points for spline baseline correction command sab, created interactively with the cursor
peaklist	peak information, generated by command ppp. To be used by the deconvolution command mdcon
reg	plot regions (same format as *intrng* files), used by the plot command if the parameter LIMITS=region. The plot limits are taken as the largest and smallest frequency values in the file

Table 5.10 *Miscellaneous* files types

may contain wildcard characters (see edpul and how to change the default text editor for details).

When such a file is generated by XWIN-NMR, it is stored in the current processed data directory

/DU/data/USER/nmr/NAME/EXPNO/pdata/PROCNO.

For example, after manual integration the regions are written into the file *intrng* in this directory. The user might want to have several *intrng* files available on disk, each with different regions. In order to save the current file, he uses the command wmisc to copy it to the directory

/u/exp/stan/nmr/lists/intrng/

under an arbitrary name. *intrng* is now used as the *file type*, and the directory may contain many range files that differ in their names, but have all the same structure. File types other than *intrng* are treated in a similar fashion. The command rmisc is used to copy a file from a miscellaneous files directory back to the current data directory, where it must be located for a corresponding XWIN-NMR command to find it.

edmisc may, of course, also be used to set up a file of these types manually. The preferred procedure, however, which ensures correct file formats, is to modify a

file that was generated by the appropriate XWIN-NMR command.

Examples of Format of miscellaneous files

1. **intrng** (Figure 5.5, generated by manual integration). The slope and bias

```
A 1.0 #regions in PPM
# low field    high field    bias slope
9.3210514094  9.1339532702 -6540000 0  # region 1
8.6147489426  8.4029001528 -7850000 0  # region 2
7.9758465528  7.8649580444 -4030000 0  # region 3
7.8097235415  7.7312765639 -6100000 0  # region 4
7.7006528774  7.5055841898        0 0  # region 5
```

Figure 5.5 Example of an *intrng* file, generated by manual integration

parameters are taken into account to baseline-correct the integrals as soon as they are calculated based on this file, either on the display or during the plot or printout.

2. **intrng** (Figure 5.6, automatically generated by command abs). No slope or bias

```
P 0
9.438358 8.147455
7.923863 7.144994
```

Figure 5.6 Example of an *intrng* file, generated by abs

parameters are present to correct the baseline of the integral. Automatic baseline correction is performed during integral plot or print based on this file, if the processing parameter INTBC=yes. 19 data points left and right of inegration intervals will then be used to determine the baseline. The 'P' at the beginning indicates ppm units. 'H' for Hertz units could also be used. A 'W' at the beginning would force the plotting or integral list command to consider the entire spectrum as a single integration interval. No ranges need be specified in this case.

3. **intrng** (Figure 5.7, set up by the user). Each interval is assigned a scaling factor by the user. When plotting or printing numeric integral values based on this

```
P 1.0
#low field  high field  weight
9.438358    8.547455   2.0
8.273884    7.523723   4.0
7.423863    7.144994   2.0
```

Figure 5.7 Example of an *intrng* file, set up by user

range file, they will be scaled such that their weighted sum will have the result specified in the first line (1.0 in this example).

4. **base_info** (Figure 5.8). The first line contains the function code for the baseline

```
0
12311
1666
3.153728e+07
5.040096e+03
1.909280e+00
1.627358e-03
8.669147e-07
```

Figure 5.8 Example of an *base_info* file

fucntion the user had selected to perform the manual baseline correction (e.g. 0=polynomial), the following two lines are file offsets defining the region to which baseline correction should be applied. The next lines contain the values of the coefficients A, B, C, D, E, for example A+Bx+Cx^2+Dx^3+Ex^4 in the polynomial case.

5. **baslpnts** (Figure 5.9). The first line contains the file size, the following lines the file offsets and corresponding ppm values within the spectrum from where the spline baseline correction command sab takes its supporting function values.

6. **peaklist** (Figure 5.10) The first line defines the frequency units (H=Hertz, P=ppm) of the line position and half width. In the right column, the gaussian percentage must be specified, if the peak list is used for mixed Lorentz-Gauss deconvolution (command mdcon).

```
16384
12438 7.699070
12591 7.677680
12737 7.657277
12892 7.635558
13034 7.615813
13361 7.570071
13653 7.529265
```

Figure 5.9 Example of an *baslpnts* file

```
H
#frequency    half width   %gauss/100.
2761.975      0.86         0.0
2760.684      0.86         0.0
2755.096      1.01         0.0
2753.802      0.86         0.0
2752.451      0.86         0.0
2746.895      0.86         0.0
2745.564      0.86         0.0
2739.110      2.47         0.0
```

Figure 5.10 Example of an *peaklist* file

5.3.10 Automation modules [edaumod]

Automation (AU) modules are, like AU programs, text files written in the C language, or using XWIN-NMR commands. However, AU modules can not be executed by typing their names on the command line, or via xau. Instead, they are called from other XWIN-NMR commands. For example, the AU module *qnmr_sx* is called by the XWIN-NMR command quicknmr, and the module *stan_sx* is called by the command run. edaumod allows you to create a new AU module, and to edit or view an existing one. When exiting from the text editor, the AU module must be compiled before it can be called. Details about writing AU modules can be found in the chapter on AU programs.

XWIN-NMR locates the C sources of AU modules in the directory

/u/exp/stan/nmr/au/modsrc/

and the executables in

/u/prog/<XWIN-NMR version>/au/modbin/

The command edaumod displays all AU programs from this directory in a dialog box from where the desired one can be selected. edaumod is identical to edau, with the difference that initially AU modules are displayed rather than AU programs. AU programs may be displayed by clicking on the *->Programs* command button at the bottom of the edaumod dialog box,.

On XWIN-NMR release media, a number of sample AU programs are delivered. After installation of XWIN-NMR, they are stored in the directory

/u/prog/<XWIN-NMR version>/au/modsrc.exam/

The command expinstall (see menu *Acquire -> Spectrometer setup*), which must be executed after installation of XWIN-NMR, will compile all sample AU modules if user requested, and install their C sources in the working directory

/u/exp/stan/nmr/au/modsrc/

and their executables in

/u/prog/<XWIN-NMR version>/au/modbin/

The command edaumod may also be entered on the keyboard, followed by the name of the desired AU program, or followed by an argument containing wildcard characters (see edpul and how to change the default text editor for details).

5.3.11 Data set info file [edinfo]

The purpose of this command is to setup or change the so-called information file *info*, a text file stored in the directory

/DU/data/USER/nmr/NAME/EXPNO/,

that is, in the same directory as the acquisition data (files *fid* or *ser*). The *info* file enables the user to store additional text information along with each acquisition data set or experiment.

The info file has the following internal structure. It consists of up to 20 *two-line* entries with up to 80 characters per line. Figure 5.11 shows an example.

The *info* file is displayed on the screen via the command edinfo. Only the *second line* of each entry may be edited by positioning the mouse cursor accordingly. The *first line* of each entry cannot be edited. Instead, these lines are predefined in a template, which is a text file called *info_item*, and which must be stored in the directory

/u/exp/stan/nmr/lists/.

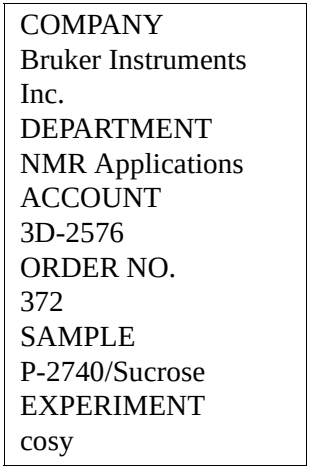


Figure 5.11 *Info* file example

The template file must be set up by the system administrator via a text editor. The template corresponding to the example in Figure Figure 5.11 is shown in Figure 5.12.

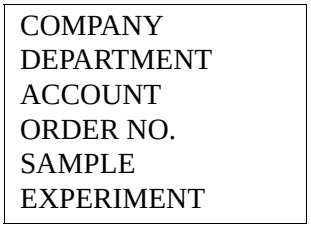


Figure 5.12 Template of an *Info* file

The *info* file can be plotted along with the data set. For this purpose, the plot parameter TITNAM (see command edg) must be set to the full path name of the *info* file, or simpler to *TITNAM=../../info*. By default, TITNAM is set to *title*, which is the title file set up via the command setti. If both, the default title and the *info* file should be plotted, you must execute a first plot using the command plots, containing e.g. spectrum, parameters, and title. Afterwards, a second plot must follow, again executed via plots, in order to put it on the same sheet of paper. For the second plot, however, spectrum and parameter output must be disabled (command edg). Only the title plot may be enabled, with *TITNAM=../../info*. The command

flplot will flush both queued plots out to the printer/plotter. Please refer to the *Output->Plotting* menu for more information on the various plot commands.

Frequently, the user has stored a large number of data sets on his disk. The *info* file may also be used to identify a particular data set by searching for an item contained in the info file. In the example above, the sample identification number is P-2740. Lets assume the user's login name is *guest*, and his data are stored in the disk partition */u*. Then, the following command would search in all *info* files of the user's data sets for the sample id

grep "P-2740" /u/data/guest/nmr/*/*/info

The two asterisks indicate that all existing data set NAMEs and EXPNOs should be included in the search. The command must be given at Unix level, usually from an Xterm window outside XWIN-NMR. If the search returns without any information, the sample id was not found. Otherwise, the complete path names of all *info* files are displayed containing P-2740. In this way, the user obtains a list of all experimental data with this sample id.

5.4 Recall last

XWIN-NMR remembers the names of the last data sets you have worked with during your session, and collects them in a table.

The first entry displays this table. Clicking on one of the data sets will make it the current data set and show it on screen. The other entries allow you to quickly return to the last used data set with a higher or lower dimension. This latter feature is frequently used when examining rows of a 2D data set, or planes of a 3D spectrum.

5.5 Search

The command search is an alternate way of finding and selecting a data set for display. It is an important tool for getting an overview of the data sets stored in different disk partitions for different users, and to switch rapidly between data sets.

5.5.1 Data set specifications

Directory
This field contains the list of directories where XWIN-NMR data sets are searched for. Assume you want to process data stored in the two directories */usr/people/guest* and */u*. Then you have to add these directories to the list by opening the *Edit* menu and selecting the command *Edit directory list*. Enter a directory name and click on the *Add* button. When you are finished setting up the list, click on *OK*. Now, whenever you click on one of the entries in the directory field, e.g. on */usr/people/guest*, XWIN-NMR will search for a subdirectory named *data*, and will display all data sets of all users located there.

User
Click on a user id to display all data set NAMES of this user in the *Name* column.

Name
Click on a name in this column to display all experiments stored under this name in the *Expno* column.

Expno
Click on an experiment number to display all processed data sets belonging to this EXPNO in the *Procno* column.

5.5.2 Command buttons

Append
Click on this button to append the selected (highlighted) data set (USER/NAME/EXPNO/PROCNO) to the end of portfolio.

Insert
Similar to *Append*, but inserts the selected data set before the highlighted portfolio entry.

Remove
Remove the highlighted portfolio entry.

Apply
Click on this button to make the highlighted portfolio data set the current XWIN-NMR data set, and display it. Alternatively, double-click on the desired portfolio data set.

5.5.3 Commands in the File menu

Load
Load a portfolio stored via the *Save* commands.

Save
Store the portfolio contents in a file of your choice.

Save as default
Store the portfolio contents in the file *default.por*, which is located in the subdirectory *.xwinnmr* of your home directory.

Reset
Resets the contents of the data set windows (Unit/User/Name/Expno/Procno) to the first data set of the logged in user.

Quit
Terminate search command and close window.

5.5.4 Other functions

You may resize the entire search window to give more space for data sets. You may also increase the size of the data set columns at the expense of the portfolio, or vice versa. Click on the button located at the right side of the line that separates the portfolio from the first part of the window. Keep the left mouse button depressed and move the mouse until you get the desired effect.

5.6 Copy

The *Copy* entry of the *File* menu (Figure 5.13) provides a number of commands to make copies of entire data sets or parts of it, i.e. of processed data or of acquisition data only. In addition parameter and other files used in XWIN-NMR can be copied to other directories for saving and retrieving.

5.6.1 Entire Data Set [wrpa]

A copy of the current data set is created. The user is prompted for the new name of the copy. The entire data set, including acquisition and processed data, are copied. The destination must be specified in the form

NAME EXPNO PROCNO DU USER

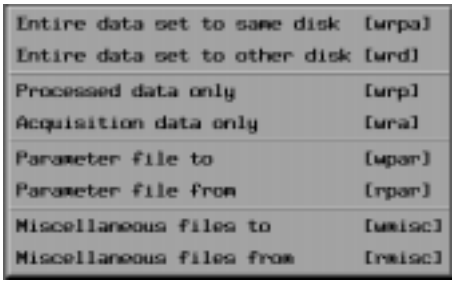


Figure 5.13 The *File->Copy* submenu

It is legal to omit parts of these specifications, which would cause the corresponding items to be taken over from the current data set. For example, if only NAME is entered, the copy will get a new name, but EXPNO, PROCNO, DU, and USER will not change.

You can also enter the command at the keyboard, optionally followed by arguments. The following options are available.

wrpa NAME EXPNO PROCNO DU USER y
The destination data set is completely specified. If the 'y' at the end is omitted and a data set of this specification already exists, XWIN-NMRwill ask whether to overwrite that data set or not. Specifying the 'y' at the end suppresses this question, and an existing data set will be overwritten unconditionally. By specifying a DU other than the current one, the data set may also be copied to another disk. A more convenient method to do this is provided by the command wrd (see below).
Example: wrpa sucrose 5 2 /v guest2

wrpa NAME EXPNO PROCNO USER
The destination data set is specified without DU and will therefore be created on the same disk partition as the current data set. If a USER other than the current one is specified, the new data set will get the ownership of the new user. Changing the user is only possible if the current and the new user are in the same user group (see your operating system documentation for users and groups).
Example: wrpa sucrose 5 2 guest2

wrpa NAME EXPNO PROCNO
The destination data set is specified without USER and DU and will therefore be

created under the same user name and on the same disk partition as the current data set. You should invoke the command in this form if not only NAME, but also EXPNO and PROCNO are to be changed when copying.
Example: wrpa sucrose 5 2

wrpa NAME EXPNO
Please use this command if you want to make a copy that differs only in NAME and EXPNO from the current data set.
Example: wrpa sucrose 5

wrpa NAME
Please use this command if you want to make a copy that differs only in the NAME from the current data set.
Example: wrpa sucrose

wrpa EXPNO
Please use this command if you want to make a copy of the current data set, and the copy should bear the same NAME, but will have been assigned the new EXPNO. The program can differentiate this command from the above one (wrpa NAME), because EXPNOs consist entirely of digits, while NAMEs *must* begin with a letter.
Example: wrpa 5

5.6.2 Entire data set to other disk [wrd]

This command is for example useful for archiving a data set on magneto-optical disk, or another device with the directory structure of a magnetic disk. While this task could also accomplished via the command wrpa described earlier, wrd is simpler to use. When entered at the keyboard, the following form is also legal.

wrd DU
In this case, the current data set is copied to the specified disk unit, and will be stored there under the same NAME, EXPNO, PROCNO, and USER.
Example: wrd /v, where /v specifies another disk (or disk partition).

5.6.3 Processed data only [wrp]

This command creates a copy of the processed data (files *1r, 1i, 2rr, 2ii, ...*, and parameter files) of the current data set under a new PROCNO, which is prompted for by the program. If the command is entered at the keyboard, additional argu-

ments may be specified:

wrp PROCNO y
The destination PROCNO is typed in explicitely. The 'y' argument tells the program to overwrite any processed data with the same PROCNO. If 'y' is omitted, the user will be prompted for a confirmation if such data exist.
Example: wrp 2

5.6.4 Acquisition data only [wra]

This command creates a copy of the acquisition data (files *fid* or *ser*, and parameter files) of the current data set under a new EXPNO, which is prompted for by the program. If the command is entered at the keyboard, additional arguments may be specified:

wra EXPNO y
The destination EXPNO is typed in explicitely. The 'y' argument tells the program to overwrite any processed data with the same EXPNO. If 'y' is omitted, the user will be prompted for a confirmation if such data do exist.
Example: wra 5

Copying acquisition data requires that the program creates a new EXPNO directory under the same NAME as the current data set. Since PROCNO directories are subdirectories of EXPNO directories, the copied data set will also have a PROCNO identical to that of the current data set without data files, but with parameter files. In this respect, wra and wrpa are different. After wrpa, the PROCNO subdirectory would additionally contain the processed data files.

5.6.5 Copy parameter files to ... [wpar]

XWIN-NMR stores acquisition data (*fid* and *ser* files) in the directory
/DU/data/USER/nmr/NAME/EXPNO/,
and processed data (*1r, 2rr, 3rrr* etc. files) in the directory
/DU/data/USER/nmr/NAME/EXPNO/pdata/PROCNO/.

The acquisition directory contains (depending on the dimension of the data) the acquisition parameter files *acqu* and *acqu2*. The processed data directory contains the processing parameter files *proc, proc2, proc3*, the plot parameter files *meta* and *meta.ext*, and the output device parameter file *outd*.

Even if no data (fid or spectrum) exist yet, the data directories already contain

parameter files. They are generated by the command new when the data directory is created, and are taken from the last data set used.

The purpose of the command wpar is to save these parameter files in a special directory. As such a parameter set (=set of parameter files) describes an entire experiment, including acquisition, processing, and plotting, it can later be used to retrieve it from the save directory to perform an NMR experiment. The storage directory is

/u/exp/stan/nmr/par/<parameter set name>/.

The name can be chosen by the user. wpar displays a dialog box containing all currently existing parameter sets. You can overwrite one, or define a new one by entering a new name. After closing the dialog box, wpar allows you to select which of the parameter files (*acqu*, proc*, meta*, outd*) are to be copied to the parameter set directory.

In order to set up an and save an experiment, proceed as follows: define the acquisition parameters using the command eda, define the processing parameters with edp, the plot parameter with edg, and the output parameters with edo. Execute the experiment to verify all parameters. Then save everything via wpar. The saved parameter files can be restored with the command rpar.

wpar may also be entered on the command line, followed by arguments, e.g.

wpar cosy all

would copy all parameter files of the current data set to the directory

/u/exp/stan/nmr/par/cosy/.

In addition to the keyword *all*, you could also use any of *acqu, proc, plot, outd* in order to copy acquisition, processing, plot, or output device parameters only.

It is also legal to use wildcard characters. For example, the command

wpar C*

would display all existing parameter sets beginning with the letter C. The command

wpar [H-Z]*

would display all existing parameter sets beginning with the letters H,I,J,...,Z.

5.6.6 Copy parameter files from ... [rpar]

The purpose of the command rpar is to overwrite the parameter files *acqu*, proc*, meta**, and *outd* of the current data set by parameter files contained in the directory

/u/exp/stan/nmr/par/<parameter set name>/,

which were stored there via the command wpar earlier by the user, or which have been installed using the command expinstall from the XWIN-NMR release (the so-called Bruker standard parameter sets). rpar displays a dialog box containing all currently existing parameter sets. After having selected a name, rpar allows you to specify which of the parameter files (*acqu*, proc*, meta*, outd*) should be copied to the current data directory, where existing parameter files are overridden. rpar will not destroy any data of the current data set, nor will it overwrite the so-called status parameter files *acqus* and *procs*, which describe the acquisition and processing status of existing data files.

rpar may also be entered on the command line, followd by arguments, e.g.

rpar cosy all

would copy all parameter files from the directory

/u/exp/stan/nmr/par/cosy/

to the current data directory. In addition to the keyword *all*, you could also use any of *acqu, proc, plot, outd* in order to copy acquisition, processing, plot, or output device parameters only.

It is also legal to use wildcard characters. For example, the command

rpar C*

would display all existing parameter sets beginning with the letter C. The command

rpar [H-Z]*

would display all existing parameter sets beginning with the letters H,I,J,...,Z.

When a parameter set has been copied to the current data directory, you can use the commands eda, edp, edg, edo, or the commands for setting individual parameters, to apply modifications.

Please note:
Parameter sets are either of type 1D, 2D, or 3D. XWIN-NMR identifies the type by looking at the plot parameter file *meta*, which contains the corresponding information. The presence of a 1D *meta* file in the current data directory will cause the 1D screen layout to be displayed, etc. Suppose your current data set is a 1D data set containing fid and spectrum data, and you invoke the command rpar to copy a 2D parameter set to your current data directory. This will turn your current data directory to 2D. In order to maintain a consistent data set, XWIN-NMR will ask you whether to delete the 1D data. If you confirm the question, all data files of the data set are deleted, the 2D parameter set is copied, and the 2D screen layout appears. Now, your current data set is a 2D data set without data files (fid or spectrum), but

with 2D parameter files corresponding to the chosen parameter set, and can be used as basis for a new 2D experiment.

5.6.7 Copy miscellaneous files to ... [wmisc]

The meaning of *miscellaneous files* is presented in the description of the command edmisc. The purpose of wmisc is to make save copies of miscellaneous files for later retrieval with the command rmisc. For example, you can integrate a spectrum and copy the *intrng* file, which contains the regions, to the directory

/u/exp/stan/nmr/intrng/

where it should get the name *intrng.1*. After calling wmisc, the available file types, *intrng, base_info, baslpnts, peaklist*, *reg* are displayed. Select the type *intrng*. The program will then show all files which are already present in the directory

/u/exp/stan/nmr/intrng/.

If you select one of those, it will be overwritten by the *intrng* file of the current data set. Alternately, you can specify a new name, e.g. *intrng.1*.

5.6.8 Copy miscellaneous files from ... [rmisc]

The meaning of *miscellaneous files* is presented in the description of the command edmisc. The purpose of rmisc is to copy a saved miscellaneous file from the directory

/u/exp/stan/nmr/<misc. file type>/

to the current data directory. rmisc is the inverse command to wmisc.

5.7 Delete Data Files

The *Delete* menu (Figure 5.14) provides commands to delete data files, parameter files, and other files used in XWIN-NMR. The delete commands display a dialog box where the data sets or other files to be deleted can be marked by clicking on them. In order to actually delete the marked files, the *Execute* button at the bottom of the dialog box must be activated.

When deleting data sets (other than parameter files etc.), the bottom of the dialog box contains a button *MODE*, which is a toggle command for the two delete modes *delete data + parameters*, and *delete data only*. Remember that an XWIN-NMR data set is a directory structure, containing data and parameter files. If the first mode is enabled, data and parameter files are deleted with the consequence, that the whole

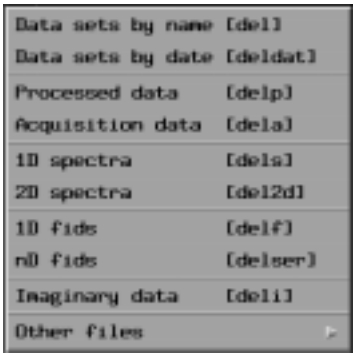


Figure 5.14 The *File->Delete* submenu

directory is removed and the corresponding data set no longer exists on disk. In the second case, only the data files are erased, while the parameter files are retained. A data set (directory) therefore still exists after executing the delete command, without any data, but containing parameter files which may be used, e.g. for acquiring new data under the same name.

Deleting data sets

del	display list of *data set names* (NAME parameter) of the current USER, stored in the user's data directory (*/DU/data/USER/nmr/*)
deldat	like del, but data sets are ordered according to their acquisition dates
delp	processed data, any dimension (e.g. *1r,1i*- and *2rr, 2ii, ...* -files)
dela	acquisition data, any dimension (*fid*- and *ser*-files)
dirs	processed one-dimensional data (*1r, 1i*-files)
del2d	processed two-dimensional data (*2rr, 2ii, ...*-files)
delf	one-dimensional fids (*fid*-files)
deli	imaginary processed data, any dimension(e.g. *1i, 2ii*)

Deleting imaginary data can save a substantial amount of disk space. After deleting the imaginary part of spectra, phase correction is no longer possible. You can get the imaginary part back by re-transforming the acquisition data, or, if those are no longer available, by applying a Hilbert Transform to the real data (see processing commands ht, xht1, xht2).

Please note that all del type commands can be used with *wildcard* arguments, such as del *, del suc*, del ??abc*, etc, which allow you to choose data sets according to a desired mask. The asterisk serves as a placeholder for a group of unknown characters, the question mark for a single character.

5.8 Delete Other Files

Figure 5.15 shows the commands to delete files other than data sets.

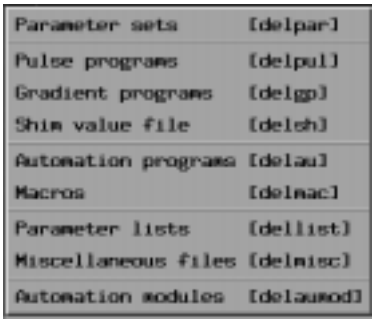


Figure 5.15 The *File->Delete->Other files* submenu

The purpose and structure of these files have already been described in the section *Open Other Files*. Please note that also these commands allow you to use wildcard characters when entered on the keyboard, e.g. delpul cosy*. If a command is executed from the menu, or typed in without argument, a list of existing files is displayed.

delpar	Delete parameter set
delpul	Delete pulse program
delgp	Delete gradient program
delsh	Delete shim value file
delau	Delete automation (AU) program. Source file and compiled (executable) file will be deleted.
delmac	Delete command macro file
dellist	Delete parameter list file
delmisc	Delete file of *miscellaneous* type, e.g. saved

	intrng (integral range) file
delaumod	Delete automation module. Source file and compiled (executable) file will be deleted.

5.9 Rename

The commands of Figure 5.16 allow you to assign new names to data sets and

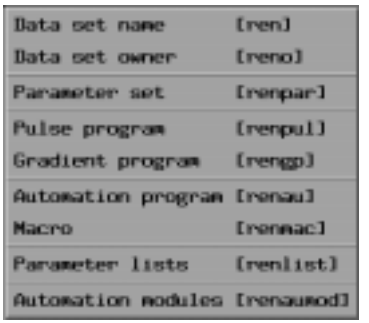


Figure 5.16 The *File->Rename* submenu

other files.

The rename commands display a list of data sets or other files. You can edit each name in the list using the cursor. Renaming is complete as soon as you terminate your change with the Return key.

ren	Change a data set NAME
reno	Change a USER name. The command displays a list of USERs, i.e. all subdirectories of /DU/data/. You can now change a name in the list. For example, if 'guest' appears in the list, and you change it to 'newguest', your data directory will be renamed from */DU/data/guest/* to */DU/data/newgues*t.
renpar	Change the name of a parameter set
renpul	Change the name of a pulse program
rengp	Change the name of a gradient program

renau	Change the name of an automation (AU) program
renmac	Change the name of a command macro
renlist	Change the name of a parameter list file
renaumod	Change the name of an automation module

5.10 Send / Receive

The *Send* submenu contains commands for transferring data sets via rcp (remote copy) or as email.

The command names shown in brackets [], may be used to quickly access the function from the keyboard.

5.10.1 Send or fetch data via rcp [tcps]

The tcps command allows you to transfer files or whole directories to a remote computer, which is connected via a network to your system. In order for tcps to work as expected the TCP/IP network environment must have been installed correctly by the system administrator. For example, a frequent problem when transferring files is that access permissions to the remote computer are not set correctly.

The command tcps calls the TCP/IP rcp utility internally. Therefore, only destinations providing an rcp server can be accessed.

tcps opens a dialog window and invites you to enter the host name of the remote computer. You can then enter the file or directory name you want to transfer. Since the wild card characters '?' and '*' may be used, more than one file or directory can be sent at once. If you want to send a complete data set, you must enter its directory name, e.g. */u/data/guest/nmr/sucrose*. The next step is to type in the destination file or directory name. For example, if you enter */u/data/guest/nmr*, the data set *sucrose* would be placed in the same directory on the host as it is stored on your local system.

The receive command is the inverse of tcps. The direction of the data transfer is from the remote to the local system.

5.10.2 Send data as email [smail]

The command smail calls tojdx and the MIME-compliant mail user agent (MUA),

zmail, internally. In order for zmail to work as expected the sendmail program must be running.

smail opens a dialog window and invites you to enter the JCAMP conversion parameters. You have to type the following electronic mail parameters as well:

- *to-list*: email address in the form user@address.
- *cc-list*: list of users to receive a carbon copy
- *subject*: subject
- *attachment*: filename will be added as an attachment to the outgoing message
- message text.

If you click on the button *SEND* the program will start the JCAMP conversion and mailing according to the parameters you have defined.

Click on the *Last Mail* command from Report menu to see whether your electronic mail was sent successfully.

If you want to start smail without dialog windows, you may type the following on Xwin-nmr's command line:

smail out=/usr/people/hb/exam1d.dx data=0 mode=3 ti=exam1d or=hb ow=hb addr=hb@bruker.de sub=Test_smail att=/usr/people/hb/exam1d.dx <cc=tgk@bruker.de>

Parameter cc=... is optional.

The tojdx parameter are:

- out=/usr/people/hb/exam1d.dx output file: name of output file
- data= 0 output type
- mode=3 compression mode
- ti=exam1d title
- or=hb origin
- ow=hb owner

The zmail parameters are:

- addr=hb@bruker.de to_list
- sub=Test_smail subject

- att=/usr/people/hb/exam1d.dx name of the attachment file eg. DX file
- cc=tgk@bruker.de cc-list

5.11 Conversion

Conversion commands are used to import data into XWIN-NMR and to export XWIN-NMR data into other formats (Figure 5.17).

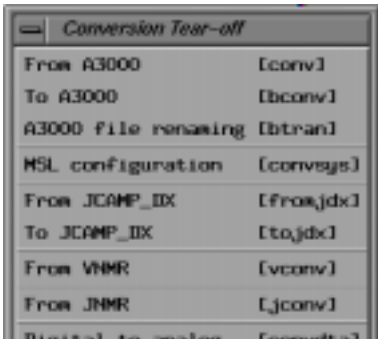


Figure 5.17 The *File->Conversion* submenu

conv and btran convert data sets, which were acquired or processed with the A2000/3000 computer series, into the XWIN-NMR format.

bconv converts an XWIN-NMR data set into the A2000/3000 format.

tojdx converts XWIN-NMR data sets into JCAMP-DX data format, which is a manufacturer independent data format.

fromjdx reads JCAMP-DX data sets and converts them into the XWIN-NMR format.

vconv converts Varian VNMR data sets into the XWIN-NMR data format.

5.11.1 Conversion between Aspect 2000/3000 and XWIN-NMR format

In principle, there are two ways of transferring data from a spectrometer equipped with an Aspect 2000/3000 to a workstation running XWIN-NMR. You can get them directly, for instance via Ethernet using the BRUKNET program. Or you get them

indirectly, for instance via a PC using NMRLINK. Other transfer programs will work in a way similar to BRUKNET or NMRLINK.
On the Aspect 2000/3000, data files are stored in a 24 bit per word format. Each word consists of three bytes in the order *high-middle-low*. A parameter sector is stored together with the data files. This parameter sector is called the -1 sector.
BRUKNET can run in four different modes which determine how the data are stored on the receiving computer. The file *SPEC.001* transferred by the user *guest* from a spectrometer called *am400* will have a different name and size on the receiving computer depending on which transfer mode of BRUKNET is used.

- Binary mode enabled, special parameter file handling disabled.
  The data are changed from 24 bit to 32 bit per word storage. The -1 sector and the data are stored in one file. The filename on the receiving computer is

  /u/bruknet/am400/guest/SPEC.001+

- Binary mode enabled, special parameter file handling enabled.
  The data are changed from 24 bit to 32 bit per word storage. The -1 sector is stored in a separate file which has the same basic name as the data set but starts with the prefix p_. The filenames on the receiving computer are

  /u/bruknet/am400/guest/SPEC.001+

  /u/bruknet/am400/guest/p_SPEC.001+

- Binary mode disabled, special parameter file handling disabled.
  The parameters and the data are stored in one file in the 24 bit per word format. The filename on the receiving computer is

  /u/bruknet/am400/guest/SPEC.001

- Binary mode disabled, special parameter file handling enabled.
  The -1 sector is stored in a separate file which has the same basic name as the data set but starts with the prefix p_. Both files are stored in the 24 bit per word format. The filenames on the receiving computer are

  /u/bruknet/am400/guest/SPEC.001

  /u/bruknet/am400/guest/p_SPEC.001

All four transfer modes swap the bytes within one word. So the byte order after the transfer will be *low-middle-(middle)-high*. The + sign indicates the change from 24 bit to 32 bit per word storage.

NMRLINK has two transfer modes.

- The data can be transferred without expansion from 24 bit to 32 bit per word. In this case, the files are not byte swapped.
- The data can be transferred with expansion from 24 bit to 32 bit per word. These files are byte swapped during the transfer. Therefore, the file format is the same as for BRUKNET (running in binary mode and special parameter file handling disabled).

When using NMRLINK, the receiving computer is a PC and the files will be stored in lower case letters. Renaming into upper case letters or appending a + sign is not necessary because conv recognizes all different file formats correctly.

The conversion of A3000 extended parameter files works, for data which were transferred with NMRLINK in 32 bit transfer mode. The conversion also works, if the data were tansferred in 24 bit mode and the extended parameter file was transferred in 32 bit mode. If the transfer of the data was done in 32 bit and the extended parameter file was transferred in 24 bit, then the extended parameters will NOT be converted. This last case cannot be implemented because it interferes with the conversion of files first transferred with NMRLINK to a PC and then to a UNIX station. In such a case, the file name convention changes. 32 bit files lose the „+“ sign at the end.

5.11.1.1 Configuration files

The number and type of components with which a spectrometer is equipped determines its software configuration. If you use the DISNMR software, this configuration information is stored in the file *DISNMR.CONF*. If you use the DISMSL software, the configuration information is stored in the file *DISMSL.SYST*. The commands conv and btran need to know the spectrometer configuration for a correct conversion of the parameters SW and OFFSET.

For DISNMR files, get the *DISNMR.CONF* file from your spectrometer. Become superuser and do the following.

1. Transfer the file from the spectrometer, e.g. an *am400*, to your workstation. The file will be put into */u/bruknet/am400/root/DISNMR.CONF+*.
2. Run the command */u/prog/< XWIN-NMR version>/mod/config*.
3. Answer the question for the station name, e.g. with *am400*. This converts */u/bruknet/am400/root/DISNMR.CONF+* to */u/conf/instr/am400/disnmr.conf*.

4. Leave the superuser mode.
You can now get and convert data files from the spectrometer.

For DISMSL files, run the command convsys in XWIN-NMR. This will generate the file /u/conf/instr/<station>/dismsl.conf. You do not need to transfer the file *DISMSL.SYST* from the spectrometer. The convsys command will ask you for the spectrometer name and its basic spectrometer frequency.

No configuration file is required if you want to convert files from a spectrometer running TOMIKON.

5.11.1.2 conv

conv will ask you for the station name.

1. Enter the name of the station, e.g. *am400.*

conv will then ask for a file name.

2. Enter one of the following.

- a file name (without the + !) to convert the file.
- ? to display a list of all files of the station available for conversion.
  You select a file by clicking on its name.
- * to convert the next available file.
  If no file is found, the program will wait until a file has been transferred. The AU program remproc makes use of this feature.
  If more than one file is present, all files will be converted, but not necessarily in the order in which they were transferred.

The conversion stores the station name into the parameter INSTRUM. You can examine this parameter with commands such as dpa, lpa or 2s instrum.

If data have been acquired through DISNMR automation (DISNMR command RUN), they will have a name in the form *YYYY0F.<nnn>*. There will also be a so called extended parameter file *YYYY0X.<nnn>*.

Important note:
Make sure that the extended parameter file is sent first, especially if you use automatic transfers from the spectrometer to the workstation. If both files are sent to the workstation for conversion, conv will first convert the extended parameter file.

The DISNMR extended parameter file contains the parameters NUCLEUS and SOLVENT which will be stored into the corresponding XWIN-NMR parameter files. If the extended parameter file is not available or if it was not transferred, the parameters SOLVENT and NUCLEUS are set to *unknown*. You can set them with the command nuso. Both parameters are needed for automatic spectrum calibration with the processing command sref.

A data set converted by conv will be stored as the following XWIN-NMR data set:

- DU is set to the disk partition on which the foreground XWIN-NMR data set is stored. If you want to store the converted data on a different disk, you must change to a data set on that disk. Alternatively, you can use the command btran for conversion.
- USER is set to the USER name of the foreground data set. Only files which were transferred by that user can be converted. btran allows you to convert data sets into a different USER directory.
- NAME is set to the file name of the Aspect 2000/3000 file without the file name extension.
- EXPNO is set to the file name extension of the Aspect 2000/3000 file.
- PROCNO is set to 1.

Example :

A file *AP010F.104* was transferred from the spectrometer *am400* by the user *guest*. It will be stored on the workstation in :

/u/bruknet/am400/guest/AP010F.104.

For the conversion, the user *guest* must run XWIN-NMR from one of his data sets, e.g.

DU *z*, USER *guest*, NAME *test1*, EXPNO *1*, PROCNO *1*.

The converted file will be stored as

DU *z*, USER *guest*, NAME *AP010F*, EXPNO *104*, PROCNO *1*.

conv is capable of converting different types of data sets by trying to determine the data type from the name of the transferred file (e.g. .SER, .SMX, .IMAG, .I001, .P001). If the type of data set cannot be determined, the programs will display a list of known data set types and you have to select the type from the list.

Conversion of MSER files (multi-slice/multi-echo data sets)

conv expects that these files are stored, after the transfer, under a name in the form NAME.MSER(+). You either have to rename the data set before you get it from the Aspect 2000/3000 or after the file was stored in the directory */u/bruknet/...*.

Example:

A multislice/multiecho experiment is stored by DISNMR in the file *RAT.SER*. After renaming the file to *RAT.MSER* and initiating the file transfer in binary mode as user *guest* from a spectrometer *am400* , the file is stored on the workstation as

/u/bruknet/am400/guest/RAT.MSER+

After the conversion, a data set of this type is stored like other 2D-data sets on the disk and has an EXPNO (experiment number) calculated from the formula :

EXPNO(IS, IE) = (IS-1)*NE + IE

IS : is the current slice number
IE : is the current echo number
NE : is the number of echo pictures

Each of these raw data sets can be transformed to an image with the command xfb. A series of images can be calculated with a suitable automation program.

Conversion of RAWD files

Another special case are n-dimensional raw data sets, acquired with the Aspect 3000 TOMIKON program. conv expects that these files are stored under a name in the form *NAME.RAWD(+)* and that the additional parameter files *NAME.PVAR(+)* and *NAME.ACQP(+)* have been transferred. They contain patient information and acquisition parameters. There are special Aspect 3000 PASCAL-programs available to create these parameter files in TOMIKON.

Conversion of Image files

Another special case are TOMIKON image files, structured as follows:

<NAME><anumber>.<class><inumber>

- <NAME>
  normally the name of a patient, which will become the name of the XWIN-NMR dataset
- <anumber>
  An optional number appended to the name, which will become the EXPNO (otherwise EXPNO is set to 1)
- <class> is a single character image class
- <inumber> is a 3-digit image number

For the different image classes, the EXPNO and PROCNO is determined in the following way:

class = I : EXPNO = <anumber>, PROCNO = <inumber>

An exception are images of multi-oblique examinations. For every subpackage a different EXPNO is taken :

EXPNO = <anumber> * 10000 + number of subpackage.

class = P : PROCNO = <inumber> + 1000
class = D : PROCNO = <inumber> + 2000
class = Q : EXPNO = <anumber> * 10000 + <inumber> + 3000,
PROCNO is the image number, stored in the original file.
class = X : PROCNO = <inumber> + 4000
class = Y : PROCNO = <inumber> + 5000
class = Z : PROCNO = <inumber> + 6000
class = S : EXPNO = <anumber> * 10000 + <inumber>, PROCNO = 1

The file *2dseq* will be created.

Editing image conversion parameters

The command edit_conv allows you to edit the parameters which control the conversion of Aspect 3000 image files to XWIN-NMR format.

The Aspect 3000 TOMIKON program normally produces single image files. The image files of one experiment can be converted either to single image files named *2rr* with different processed data set numbers PROCNO, or to one multi-image file named *2dseq*.

The TOMIKON 3D reconstruction produces one 3D image data file. This file is

converted to a *2dseq* file.

The Aspect 3000 DISNMR and DISMSL programs produce files with the extension .IMAG. They may contain several images. The format is similar to the 3D image data file of TOMIKON. The data can be converted either into *2rr* files with different processing numbers for every slice or to one *2dseq* file.

If you create

- a *2rr* file, the pixels are converted into 32-bit words.
- a *2dseq* file, the pixels can be converted into unsigned bytes, 16-bit words or 32-bit words.

The file */usr/local/lib/bruknet/conv.par* contains the parameters which determine the type of the output file. You must create this file as superuser.

5.11.1.3 Back conversion - bconv

Certain XWIN-NMR data sets can be converted back to DISNMR format with the command bconv. The syntax of bconv is:

bconv <station name> <data type> .

If you type :

- bconv with both arguments, the conversion is immediately started.
- bconv <station>, you will be asked for the data type.
- bconv without any argument, you will be asked for the station as well as the data type.

Example: *bconv am300 fid*

Depending on which data type you specify, bconv either converts the acquired data or the processed data of the current data set together with the corresponding parameters. Currently, the data types according to Table 5.11 are implemented. Please note that back conversion of 2D and 3D transformed spectra is not supported.

The conversion writes two files into the current network directory. Its base name must be specified in the file */usr/local/lib/bruknet/destination*, e.g. */u/bruknet*. The station name specified with bconv and the user name under which the current data set is stored are appended.

data type	file to be converted
fid	*fid* (1D acquisition data)
ser	*ser* (multi dimensional acquisition data)
proc1d	*1r* and *1i* (1D processed data)
2rr	single image data file
2dseq	multi image data file

Table 5.11 Data types supported by back conversion

Example: bconv am300 fid on the data set :

NAME=*sucrose*, EXPNO=*2*, PROCNO=*3*, DU=*u*, USER=*guest*

generates the two files *SUCROSE.002+* and *p_SUCROSE.002+* in the directory */u/bruknet/am300/guest/* . The first file contains the fid, the second file the spectral parameters. The plus sign indicates that the files are stored in 32-bit format.

The files can now be sent to the destination station, e.g. with BRUKNET. Make sure that BRUKNET runs with special parameter file handling enabled and in binary mode.

The parameters, particularly the measurement frequencies, correspond exactly to the configuration of the station stored in

/u/conf/instr/<station>/disnmr.conf

If bconv cannot find the *disnmr.conf* file, all frequencies correspond to those of the current AMX or Avance configuration.

Back conversion of 2rr image data files

bconv supports conversion of images into the Aspect 3000 TOMIKON format. Contrary to *fid*, *ser* or *proc1d* types, no special parameter file will be created. Therefore, data must be transferred with special parameter file handling disabled.

bconv creates an output file using the following naming convention :

<NAME><EXPNO mod 1000>.<class><PROCNO mod 1000>

where class is the image class depending on EXPNO and PROCNO:

class = I : default
class = P : if (PROCNO / 1000) = 1
class = D : if (PROCNO / 1000) = 2
class = Q : if (PROCNO / 1000) = 3 and in versions later 930101
if ((EXPNO mod 10000) / 1000) = 3
class = X : if (PROCNO / 1000) = 4
class = Y : if (PROCNO / 1000) = 5
class = Z : if (PROCNO / 1000) = 6

Back conversion of 2dseq image data files

bconv supports conversion of these images into the Aspect 3000 TOMIKON format. Contrary to *fid*, *ser* or *proc1d*, no special parameter file will be created. Therefore, data must be transferred with special parameter file handling disabled.

bconv creates an S-image if (PROCNO/1000) = 10, and, in versions newer than 930101, if (EXPNO/1000) = 10.
bconv creates an output file using the following naming convention :

<NAME><EXPNO mod 1000>.S<PROCNO mod 1000>

Otherwise, a TOMIKON 3D image file will be created and the following naming convention is used :

<NAME><EXPNO mod 1000>.<PROCNO mod 1000>S

5.11.1.4 Flexible conversion with Bruktran

Introduction
The *Bruktran* program (command btran) provides a flexible method to convert data from an Aspect 2000/3000 (referred to as Aspect in this description) into the XWIN-NMR data format.
You can convert data from any station and any user. You can also individually specify the destination disk or user for the converted data.

Bruktran also allows you to rename files thus enabling you to make use of the 14 characters available for a filename. Renaming can be done either automatically or for each set of files with the same experiment name. To rename files automatically, the program needs to know the filename formats for both the Aspect filename and the XWIN-NMR filename. Using these formats, it is possible to extract a group of characters from the Aspect filename, check these characters

against a list of expanded tokens and place the expanded token in the XWIN-NMR filename.

Automated renaming is only really useful for those users who name their Aspect files in a specific manner. Other features of the program include date stamping of a filename (where the date of acquisition is inserted in the XWIN-NMR filename) and conversion and transfer to a Unix workstation user specified in the Aspect filename.

Files used in the Bruktran program

For correct operation of the Bruktran program, the following files should reside in the home directory of the user:

- *Btran_defs*
  This file contains the program settings such as current BRUKNET station and user, Aspect and XWIN-NMR filename formats, etc. This file can be regenerated with default values by selecting the *Reset Defaults* button.
- *Bt_tokens*
  This file contains the tokens (elements which constitute the Aspect filename) and their replacement values in the XWIN-NMR file name.

Permissions required by the user

Since BRUKNET data directories are set up with read/write privileges for all users, there is no special permission required by the user to convert data. However, if you wish to convert data sets and copy them to data directories of another user, you must have permission to do so (i.e. you must belong to the same user group).

Bruktran program options

During program execution, the Bruktran program accesses various parameters stored in the Btran_defs file. These parameters include the current BRUKNET directory path, the Aspect and XWIN-NMR filename formats etc. Parameters may be accessed by clicking on the Edit Program Options menu item in the main Bruktran window and are described below:

- Current Destination Directory.
  This specifies the BRUKNET destination of data directory path. The default value is obtained from the contents of the file:
  /usr/local/lib/bruknet/destination.

- Current BRUKNET Station.
  This sub-directory refers to the station from which the data was sent.
- Current User.
  This sub-directory contains the transferred data files.
- Aspect File Format.
  This string defines the way in which the tokens (groups of characters) are arranged to form the filename. A token is defined by a set of digits (...111.., ..22...., ...33.. etc) and each digit refers to a particular token. The digits 0 and 9 have special meanings (see Bruktran automated renaming). The length of the format string must not exceed 8 characters.
- X32 File Format.
  As above. The length of the format string must not exceed 14 characters.
- Owner Of The Converted Data Set.
  This specifies the user directory into which files will be copied after conversion. This parameter is globally defined although the destination directory may be edited for individual files (with preview mode selected).
- Rename Enable Flag.
  This globally defined parameter determines whether the file is to be renamed from the Aspect filename to the X32 filename. In preview mode, it is possible to change this option for each entry in the table.
- Preview Enable Flag. If preview mode is set to on, then a list of files in the current BRUKNET directory with different experiment names is produced along with the following:
  - the name as it would appear in the XWIN-NMR data tree,
  - the XWIN-NMR user directory,
  - the rename flag (whether the name specified in 'a' is used)
  - whether the file is to be converted.

  Each of these items can be edited individually for each entry in the preview table.
- Hard Copy Flag.
  With this parameter set, the contents of the current BRUKNET directory are listed on the default printer.
- Date Format Flag. This parameter indicates the format in which the date is presented in the renamed file. The following formats are available:
  YYMMDD DDMMYY MMYY

For date strings to appear in the XWIN-NMR filename, the XWIN-NMR file format string must contain between two and six '0's.

Bruktran Preview Mode

Using The Preview Mode

The preview mode is enabled by setting the Preview enabled flag to on in the Edit Program Options window and then by selecting the Convert (Rename enabled) flag in the main menu. Preview allows the user to inspect and edit the results of the renaming routine as well as changing the destination directory, the renaming select flag and the conversion select flag.

The Preview Window

The layout of the Preview window shows the current BRUKNET data directory, the station and the user as specified in the program options. Each entry in the table corresponds to one or more files with the same experiment name. Each entry contains the following:

- Aspect filename (without the extension).
- XWIN-NMR filename.
  The name as it would appear in the XWIN-NMR data directory. May be edited manually.
- Destination User Directory.
  This indicates the data directory into which the converted dataset will be stored, i.e. in the directory:
  / < disk > /data/ < Destinationuserdir > /nmr/.
  The value of this field is initially set in the Edit options window although it may be edited for each entry. This field may also be extracted from the Aspect filename if option '9' has been used in the format.
- Rename Flag.
  This determines whether a file is to be renamed to the XWIN-NMR filename string. If this flag is set to off, then the original Aspect filename is used in the XWIN-NMR data tree and the XWIN-NMR filename is completely ignored. If renaming has been enabled, then the old Aspect filename is stored in the title parameter (TI) with the processed parameters. The flag is initially set in the Edit Options window.
- Conversion Flag.
  If this flag is set to off then the file will be unconverted and will remain in the

BRUKNET directory.The value of this flag is initially set by the Select button in the main menu although it may be altered individually for each entry.

Bruktran automated renaming

Renaming Files

The Bruktran program offers two possibilities for the user to rename Aspect data files: either manual editing or automatic editing. Manual editing simply involves entering the new XWIN-NMR filename for each entry in the Preview table. Automated renaming, useful when Aspect file formats are consistent, involves defining both the Aspect and the XWIN-NMR filename format and possibly defining entries in the tokens file.

Filename Formats

Filenames are often represented by a string comprising one or more sets of characters. For example, you may identify a file by assigning the first 3 characters to mean one thing, the next 2 characters to mean another thing and the last 3 characters to mean something else. The filename format may be represented as follows:

Aspect file format : 11122333

Supposing you wish to rename the Aspect file. In this case, you have to decide on a format for the renamed file. One possibility is to expand :

1. the first 3 characters in the Aspect filename to 5 characters

2. the next 2 characters in the Aspect filename to 3 characters

3. the last 3 characters in the Aspect filename to 5 characters

The XWIN-NMR file format is : 1111122233333

Automatic renaming is then performed by extracting groups of characters from the Aspect filename (according to the Aspect file format), attempting to match these characters with an entry in the tokens file and then replacing these characters with the replacement string in the XWIN-NMR filename. If no match is found in the tokens file, then the original set of characters is used in the XWIN-NMR (possibly padded out with '_' characters).

Example:

1. determine formats

Aspect file format: 11122333
XWIN-NMR file format: 11113333222

2. get aspect filename

Aspect filename: *CSYEC104*

3. process filename attempting to match entries in the tokens file

Tokens file entries: *CSY=cosy* *EC=ehc* *104=chcl*

4. generate new filename

XWIN-NMR filename: *cosychclehc*

Filename Protocols

When defining a file format, the following points should be noted:

- Digits (0-9) must be used to specify a file format. All other characters (including spaces) are illegal. The characters 1 to 8 may be used to define a single or group of characters. The digits 0 and 9 have a special meaning described below.
- The same digit may not be used to define two different sets of characters i.e. the format 11122111 is ILLEGAL.
- The digits '0' and '9' have a special meaning.
  '0' is only meaningful in the X32 file format and reserves a character space for the date stamp.
  '9' is used in the Aspect file format to define a USER name.
  For example if the Aspect file format is defined as 11229994 the characters 5, 6 and 7 are extracted and stored in the Destination User Directory string. WARNING : Files will only be copied to this user's directory if the user exists on the system AND already has a valid user data directory structure (i.e. */DU/data/USER/nmr/*).
- The number of characters substituted into the XWIN-NMR filename depends on how many characters were allocated in the XWIN-NMR file format. For example, for the following Aspect fragment *CSY,* an entry in the tokens file is found to be: *CSY=cosy90*. The XWIN-NMR file format must therefore reserve 6 characters for the string *cosy90*. If less then 6 characters are reserved, then truncation will occur. If more than 6 characters are reserved, then the string *cosy90* is padded out with the underscore character.

Bruktran Tokens File

The Bruktran program attempts to access the file *Bt_tokens* which should be stored in the home directory of the currently logged in user. If this file is not found, then the program accesses the system editor to create this file, thus prompting you to input tokens. If you do not wish to edit this file, then the filename formats are reset to their default values and no automatic renaming is possible.

Token entries MUST be of the following format:

TOKEN1=TOKEN2

- TOKEN1 refers to the group of characters in the Aspect file name
- TOKEN2 refers to the characters to be inserted into the X32 filename

Important note :

1. THERE MUST BE NO SPACES BETWEEN THE '=' SIGN AND EITHER OF THE TOKENS.
2. THE ENTRY MUST BE LEFT JUSTIFIED
3. THERE MUST BE NO COMMENTARY IN THE TOKENS FILE

An example Bt_tokens file would contain the following entries:

CS=cosy
TC=tocs
AA=H1
AB=C13
AC=N15
10=fred
11=jane
12=jim
....
....

5.11.1.5 Parameter Plot of converted Apect 2000/3000 Data

When plotting data transferred from Aspect 2000/3000 computers and converted to XWIN-NMR format, you usually want to plot the parameters as well. However, only those acquisition parameters should appear that were actually used in the experiment's pulse program. This is possible by using a CONDITIONAL state-

ment entered into the format files for the parameter plot.

XWIN-NMR needs the corresponding Aspect 2000/3000 pulse program to generate a list of the parameters to be plotted. The nameof the pulse program must be contained in the parameter PULPROG, usually set automatically during the conversion of the Aspect 2000/3000 data set. The XWIN-NMR release tape contains the required Aspect 3000 pulse programs. After the installation of the release, they are stored in the directory :

/u/exp/stan/nmr/lists/pp.3000

However, these pulse programs have a special format which can be used only for plotting and they are no longer suitable for experiments. The special format can be recognized by the character sequence *;@* at the beginning of these files. You can add your own pulse programs to this directory. As soon as a plot is requested, they are converted automatically into the corresponding format and will stay in the directory given above.

5.11.2 JCAMP-DX Format

XWIN-NMR supports the JCAMP-DX Standard 5.0, Implementation Version 1.0 (draft), but currently only for the exchange of one-dimensional FIDs and spectra. The two conversion directions are available through the following commands :

XWIN-NMR data set *to JCAMP file : tojdx*

JCAMP file to XWIN-NMR data set : *fromjdx*

Their syntax and functionality is as follows:

tojdx [output-file [output-type [compression-mode [title [origin [owner]]]]]]

The current data set is converted to JCAMP format.

The parameters have the following effects:

- *output file*: name of the output file. If it is omitted the name of the current data set is taken and the suffix '.dx' will be appended.
- *output type*: 0=FID, 1=real spectrum, 2=complex spectrum.
- *compression mode*: this can be one of the following formats :
  - FIX (table format)
  - PACKED (no spaces between y values)

- SQUEEZED (sign coded into the first digit of the value, no spaces)
- DIFF/DUP (coding the differences of subsequent values in the SQUEEZED format with suppression of repeated values)

The format must be specified as an integer value with :
- 0=FIX
- 1=PACKED
- 2=SQUEEZED
- 3=DIFF/DUP.

Because of its maximum compression factor DIFF/DUP is the default for both FIDs and spectra.

- *title*: The text specified here appears as ##TITLE= .... in the output file. The processing parameter TI is taken as default. If TI is not specified, the current data set parameter NAME is used.
- *origin*: The text specified here appears as ##ORIGIN= .... in the output file.
- *owner*: The text specified here appears as ##OWNER= .... in the output file.

If one or more of the parameters (which are optional on the command line but required for the output) are omitted, a menu is brought up in which the missing entries must be entered. In this menu, you can terminate the conversion with the *Abort* button.

fromjdx [input-file [o]]

The file *input-file* is converted to an XWIN-NMR data set. If the file name is omitted, you will be asked for it in a dialog window. The data set to be created is stored in the path

/DU/data/USER/nmr/<input-file>/EXPNO/pdata/PROCNO

If the JCAMP file has the suffix *.dx,* the XWIN-NMR data set is stored without this suffix. The experiment number and the processing number are determined with the following procedure:

- If the option *o* (overwrite) was specified, EXPNO and PROCNO are both set to 1. Any existing data set is overwritten.
- If the option *o* was not specified, EXPNO and PROCNO are initially set to 1.
  If the JCAMP data set contains a fid, the EXPNO is incremented until an unused number is found.
  If the JCAMP data set contains a spectrum, the PROCNO is incremented.

In this way, the lowest possible EXPNO or PROCNO is assigned to the new data set.

During the conversion an error logging is performed. Any errors are logged into the file *jcerr.<input file>* in the current directory. Even if the file seems to be successfully converted, there could have been errors which are logged to this file.

After successful conversion, the converted data set becomes the current data set.

5.11.3 From VNMR

The command vconv converts Varian Unity and Gemini data sets to XWIN-NMR format. Only acquired (raw) data can be converted.
Up to now, data sets of VNMR 4.1 have been successfully converted. It could well be that data from older VNMR versions can be converted also but we have not tried it.
Comments and suggestions for a further improvement of the conversion routine are welcome.

5.11.3.1 File formats and setup

The VNMR data must reside in a directory with the extension *.fid*. Usually, the VNMR store functions will generate a directory with this extension. It contains the files *fid*, *procpar* and *text*. All three files are interpreted by vconv.
vconv will first ask for a file name. You can enter the file name in two ways.

- With its full path name, e.g. */usr/people/guest/eva.fid*. In this case, vconv only checks for the existence of this file.
- With its relative path name, e.g. *eva.fid*.
  In this case, vconv first checks for the file in the current directory (from which XWIN-NMR was started).
  If the file cannot be found, vconv will check for the environment variable VNMR. If VNMR is set (e.g to */usr/people/guest/varian*), the directory path is added to the file name and vconv checks for its existence.
  If this check also fails, vconv will ask for the file name again. During this interactive session, vconv can be terminated with the Delete key from the keyboard.

The extension *.fid* can be omitted when you enter the file name. vconv will append it if a file without the extension can not be found.

vconv reads the relevant parameters from the file *procpar* and converts them into the corresponding XWIN-NMR parameters. Because of the hardware and software differences, not all parameters have equivalents. The file *procpar* will be stored in the XWIN-NMR processing directory so that missing or uninterpreted parameters can be inspected.

During the conversion, vconv will check the parameter *seqfil*. If it is set, e.g. to *s2pul*, vconv checks for the parameter set */u/exp/stan/nmr/par/s2pul*.

- If the parameter set exists, it will be copied into the XWIN-NMR data set. Those parameters from the *s2pul* parameter set which are also interpreted and converted from the original *procpar* file will be overwritten by vconv.
- If the parameter set does not exist, standard parameters will be copied (for instance from */u/exp/stan/nmr/par/standard1D)*. Those parameters from the *standard1D* parameter set which are also interpreted and converted from the original *procpar* file will be overwritten by vconv.

5.11.3.2 Parameter conversion

Table 5.12 lists the *procpar* parameters and their equivalent in XWIN-NMR.

VNMR	XWIN-NMR	VNMR	XWIN-NMR	VNMR	XWIN-NMR
seqfil	PULPROG	pw90	P1	ss	DS
ct	NS(s)	pslabel	AUNM	temp	TE
d1	D1	pw	P0	sw1	SW_h(2D)
date	DATE	pp	P3	tn	NUCLEUS
dfreq	BF2	np	TD	tof	O1
dmf	P31	nt	NS(f)	rp	PHC0
dn	DECNUC	sfrq	BF1	lp	PHC1
fn	SI	solvent	SOLVENT	rfl/rfp	OFFSET
fb	FW	sw	SW_h		
dof	O2	spin	RO		

Table 5.12 Parameters converted from VNMR to XWIN-NMR

(s) status parameter
(f) foreground parameter
(2D) a parameter which is only set in 2D data files

5.11.3.3 Running vconv

vconv has five arguments. The command syntax is :

vconv <VNMR-name> <NAME> <EXPNO> <DU> <USER>

You can enter these arguments on the command line and they are interpreted in this order. If an argument is missing, vconv will ask for it. If all arguments are specified on the command line and the VNMR data set is found, vconv will do the conversion silently. The meaning of the arguments is as follows.

- **VNMR-name**
  The directory name under which the VNMR data set is stored
- **NAME**
  The XWIN-NMR directory name under which the converted data set is stored
- **EXPNO**
  The experiment number under which the converted data set is stored
- **DU**
  The disk on which the converted data set is stored
- **USER**
  The user name under which the converted data set is stored

vconv can also be used in automation. The command is VCONV and it must be called with the five arguments decribed above. Example :

VCONV(“eva.fid”,”eva.fid”,7,”u”,”guest”)

This would convert the VNMR data set *eva.fid* into an XWIN-NMR data set called

/u/data/guest/nmr/eva.fid/7/pdata/1

All arguments are of type string except for the EXPNO which is of type integer. Foreground and automatic conversion will set the PROCNO to 1.

The conversion of 2D HSQC and 3D data has also been added to the VNMR data conversion routine, as well as the conversion of fids acquired as float.
If the environment variable VNMR is set, all data from that directory are automat-

ically displayed if you just type vconv. The options were vconv is followed by a name (and other additional options) remain the same.

5.11.4 From JNMR

The command jconv converts Jeol data sets to XWIN-NMR format. It allows conversion of Jeol EX, GX and ALPHA 1D, 2D and 3D data. Only FID type data are converted. The conversion of processed data is not supported. If the environment variable JNMR points to a directory, jconv automatically displays all *.gxd* and *.nmf* files for selection.
The conversion of FX FID data has been implemented. FX data must have a numerical extension (like in proton.1) and the name must be specified on the command line, e.g. jconv proton.1. No parameter file is needed for the conversion, the most relevant parameters are extracted from the header of the data file.

5.11.5 Digital to analog

The command convdta converts digitally filtered Avance acquisition data to analog data format. The purpose of this command is to export data to external NMR processing programs not including the processing tools for digital data. Please note that by this conversion the quality of the baseline of the data may be slightly affected. It is therefore recommended to use the proper digital processing algorithms if available. XWIN-NMR recognizes the data type automatically and invokes the appropriate routines. convdta uses the current acquisition data as input, and generates a new EXPNO where it places the converted data. The destination EXPNO is asked for by the program. It can be specified as an argument also :
convdta <EXPNO>.

5.12 Execute

The *File->Execute* submenu (Figure 5.18) contains the commands which start automation (AU) programs and command macros. Please refer to the description of the commands edau and edmac and how to set up AU and macro files. There is also a special chapter in this manual which describes the syntax of AU programs in detail.

xau
The command xau executes an arbitrary AU program if it is available as an execut-

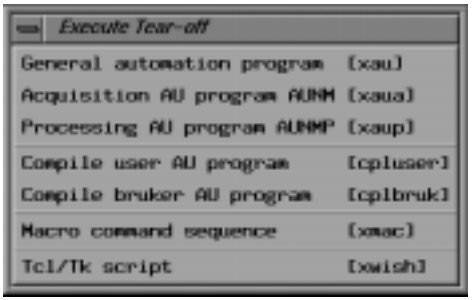


Figure 5.18 The *File->Execute* submenu

able version. For example, when terminating the edau command compilation is offered as an option. If the compiled version is not available, but the source code, XWIN-NMR will automatically start the compiler and generate the executable code.

When you call xau from the menu or when entering it on the keyboard, the list of compiled AU programs is displayed (edau, in contrast, shows all AU programs available as source code). Select one from the list for execution. If you already know the name of the program, you can start it via xau <name>, or just with <name>, leaving off xau. A compiled AU program behaves for this reason just like a normal XWIN-NMR command. You should avoid giving an AU program the same name as an existing XWIN-NMR command. If you did, and typed in the command name, the internal command would be executed instead of the AU program. However, you could still start the AU program by calling it via xau.

AU programs are, like all XWIN-NMR processing commands, executed in the *background*. This means that you can start several AU programs, and they will run in parallel.

In order to see which AU programs are currently running, execute the command *Display->Active Commands->Show*. If you want to interrupt the execution of an AU program before it terminates regularly, or if it hangs up for some reason, invoke the command *Show & allow for killing* in the same menu.

AU programs can also be called from within AU programs. The command xau(„t1calc“), inserted in an AU program, would execute the AU program *t1calc*. Instead of specifying an explicite name between quotes, a character array variable

initialized with a name would serve the same purpose.

xaua, **xaup**
These two commands provide an alternate method of starting an AU program. Its name is taken from the acquisition parameter AUNM (for xaua) or from the processing parameter AUNMP (for xaup) of the current data set. The length of the name is limited to 15 characters. These commands are mainly employed to start an AU program from within a higher level AU program, which can remain unmodified, but nevertheless call different lower level programs depending on the current data set and the value of the parameters AUNM or AUNMP.

cpluser, **cplbruk**
cpluser compiles all user-owned AU progarms, cplbruk all AU programs delivered by Bruker. cpluser may also be specified with an argument: cpluser auprog1 would only compile *auprog1*.

xmac
Invoking xmac from the menu or entering it on the keyboard will display a list of available command macros. Select one from the list for execution. If you already know the name of the macro, you can start it via xmac <name>, or just with <name>, leaving off xmac. A macro behaves for this reason just like a normal XWIN-NMR command. You should avoid giving a macro the name of an existing XWIN-NMR command. If you did, and typed in the command name, the internal command would be executed, not the macro. You could still start the macro by calling it via xmac.

xwish
Execute a Tcl/Tk script by typing xwish -f <script> on the keyboard. Tcl/Tk provides a programming system for developing and using graphical user interface applications; its name stands for *tool command language*. Please refer to the book *Tcl and the Tk Toolkit* by *John K. Ousterhout, Addison-Wesley Publishing Co.* Table 5.13 shows a simple Tcl/Tk script *example.tcl*, stored in the user's home directory. If you type xwish -f example.tcl on XWIN-NMR's command line, a window according to Figure 5.19 will be opened, consisting of a title, a text and an entry field. If you enter a number, followed by Return, the XWIN-NMR command processor CPR will be called using the procedure CPR_exec, and an exponential multiplication em with the entered number as line broadening factor LB will be executed on the current data set, followed by a Fourier transform ft. Then the window will be closed.

```
wm title . “xwish -f example.tcl“
label .label -text “Please enter line broadening factor LB:“
entry .entry -width 10 -relief sunken -bd 2 -textvariable lb
pack .label .entry -side left -padx 1m -pady 2m
bind .entry <Return> {
  CPR_exec “em $lb“
  CPR_exec “ft“
  Exit_unimar
}
```

Table 5.13 Tcl/Tk example script

Figure 5.19 Window generated by Tcl/Tk example script

This example shows that you can write your own windows applications with Motif style look and feel without the knowlege of X Windows, Motif and the C language. In a simple way you can tailor your own user interfaces for XWIN-NMR, including graphics drawing. You may combine Tcl/Tk scripts with AU programs. For example, from an AU program a Tcl/Tk script can be executed with the AU command *CPR_exec(“xwish -f <script“, WAIT_TERM)*. From a Tcl/Tk script, an AU program can be executed with *CPR_exec “xau <AU program name>*“. Table 5.14 shows the commands which may be used within Tcl/Tk scripts to access XWIN-NMR functions.

xwish2

The Tcl/Tk language has changed significantly since the introduction of the xwish command. For this reason, the command xwish2 has been introduced. If you want to write Tcl/Tk scripts based on Tcl 7.5 and Tk 4.1, you must invoke them with xwish2 rather than xwish. We recommend to store such scripts in the directory

XWINNMRHOME/prog/tcl/xwish2_scripts/

(the environment variable XWINNMRHOME=/u for XWIN-NMR standard installa-

CPR_exec “XWIN-NMR command“ e.g. CPR_exec “zg“	Execute XWIN-NMR command from a Tcl/Tk script. Continue executing the script without waiting for the command to be finished.
CPR_exec_wait “XWIN-NMR command“ e.g. CPR_exec_wait “swh 5000“	Execute XWIN-NMR command from a Tcl/Tk script. Continue executing the script after the command is finished.
Getstan	Return the current experiment path (usually */u/exp/stan/nmr/*)
Getinstr	Return the acquisition parameter INSTRUM (the name of the configured spectrometer)
Getversion	Return XWIN-NMR’s version number
Getcurdir	Return the path name */u/prog/<XWIN-NMR version>/curdir/<user>/*
Exit_unimar	Terminate current Tcl/Tk script

Table 5.14 Tcl/Tk interface commands to XWIN-NMR

tions). Then, in order to execute the script, you just enter its name on the XWIN-NMR command line.

Chapter 6

The *Process* Menu for 1D Data

Basic 1D processing is presented in the chapter 1D Data Processing: Tutorial for Beginners. This chapter describes the commands located in the Process Menu.

6.1 General parameter setup [edp]

The edp command opens a dialog box which allows you to set the parameters required by the various commands in the Process menu. The parameters are described along with the commands which make use of them.

6.2 Line broadening factor [lb]

The LB parameter is set by calling this command. LB is used by the commands em and gm. It can also be set in the edp window.

You can also type lb on the keyboard, followed by Return, or specify the desired value on the command line, e.g. lb 0.3 , followed by Return.

6.3 Exponential multiply [em]

The command em multiplies the data point i by the factor

$$\exp(-(i - 1) * LB * P I/(2 * SW))$$

The processing parameter LB (in Hertz) determines the resultant line broadening. LB is normally greater than zero and improves the sensitivity, but with the disadvantage of line broadening. The acquisition parameter SW (in Hertz) is the width of the spectrum.

Execution of em is automatically preceded by bc, i.e. a base-line correction is always executed before the multiplication, as long as the parameter BC_mod is not set to 'no'.

6.4 Manual window adjustment

With this command, interactive window adjust mode is entered. You will find a detailed description of this mode in the chapter 1D Data Processing, Tutorial for Beginners.

6.5 Real spectrum size [si]

This command defines the number of points in the real spectrum after Fourier transformation of the fid. You can also type in the command, or set the parameter SI from the edp processing parameter editor.

6.6 Fourier transform [ft]

The command ft transforms the acquisition data of the current data set, which may already have been processed using a filter function. The result is a spectrum consisting of SI data points in the real part and SI data points in the imaginary part. The value of the processing parameter, SI, must be set by the operator and must be a power of 2. Alternatively, the default value may be used which is set equal to the acquisition parameter TD/2, in the case where TD/2 is also a power of 2. SI values which are not powers of 2 are automatically rounded up to the next highest power of two. TD is the number of acquired data points, i.e. the length of the FID. If the user sets the value of SI < TD/2, then the resulting spectrum is correspondingly smaller. If the value of SI is greater than TD/2, then the Fid is filled out with 2*SI -TD zeroes before the transformation.

The processing parameter TDeff (effective TD) defines the number of points of the FID that will contribute to the transform. The default value of TDeff is zero which means that all TD data points of the fid will be taken into account (but maximum 2*SI).

The command ft can transform data which have been acquired in one-channel mode or two-channel mode (quad). There are two cases in quad mode. Sequential acquisition is where the data points from the two channels are sampled one after the other with a time interval set by the dwell time. The second case is a simultaneous acquisition where data points from the two channels are sampled at the same time. The type of acquisition used is recognized by the transform software from the value of the status acquisition parameter, AQ_mod.

The first data point is multiplied by the parameter FCOR immediately before the transform is executed. Values, from 0 to 2 are allowed, with the default being 0.5.

For AMX/ARX type spectrometers, if the processing parameter PKNL is set to the value true, a 5th order phase correction is performed after the Fourier transform to compensate phase errors introduced by the Butterworth filters. For AVANCE type instruments, PKNL must always be set to true. In this case, no 5th order correction is applied, but a special handling for digital spectrometers is applied.

If the processing parameter REVERSE is set to true, both the real and the imaginary part of the spectrum are reflected by a vertical line through the center of the spectrum. In other words, the spectrum is reversed. The same effect can be achieved by the command rev.

6.7 Automatic phase correction [apk apk0 apk1]

Please refer to the chapter 1D Data Processing: Tutorial for Beginners.

The command apk0 performs an automatic phase correction of zero order phase and leaves first order phase unchanged. The command apk1 performs an automatic phase correction of first order phase and leaves zero order phase unchanged.

6.8 Manual phase correction

Please refer to the chapter 1D Data Processing: Tutorial for Beginners.

6.9 Phase correction with constants PHC0 and PHC1 [pk]

The data, consisting of real and complex points (R(i); I(i)), is phase corrected using the formulae:

$$R0(i) = R(i) * \cos(a(i)) - I(i) * \sin(a(i))$$

$$I0(i) = I(i) * \cos(a(i)) + R(i) \sin(a(i))$$

where R and I represent the corrected values and

$$a(i) = PHC0 + (i - 1) * PHC1; \text{ where } i > 0$$

PHCO and PHC1 are the zero and first-order phase parameters, respectively, and may be set by the user (in degrees) as processing parameters. The values may also be obtained using the interactive phase correction procedure as described elsewhere in this manual. pk is mainly used to phase a number of similar spectra with the same phase parameters.

pk always operates on the processed data if available (e.g. after em or ft), otherwise the acquisition data (Fid) are used. In order to force pk to use the acquisition data even though processed data exist, use trf instead of pk by setting the mode parameters properly as described with the trf command.

6.10 Magnitude spectrum [mc]

The command mc replaces the real part of the spectrum by its absolute value according to the formula

$$ABS(i) = SQRT (R(i) * R(i) + I(i) * I(i))$$

where R is the real part and I is the imaginary part of the spectrum. It can also operate on the Fid like pk (see the description of the pk command).

6.11 Power spectrum [ps]

The real part of the spectrum is replaced by the square of the absolute value ABS(i) *ABS(i), (see mc).
It can also operate on the fid like pk.

6.12 Special window functions

6.12.1 Exponential [em]

See “Exponential multiply [em]” on page 139.

6.12.2 Gaussian [gm]

gm multiplies the data by the function

$$\exp(-a*t - b*t**2)$$

where a is negative and b is positive.

The Fid originally has the exponential envelope of the form exp(-t/T2). Therefore a gaussian line shape exp(-b * t * t) results when a =-1/T2. Since this line shape has less extensive wings than a normal line, the resolution of overlapping resonances is improved greatly. A suitable choice of parameters can lead to further reduction in the line width. a and b have the form

$$a = PI*LB \quad \text{and} \quad b = - a /(2*GB*AQ);$$

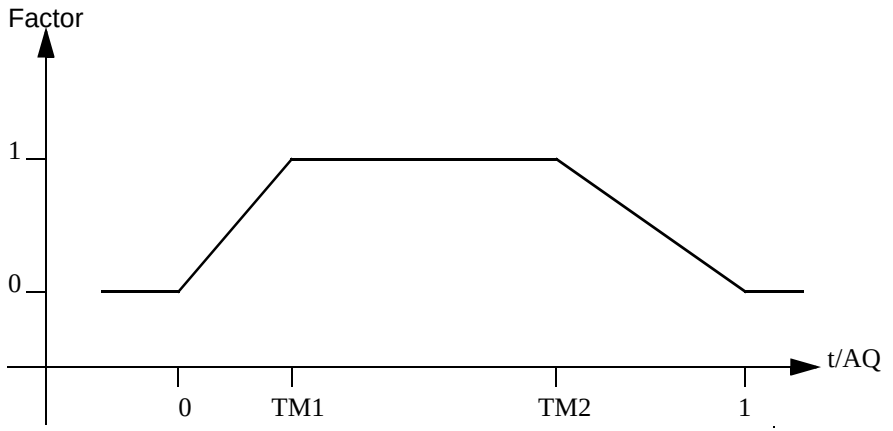
where LB and GB are processing parameters and the acquisition parameter AQ is the acquisition time of an Fid (without accumulation).

The function has a maximum value of exp(-PI*LB*AQ/2) when Tmax = GB*AQ. Tmax, and therefore GB, may be determined (once the data has been weighted by a suitable em LB exponential and transformed) by taking the reciprocal of the half line width of the line to be sharpened. To execute gm, LB should be entered as a negative value. Valid values for GB lie between zero and one. If GB=0.33 the Gaussian function has a maximum after 1/3 of the acquisition time.

gm is automatically preceded by bc - baseline correction, if the parameter BC_mod is not set to no.

6.12.3 Trapezoidal [tm]

tm multiplies the data by a function of the following form:



The processing parameters TM1 and TM2 determine which region of the data is multiplied by a constant. Outside of this region, the data is multiplied by a straight line, the gradient of which is given by TM1 and TM2. TM1 and TM2 must lie between zero and one, and are entered as a fraction of the acquisition time, AQ, of an Fid.

6.12.4 Sine [sinm], Squared sine [qsin]

The command sinm multiplies the data by a sine function of the form

$$\sin((PI - PHI) * (t/AQ) + PHI),\ 0 < t < AQ;$$

where PHI = PI/SSB.

The processing parameter SSB may have the values 0; 1; 2; 3; . ... A pure sine wave is obtained when SSB=0 or 1; a pure cosine wave when SSB=2. PHI approaches zero for greater values of SSB.

The command qsin multiplies the data by the square of this function.

6.12.5 Sinc [sinc], Squared sinc [qsinc]

sinc multiplies the data by
$\sin(t)/t\ , \quad -2\pi \cdot SSB \cdot GB < t < 2\pi \cdot SSB \cdot (1 - GB)$

6.12.6 User defined [uwm]

The command uwm multiplies the fid of the current data set with the fid of the second data set (to be defined with edc2), and stores the result as processed data (files 1r, 1i) of the current data set. The original fid is retained.

The fid of the second data set is considered as the user defined window function. Its size (TD) must be equal or greater than the size TD of the fid of the current data set. In the latter case the window is truncated before multiplication is applied.

In order to generate a user defined window, proceed as follows:

1. Type new and define a new data set, whose fid will serve as the window.
2. Execute the AU program *calfun*. This AU program calculates an arbitrary function and stores it as the file *fid* in the current data set. This will become the user defined window. *Calfun* is set up so that you can modify it by adding your desired window function. You must recompile it after changing it. Please read the header of *calfun* how to do that. *Calfun* also contains examples, e.g. how an exponential window is programmed.

6.12.7 Traficante window multiplication [traf, trafs]

Window function according to D.D. Traficante and G.A. Nemeth,
using parameter LB
J. Magn. Res., **71** (1987) 237.

6.13 Sequential operations

These commands provide abbreviations for a sequence of frequently used processing steps.

ef corresponds to the execution of em, followed by ft
efp corresponds to the execution of em, followed by ft , followed by pk
gf corresponds to the execution of gm, followed by ft
gfp corresponds to the execution of gm, followed by ft , followed by pk

fmc corresponds to the execution of ft, followed by mc

6.14 Baseline correction operations

6.14.1 Manual abs, absf, absd, sab

Please refer to the chapter 1D Data Processing: Tutorial for Beginners.

6.14.2 Fid dc offset correction [bc]

bc applies a baseline correction to the acquisition data, i.e the file fid of the current data set, depending on the processing parameter BC_mod. The result is stored as a processed data set in the files *1r, 1i*, and is ready to be further processed, usually by applying a window function and a fourier transform.

Effect of the parameter BC_mod:

BC_mod	Effect
no	bc performs no operation
single/quad	a constant is subtracted from the fid
spol/qpol	a polynomial of degree 5 is subtracted from the fid
sfil/qfil	filtering of the fid according to Bax/Marion

Table 6.1

single/quad The constant is determined by calculating the average value of the last quarter of the Fid, for each channel separately if quadrature detection is active.

spol/qpol The polynomial of degree 5 is the result of a least squares fit, again separate for each channel. This mode leads to a reduction of strong water signals in the center of the spectrum.

sfil/qfil This mode suppresses signals (e.g. water) in the center of the spectrum. It is effective over a range defined by the processing parameter BCFW (in

ppm). The algorithm is derived from Marion, Ikura, Bax, J. Magn. Res. 84,425-430(1989).

The latter two cases can also be used to suppress off center signals if the processing parameter COROFFS is different from 0.

bc is called automatically when the commands em , gm or ft are executed.

The parameter pairs single/quad, spol/qpol, and sfil/qfil are provided for single detection and quadrature detection data, respectively. bc is not affected, however, because XWIN-NMR knows the data type and uses the correct mode internally. The difference is only valid for the command trf, where the user can define the type of calculation to be performed.

6.15 File algebra

6.15.1 Define multiplier [dc]

The floating point parameter DC is used by various algebra commands as a multipplier or term of a sum. Use dc to set DC, or set DC from the edp processing parameter editor.

6.15.2 Define second and third data set [edc2]

This command allows you to define the two data sets used by the add command (or some other commands that apply to two data sets).

6.15.3 Add data sets [add]

These commands multiply the third current data set (defined using edc) by the processing parameter DC, then add to that the second current data set and store the result as the current data set: currentdataset = second + DC * third .

add may also be used so that the current, second or third data set is substituted (overwritten) by the result.

The processing parameter DATMOD determines whether add or at uses acquisition data (DATMOD = *raw*) or processed data (DATMOD = *proc*).

6.15.4 Add raw data sets [addfid]

Like command add above but works on fids only. addfid is independent of parameter DATMOD and generates a new fid in the current data set.

6.15.5 Add constant DC to current data [addc]

addc adds the value of the processing parameter DC to the acquisition data or the processed data, depending on the value of DATMOD.

6.15.6 Multiply current data by DC [mulc]

mulc multiplies the current acquisition or processed data by the processing parameter DC, depending on the value of DATMOD.

6.15.7 Multiply two data sets [mul]

mul multiplies the second and third current data set (to be defined with edc2). The result is stored as the current processed data set. The acquisition or processed data are used depending on the value of DATMOD.

6.15.8 Divide two data sets [div]

With div, the second current data set is divided by the third data set (to be defined with edc2) and the result is stored as the current data set. The acquisition or processed data are used depending on the value of DATMOD .

6.15.9 Logical Or, And, XOR [or, and, xor]

The data in the second and third current data set (to be defined with edc2) are combined with each other bitwise using the corresponding boolean operation and the result is stored as the current data set. The processing parameter DATMOD determines whether this unification is carried out on the acquisition (raw) or processed data. The following commands are available:

and = logical *and* of the two data sets
or = logical *or* of the two data sets
xor = logical *exclusive or* of the two data sets

6.15.10 Negate data [nm]

nm negates the current acquisition or processed data, i.e. they are multiplied by -1, depending on the value of DATMOD .

6.16 Special transforms

6.16.1 Inverse ft [ift]

In some cases it is useful to transform a spectrum back to the Fid. For example, a row of a 2D spectrum can be back-transformed to produce a Fid that can again be forward transformed to produce a spectrum of a different size afterwards. Back Transformation is accomplished with the command ift. It must be noted, however, that the Fid thus obtained replaces the spectrum (i.e. the real and imaginary parts *1r, 1i*) of the current processed data set number, but never the original Fid of the current experiment number. The length of the Fid obtained through ift equals twice the length of the real part of the back-transformed spectrum.

6.16.2 User defined ft [trf trfp]

trf executes a sequence of commands in the order: baseline correction bc, window multiplication, Fourier transform and phase correction, depending on the values set for the processing parameters BC_mod, WDW, FT_mod and PH_mod. This command is essentially intended for the experiments performed by the more experienced operator. trf operates on the acquisition data and generates a processed data set, leaving the Fid unchanged. Unlike the trf command the trfp command uses data that have already been processed. The trfp command can be applied to the same data as often as needed, with different processing parameters each time.

The parameters may have the following values:

BC_mod	effect
no	bc baseline correction will not be executed
single	subtract bc constant from single detection data
quad	subtract bc constant from quad detection data
spol	subtract bc polynomial from single detection data
qpol	subtract bc polynomial from quad detection data
sfil	Bax/Marion bc filter for single detection data
qfil	Bax/Marion bc filter for quad detection data

Table 6.2

WDW	effect
no	no filter function used
EM	exponential multiplication
GM	gaussian multiplication
SINE	sine multiplication
QSINE	multiplication with sine-squared
TRAP	trapezoidal multiplication
SINC	sinc multiplication
TRAF	traficante window multiplication
TRAFS	traficante window multiplication

Table 6.3

6.16.3 Hilbert transform [ht]

The dispersion relations or Kramers-Kronig relations (see, for example, R. R. Ernst, G. Bodenhausen and A. Wokaun, Principles of nuclear magnetic resonance in one and two dimensions, Clarendon Press, Oxford, 1987) apply to the real and imaginary parts (files *1r* and *1i*) of a spectrum. They state that the real and imaginary part can be calculated from each other via a Hilbert transformation. In XWINNMR the command ht provides this transformation. ht calculates the Hilbert transformation of the file *1r*, which is created during the Fourier transformation and

FT_mod	effect
no	no FT will be executed
fsr	real FT of one channel (forward, single, real)
fqr	real FT of quad data (forward, quad, real)
fsc	complex FT of one channel data (forward, single, complex)
fqc	complex FT of quad data (forward, quad, complex)
isr	real inverse FT of one channel data (inverse, single, real)
iqr	real inverse FT of quad data (inverse, quad, real)
iqc	complex inverse FT of quad data (inverse, quad, complex)

Table 6.4

PH_mod	effect
no	no phase correction will be executed
pk	phase correction using the parameters PHC0, PHC1
mc	magnitude calculation
ps	power spectrum

Table 6.5

ME_mod	effect
no	no Linear Prediction
LPbr,LPbc	Linear Backward Prediction (see chapter on 2DFT for details)
LPfr,LPfc	Linear Forward Prediction (see chapter on 2DFT for details)

Table 6.6

contains the real part of the spectrum. The result is stored as the new imaginary part *1i*.

ht is of use whenever the real part of the spectrum was further processed alone (for example with baseline correction), and afterwards the imaginary part is needed again, e.g. to re-phase the spectrum or to calculate a magnitude spectrum, etc.

For multidimensional spectra, it can be advantageous to only store the real part on disk (in order to save disk space) and recalculate an imaginary part via the Hilbert transformation when needed.

6.16.4 Make fid from 1r, 1i files [genfid]

If an Fid obtained by ift is to be treated like an actually acquired Fid, it must be stored under a suitable experiment number as the file fid. This can be done with the command genfid.

Syntax of genfid:

- genfid
- genfid EXPNO <option>
- genfid EXPNO <NAME> <option>
- genfid EXPNO <NAME> <USER> <option>
- genfid EXPNO <NAME> <USER> <DU> <option>

genfid
The new EXPNO is asked for by the program. If it already exists the user is asked if the data can be overwritten (all raw data and PROCNO 1 of the new EXPNO will be overwritten). The new EXPNO will become the current data set and will appear on the display.

genfid EXPNO
The new EXPNO is specified directly. If it exists already, the user is asked whether the data can be deleted.

The processed data set number PROCNO of this new data set is set to 1.

If option is the character y existing data are overwritten without warning.
If option is the character n the program does not switch to the new data set.
Both options can be combined.

6.17 Digital filtering (smooth) [filt]

filt smooths the data by replacing each point with a weighted average of its surroundings. The coefficients must be in a text file in the directory */u/exp/stan/nmr/filt/1d* and have, for example, the following format: *3; 1; 2; 1:*

The first number must be odd and gives the number of coefficients, followed by the coefficients themselves. In the above example, the data point p(i) is replaced by

$$1* p(i -1) + 2 * p(i) + 1 * p(i + 1).$$

The coefficients file name of the smoothing filter to be used must be entered in the processing parameter DFILT. filt may be applied to the acquisition data as well as processed data of the current data set, depending on how the processing parameter DATMOD is set (*raw* or *proc*).

6.18 Miscellaneous operations

6.18.1 Zero file contents [zf]

zf produces a processed data set (files *1r* and *1i*) which contains only zeroes. The current fid remains unchanged. The zf command evaluates the processing parameter DATMOD. If the value of DATMOD is raw, processed data sets of size SI will be generated. If the value of DATMOD is proc, available processed data will be zeroed and their size remains unchanged.

6.18.2 Define NZP parameter [nzp]

The processing parameter NZP is used to define an integer required by the command zp. Type nzp, or set it NZP from edp.

6.18.3 Zero first NZP points of data [zp]

The first NZP points of the data are set to zero by zp. NZP is a processing parameter. Depending on the value of the processing parameter DATMOD acquisition data or the processed data are used.

6.18.4 Define NSP parameter [nsp]

The processing parameter NSP is used to define an integer required by the commands ls and rs.

6.18.5 Left shift data by NSP points [ls]

With ls, the complete data field is shifted NSP points to the left. NSP is a process-

ing parameter. The end of the data field is zero-filled. Depending on the value of the processing parameter DATMOD acquisition data or the processed data are used.

6.18.6 Right shift data by NSP points [rs]

With rs, the complete data file is shifted NSP points to the right. The left of the field is zero-filled. Depending on the value of the processing parameter DATMOD acquisition data or the processed data are used.

6.18.7 Reverse data [rv]

By rv the data are reversed about a line vertical to the time or frequency axis and going through the middle of the data set. The real and imaginary part of the spectrum are thereby reversed. Depending on the value of the processing parameter DATMOD acquisition data or the processed data are used.

6.18.8 Build first derivative [dt]

The first derivative of the current data set is calculated by dt. Depending on the value of the processing parameter DATMOD acquisition data or the processed data are used.

6.18.9 Quadrature image correction [qc, qk]

qk - quadrature correction using the previously calculated constants ALPHA and GAMMA
qc - quadrature correction using the current constants ALPHA and GAMMA

qk and qc implement Parks and Johannesen's algorithm, J. Mag. Res. 22, 265-267 (1976). In quadrature detection, that is AQ_mod=qseq or AQ_mod=qsim, the phase difference between the two channels of the receiver, A and B, may be exactly 90 degrees and the signal amplitude may differ slightly between them. These two errors can cause mirrored signals to arise around the center of the spectrum.

This correction method replaces the signal A with a corrected signal

GAMMA * (A - ALPHA * B),

that is, a linear combination of A and B, where GAMMA and ALPHA are so cal-

culated that the corrected signal A is orthogonal to B (in vector notation):

ALPH A = A * B/B ** 2, GAMMA** 2 = B ** 2/(A - ALPH A * B) ** 2

The algorithm can fail when there are signals in the spectrum symmetrical about the center. Therefore, the processing parameters ALPHA and GAMMA should be determined with qk using a suitable reference sample, and then further spectra should be corrected with qc using these parameters. In time, and using different experimental conditions, it will be necessary to redetermine ALPHA and GAMMA.

qk and qc are always applied to the processed data if they exist (eg. after em or ft). Otherwise, they are applied to the raw data (Fid). If one wants to apply the quad correction to the fid even though the processed data already exists, then the processed data must be deleted (delp).

In our experience quad correction works best if applied directly to the fid, because the window function disturbs the algorithm.

If the data is acquired with AQ_mod=*qsim*, the quad correction may be applied either to the spectrum or to the fid. If data acquisition is performed using AQ_mod=*qseq*, the spectrum must be corrected, because the fid does not have the correct format.

6.19 Linear Prediction

The parameter ME_mod, which is set to *no* by default, controls whether a linear prediction is to be applied to the data. Details about ME_mod are described in the chapter on 2D transforms.

A linear prediction of the Fid is executed if a window function or ft command is issued, but only if the parameter ME_mod is not set to *no*.

For backward linear prediction the processing parameter TDoff is evaluated. If TDoff is set to a positive value, the first TDoff points are replaced by predicted points. If TDoff is set to a negative value, -TDoff predicted points are added to the beginning of the fid. The added -TDoff points are discarded from the end of the zerofilled fid. Note that fid points are lost if more points are added at the beginning than zero filled at the end -TDoff > 2*SI -TD.

Chapter 7

The *Process* Menu for 2D data

7.1 General parameter setup [edp]

All parameters required for 2D processing can be displayed and modified using this command. Most parameters exist for both dimensions. The effect of a parameter is described along with the command that it influences.

7.2 2D Transform [xfb]

The command xfb executes the two commands xf2 and xf1 in sequence, but requires less computation time than the separate execution of the two commands.

After completion of an xf1 transformation the middle point of the *2rr* file is corrected for a quad spike, i.e. it is replaced by the median of its two neighboring points (in F1).

7.3 Transform of F2 rows [xf2]

The xf2 transformation acts on the raw acquisition data (the *ser* file) if no data exist that resulted from an F1 transform (executed with xf1). In this case a data set is

created which is only transformed in the F2 dimension, and can be further processed with xf1 to complete the transform. The file *2rr* contains the real data. The imaginary data are stored depending on the xf1 processing parameter MC2, i.e. in the case of qf in the file *2ii* and in all other cases in the file *2ir*. The transformation mode is independent of the processing parameter FT_mod, but depends on the acquisition status parameter AQ_mod. If it is equal to *qsim*, a complex forward transformation is performed, otherwise a real one.

7.3.1 Processing Parameters TDeff (effective TD) and TDoff

TDeff determines the number of points of the Fid which are to be used in the transformation. The default value for TDeff is zero which means that all TD points of the Fid are used in the transformation, but at the most 2*SI. This number is reduced when TDeff is set to a non-zero value. The transformation will begin at the data point with number TDoff (starting at 0), which corresponds to a left shift by TDoff points. If TDoff is negative, -TDoff zero points are inserted at data start, which corresponds to a right shift.

7.3.2 Processing Parameter SI (Size)

SI determines the number of data points with which the Fourier transformation is calculated. After the transformation there are SI real data points and SI imaginary data points. The default value for SI is TD divided by 2 rounded to the next higher power of 2. The user can set SI to a larger or smaller value. The minimum value allowed for SI is 4. If necessary, the data are zero-filled or truncated.

7.3.3 Strip transform

The default values of the processing parameters STSR **(**strip start) and STSI (strip size) are zero, which means that all data points (SI real and SI imaginary points) after the Fourier transformation are stored in the data sets for the processed data. If STSI is different from zero, then only a strip of STSI data points starting from STSR is stored. All subsequent data processing such as phase correction (whether within xf2 or as a separate command) only acts on these STSI data points. Since XWIN-NMR stores 2D spectra as a sequence of submatrices, arbitrary values of STSI are not allowed. The strip must consist of complete submatrices. XWIN-NMR therefore rounds the value entered by the user, so the calculated strip is usually somewhat larger than specified. The values actually used are stored after the transformation in the status parameters (see command dpp). The desired strip can also

be set interactivly. Enter the 2D utilities menu, define a region using the drag-and-draw button, then click the set strip command button. This will set STSR and STSI for both dimensions.

7.3.4 Processing Parameter BC_mod (baseline correction mode)

The first step of processing is the baseline correction of the fid to be transformed, by subtracting either a constant or a 5th order polynomial or by applying a filter according to Marion, Ikura & Bax (1989) (BC_mod=*quad/qpol/qfil*). The latter two cases are used to suppress water signals in the center of the spectrum or off center if the processing parameter COROFFS is different from 0. BC_mod=*no* means that no baseline correction is performed (default in FI dimension), which is sometimes to be preferred if the signals have not decayed strongly towards the end. Further details on BC_mod can be found in the description of bc.

7.3.5 Processing Parameter ME_mod (linear prediction mode)

Instead or in combination with zerofilling forward or backward linear prediction can be applied to the data by setting the parameter (ME_mod=LPfr/LPbr). Linear Prediction will be described in detail in a later section.

7.3.6 Processing Parameter WDW (window type)

Data are digitally filtered, with the parameter WDW determining the type of filter:

no	no filter
EM	exponential multiplication
GM	gaussian multiplication
SINE	sine multiplication
QSINE	multiplication with squared sine
TRAP	trapezoidal multiplication
USER	user defined window function
SINC	sinc multiplication
QSINC	squared sinc multiplication
TRAF	traficante window multiplication
TRAFS	traficante window multiplication

These filters function exactly like the 1D window functions.

In the case of USER the user himself must provide the window function in the

form of an Fid file of a 1D data set. This data set can be created using the command edc, with TD set according to the desired size. The Fid file must be provided by the user, and must contain the user's window function. Now, before starting the transform, the program must know where to find the window function. Enter it as second data set via the command edc2.

7.3.7 Processing Parameter PKNL (non-linear phase mode)

For AMX/ARX type spectrometers, if the F2 processing parameter PKNL is set to the value true, a 5th order phase correction is performed after xf2 to compensate phase errors introduced by the Butterworth filters. For AVANCE type instruments, PKNL must always be set to true. In this case, no 5th order correction is applied, but a special handling for digital spectrometers is applied.

7.3.8 Processing Parameter PH_mod (phase mode)

After the transformation of a signal the following processing is done depending on the value of PH_mod:

no = no further processing
pk = phase correction with the parameters PHC0, PHC1
mc= calculate the magnitude spectrum
ps = calculate the power (squared magnitude) spectrum

In the case of mc or ps, no Fourier transformation can subsequently be performed in the F1 direction.

7.3.9 Processing Parameter FCOR

The first data point is multiplied by the parameter FCOR immediately before the transform is executed. The default value is 0.5, numbers between 0 and 2 are legal.

7.3.10 Processing Parameter REVERSE

If the processing parameter REVERSE is set to true, both the real and the imaginary parts of the spectrum are reflected by a vertical line through the center of the spectrum. In other words, the spectrum is reversed. The same effect can be achieved by the command rev2 .

7.4 Transform of F1 columns [xf1]

The xf1 transformation acts on the raw acquisition data (the *ser* file) if no data exist that resulted from an F2 transform excuted by xf2. In the first case a data set is created which is only transformed in the F1 dimension, and must be further processed with xf2 to complete the transform. In the second case xf1 completes a partial F2 transform.

The processing parameters SI, STSR, STSI, TDeff, TDoff, BC_mod, ME_mod, WDW, PH_mod, FCOR and REVERSE for this dimension work just like the ones described above for the F2 dimension. However, in addition to these, the following parameter is important:

7.4.1 Processing Parameter MC2

The type of transformation is determined from the parameter MC2 according to Table 7.1.

MC2	transformation type
QF	fqc = forward quad complex
QSEQ	fqr = forward quad real
TPPI	fsr = forward single real
States	fqc = forward quad complex
States-TPPI	fsc = forward single complex
echo-antiecho	fqc = forward quad complex

Table 7.1

Through appropriate choice of MC2 the transformation is matched to the experiment which created the raw data:

QF: complex 2-quadrant transformation is applied when only the magnitude or power spectrum is to be obtained. The transformed result is stored in the 2 files *2rr* (real matrix) and *2ii* (imaginary matrix) of the current data set.

QSEQ: phase sensitive 4-quadrant transformation is applied when two successive

fids (separated by the incrementable delay) were acquired with the pulse phases 0 and 90 degrees (sequential quadrature detection in F1), with t for the second fid being incremented by IN.

TPPI: Time Proportional Phase Increments, phase sensitive 4-quadrant transformation is applied when four Fids separated by D0 with phases 0, 90, 180, 270 degrees were acquired (this method is also sequential in F1).

States (States, Ruben, Haberkorn), phase sensitive 4-quadrant transformation is applied when two successive FIDs (not separated by D0) with phases 0 and 90 degrees were acquired, and the next two also with 0 and 90 degrees. The FIDs are acquired with the same t (simultaneous quadrature detection in F1).

States-TPPI (States, Ruben, Haberkorn), phase sensitive 4-quadrant transformation, is applied when two successive FIDs (not separated by D0) with phases 0 and 90 degrees were acquired, and the next two with 180 and 270 degrees. The FIDs are acquired with the same t (simultaneous quadrature detection in F1).

Echo-Antiecho (States, Ruben, Haberkorn), phase sensitive 4-quadrant transformation for gradient enhanced spectroscopy. Special handling is applied to the raw data:
Each two consecutive FIDs are replaced by

re0 = -im1 - im0
im0 = re1 + re0
re1 = re1 - re0
im1 = im1 - im0

processing then continues as in the States case.

The result of the phase sensitive transformations is stored in the four files *2rr*, *2ri*, *2ir* and *2ii* in the current data set directory.

2rr = purely real data
2ri = real in F2, imaginary in F1
2ir = imaginary in F2, real in F1
2ii = purely imaginary data

7.5 General 2D processing options

xfb as well as xf1, xf2, xtrf, xtrf2 (the latter 2 to be described later) may be called

with the option n, e.g. xfb n. The effect is that the imaginary parts *2ri*, *2ir*, *2ii* will not be stored on disk. This option can be useful if disk space is restricted or to improve processing time if phase constants are already known (e.g. after delp).

All 2D processing commands which do baseline correction, linear prediction, window multiplication or Fourier transform evaluate the option nc_proc followed by a decimal value. If possible data are scaled such that the resulting processing status parameter NC_proc equals the preset; check status parameters for success. Instead of a numeric value the keyword *last* may be specified to use the value of the current status parameter NC_proc as a preset.

Commands working on acquisition data can be forced to generate big or little endian processed data by the options big and little. Also with commands working on acquisition data with the xdim option the 2rr submatrix dimensions can be defined by the processing parameters XDIM. These options should only be used when processed data are exported to 3rd party programs that do not evaluate the processing status parameter BYTORDP respectively XDIM.

The only commands always using raw data (*ser* file) are xtrf and xtrf2. The commands xf2, xf1, xfb work either on raw or processed data, all other commands work on processed data. Processed data is used with xf2, xf1 and xfb unless:

1. there is no *2rr* file, or *procs* or *proc2s* are not readable (missing),

2. a negative number is found in the file dsp.hdr entry LOC (indicating last processing command was killed),

3. a Fourier transform is requested in a dimension that already contains frequency domain data, or

4. data are power or magnitude data (processing status parameter PH_mod is *ps* or *mc* in any dimension).

To make the behavior more transparent the raw option with 2D processing commands forces use of raw data. Also supported is the proc option, causing processing to stop with an error message if it is impossible to continue on processed data.

Note: Do not confuse raw data (*fid* or *ser*) and time domain data. Processed data may be in time domain in any dimension (if no Fourier transform has been performed in this dimension).

7.6 Linear Prediction

The transform commands described so far execute a number of processing steps in the following sequence:

baseline correction, window multiplication, Fourier transform, phase correction.

If linear prediction is to be applied, this sequence looks as follows:

baseline correction, linear prediction, window multiplication, Fourier transform, phase correction

This means, that linear prediction is also part of the transform commands in either dimension, and is applied to the acquisition data after the baseline correction. Usually, linear prediction is disabled, i.e the default value of the processing parameter ME_mod is set to *no*. In order to enable it, Me_mod must be set to:

LPfr for forward LP, real data
LPfc for forward LP, complex data
LPbr for backward LP, real data
LPbc for backward LP, complex data

Real or complex data types depend on the acquisition type. See the Table 7.1 in the section describing xf1/MC2. This distinction is, however, only important for xtrf type transforms. For the commands xf2, xf1 and xfb, the difference between real and complex LP is handled internally, i.e *LPfr* and *LPfc* will give the same result. The same is true for *LPbr* and *LPbc*.

The processing parameter NCOEF represents the number of coefficients used for the LP-calculation. Ideally this parameter should be set to 2-3 times the number of expected peaks. A default value of 100 for a TD of 256 is recommended.

7.6.1 Forward Linear Prediction

Forward linear prediction is performed if the processing parameter ME_mod is set to *LPfr* or *LPfc* (see above). The number of points used for the prediction is TD. This number is reduced to TDeff if TDeff > 0. If this is the case, replace TD by TDeff in the following description. Forward LP is only performed if 2*SI>TD. If the parameter LPBIN is zero, the number of predicted points is the maximum of TD and 2*SI - TD. All points exceeding 2*TD are set to zero. If LPBIN is set between TD and 2*SI , it represents the number of output points (original + pre-

dicted) of the forward LP. The remaining points 2*SI - LPBIN are set to zero.

7.6.2 Backward Linear Prediction

Backward linear prediction is performed if the processing parameter ME_mod is set to *LPbr* or *LPbc*. In the following description, TD must be replaced by TDeff if TDeff > 0 (see forward LP). The parameter LPBIN has a meaning different from the case of forward LP. The number of points contributing to backward LP (input points) can be reduced to LPBIN, if LPBIN is set between 0 and TD. The processing parameter TDoff represents the number of points to be predicted. If TDoff is set to a positive value, the first TDoff points are replaced by predicted points. If TDoff is set to a negative value, -TDoff predicted points are added to the beginning of the fid. The added -TDoff points are discarded from the end of the zerofilled fid. Note that fid points are lost if more points are added at the beginning than zero filled at the end -TDoff > 2*SI -TD.

7.6.3 Combination of Forward and Backward Linear Prediction

Forward and backward Linear Prediction can be combined with two succesive processing commands. The first command must be xtrf type and performs baseline correction and backward LP. Set the flags BC_mod and ME_mod to the desired values, the others to no. The second command xtrfp, xf2, xf1 or xfb performs forward LP, window multiplication, FT and phase correction. Set ME_mod, WDW, FT_mod and PH_mod to the desired values and BC_mod to *no*.

7.6.4 Zerofilling, right-shift, left-shift

If ME_mod is set to *no* or NCOEF is 0, no linear prediction is done. Ordinary zerofilling is performed if 2*SI >TD. If TDoff is positive, the first TDoff points of the Fid are discarded, and zeroes are added and the end. This corresponds to a left-shift in 1D processing. If TDoff is negative, zeroes are added at the beginning of the Fid, the last TDoff points of the (zerofilled) Fid are discarded. This corresponds to a right-shift in 1D processing.

As follows from the previous explanation, forward LP and backward LP with negative TDoff will increase the number of Fid-points contributing to window multiplication and further processing steps. This number is documented in the processing status parameter TDeff.

7.7 Phase correction

7.7.1 Manual

This command lets you enter manual 2D phase correction mode.

7.7.2 Phasing using constants PHC0, PHC1

After a phase sensitive Fourier transformation, the data may be submitted to a phase correction in the F1 or F2 dimensions with these commands. The phase correction is based upon the phase parameters PHC0 and PHC1 in the dimension in question. If a phase correction has already been carried out, then the status parameters PHC0 and PHC1 give the sum of all the phase corrections in the F1 and F2 directions, respectively. Commands xf1p and xf2p are analogous to the 1D-command pk. Both dimensions can be phased with a single command xfbp, thereby saving computing time.

7.8 Magnitude Spectrum [xf1m, xf2m, xfbm]

After the 2D Fourier transformation of a 2D acquisition data set (*ser* file), the spectrum is stored in the directory of the current data set in the files *2rr*, *2ri*, *2ir* and *2ii*. *2rr* is the real part which is displayed on the screen or output to the plotter. The other files contain imaginary parts, with *2ri* and *2ir* only occuring for phase sensitive spectra. There are various commands for the calculation of magnitude spectra which have the following effects:

xf2m: The contents of *2rr* and *2ir* are substituted (pointwise) by

$$2rr = \sqrt{(2rr^2 + 2ir^2)}$$

$$2ri = \sqrt{(2ri^2 + 2ii^2)}$$

xf1m: The contents of *2rr* and *2ir* are substituted (pointwise) by

$$2rr = \sqrt{(2rr^2 + 2ri^2)}$$

$$2ir = \sqrt{(2ir^2 + 2ii^2)}$$

If the command xf2m has been applied before, the final result is thus

$$2rr = \sqrt{(2rr^2 + 2ir^2 + 2ri^2 + 2ii^2)}$$

xfbm: This command has the same effect as xf2m, followed by xf1m, so that the result equals the last formula above.

7.9 Power Spectrum [xf1ps, xf2ps, xfbps]

These commands are analogous to xf1m, xf2m, xfbm, except that the square root is not taken for the calculation of the power spectrum.

7.10 COSY-Type symmetrization [sym, syma]

These commands symmetrize the data points on both sides of the spectrum diagonal (which runs from below left to above right). The command sym replaces the larger of two symmetrical points with a point which has the same magnitude as the smaller point, while syma does the same thing but the replaced point retains its sign (positive or negative). syma must therefore be used for phase sensitive spectra.

7.11 J-resolved-Type symmetrization [symj]

It is recommended that the spectrum subjected to tilt before symj is applied.

symj symmetrizes the data points on both sides of a horizontal line which runs through the middle of the matrix. The data point with the larger magnitude is replaced by one which has a magnitude equal to that of the smaller data point.

The status parameter SYMM documents which kind of symmetrization was applied to a given 2D spectrum. It may take on the values no, sym, syma or symj.

7.12 Tilt of a 2D-spectrum [tilt, ptilt, ptilt1]

Again, due to experimental conditions, 2D J-resolved spectra may be tilted along a horizontal line through the center of the matrix. This distortion may be removed by tilt, which shifts the data in points above this line to the right and data below the line to the left: n = tiltfactor * (nsrow/2 - row), where

tiltfactor = (SW_p1/SI1) / (SW_p2/SI2)
nsrow = no. of rows in the spectrum
row = the row which is to be corrected

The two edges at both sides are filled with those data that are thrown away on the opposite side due to the shifting.
The status parameter TILT (*true* or *false*) indicates whether tilt was applied to a given 2D spectrum.

Tilting with arbitrary angles can be achieved with the command ptilt and the processing parameter ALPHA(F2) between -1 and 1, resulting in a tilt between -45 and +45 degrees. ptilt can be applied more than once to the same spectrum.

The command ptilt1 tilts along a vertical line through the center of the matrix using the processing parameter ALPHA(F1). With processing parameters ALPHA(F2) = ALPHA(F1) = 1 the command sequence ptilt, ptilt1, ptilt rotates the 2D spectrum by 90 degrees.

7.13 Manual Baseline Correction [bcm1, bcm2]

If a row or column is extracted from a 2D spectrum a manual baseline correction can be applied within the basl menu. The same correction can be applied to all rows or columns of the 2D spectrum with the commands bcm2 and bcm1.

7.14 Automatic Baseline Correction [abs1, abs2]

The command abs1 performs a baseline correction of all columns for the current 2D spectrum, while abs2 corrects all rows. This is done in a manner analogous to the 1D command absf. Only that part defined by the 2D and 1D processing parameter pairs ABSF1 and ABSF2, respectively, is used. No region files are created.

7.15 Trapezoid Baseline Correction[abst1, abst2]

The effect of these commands is similar to abs1, abs2. The corrected region, however, is not the same for all 1D sections, but is shifted from section to section by a constant amount. For the first section it is given by ABSF2, ABSF1, for the last one by the parameters SIGF2, SIGF1. The region limits for the sections in between result from an interpolation. Only the case ABSF2 less than ABSF1 is treated. In the orthogonal dimension the corrected region is limited by the orthogonal parameters ABSF2, ABSF1.

The 2D basline correction commands absot2, absot1 are similar to abst2, abst1 but use a modified algorithm that gives a larger corrected region at the cost of a worse baseline.

7.16 Subtraction of 1D Spectra [sub1, sub2, sub1d1, sub1d2]

The commands sub1 and sub1d1 subtract the real part *1r* of the 1D spectrum, defined as second data set with the command edc2, from each column of the real part *2rr* of the current 2D spectrum.

sub2 and sub1d2 subtract from the rows of the 2D spectrum.

Note: When using the commands sub1 and sub2 the subtraction only takes place if the two points to be subtracted from each other have the same sign. All points will be set to zero for which the subtraction caused a change in sign.

7.17 Extracting Rows and Columns [rsr, rsc]

These commands allow a desired row (rsr) or column (rsc) to be extracted from the current 2D spectrum and to be stored as a 1D spectrum (see also chapter Interactive Manipulation of 2D Data).

Syntax:

- rsc
- rsc <column number>
- rsc <column number> <PROCNO> <switch>
- rsr

- rsr <row number>
- rsr <row number> <PROCNO> <switch>

The program will ask for the row or column number if only the command itself was entered (case 1). The user can also specify the number directly (case 2). The extracted data will then be stored as a temporary data set using the name ~ *TEMP* with EXPNO=PROCNO=1. Furthermore, it is possible to specify a processed data number PROCNO (case 3). The data will then be stored under the name of the current 2D data set, but with the entered PROCNO. If the optional switch parameter is the character n, the 2D data set remains the current data set to allow for additional rsr/rsc commands. Otherwise, the extracted row or column will become the current data set and will appear on the display.

WARNING: This description is valid for the case where the 2D data were generated by a complete 2D transform. If only one dimension was transformed, then the following descriptions apply.

If only xf2 was performed, rsc generates a new EXPNO containing the section as an Fid. In contrast, rsr creates a PROCNO. Make sure that all transformation parameters are set up correctly. If only xf1 was performed, both rsc and rsr generate a new PROCNO. An fid may be obtained using the command genfid.

7.18 Extracting an fid from 2D Acquisition Data [rser]

If the current data set contains multidimensional acquisition data, i.e. a series of fids, rser allows one to extract one fid and store it as a 1D data set.

Syntax:

- rser
- rser <fid number>
- rser <fid number> <EXPNO> <PROCNO> <switch>

In case 1, the program will ask for the number of the fid within the serial acquisition data. Case 3 allows one to explicitly specify an experiment and a processed data number under which to store the extracted fid. The <switch> has the same meaning as with rsr/rsc.

7.19 Extracting Projections [rhpp, rvpp, rhnp, rvnp]

All 2D processing commands automatically calculate horizontal and vertical projections of the 2rr file. They are stored in the current data directory. In case of magnitude or power data only positive projections are calculated, in all other cases also negative projections. If projections are not available or data have been manipulated outside XWIN-NMR the command proj will recalculate all projections.

With the following commands projections can be extracted from a 2D data set and stored as a 1D spectrum

rhpp read horizontal positive projection

rvpp read vertical positive projection

rhnp read horizontal negative projection

rvnp read vertical negative projection

Syntax (valid for all above commands):

- rhpp
- rhpp <PROCNO> <switch>

Handling of PROCNO and <switch> is the same as with rsr/rsc .

7.20 Replacing Rows and Columns [wsr, wsc]

wsr and wsc are the complimentary operations to rsr and rsc, and replace the specified row or column of the current 2D spectrum by the 1D spectrum ~*TEM*P 1 1 (cases 1, 2) or by the 1D spectrum with the specified processed data number PROCNO (case 3). The program will print an error message if the 1D spectrum and the 2D spectrum are of incompatible size. The <switch> has the same meaning as with rsr/rsc.

Syntax:

- wsc
- wsc <column number>
- wsc <column number> <PROCNO> <switch>

- wsr
- wsr <row number>
- wsr <row number> <PROCNO> <switch>

7.21 Replacing an fid within a *ser* File [wser, wserp]

wser replaces the n-th fid of multi-dimensional acquisition data (file *ser*) by the original acquisition data (file *fid*) of a one-dimensional data set. This enables one, for instance, to store a repeated measurement of a single fid of a 2D experiment in a *ser* file.

wserp replaces the n-th fid of multi-dimensional acquisition data (file *ser*) by already processed acquisition data (files *1r, 1i*) of a one-dimensional data set. This command is usually employed when a fid that was read from a *ser* file using rser, and that was then processed by ls or a similar command, is now to be written back into the *ser* file.

Syntax (also valid for wserp):

- wser
- wser <fid number>
- wser <fid number> <EXPNO> <PROCNO> <switch>
- wser <fid number> <EXPNO> <PROCNO> <NAME> <USER> <DISK>

<fid number> is the number of the fid to be replaced in the *ser* file, while the remaining arguments define the data set of the *ser* file, if the current data set is the single fid to be stored. If the current data set, however, is the data set of the *ser* file, the arguments must define the fid to be stored.

If EXPNO and PROCNO are not specified, the data set *~TEMP* 1 1 replaces the fid in the *ser* file. This is useful if, for instance, *~TEMP* 1 1 was obtained by rser.

With wser, a conflict can arise if one tries to write a data set into the *ser* file that is larger than the fid to be replaced. XWIN-NMR solves this problem by truncating the data set to the suitable size before executing wser.

<switch> has the same meaning as with rsr/rsc.

Please check the AU command table in this manual if you want to use wser or

wserp in AU programs.

7.22 User defined 2D Fourier Transform [xtrf, xtrf2]

Unlike with the xfb transformation, the type of transformation to be performed can (and must) be set with the processing parameter FT_mod. Table 6.4 . The baseline correction mode for the fids, BC_mod, must specify single or quad data. ME_mod must specify real or complex operation. xtrf is the two-dimensional analogue of the trf command and executes the commands:

- bc correction of the fid, linear prediction, window multiplication, fourier transformation and phase correction

for each dimension. As with trf, the individual steps are controlled by the processing parameters BC_mod, ME_mod, WDW, FT_mod and PH_mod for the respective dimension. FT_mod can even be set to *no* if no Fourier transformation is wanted. xtrf2 only evaluates the processing parameters in the F2 dimension. It behaves like the xf2 transformation with freely selectable transformation type.

If all XF2 dimension parameters are set to *no*, then xtrf2 only causes the *ser* file to be re-arranged into the submatrix format (including zero filling or truncation, depending on the parameters TDoff, TDeff, and SI). The *ser* file itself remains unchanged and the result is stored in the files *2rr* (even numbered points) and *2ri* (odd numbered points) of the current PROCNO (non-phase sensitive spectra: *2rr*, *2ii*). This way, a two dimensional view of a *ser* file can be obtained on the screen. It appears in the lower left quarter of the display.

xtrf with all F1 parameters set to *no* only evaluates the processing parameters in the F2 dimension. It behaves like xtrf2 and does no processing or data rearrangement in the F1 dimension. xtrf and xtrf2 always use the acquisition raw data as input data and create new data sets for the processed data.

7.22.1 User defined 2D Fourier Transform [xtrfp, xtrfp2, xtrfp1]

The xtrfp transformation differs from the xtrf transformation only in the type of data used as input data. Unlike the xtrf transformation, the xtrfp transformation uses data that have already been processed with the transformations described above. The xtrfp command can be applied to the same data as often as needed, with different processing parameters each time, e.g. to apply several filters or to transform the data back and forth. xtrfp2 only evaluates the processing parameters

in the F2 dimension, xtrfp1 only in the F1 dimension.

7.23 Inverse Fourier Transformation [xif2, xif1]

These commands correspond to xtrfp2 and xtrfp1 transformations with the processing parameters BC_mod, ME_mod, WDW and PH_mod set to no. The inverse of the processing status parameter FT_mod is used as the transformation mode.

7.24 2D Hilbert Transformation [xht2, xht1]

The commands xht1 and xht2 are the two dimensional analogues of the command ht (see description there).

For phase-sensitive spectra the following rules apply:

From the real part *2rr* of the 2D spectrum xht1 creates the imaginary part *2ri*. Additionally, it creates *2ii* if the Fourier transformation has created the imaginary part *2ir.*

Likewise, xht2 creates the imaginary part *2ir* from the spectrum's real part *2rr*, and in addition the part *2ii* if the Fourier transformation has created the imaginary part *2ri*.

Choose the Hilbert transformation command which creates those imaginary parts that are needed for the subsequent phase correction (xf1p , xf2p).

7.25 Reflection of a 2D-spectrum [rev1, rev2]

rev1 reflects a 2D-spectrum about a horizontal line at the center, and rev2 about a vertical line at the center. Further processing of the data is possible as both the real and imaginary parts are handled.

7.26 Create a *ser* file from processed data [genser]

If an inverse transformed 2D spectrum is to be further processed as an original *ser* file (i.e stored under a certain EXPNO) then the command genser is to be used. It is

the two-dimensional analogue to the 1D genfid command.

Similar to the command genfid, the command genser generates a *ser* file from processed 2D data (files *2rr*,..) and stores it under a new EXPNO. genser is required if, for futher processing of the data, one of the parameters SI, STSR, STSI, TDoff or TDeff must be changed, for example in order to combine strip transformation and inverse Fourier transformation.

Syntax of genser:

- genser
- genser <EXPNO> <option>
- genser <EXPNO> <NAME> <option>
- genser <EXPNO> <NAME> <USER> <option>
- genser <EXPNO> <NAME> <USER> <DISK> <option>

genser
The new EXPNO is asked for by the program, it must be different from the current EXPNO. If it already exists the user is asked if the data can be overwritten (all raw data and PROCNO 1 of the new EXPNO will be overwritten). The new EXPNO will become the current data set and will appear on the display.

genser EXPNO
The new EXPNO is specified directly. If it already exists the user is asked if the data can be overwritten.

genser <EXPNO> <NAME> <USER> <DISK>
After EXPNO also NAME , USER and DISK can be specified.

If option is the character **y** existing date are overwritten without warning.
If option is the character **n** the program does not switch to the new data set.
Both options can be combinad.

All arguments of genser can be interchanged.
The processed data set number PROCNO of this new data set is set to 1.

If a 2D dataset was reduced in size with xtrf2 and all processing flags set to *no* with appropriate values of TDoff and TDeff, genser creates a new EXPNO with reduced TD values.

7.27 Convert AVANCE data to AMX format [convdta]

Generates an AMX *ser* file from an Avance *ser* file. Uses the same syntax as genser. Use this command before processing Avance data with a Fourier transform written for AMX data.

7.28 Add and subtract 2D spectra [add2d]

Processed data of the current data set (curdat) are replaced by:

ALPHA*curdat + GAMMA*curdat2.

Set both values to 1 for addition and use the values 1 and -1 for subtraction. Both real and imaginary part are handled.

7.29 Calculate level file [levcalc]

After the Fourier transform or any other operation on a 2D spectrum, the standard deviation and maximum and minimum of the 2D spectrum are calculated and stored as processing status parameters S_DEV, YMAX_p, and YMIN_p. They are used by the command levcalc to determine a suitable threshold for plotting contours of the spectrum on either display or plotter.

The automatic contour level calculation (levcalc) can be influenced by the processing parameters NLEV, LEV0 and TOPLEV. Levels are calculated between the bottom and top levels. The top level must be specified in TOPLEV in per cent of the absolute maximum intensity value in the spectrum, the bottom level is a multiple of the standard deviation S_DEV with the factor LEV0.

levcalc generates a level file with the specified number of contours and must be called in AU programs after the 2D processing commands if automatic contour plotting follows.

This method ensures that no spectral peak will affect the result.

7.30 Quadrature image correction [xqc1, xqc2]

xqc1 and xqc2 are the two-dimensional analogues to the quadrature correction command qc for 1D spectra. (See qc for description).

The needed correction value ALPHA and GAMMA are determined as in the 1D case with the command qk. It is applied to a row or column of the 2D spectrum and the resulting ALPHA and GAMMA are entered as processing parameters for the F1 or F2 dimension of the 2D spectrum before issuing the command xqc2 or xqc1.

7.31 Zero region[zert1, zert2]

These commands zero a trapezoidal region of 2rr, 2ir, 2ri and 2ii. The region definition is the same as for the abst1, abst2 commands.

Chapter 8

The 3D *Process* Menu

8.1 General parameter setup [edp]

The command edp invokes the parameter editor for the 3D processing parameters. It displays three columns for the dimensions F3, F2, and F1. Set the parameter PPARMOD to the value *3D*, if this is not yet the case, to ensure that parameter files are created for all three dimension. The corresponding aquisition parameter is PARMOD, which is usually set prior to acquisition with eda. Acquisition and processing parameters files are stored in the current acquisition or processed data directory. Table 8.1 shows the file names and the associated commands to edit or

Dimension	**Acquisition parameters (eda)**	**Acquisition status parameters (dpa)**	**Processing parameters (edp)**	**Processing status parameters (dpp)**
F3	acqu	acqus	proc	procs
F2	acqu2	acqu2s	proc2	proc2s
F1	acqu3	acqu3s	proc3	proc3s

Table 8.1 3D parameter files

display the parameters. Individual parameters may also be examined by means of keyboard commands such as 1s td, 2s td, 3s td, si, 2 si 64, etc. *s* denotes status parameters.

The processing parameters to be set before a 3D transform are straightforward extensions to the 2D case and will therefore not be explained again in this chapter. Please refer to the chapter *The 2D Process Menu*. There is one particular status parameter, however, which is unique to 3D: AQSEQ. It describes the order *321* or *312*, in which the 1D fids of a 3D acquisition are stored in the *ser* file (3 = the acquisition dimension, 1 and 2 = the orthogonal dimensions). On Avance type systems, 3D pulse programs will set AQSEQ automatically if td and td1 are used consistently within the pulse program. However, you may explicitly define AQSEQ in the pulse program. For this purpose, insert one of the following statements in the pulse program header: aqseq 321 or aqseq 312. On both, AMX/ARX and Avance type systems, you can set or modify AQSEQ using the command 3s AQSEQ before starting the transform.

8.2 Fourier Transformation [tf3, tf2, tf1]

The commands tf3, tf2, tf1 execute the transform in the dimensions F3, F2, F1, and must be invoked in this order to obtain a completely transformed 3D spectrum. Depending on the setting of the processing parameters, the transform includes the baseline correction of the fid and the application of a window function before the fourier transform, and phase correction or magnitude/power spectrum calculation after the transform. This is just as in the 2D case.

The 2D commands xfb, xf2, and xtrf can be executed on a 3D acquisition data set in order to obtain a selected plain. When typing one of these command the transform direction and the PROCNO (where the 2D result is to go) are requested.

Table 8.2 and Table 8.3 show the files involved. The files beginning with the character *3* are stored in the current processed data directory. *3rrr* contains the real spectrum, the others include imaginary components. The *ser* file contains the acquisition data and is stored in the current acquisition data directory. tf2 and tf1 only transform real data if MC2 is not equal to *qf*. All transformations only create those imaginary output data needed for the phase correction in the corresponding direction. This prevents the creation of 7 more files with imaginary data in addition to the real data (*3rrr*), which would consume disk space and slow down the transformation.

Command	Input files	Output files
tf3	ser	3rrr, 3irr
tf2	3rrr	3rrr, 3rir
tf1	3rrr	3rrr, 3rii
xfb, xf2, xtrf	ser	2rr, 2ri, 2ir, 2ii

Table 8.2 Files read and written by transform if MC2 ≠ *qf* ¼ ¼¼

Command	Input files	Output files
tf3	ser	3rrr, 3iii
tf2	3rrr, 3iii	3rrr, 3iii
tf1	3rrr, 3iii	3rrr, 3iii
xfb, xf2, xtrf	ser	2rr, 2ii

Table 8.3 Files read and written by transform if MC2=*qf*

After invoking a transform command, you will be asked whether the imaginary parts should be stored on disk. If not, disk storage capacity will be saved, but an additional phase correction at a later time with the commands tf3p, tf2p, tf1p will not be possible. However, you may use the Hilbert transform commands tht3, tht2, tht1 at any time to calculate the required imaginary parts from the real part *3rrr*. From the keyboard, the transform commands may also be entered in the form tf3 y or tf3 n in order to suppress the question by specifying the answer directly. Another option has the form tf3 y a, telling the program to store imaginary parts only if enough disk space is available.

Each of the files *3rrr* etc. created during the transformation contains the spectrum in a *subcube* format, a generalization of the 2D submatrix format for 3D. The subcube size is determined automatically by the program according to the amount of available memory in order to minimize transformation time. After the transformation the size can be examined with the commands 3s xdim, 2s xdim, 1s xdim. The maximum subcube size is 64K data points (=256K bytes). If it is important for you to define the subcube dimensions yourself, append the option c to the transform command, e.g. tf3 y a c or tf3 y c. The program will ask for the sizes in a dialog.

3D data sets often cover a large portion of the disk capacity. tf3 usually stores the

transformed data on the same disk partition where the acquisition data are located. This would restrict the size of the raw data to about half of the available space. In order to circumvent this problem, tf3 may be called with another option *p*. The command tf3 p requests the disk partition where the transformed data are to be stored. This partition may be located on the same or on a different disk. Alternately the partition may be specified directly after the *p*: tf3 pu2. The option *p* may be given together with the options decribed earlier. Their sequence does not play a role.

The 3D transform commands may also be invoked from AU programs. Two arguments are required. The first one specifies whether to store the imaginary parts, and the second one corresponds to the p option described above.
Example: tf3("y", "pu").

8.3 Phase correction [tf3p, tf2p, tf1p]

The purpose of tf3p, tf2p, tf1p is to apply an additional phase correction to a transformed 3D spectrum, provided the required imaginary parts are available. The storage of imaginary parts can be selected when giving the transform commands. Alternately, the imaginary part can be generated from the real 3D spectrum by means of a *Hilbert* transform (commands tht3, tht2, tht1).

Details about phase correcting a 3D spectrum can be found in the chapter *3D Data Processing Tutorial*.

8.4 Baseline correction [tabs3, tabs2, tabs1]

The commands tabs3, tabs2, tabs1 are the 3-dimensional analoga to the 2D commands abs2 and abs1, and the 1D command absf.

Please refer to the chapter *3D Data Processing Tutorial* for a more detailed dicussion.

8.5 Calculate projections [r23p, r13p, r12p]

The commands r23p, r13p, r12p calculate the F2-F3, F1-F3, and F1-F2 projections of the real part *3rrr* of the current 3D spectrum, respectively. The result is stored as

a 2D spectrum under a new EXPNO requested by the program, or specified on the command line (e.g. r23p 5).

8.6 Hilbert transform [tht3, tht2, tht1]

The commands tht3, tht2, tht1 calculate the imaginary parts of a 3D spectrum required for the phase correction commands tf3p, tf2p, tf1p, respectively, from its real part *3rrr*.

8.7 Read planes

8.7.1 Orthogonal planes [r23, r13, r12]

The commands r23, r13, r12 store a 2-dimensional plane of the current 3D spectrum parallel to the F2-F3, F1-F3 and F1-F2, respectively, as a regular 2D spectrum under a new EXPNO with PROCNO=1. They may be invoked in one of the following three forms

r23
r23 <plane no.> <EXPNO>
r23 <plane no.> <EXPNO> no

In the first form, the arguments are requested by the program. The second form creates a 2D spectrum including the imaginary part, while in the third form only the real part is stored. If MC2 is equal to *qf*, the imaginary part of the 2D spectrum *2ii* is taken from the 2D file *3iii*. If MC2 is not equal to *qf*, the generated 2D imaginary files depend on the transformation sequence. The resulting 2D files and the 3D files from which they originate are given in Table 8.4.

In AU programs the commands for extracting planes must always be given with the described arguments e.g. r23(5, 2, “pu“). Here, the F2-F3 plane with number 5 including its imaginary part is extracted and stored as 2D spectrum under EXPNO=2 on disk partition *u*.

8.7.2 3D Diagonal planes

The commands r23d, r13d, r12d extract the diagonal planes F2=F3, F1=F3, F1=F2, respectively, from the real part *3rrr* of the current 3D sepctrum. The result

tf sequence	r23		r13		r12	
	from	to	from	to	from	to
tf3	3irr	2ir	3irr	2ir	-	-
tf3, tf2	3rir	2ri	3rir	2ir	3rir	rir
tf3+tf1	3rri	2ir	3rri	2ri	-	-
tf3, tf2, tf1	3rri	2ri	3rri	2ri	3rri	2ri
tf3, tf1, tf2	3rir	2ri	3rir	2ri	3rir	2ri

Table 8.4 2D imaginary files generated by read plane commands

is stored as a 2D spectrum *2rr*. It is saved under a new EXPNO requested by the program, or specified on the command line, e.g. r23d 5.

Chapter 9

The *Analysis* Menu

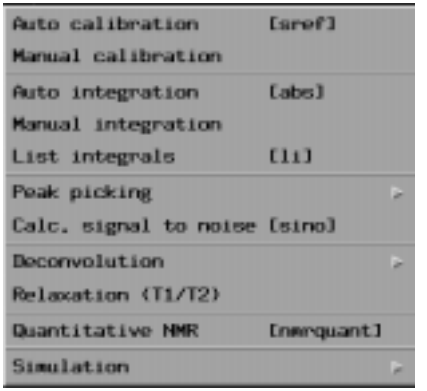


Figure 9.1 The *1D Analysis* menu

9.1 Manual calibration

The manual calibration of a spectrum's frequency axis is described in the chapters *1D Processing Tutorial* and *2D Processing Tutorial*.

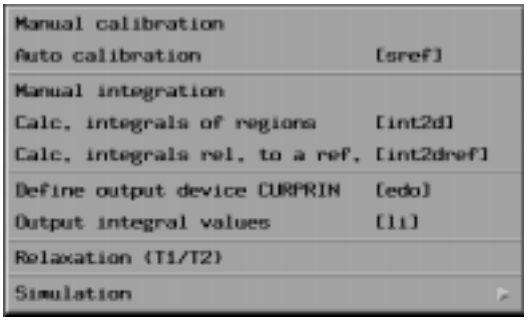


Figure 9.2 The *2D Analysis* menu

9.2 Auto calibration [sref]

The command sref calibrates the TMS peak to 0 ppm. It is one of the most important commands used during automatic spectrometer operation, where manual calibration (described in the chapters *1D/2D Processing Tutorial*) is not possible. sref may be invoked to calibrate 1D or 2D data sets. In order for sref to work as expected, the lock nucleus (parameter LOCNUC) and the solvent (parameter SOLVENT) must be set in either dimension. The lock nucleus determines, which table is used for the lock command, containing parameters such as lock power, field value, and frequency shift for the solvent. The table is located in the directory */u/conf/instr/INSTRUM/*. Its file name is built by appending *lock* to the lock nucleus name, e.g. *2Hlock*. INSTRUM is the parameter where the configuration name of the spectrometer is stored. You may examine it by entering the command instrum on the keyboard, or by calling the dpa parameter display. The lock table may be edited with the command edlock.

SOLVENT is usually defined with the lock command. During lock-in, the lock power, field value, and frequeny shift for the solvent are set according to the values in the lock table. The lock-in procedure sets the frequeny shift to the exact frequency shift value for the given solvent as listed in the table, and then adjusts this slightly to achieve lock-in. As a result, the absolute magnetic field is now nearly the same no matter what lock solvent is used. This has the advantage that offsets can now be defined in ppm, since the absolute frequency corresponding to a given ppm value no longer depends on the lock solvent (i.e. SF=BF1, where SF is the

absolute frequency of the reference signal, and BF1 is the basic spectrometer frequency). The checklockshift command sets the parameter LOCSHFT to TRUE. The parameter LOCSHFT is checked during sref. checklockshift is invoked automatically at the end of an acquisition (zg). The only task of sref is to correct the remaining error of a few Hertz by searching for the reference signal in a small interval around SF. The interval size may be specified in the *Width* column of edlock. Finally sref sets the corrected value of SF, and the processing status parameter OFFSET, which is the chemical shift of the first data point of the spectrum. From this value, the program can calculate the shift value of any data point.

The sref procedure described so far is valid for data acquired with instruments equipped with the BSMS digital lock. If LOCSHFT is FALSE, the algorithm originally supplied for data sets measured with spectrometers equipped with an SCM unit is employed. In this case, the basic setting of SF is computed from the *Distance* parameter in the lock table. For BSMS type data sets, this algorithm delivers incorrect SF values.

9.3 Integration and peak picking

These commands are described in detail in the chapters *1D Processing Tutorial* and *2D Processing Tutorial*.

9.4 Calculate signal to noise ratio [sino]

The command sino calculates, sets and prints out the processing status parameter SINO, the signal-to-noise ratio of a 1D spectrum, according to the formula *SINO=maxval/(2*noise)*. *maxval* is the largest intensity value in the frequency range between SIGF1 and SIGF2. The noise is computed from the data points in the frequency range between NOISF1 and NOISF2 according to the algorithm in Table 9.1. These parameters are processing parameters in ppm units. They may be entered by typing sigf1 etc. on the keyboard, or using the edp editor.

You may also set the parameters interactively from the screen. Enter the *utilies* mode by clicking on the *utilities* button of the button panel. Expand the desired signal region so that it fills XWIN-NMR's data area, and click on the *sigreg* button. This will set SIGF1 and SIGF2 to the displayed region. Reset the horizontal scaling to redisplay the whole spectrum. Expand the desired noise region to full

noise=sqrt((Y2 - (Y*Y + 3*XY*XY/(N*N - 1))/N)/(N - 1)),
with
N = number of points in the noise interval [NOISF1, NOISF2],
Y =sum(-n<=i<=n)[y(i)]
Y2 =sum(-n<=i<=n)[y(i)*y(i)]
XY =sum(1<=i<=n)[i*(y(i) - y(-i))],
where
y(-n)=first intensity value in noise interval
....
y(0) =central value
....
y(n) =last value

Table 9.1 Algorithm of noise calculation

screen, and click on the *noisereg* button. This will set NOISF1 and NOISF2 to the displayed region.

The parameter SINO is sometimes used to control the duration of an acquisition. While normally the number of scans (NS) determine, when acquisition is terminated, in such experiments measuring continues until a desired signal-to-noise ratio is achieved. It may be specified by setting the parameter SINO accordingly (use the edp parameter editor). An example is the AU program *au_zgsino*, which is part of the XWIN-NMR distribution. Since signal and noise regions are not known in advance for arbitrary samples, XWIN-NMR provides the following special conventions.

If NOISF1 and NOISF2 are both zero, the first 1/16th of the spectrum is used as the noise region. If both, SIGF1 and SIGF2, are zero, the maximum value is searched for in the whole spectrum, minus its first 1/16th part. You can force the program to exclude solvent or other signals from this region, depending on the nucleus and the solvent. The spectral regions to be excluded for a given NUCLEUS and SOLVENT must be specified in a text file, whose name is built from these two parameters: *<NUCLEUS>.<SOLVENT>*, e.g. *13C.Aceton*. The file must be located in the directory

/u/exp/stan/nmr/lists/scl/.

For the most convenient nucleus/solvent combinations, such region files are distributed with XWIN-NMR in the directory

/u/exp/stan/nmr/lists/scl.exam/.

Its contents are copied to the *scl* working directory during execution of the command expinstall. Whenever the working directory *scl* contains a suitable region file, signal exclusion according to the regions contained in the file is enforced automatically. The structure of a regions file is identical with the structure of *intrng* files generated by the command abs, and is described in the chapter *The File menu*.

The command sino uses the magnitude spectrum to calculate the signal-to-noise ratio. Correct phasing of a spectrum is therefore not required for sino to work properly. You may, however, force the program to perform its calculation on the spectrum's real part rather than on the magnitude spectrum by entering the command sino real on the keyboard. Assume you have applied data manipulations to the spectrum affecting the real part only, e.g. a baseline correction. The imaginary part then is no longer valid, giving a wrong magnitude spectrum and therefore a wrong SINO value. Use sino real in this case, or execute a Hilbert transform ht before calling sino in order to re-generate a correct imaginary part.

The commands sino noprint and sino real noprint calculate and set the status parameter SINO without displaying the result (you may examine the result by typing 2s sino, or dpp). The *noprint* option is automatically set when including a sino command in an AU program.

9.5 Deconvolution

The commands ldcon, gdcon, and mdcon decompose a specified spectral region into individual signals by approximating the experimental spectrum in this region with a sum of Lorentzian, Gaussian, or mixed Lorentzian/Gaussian lineshapes, respectively. You should be familiar with peak picking (pp) before trying to deconvolve a spectrum.

Please prepare deconvolution as follows (skip item 2 for mdcon):

1. Define the desired spectral range from the currently displayed region by choosing Define region from the *Analysis->Peak picking* menu, or from the *Analysis->Deconvolution* menu.
2. Define the MI, MAXI, and PC peak picking parameters. Execute a pps peak picking command and check whether the output contains exactly the lines you want to include in a deconvolution. Exclude peaks that are too big or too small by adjusting MI and MAXI. Increase or decrease peak picking sensitivity with

respect to noise by adjusting PC. This will also enable you to include/exclude peak shoulders or broad lines.

3. Set up the parameter AZFW with a number in ppm units. AZFW influences the way deconvolution is performed. The spectral region you have selected for deconvolution may contain peaks or peak groups separated from others in that they fall back under noise level before the next group starts. You may force the deconvolution commands to fit these peaks or groups separately, as if you were to execute an individual deconvolution of each of them. For this case set AZFW=0. Otherwise, if AZFW is larger than 0, it determines how far the peaks or groups may be apart so as to still consider them as a whole region to be deconvolved together. A recommended value is AZFW=0.5ppm. While AZFW=0 executes significantly faster, AZFW>0 will give a better fit of the peak edges and of the region between the signals or signal groups.

4. Set up the so-called *second data set* using the command edc2. The result of the deconvolution, which is the sum of iterated Lorentzian or Gaussian lines, will be stored there. Since this data set is displayed above the current data set in *Dual Display* mode, you can compare the two spectra after termination of ldcon, gdcon or mdcon by simply clicking on Dual display in the *Display* menu. Usually, one uses the next available PROCNO for the second data set, while taking over the remaining parameters NAME, EXPNO, DU, and USER from the current data set. It is convenient to store a series of deconvolutions by incrementing the PROCNO numbers. This facilitates comparing the spectra using the Incr. PROCNO commands in the *Display* menu of the dual display routine.

5. Upon termination of a fit, XWIN-NMR will output the result (the position/intensity/integral of the iterated lines). Set up the parameter CURPRIN using the command edo to define the output device for the fit result (CURPRIN=*$screen*=output to screen, CURPRIN=*$<printer>*=output to specified printer, CURPRINT=<*file path name*>=output to a file).

6. Now you are ready to start deconvolution with ldcon or mdcon. Caclulation speed depends on the selected spectral region, and of the number of lines therein. The iteration cycles are displayed on XWIN-NMR's status line, informing you that the calculation is still in progress.

As opposed to the commands ldcon and gdcon which pick the peaks to be deconvolved automatically based on the peak picking parameters, mdcon extracts the signals from a file. The file must contain one line per peak, specifying peak start,

peak center, and peak end. In addition, since mdcon supports mixed Gaussian/ Lorentzian line shapes, you must specify the desired Gaussian fraction (0...1). The name of the file is *peaklist* (see Table 9.2). It must be located in the current data

#frequency	half width	%gauss/100.
20969.627	5.18	0.8
20962.563	3.74	0.0
20960.182	2.02	0.1
20957.709	4.03	0.5
20950.609	5.18	1.0

Table 9.2 Example of a *peaklist* file:

directory (where the spectrum file is). The simplest way to create the file is with the command ppp (available in the *Analysis->Peak picking* menu), after having defined the spectral region as descibed above. Then you can modify the file contents using your preferred text editor.

9.6 The T1/T2 relaxation menu

9.6.1 Introduction

The T1/T2 relaxation menu has its name for historical reasons. However, in addition to T1/T2, this XWIN-NMR version allows fitting of data points from a variety of experiments.

FCTTYPE	Experiment
uxnmrt1t2	T1 or T2
invrec	inversion recovery
satrec	saturation recovery
cpt1rho	CP T1rho/TIS
expdec	exponential decay (T2 or T1rho)
gaussdec	Gaussian decay
lorgauss	combined Lorentzian/Gaussian decay
linear	linear
varbigdel	diffusion: variable Big Delta
varlitdel	diffusion: variable Little Delta
vargrad	diffusion: variable Gradient

Two different fitting algorithms are available. The first algorithm estimates the start parameters from the data points. This algorithm only works where one single component is the source of the intensity of the data points. The commands to fit these types of experiments are ct1, dat1, ct2 and dat2.

Command	FCTTYPE	Number of Components	Equation
ct1, dat1	uxnmrt1t2	1	I[t] = I[0]+P*exp(t/T1)
ct2, dat2	uxnmrt1t2	1	I[t] = I[0]+exp(t/T2)

The second algorithm is a Simplex method. This algorithm can fit data where one or more than one component contributes to the intensity of the data points. The fit command is always simfit and the following fit equations are currently implemented.

FCTTYPE	Number of Components	Equation
uxnmrt1t2	1	I[t] = I[0]+P*exp(t/T1)
invrec	1 - 4	I[t] = I[0]*(1-2A*exp(-t/T1))
satrec	1 - 6	I[t] = I[0]*(1-exp(-t/T1))
cpt1rho	1 - 4	I[t] = I[0]/(1-TIS/T1rho)*(exp(-t/T1rho)-exp(t/TIS))
expdec	1 - 6	I[t] = I[0]*exp(-t/T)
gaussdec	1 - 6	I[t] = I[0]*exp(-SQR(t/T))
lorgauss	1 - 3	I[t] = IL*exp(-t/TL)+IG*exp(-SQR(t/TG))
linear	1 - 6	I[t] = A+B*t
varbigdel	1 - 6	I = I[0]*exp(-D*SQR(2*PI*gamma*G*LD)*(BD-LD/3)*1e4)
varlitdel	1 - 6	I = I[0]*exp(-D*SQR(2*PI*gamma*G*LD)*(BD-LD/3)*1e4)
vargrad	1 - 6	I = I[0]*exp(-D*SQR(2*PI*gamma*G*LD)*(BD-LD/3)*1e4)

The fitting procedure consists of two basic steps.

g) The data points must be extracted from a data set. This step is described in Section 9.6.2 (Picking points).

h) The variables of the fit equation are calculated. This step is described in Section 9.6.4 (The fitting procedure)

For both steps all relevant parameters are set with the command edt1.

9.6.2 Picking points

Points for fitting can be extracted from three different data types.

a) from a single FID
A typical example is the Carr-Purcell experiment (pulse program cpmg)

b) from rows of a 2D experiment
A typical example is the pulse program t1ir where several FIDs are acquired into a 2D series file. This ser file is processed with xf2 and phase and baseline corrected. The points as either intensities or as integrals are extracted columnwise from each row of the 2D data file.

c) from an ASCII file
You can enter the x- and y-coordinates of the data points in an ASCII file using your preferred text editor.

9.6.3 pft2 - Pick points from a single FID

You do not need to process the FID. You simply switch to the T1/T2 relaxation menu (*Analysis->Relaxation*). In this menu you select the edt1 command. The following parameters must be set.

NUMPNTS
Enter the number of points to be picked from the FID. A maximum of 256 points can be fitted.

LISTTYP

a) Enter *dw* if you want to use the dwell time DW for the calculation of the x-axis time values.

b) Enter *auto* if you want to use individual time values on the x-axis. In this case the parameters X_START and LISTINC must be set.

START
Enter the first point to be picked. The first point of the FID has the number 0. But you can start the point picking at any point in the FID.

INC
Enter the increment between the picked points. INC = 1 means that NUMPNTS are picked starting with START. INC = 2 means that every second point starting with START is picked until NUMPNTS are reached.

X_START
Enter the start value of the x-axis. Usually you will set X_START to 0 but you can introduce an offset by setting X_START to a value greater than 0. X_START is used only if LISTTYP = auto.

LISTINC
Enter the increment between two adjacent data points. LISTINC is independent from INC. The correct distance between points picked with INC > 1 is calculated automatically. LISTINC is used only if LISTTYP = auto.

After you have set all relevant parameters, leave the editor with SAVE and execute the command pft2. The display should be updated with the picked points (see Figure 9.3). All points belong to one series or, if you compare this case with the one described in Section 9.6.3.1, to one peak. You can now immediately proceed with the fitting described in Section 9.6.4.

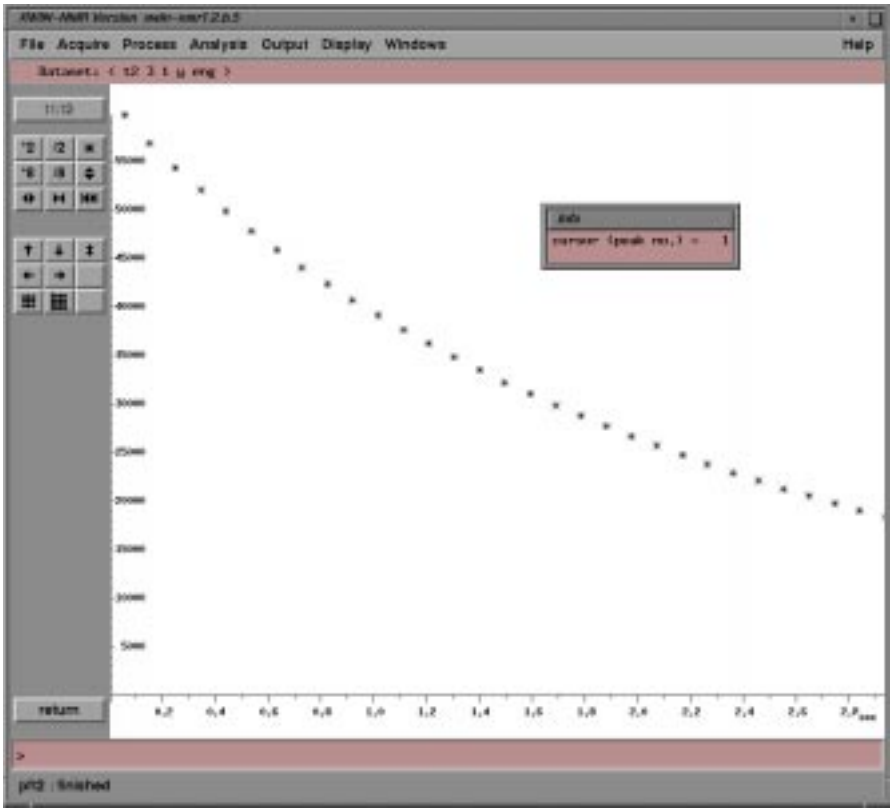


Figure 9.3 The display after executing pft2

9.6.3.1 pd and pd0 - pick points from rows of a 2D experiment

- ***edt1* - Setting up the parameters, *rspc* - read a slice from the 2D**

  The 2D series file must be processed at least in F2 dimension. The data should be properly phased and baseline corrected. Having done this, switch to T1/T2 relaxation menu (*Analysis->Relaxation*). Enter the command edt1 (*Process->edt1*) and set the following parameters.

  NUMPNTS

Enter the number of rows from which you want to pick points. When you switch to the T1/T2 menu on this data set for the first time, then NUMPNTS is automatically set to the processed number of rows (SI{F1}). You can reduce NUMPNTS if you want to restrict the number of rows from which data points are picked.

FITTYPE

a) Select *intensity* if you want to fit the peak maxima.

b) Select *area* if you want to fit the peak integrals.

LISTTYP

a) Enter *vdlist* if you want to extract the x-coordinates from the vdlist file stored in the acquisition data directory. If the file *vdlist* is not found there, the file specified in the status acquisition parameter VDLIST is read from the */u/exp/stan/nmr/lists/vd* directory. If this file cannot be found either, an error message is printed and the point picking command is terminated.

b) Enter *vplist* if you want to use this file. The handling is the same as for the case *vdlist*.

c) Enter *vclist* if you want to use this file. The handling is the same as for the case *vdlist*.

d) Enter *dw* if you want to use the dwell time DW for automatic calculation of the x-coordinates.

e) Enter *auto* if you want to use individual time values as x-coordinates. In this case the parameters X_START and LISTINC must be set.

f) Enter any filename in which the x-coordinates are stored. The file must reside in the acquisition data directory.

DRIFT

Enter the number of drift points. The maximum of a peak will be picked if it lies within DRIFT points of the position specified for the point picking.

START

Enter the row number on which you want to define the point positions where later the peak maxima will be searched. Normally you will set START to 1.

INC

Enter the row increment for the point picking. Normally you will set INC to 1 which means that the data points are picked from the rows in ascending order.

X_START

Enter the start value of the x-axis. Usually you will set X_START to 0 but you can introduce an offset by setting X_START to a value greater than 0. X_START is used only if LISTTYP = auto.

LISTINC

Enter the increment between two adjacent data points. LISTINC is used only if LISTTYP = auto.

After you have set all relevant parameters, leave the editor with SAVE and execute the command rspc. This will read the selected slice (START) from the 2D file and you will be put into the main 1D processing menu.

- **Create the *baslpnts* (baseline points file) for intensity fitting**

  You must now define the positions from where peak maxima will be picked along columns from all rows of the 2D data set. Switch to the interactive baseline correction menu (*Process->Special processing->Baseline correction* or type basl). Click on define points. The cursor will be put onto the spectrum. Move the mouse cursor to the top of a peak and click the middle mouse button. This selects this peak position for the point picking on the 2D data set. Move the cursor to the next peak and click again the middle mouse button. You have selected a second position for the point picking. After you have selected all positions, click the left mouse button. The cursor is now free again and you can click on return to return to the main 1D processing menu. You can also do this at any stage during the selection of peaks, for instance if you want to select a different area for the point selection or if you want to zoom into a region for a more accurate point selection. Re-enter the baseline menu, click again on define points and answer the question with a (for append) to append more points to the already selected ones. Proceed from there as decribed above. After you have selected all positions where you want to pick points, you click the left mouse button to free the cursor followed by a click on return to return to the main 1D processing menu. You can now proceed with the next chapter if you want to define an integral range file for area fitting. Or you can immediately return to the T1/T2 menu (click on *Analysis->Relaxation*). Without an integral range file, the commands pd and pd0 will complain about the non-existence of the *intrng* file. You can ignore this warning if you don't intend to do an area fit at a later stage.

- **Create the *intrng* (integral range) file for area fitting**

Switch to the interactive integration menu and define integrals around all peaks which you have selected for the maximum search in the previous step. For more details on how you define integrals, see Chapter 2.9.2 of the XWIN-NMR Manual. A slope and bias correction of the displayed integrals is not necessary, since each row of the series file has already been Fourier transformed, phased, and baseline corrected in the F2 dimension. After all integral ranges have been defined, return to the main 1D processing menu. From there, switch to the T1/T2 relaxation menu (*Analysis->Relaxation*).

- **ppt1 - Create *baslpnts* and *intrng* file automatically**

  An alternative to the steps described in the two previous chapters is the command ppt1. This command performs a peak picking like the pps command (peak picking on screen) which is described in Chapter 2.10 of the XWIN-NMR Manual. But in addition to the report box shown on the screen, the two files *baslpnts* and *intrng* are written into the processing data directory. These files can now be used immediately for the next step, the actual maximum search in the 2D data set. Return to the T1/T2 relaxation menu by clicking on *Analysis->Relaxation*.

- **pd - Pick peak maxima along columns from all rows**

  Enter the command pd to pick the maxima along the defined columns from all rows of the 2D data set. A peak picking is done in each row of the 2D data file. A maximum will be found

  a) if at the given position the peak maximum lies within the allowed drift range and

  b) if the peak maximum is bigger than MI (minimum intensity) taking into account the peak picking sensitivity constant PC.

  At the same time, all data points within the specified integral range will be added up (if the file *intrng* exists).

  The display will be updated with the maxima which were picked at the first peak of each row of the 2D data set. The total number of peaks is given by the number of positions defined in the *baslpnts* file. Each series of maxima (points) for one peak consists of at the most NUMPNTS points. A little window on the screen will contain information about the currently displayed peak (number and position). If you want to see the next peak, type nxtp (see Section 9.6.4.2).

  It is possible that maxima at a certain position in one or several rows of the 2D

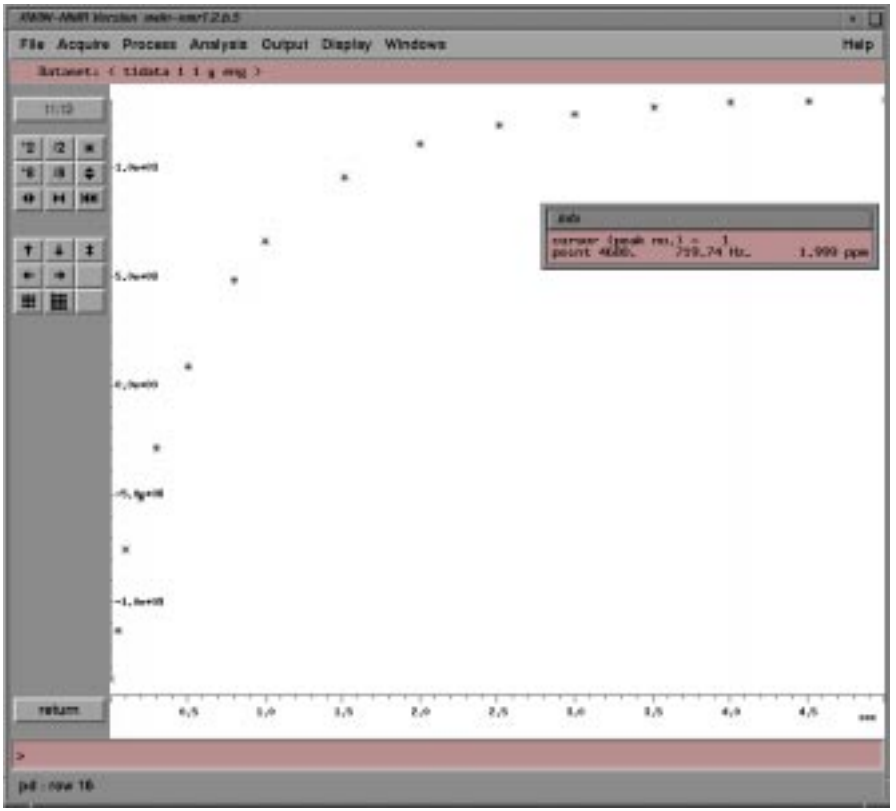


Figure 9.4 The display after executing pd

data set were not found by the maximum search with pd. Please check the following in such a case.

a) Is the intensity at this position 0 or almost 0. If so, the peak picking routine cannot find the peak and there is nothing you can do.

b) The point (or the maximum of the peak) lies outside the allowed drift range. Increase the number of drift points DRIFT (command edt1) and execute the command pd again.

c) The point is the maximum of a relatively small peak or is not fully resolved from an adjacent peak. Type PC and decrease the peak picking sensitivity

constant. Execute the command pd again and check whether the point was now found. If not, decrease PC even further and execute the pd command again.

d) It could be that MI (the minimum intensity) for peak picking is set too high. Type MI and set it to 0 and execute the command pd again.

- **pd0 - Pick all points at the exact same position from all rows**

  The pd0 command is an alternative to the pd command. The values of DRIFT, MI and PC are irrelevant when you use the command pd0. No peak picking (maximum search) is done. Instead, the intensity values in the rows of the 2D data file at the positions defined in the *baslpnts* file are simply read. If the peaks do not drift at all, then the result of the commands pd and pd0 is the same. The pd0 command can be used if points to be picked lie on the shoulder of bigger peaks. The normal maximum search will not be able to pick these peaks. But bear in mind that in such cases the intensity value might not be the real height of the peak. And this might obscure your fitting results.

9.6.3.2 Pick points from an ASCII file

The only thing you have to do is to set up an ASCII file named *t1ascii*. This file must reside in the processing data directory. The format of the file *t1ascii* is the following.

1.	line	: SIMFIT	<number of peaks>			
2.	line	: x(1)	y1(1)	y2(1)	y3(1)	y(number of peaks)(1)
3.	line	: x(2)	y1(2)	y2(2)	y3(2)	y(number of peaks)(2)
4.	line	: x(3)	y1(3)	y2(3)	y3(3)	y(number of peaks)(3)
.		.	.	.	.	.
.		.	.	.	.	.
.		.	.	.	.	.
.		.	.	.	.	.
(n+1)th	line	: x(n)	y1(n)	y2(n)	y3(n)	y(number of peaks)(n)

The maximum number of peaks for fitting with simfit asc is 50. Once the file *t1ascii* is set up, you can immediately continue with the fitting procedure described

in Section 9.6.4.5.

9.6.4 The fitting procedure

9.6.4.1 General information

The parameter CURPRIN determines where you will get the output of your fitting results. Enter the command edo and set CURPRIN.

$screen	If you want to see the output on the screen.
$printer	If you want to get output on the specified printer.
filename	If you want to get the output in a file named filename. The file will be created in the current processing data directory. This option is not really needed because an output file called *ct1t2.out* or *simfit.out* with all results is always written to the processing data directory.

After you have fitted a series of data points, the fit parameters are stored in the edt1 editor. Type edt1 and click on EDPARM to see the fitted parameters. The same parameters are, of course, also documented in the output file created by the fit command.

9.6.4.2 **nxtp** - Display the next peak

If you have picked the points with the command pd or pd0 or if the points are read from the ASCII file *t1ascii*, then there might be more than one series of data points, each belonging to one peak position. In this case, only one series of data points will be displayed on the screen at a time. If you want to inspect the next series of data points, type nxtp. If nxtp is used on the last available series of data points, then the first series of data points will be displayed (wrap around). A little window on the screen will contain information about the currently displayed peak.

Every time you use the nxtp command, the parameter CURSOR will be updated.

The command nxtp will always delete the fit curve (if there is one on the screen).

The command nxtp will always display the next series of data points at maximum vertical expansion. A subsequent fit with simfit will rescale the point and fit display. This is done because simfit scales all data points from all positions to their

global maximum. The commands ct1, dat1, ct2 and dat2 only read one series of data points and scale to their maximum.

If you have picked the points with the command pft2, then all available data points are displayed on the screen because they all belong to one series. In this case, the command nxtp is not applicable.

9.6.4.3 ct1, dat1, ct2, dat2

You can use these commands if the following two conditions apply.

a) The data points were picked with the commands pft2, pd or pd0.

b) The intensities of the data points originate from a single component.

The commands ct1, dat1, ct2 and dt2 do not require a specific parameter setup. Nevertheless, these commands update certain parameters. If you want to use the command simfit after you have used ct1, dat1, ct2 or dat2, make certain to set the parameter FCTTYPE properly (and check other parameters as necessary) for the simfit run (see Section 9.6.4.4).

The commands ct1 and ct2 will fit the data points currently displayed on the screen. If there is another series of data points, you can switch to it with the command nxtp (see Section 9.6.4.2). You can now type again ct1 or ct2 to fit this series of data points. If you want to switch to a specific peak number, enter edt1 and set the parameter CURSOR to the desired peak number. Leave the editor with SAVE and type ct1 or ct2. The display will be updated with this peak and its fit curve.

If you want to fit all peaks, then use the command dat1 or dat2.

If you have defined integral ranges, then toggle the parameter FITTYPE from *intensity* to *area* if you want to fit the peak integrals. Any command typed in next will now work on the so selected fit type. If you want to fit the peak maxima, toggle FITTYPE back to *intensity.*

Important notes :

The commands nxtp, ct1, dat1, ct2 and dat2 will always show the points and the fit curve in full scale mode. The display is scaled according to the maximum and minimum of the data points of the current peak.

The simfit commands will always show the data points and the fit curve in an

absolute scaling mode. The display is scaled according to the maximum and minimum of the data points of all peaks.

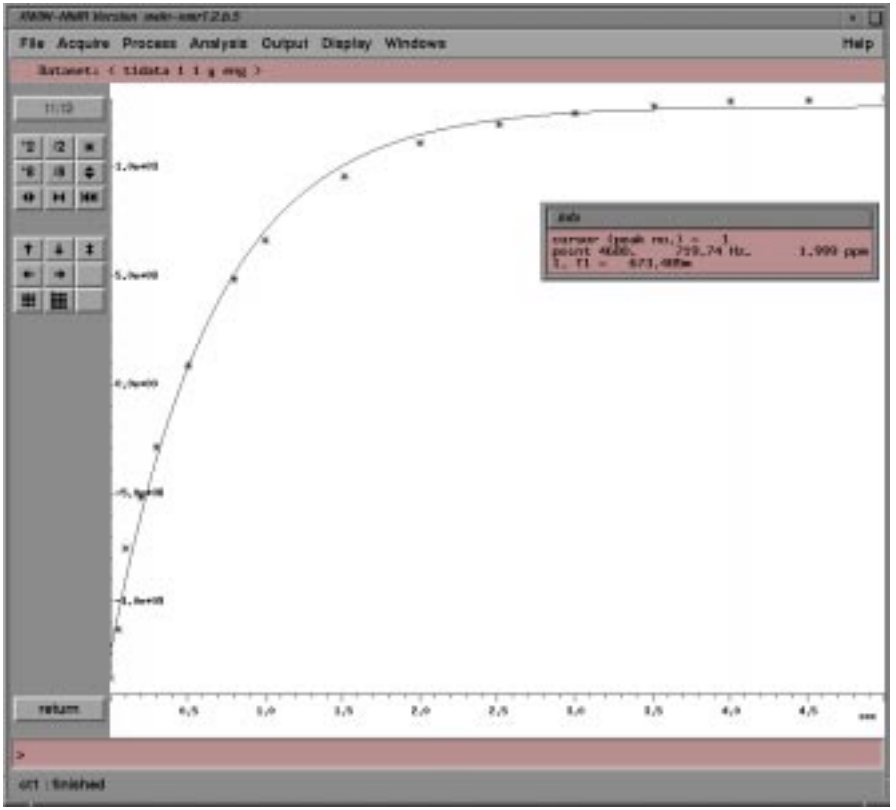


Figure 9.5 The display after executing ct1

9.6.4.4 simfit

The following parameters must be set in the editor invoked with edt1.

CURSOR
Select the peak you want to fit. Normally, you will set CURSOR to 1. You can later switch to other peaks with the command nxtp.

FITTYPE
a) Select *intensity* if you want to fit the peak maxima.
b) Select *area* if you want to fit the peak integrals.

FCTTYPE
Select the appropriate fit function.

COMPNO
Set the number of components which contribute to the intensities of the data points.

EDGUESS
Set an initial estimate for the parameters to be fitted. Make certain that the sum of all I[0] values is not bigger than 1. If COMPNO was set to 1, then the best guess for I[0] is usually 1.

The parameter NUMTERM is set automatically and depends on the selected fit function. NUMTERM determines the maximum number of components which can be fitted. The following rule applies.

COMPNO(maximum) = 12 / NUMTERM

After all parameters are set, save your changes with SAVE and type

a) simfit if you want to fit the currently displayed points or

b) simfit all if you want to fit all peaks with one command.

The command nxtp (see Section 9.6.4.2) switches to the next peak and updates the parameter CURSOR.

Important notes :

The simfit commands will always show the data points and the fit curve in an absolute scaling mode. The display is scaled according to the maximum and minimum of the data points of all peaks.

The commands nxtp, ct1, dat1, ct2 and dat2 will always show the points and the fit curve in full scale mode. The display is scaled according to the maximum and minimum of the data points of the current peak.

For each variable from the fit equation you must enter an initial start value for the iterative refinement. If you have more than one component, then each variable

appears as often as you have components. The sum of all I[0] guesses should not be greater than 1 because simfit rescales all data points of all peaks to a maximum of 1. You must also enter an initial step rate with which the variables are modified. A good guess for the step rate is 10% of its corresponding variable. The editor defaults to this value automatically if you set the step rate to a value bigger than its corresponding variable. During the iterative refinement, the step rate is adjusted according to the change of the variable.

If you set the step rate of a parameter to 0, then the variable is not iterated (kept constant). This feature can be used, for instance, if you want to fit a two component system with the combined Lorentz/Gauss function. But for the first component, the Gaussian contribution is 0 and for the second component, the Lorentzian contribution is 0. In the EDGUESS menu, set the parameters as follows.

GC1IL	0.5	SC1IL	0.05
GC1TL	0.1	SC1TL	0.01
GC1IG	1e-20	SC1IG	0.0
GC1TG	0.5	SC1TG	0.0
GC2IL	1e-20	SC2IL	0.0
GC2TL	0.5	SC2TL	0.0
GC2IG	0.5	SC2IG	0.05
GC2TG	0.05	SC2TG	0.005

Wrong guesses can lead to completely wrong parameters. This is an inherent SIMPLEX problem. Always check the RSS (root mean square) and SD (standard deviation) values after a simfit run. If you are in doubt about the fit, change one guess (or more) by at least a factor of 5 and repeat the fit. If your results remain the same, change the guess or guesses by a factor of 5 in the opposite direction, repeat the fit and compare all three results. Usually, the match between the points and the fitted curve will allow you a good judgement of the quality of the fit.

9.6.4.5 simfit asc - Fit data from the ASCII file *t1ascii*

You have to set the same parameters as described for simfit (see Section 9.6.4.4).

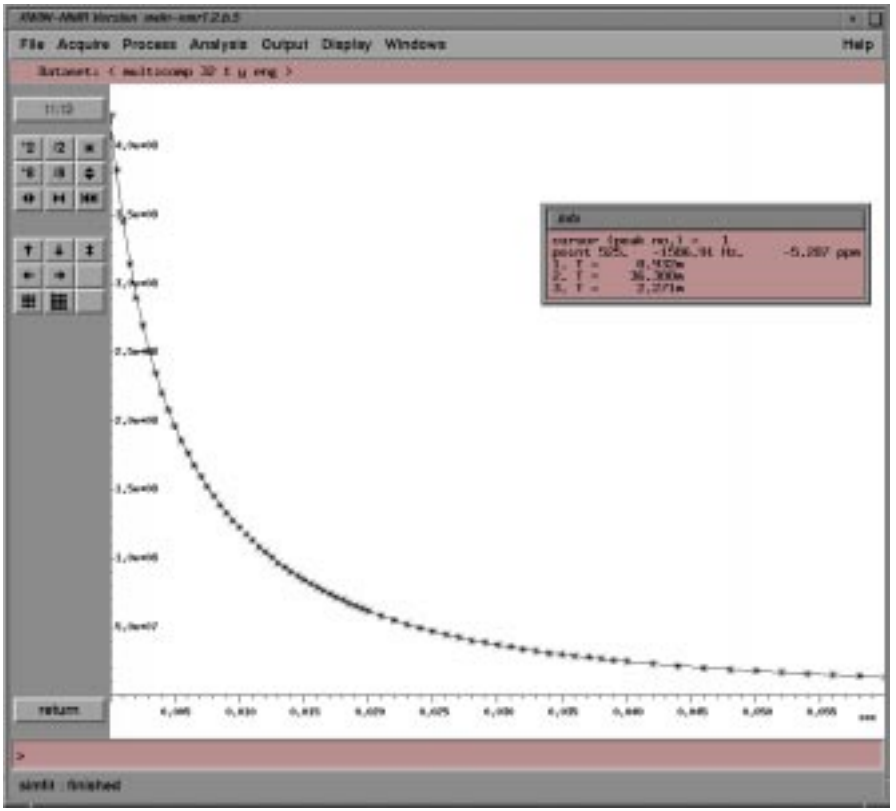


Figure 9.6
The display after executing simfit. Three components contribute to the peak intensities (COMPNO = 3).

The command simfit asc will do the following.

a) It will read the file *t1ascii*.

b) It will create all auxiliary files (*t1_disp*, *t1elim*, *t1ints* and *t1peaks*) which are otherwise created by the commands pd, pd0 or pft2. But these commands are not applicable if you want to fit from the ASCII file *t1ascii*.

c) It will update the display with the points of the peak specified by the parameter CURSOR.

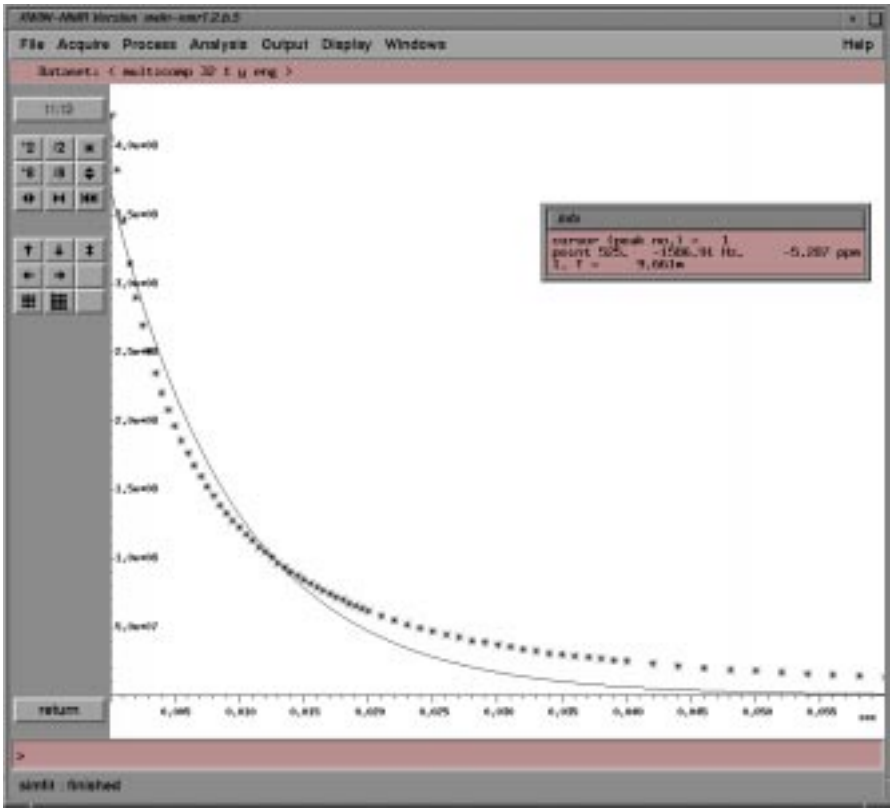


Figure 9.7
The display after executing simfit. Three components contribute to the peak intensities but only one was specified in edt1 (COMPNO = 1). This serves basically also as an example for a bad guess.

d) It will fit the data and display the fit curve.

You can use simfit asc all to fit all peaks with one command.

The command simfit asc works exactly like the command simfit. After you have typed simfit asc once, you can switch to the next peak with the command nxtp. The new peak can now be fitted by typing simfit or simfit asc. You can also switch to a specific peak by setting the parameter CURSOR. Again, you can fit this peak

either with simfit or with simfit asc.

Here is an example you can try to fit. Enter the following lines in the file *t1ascii*.

SIMFIT 2

0.1	-99	-48
0.5	-90	-44
1.0	-81	-39
2.0	-64	-33
3.0	-48	-23
4.0	-34	-17
6.0	-10	-6
8.0	10	5
10	27	13
15.0	55	26
20.0	73	38
40	97	50

Use the following parameter setup in edt1 to fit the above data points.

FCTTYPE	:	invrec
COMPNO	:	1

Click on EDGUESS and enter the following guesses.

GC1I0	:	1.0		SC1I0	:	0.1
GC1A	:	1.0		SC1A	:	0.1
GC1T1	:	1.0		SC1T1	:	0.1

Leave the editor and type simfit asc. The display should be the same as the one shown in Figure 9.8. Type nxtp and then simfit. The display should be the same as the one shown in Figure 9.9.

Important notes :

Simfit and simfit asc cannot be used alternatively if a *baslpnts* and a *t1ascii* file exist and the two files contain different numbers of peaks.

If you have placed the *t1ascii* file into a processing data directory where you previously used pd, pd0 or pft2 to pick points, then the simfit asc command will overwrite all auxiliary files created by these commands. With the command pd, pd0 or pft2 you will, on the other hand, overwrite the auxiliary files created by the simfit asc command. This does not really present a problem - you can switch between different ways of picking points this way. But be aware of this alternating overwrite situation.

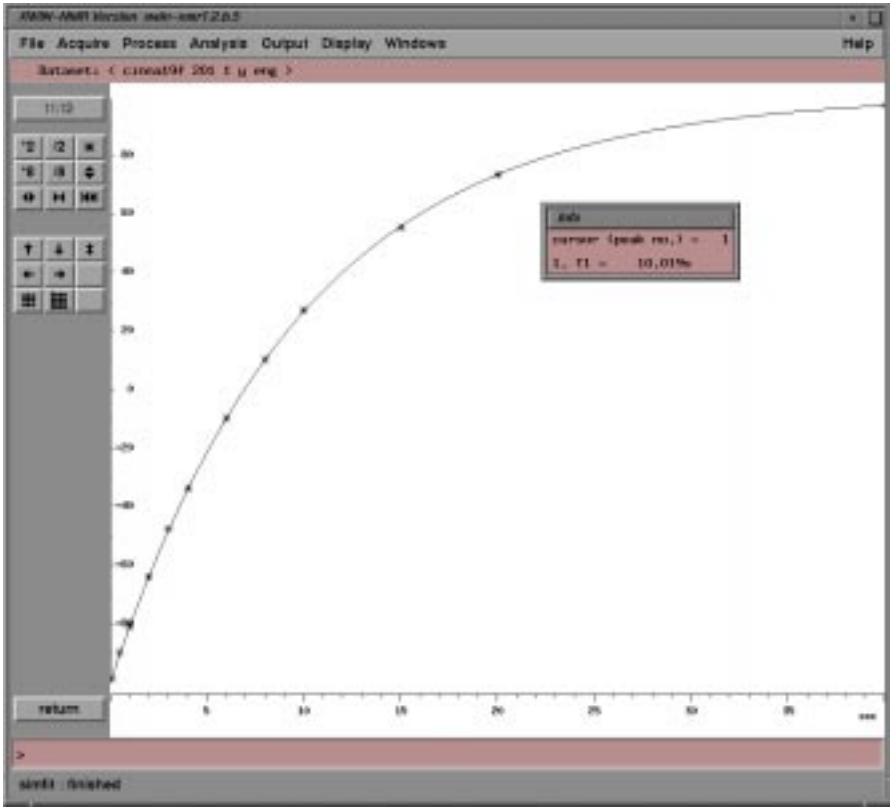


Figure 9.8 The display after the first simfit asc command

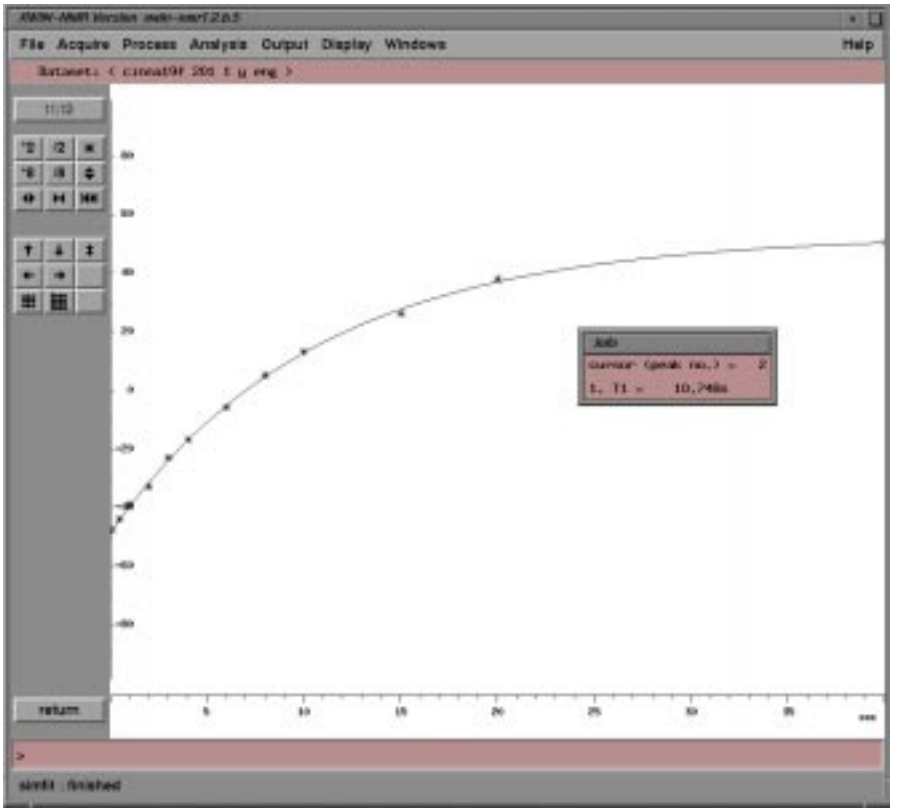


Figure 9.9 The display after the second simfit command

9.6.5 Interactive manipulations

9.6.5.1 General information

The interactive manipulation of the displayed points and the fit curve works exactly like the interactive manipulation of 1D data sets.

If you have managed to shift all points (and the fit curve) outside the display region, then clicking the following three icons will always bring the data points back into the display region.

Click on :



9.6.5.2 Interactive elimination of data points

If you want to temporarily eliminate points from fitting, click the left mouse button in the data display area. The mouse cursor will be put onto the nearest point. You can now move the cursor along the points. Clicking the middle mouse button on a point will eliminate the point temporarily. The cursor is freed again afterwards. To eliminate another point, repeat the above procedure. If you want to fit the remaining data points, type the corresponding fit command. If you want to get the eliminated points back, type rspc (restore eliminated points). rspc will only bring back the points eliminated from the currently displayed peak. Eliminated points will remain eliminated until you type rspc. Switching between different peaks will not bring eliminated points back. Eliminated points are either marked with ******* in the output listing (commands ct1, dat1, ct2 and dat2) or do not appear (command simfit).

9.6.6 Supplementary commands

prxy	prints the list of x,y pairs currently displayed on the screen (CURPRIN is used).
lstp	lists the complete information for the points of the currently displayed peak (CURPRIN is used).
elim	eliminates a point from the currently displayed peak.
rstp	restores all eliminated points of the currently displayed peak.

9.6.7 Files

The following files are used by the T1/T2 commands. They all reside in the processing directory (PROCNO) of the data set. Depending on the fit function and the point picking command used, some files might not exist.

baslpnts	Contains the positions from where points are picked if you work with a 2D data file. *baslpnts* is an ASCII file and can be edited with the command edmisc baslpnts (see XWIN-NMR Manual Chapter 5.3.9).
intrng	Contains the left and right limit of the range around the position defined in the *baslpnts* file. Every position must have a corresponding range in the *intrng* file. Between the limits all data points are added up for area fitting. *intrng* is an ASCII file and can be edited with the command edmisc intrng (see XWIN-NMR Manual Chapter 5.3.9).
t1ascii	Contains data points in ASCII format for fitting with the command simfit asc.
ct1t2.out	Output listing of the commands ct1, dat1, ct2 or dat1.
simfit.out	Output listing of any simfit command.
t1par	Contains all parameters shown by the command edt1.
t1_disp	Contains the points currently displayed on the screen.
t1elim	Contains the information which points were eliminated.
t1ints	Contains all integrals of all peaks after the point picking (or maximum search) was done.
t1peaks	Contains all points of all peaks after the point picking (or maximum search) was done.

The auxiliary files *t1_disp*, *t1elim*, *t1ints* and *t1peaks* are created by the point picking commands. They cannot be edited because they are binary files.

9.7 Quantitative NMR [nmrquant]

9.7.1 Program requirements

The nmrquant command is required to perform reliable quantitative analysis of 1D spectra comprising complex overlapping spectra. The program must be able to :

- Perform mole ratio calculations given integral heights and number of protons per signal.

- Perform absolute weight calculations given the information detailed in (1) plus molecular weights of all components and the weight of a reference component.
- Permit analysis of spectra characterised by poor baseline.
- Run in automation.

9.7.2 Typical interactive session

The procedures for quantitative determination are described in example sessions for interactive and automated execution.

For quantitative analysis of a data set in interactive mode, the following procedure is adopted :

1. Select the 1D spectra and start the {nmrquant} program either by typing the command or by selecting from the pull down menu.

2. Click on the *Define* button on the left hand side of the display and you will be given a choice of defining the integral/baseline regions, the proton matrix file or all three. By selecting the *integral* option from the enumeration window, instructions for defining integral regions and reentering the program are displayed.

3. Define the integral regions (as discussed in the chapter *1D Data Processing Tutorial*) and store with the *Save as 'intrng' & return* integral menu item. Reenter the program by typing the command nmrquant on XWIN-NMR's command line, or by calling it from the *Analysis* menu. The main display now shows the number of regions defined on the right hand side.

4. Click on the *Define* button on the left hand side of display and select the *baseline* option from the sub window. Again, instructions for defining base line regions and reentering the program are displayed.

Internally, the integral regions defined in step 3 are renamed to prevent being overwritten by the *intrng* file which is set up when defining the base line regions. The original integral file is restored after renaming the base line *intrng* file defined in step 4 to *basereg*.

5. Click on the item *Define Components* to set up the matrix describing the proton contribution from each component in each integral region. If a matrix file already exists in the processed data directory, you will be asked whether you wish to define a new matrix or edit the existing file. For new matrix files, you must first define the number of components which contribute to the signals

within the integral regions. In order to arrive at a solution, the number of components must be greater or equal to the number of integral regions. Next, fill in the matrix with the number of protons and optionally, the component names.

After defining the number of protons, select either the *Continue* button to return to the main nmrquant window or the *Define Weights* button to for defining component molecular weights and reference weights for weight calculations. If the define weights button is selected, then all component molecular weights as well as a reference sample weight must be defined otherwise only mole ratios will be calculated.

6. On returning to the main nmrquant window (by selecting the *Continue* button from both proton contribution and component weight dialogue windows), the files can be saved for later recall by selecting the *Save* item on the left hand display. As in the case of the define option, the choice of saving the integral, baseline, matrix or all three files is offered.
7. Finally, click on the process button to perform the calculation. The output is redirected to the device specified in the edo parameter CURPRIN.

9.7.3 Typical automated execution

nmrquant can be easily incorporated into AU-programs thus making automated analysis of a range of similar samples possible. To run the program in automation mode, the user must enter the command

nmrquant <filename>.

The program then searches for the files

/u/exp/stan/nmr/lists/intrng/<filename>
/u/exp/stan/nmr/lists/basereg/<filename>
/u/exp/stan/nmr/lists/nmrquant/<filename>

for the integral region, baseline region and proton matrix information.The integral region and proton matrix files must be readable but the baseline region file need not exist for the analysis to proceed.

9.7.4 Input specification

The basic information required by nmrqant is stored in 2 files - the *intrng* file and the *nmrquant* file which must be stored in the processed data directory (where the

spectrum is stored).

The *intrng* file must contain at least two entries and may be of the A, P, H type. A W type *intrng* file is invalid as it contains no explicitly defined integral regions. See the section on integration in the chapter *The File Menu* for more information about *intrng* file formats.

The *nmrquant* file contains the proton contribution to each integral region from a given (named) component. This file is set up automatically when running the nmrquant command and the user need not worry about the file format which is defined according to Table 9.3.

n # number of components
m # number of integral regions
comp 1 id:<comp 1 mol wt>:<comp 1 weight>
....
comp n id:<comp n mol wt>:<comp n weight>
np(region 1):..:np(region m) # Proton contributions for comp 1
np(region n):..:np(region m) # Proton contributions for comp m

Table 9.3 Matrix file format

The first line contains the number of components to be analysed which must be greater or equal to the number of integral regions defined on the next line. The number of integral regions must match the number of integral regions defined in the *intrng* file otherwise the matrix becomes undefined.

The next n lines contain three fields reserved for component identifiers, molecular and component weights (in grams) separated by colons. Component identifiers contain the defaults *Sample 1, Sample 2,* etc. The molecular and component weight fields are empty by default and need not be set when calculating mole ratios. To peform weight calculations, *all* component molecular weights must be defined as well as the weight of at least one reference component.

The last part of *nmrquant* file consists of values for a matrix of dimension *number of integral regions* times *number of components*. The values represent the number of protons from each component for each integral region defined and each line consists of the number of protons for each integral region (separated by colons) for a given component.

The *intrng* and the *nmrquant* files contain the minimum information required by nmrquant. However, the program may also make use of the *basereg* file. The *basereg* file can be set up to define two regions of baseline on either side of the signals which are then used for slope and bias calculations rather than the regions immediately to the left and the right of each integral region.

Basereg files are defined in exactly the same way as integral regions i.e. by temporarily exiting the nmrquant program, and entering integration mode. Note that the program saves any existing *intrng* file which is then restored on re-entering the program. Within integration mode, instead of selecting regions of signal, regions should be defined which contain two regions of flat baseline. Note that the integral trails displayed on the screen should be ignored and correcting for slope and bias is not necessary. After defining the baseline regions and storing with the *Save to 'intrng' & Return* option, nmrquant must be restarted in order to handle the basereg and integral files correctly.

To avoid defining integral regions, baseline regions and proton information each time a different spectrum is processed, the user can save the files *intrng*, *nmrquant*, *basereg* for subsequent recall. Indeed, this must be carried out prior to running the program in automation mode. The procedure for saving integral ranges is well established within XWIN-NMR (wmisc) with files being stored in the directory

/u/exp/stan/nmr/lists/intrng/.

(See chapter *The File Menu*). nmrquant requires two additional directories

/u/exp/stan/nmr/lists/basereg/
/u/exp/stan/nmr/lists/nmrquant./

containing globally accessible baseline region and nmrquant workfiles. When used in automation, the *intrng* and *nmrquant* (and *basereg*, if required) files have the same name.

9.7.5 Output specification

All output from the nmrquant command is redirected to the file *quant* stored in the processed data directory. If the molecular weight and a reference weight is entered, an extended output is stored.

9.7.6 The user interface

The options list on the left hand side of the main nmrquant display can be divided into 4 distinct groups, namely

1. manual setup of parameters
2. reading parameters from disk
3. writing parameters
4. others

The first three groups can be considered as separate steps which must be performed sequentially in order to specify all the information required by the program. The last group of options are separate from the other functions as they relate to functions which are not involved with the computation.

Selecting the *Define Components* field replaces the main nmrquant display with a prompt for the number of components to be defined. If the program verifies the number entered is less than the number of number of integral regions, a figure is presented containing the proton contribution of each component for each integral range (default = 0). After entering proton values, the display is replaced with the Component identification dialog where additional information can be entered such as identification string, molecular weight and reference weight.

9.8 The *Simulation* submenu

The remaining sections of this chapter introduce a collection of simulation programs.

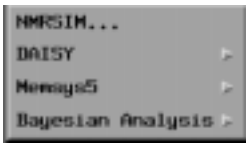


Figure 9.10 The *Simulation* submenu

9.9 NMRSIM

NMRSIM is a program for the simulation of NMR experiments. The user must

specify a spin system and a pulse program in the standard Bruker pulse program language for AMX/ARX or AVANCE series spectrometers. Based on the numerical solution of the quantum mechanical Liouville equation, NMRSIM applies the pulse program to the spin system and calculates the resulting Fids. A wide range of NMR experiments can be simulated, including selective excitation and magnetization transfer (HOHAHA, TOCSY, multiple quantum and filtered COSY, heteronuclear 3D). Mixtures can also be simulated. NMRSIM is equipped with a pulse program display facility and a Bloch simulator. It is ideally suited for both, scientific calculations and educational purposes.

NMRSIM stores the calculated data in XWIN-NMR format. When the calculation is finished, the result is automatically displayed by XWIN-NMR, and the user can apply a fourier transform and all other XWIN-NMR display, processing, and plotting utilities.

For details please refer to the NMRSIM manual.

9.10 DAISY

The DAISY program package was developed in cooperation with the Institute of Inorganic Chemistry I, University of Düsseldorf, Germany.
It allows simulation of 1D NMR spectra, taking into account general magnetical *and* chemical equivalence. No line assignments are required, because DAISY uses the original experimental line shape. Therefore, highly symmetrical spin systems can be iterated, where packages that require line assignments will fail due to the uncertainty of the assignments.
Parameter set-up and execution take place from within XWIN-NMR, and the generated data are immediately accessible from there. For example, the experimental and refined spectrum can be viewed in dual display mode.

DAISY is *not* a standard component of the XWIN-NMR package and must be purchased as a separate option. A complete manual is delivered with the program.

The command ded (available in the *Simulation->DAISY* menu) is the XWIN-NMR interface for the DAISY program package.

9.10.1 General hints for working with ded

9.10.1.1 The main menu

The main menu of the editor has the following contents and options :

- header line with the name of the program
- status of the editor with
  - selected program type
  - filename of the actually loaded data set
  - title of the data set
- menu line with the commands :
  - Edit : edit loaded set
  - Save : save data set to disk
  - Run : quit the editor and run the selected program (the programs run in background)
  - Quit : quit the editor

9.10.1.2 Select the program type

Use the *Daisy Program Type* button to specify the program for which parameters should be loaded. You must choose a program before a data set can be loaded and edited.

9.10.1.3 Select a data set

The editor uses the following from the XWIN-NMR foreground data set :

- the experiment name
- the experiment number
- the processing number

The full path is displayed in the header line. Internally, the programs have continuous numbers :

1 for SPECPREP
2 for DSYMPLOT
3 for DCYMPLOT
4 for DAVSYM1
5 for DAVSYM2
6 for DAVCYM1
7 for DAVCYM2

The concept of the directory naming convention is the following. The files of an NMR-experiment of the user *guest* are stored in

/u/data/guest/nmr/135TFB/5/pdata/4

which means:

the name of the experiment is : *135TFB*
the experiment number is : *5*
and the processing number is : *4*.

To make this experimental data set readable for the iterators, SPECPREP creates files in the correct format.

The name of a SPECPREP data set contains the following parts :

experiment name : same as the experiment
experiment number : e-number of experiment + 1000
processing number : p-number of experiment*10
+SPECPREP processing number

This results in the following path :

/u/data/guest/nmr/135TFB/1005/pdata/41

The additional SPECPREP processing number was introduced because one experiment can be manipulated by SPECPREP in several ways.

The path to store the results of a simulation with DSYMPLOT is generated as follows.

e-name : same as the experiment
e-number : expno (exp) + 2000
p-number : procno (exp)

This results in the following path :

/u/data/guest/nmr/135TFB/2005/pdata/4

The path to store the results of an iteration with DAVCYM2 is generated as follows.

e-name : same as the experiment

e-number : expno(exp) + 7000
p-number : procno (exp) * 1000 + SPECPREP-p-number * 100
+ DAVCYM2-p-number * 10 + output number (1/2/3)

The output numbers 1, 2 and 3 are generated by DAVSYM1/2 and DAVCYM1/2 automatically. They contain the following spectra :

/u/data/guest/nmr/135TFB/7005/pdata/4111 : experimental
/u/data/guest/nmr/135TFB/7005/pdata/4112 : first simulated
/u/data/guest/nmr/135TFB/7005/pdata/4113 : final output

The predominant rules for creating a path name are:

expno (all programs) = expno (exp) + program number * 1000
procno (SPECPREP) = procno (exp) * 10 + runprocno (SPECPREP)
procno (simulators) = procno (exp)
procno (iterators) = procno (exp) * 1000
+ runprocno (SPECPREP) * 100
+ runprocno (iter) * 10 + output number

If data sets exist for the selected program, you can load them by clicking on the *Experiment number* or the *Experiment processing number.* The existing numbers are displayed in a menu. If you select one of these numbers, the corresponding parameter set is loaded.

With *new* a new parameter file with the next available processing number is created. The new path name for this parameter file *params* is displayed in the header line.

9.10.1.4 Title

The title field displays NEW_FILE to indicate that you have not yet edited the parameters.

When you return to the main menu after you have edited parameters, the title is updated with the title specified in the *Miscellaneous parameters* menu (see : Input pages).

9.10.1.5 Edit

During an editor session, several pages of input are shown, depending on the type of program. For SPECPREP, you can set all relevant parameters on one page. For all other programs, you must enter the parameters on several pages.

Use the buttons :

next page : to display the next parameter page
prev. page : to display the previous parameter page
main menu : to return to the main menu page
comment : to open the UXNMR editor of your choice

The *comment* button allows you to write any information which you want to store together with the parameters into the *params* file. The DAISY programs ignore these comment lines until the first line with the first character not equal to 'C' or 'c' is found. From there on, the format is fixed (due to the nature of the Fortran programs). This means that a comment line or line with no input somewhere in the parameter section will terminate the program (probably with an error message). If you want to edit a *params* file by hand (in principle this is possible) you have to be very careful!

If all parameters are set, you must return to the main menu to save them. After saving, you can easily set up new parameters or modify parameters by incrementing the *Experiment processing number.* The parameters of the previously edited data set are used to initialize the new parameter file.

9.10.1.6 Save

With *save*, the editor creates two files for each data set.

params : contains the parameters for the selected program
callfile : contains all filenames needed for the run

If the data set does not exist, these files are stored to disk. If the data set exists already, you can either overwrite it or cancel the save operation. In either case, the program returns to the main menu page.

9.10.1.7 Run

The following files resulting from DAISY runs are written into the same directory

as the *params* and *callfile*.

.lst files	: the output results of a DAISY program run
1r	: the simulated data set
proc, procs	: processing parameter files
outd	: output device file for XWIN-NMR
meta, meta.ext	: the plot parameters for XWIN-NMR

9.10.2 Input pages for the programs

Parameters (written in capital letters and enclosed in brackets) refer to the original parameter names used in the DAISY programs. Detailed information can be found also in the files of the documentation directory :

/u/prog/daisy/doc/english

9.10.2.1 Program SPECPREP

- **Miscellaneous parameters**
  - Title of the data set (TEXT)
  - No. of points used for noise reduction (NS)
  - No. of points in resulting spectrum (NOF)

  Usually, you should set the number of noise reduction points to 0. Nowadays, NMR spectra are good enough for DAISY without further manipulation within SPECPREP.
  The maximum number of points is 16000. Bigger numbers would result in enormous calculation times for the iterators. We recommend that you limit the total number of points as follows :

  - specify a close overall range (see Spectral limits)
  - cut out irrelevant sections (See Cut outs)

- **Baseline correction**
  - Enter baseline correction mode
  - Number of Fourier terms (displayed only if the mode requires Fourier terms)
  - Do you want step control output (Y/N)

  When you save the data set, the editor converts the correction mode into the

parameter IFLAT.
If you select a correction mode, you must set the number of Fourier terms (corresponding to the variable NFC).
If you did not select a correction mode, this parameter input line will not be displayed.
Usually, you will already have corrected the baseline with XWIN-NMR. In that case, set the correction mode to *no correction*.

To get a more detailed protocol output, you can activate the step control (variable IPLOT). After each operation the program prints an X/Y data section allowing you to check the single steps.

- **Spectral limits**
  The frequency limits and the sweep width of the spectrum cannot be changed because they are read from the experimental data. They are only displayed to guide you when you set the next parameters.

  The different ranges are :
  - Frequency range of iteration (FSTRUN and FETRUN)
  - The region for calculating the noise (FSNOIS and FENOIS) (this region must not contain any peak)
  - The range where the baseline should be smoothed (FSFL und FEFL)

  Select the limits (FSTRUN and FETRUN) in such a way that only the relevant part of the spectrum is covered. In this way, you reduce the total number of points used for the iteration and therefore speed up the calculations.

  If the *Number of points used for noise reduction* was set, you must enter the region limits for the calculation of the statistical noise. Otherwise, you can ignore this option.

  If a *Baseline correction mode* was selected, you must enter the region limits for determination of the baseline. Otherwise, you can ignore this option.

- **Cut outs**
  - The number of the cut out segments (ICUT)

  The following line will be repeated ICUT times :

- The limits of the cut out segments (FSCUT(I) and FECUT(I)

Cut outs allow you to reduce the number of points used for the iteration even further. You can cut out up to five ranges without (interesting) signals, because the iterators can handle six different spectral segments.

Important note :
If you omit five spectral ranges, there are six left !

9.10.2.2 Programs DSYMPLOT and DCYMPLOT

The input parameters for DSYMPLOT and DCYMPLOT are very similar. Any differences will be pointed out.

- **Miscellaneous parameters**
  - Title : individual title for this simulation
  - Output experiment name :
  - Output experiment number :
  - Output experiment processing number :

The last three variables are automatically taken from the main menu page. If you modify them, the simulators will store their results in a different data set from the one you are currently editing. This is useful if you want to simulate several spectra which differ only with respect to the chemical shifts, for instance.

- Spectrometer frequency : is read from the XWIN-NMR data set

Usually you will not change this value since it reflects the status of your experiment. You should not change the value unless you want to simulate fictitious data.

- Low frequency limit : right end of the simulated spectrum
- High frequency limit : left end of the simulated spectrum
- Minimum intensity : weaker lines are suppressed
- Halfwidth : of the lines in the simulated spectrum
- Number of Points : in the simulated spectrum
- Number of Nuclei :

The low and high frequency limits are interpreted as ‘Hz’ values. If a value is followed by ‘p’ or ‘P’, it is accepted as a ‘ppm’ value and immediately converted to ‘Hz’. In DSYMPLOT the maximum number of nuclei (with spin 1/2) is 7. In DCYMPLOT this number reflects the number of magnetically independent groups of nuclei. The maximum number of groups is 7. Each group can contain up to 9 magnetically equivalent nuclei. Each nucleus can have spin 1/2 or greater.

- Symmetry group (DSYMPLOT) :
- Number of symmetry groups (DCYMPLOT) :

DSYMPLOT works with only one symmetry group which you enter here. DCYMPLOT can work with more than one symmetry group and you enter the number of symmetry groups here. The ’next page’ button displays the menu where you enter the actual symmetry groups.

Important note:
The symmetry groups must be entered in decreasing order; otherwise the simulators will not work correctly.

- **Output list generation**
  The menu items listed here determine the contents of the output files. To generate a spectrum which can be displayed by XWIN-NMR, *the write spectrum to disk in XWIN-NMR format* button must be set to *include*.
  When you save the data set, the editor converts all *included* options into the variable IPRINT.

- **Permutation operators**
  This menu is displayed only if a symmetry group other than C1 was selected (see *Miscellaneous parameters*).
  The first line displays the E (identity) symmetry operation for the total number of nuclei specified in the *Miscellaneous parameters* menu. One line each is displayed for all other symmetry operations of the character table. In each field of these lines, you specify the logical number of the nucleus into which the nucleus given in the first line is converted by applying the symmetry operation.

- **Resonance frequencies**
  All Iso-values and resonance frequencies are listed in this menu. The input values for resonance frequencies are interpreted in ‘Hz’. If a value is followed by ‘p’ or ‘P’, it is accepted as a ‘ppm’ value and immediately converted to ‘Hz’.
  If you change a resonance frequency of a nucleus (or a group) which is symmetrically equivalent to other nuclei (or groups), all lines with corresponding values (including the Iso-values) are adjusted automatically. Nuclei or groups with the same Iso-value are treated by a second order approximation; nuclei or groups with different Iso-values are treated by first order approximation.

  For DCYMPLOT, also the number of nuclei in each composite particle (CP) group and their single spin is required. To allow easier setting of the spin, you have to enter ’1’ for a 1/2 spin, ’7’ for a 7/2 spin nucleus.

- **Spin/Spin coupling constants**
  All spin/spin coupling constants are displayed in this menu. Chemically equivalent values are updated automatically.

- **Dipol/Dipol coupling constants**
  The default value for all coupling constants in this menu is 0.0. This is the correct setting if you want to simulate a spectrum in isotropical phase. However, if you want to simulate a spectrum in an anisotropical phase, you set the dipol/dipol coupling constants to values not equal to 0.0. The order of magnitude of the dipol/dipol coupling constants follows the conventions used in the literature.

9.10.2.3 Programs DAVSYM1/2 and DAVCYM1/2

In addition to the menu pages for DSYMPLOT and DCYMPLOT, the editor displays the following menu pages to set up all parameters needed for an iteration of a spectrum.

- **Iteration limits**
  This menu page displays a list of all chemically independent chemical shifts, spin/spin and dipol/dipol coupling constants. A parameter will be iterated if the *iterate* flag is set to *yes*. In that case, set the limits between which the parameters should be varied. The default lower and upper limits are calculated from the initial parameter value.

- **Iteration parameters**
  - Correlation matrix factor mantissa :
  - Correlation matrix factor exponent :

  These two values lead to the

  - Correction matrix factor : (A1)

  It defines the initial strength of the smoothing function for the error hyper plane.

  - Multiplier : (A2)
  - Convergence criterium : (SETOUT)
  - Speed factor : (SPEED)
  - Negative gradient seek : (ISD)
  - Maximum number of cycles : (ITMAX)

  During the iterative refinement of the parameters, several cycles are stepped through to find the best match between the simulated spectrum and the experimental data. If the *Convergence criterium* is reached, the *Correction matrix factor* is multiplied with the *Multiplier*. The iteration is now continued with a stronger smoothing function for the generation of the error hyper plane. (This basically means that the valleys in the error hyper plane are now deeper.)

  The *Speed factor* determines the number of points which are summed up at the beginning of the iteration to speed up the calculation. The iteration stops if no better match can be found by the program or if the *Maximum number of cycles* is reached.

  - Halfwidth threshold : (THRLW)
  - Baseline offset threshold : (THRBI)
  - Baseline ascension threshold : (THRBT)

  If the *Convergence matrix factor* becomes equal to one of the above values, iteration is started on the corresponding parameter.

  Important note:
  If your simulated and experimental spectra match well, start with a relatively

high *Correction matrix factor,* e.g. mantissa 10, exponent -2.
If they do not match well, start with a smaller exponent, e.g. -5.

- **Region parameters**
  This page is displayed only for DAVSYM2 and DAVCYM2 where the SPECPREP spectrum may contain more than one region of spectral information. The number of lines displayed on this page is given by the number of regions.
  If you want to exclude a region, set the *using range* to *no*.
  If you want to include the region, set the *using range* to *yes*.
  In that case, you can also specify a statistical weight (or importance factor) with which this region is treated during the iteration. Regions with rather little information about couplings, for instance, but with a relatively high overall intensity can be reduced in their importance by setting the statistical weight to a value less than 1.0.

9.10.3 Requirements for the installation of the DAISY package

To run the DAISY program the following files must exist.

directory/program name	**data type**	**contents/function**
/u/prog/daisy/mkux	executable	converts data formats
specprep	executable	SPECPREP
dsym	executable	DSYMPLOT
dcym	executable	DCYMPLOT
ds1	executable	DAVSYM1
ds2	executable	DAVSYM2
dc1	executable	DAVCYM1
dc2	executable	DAVCYM2
/usr/bin/specux	shellscript	starts SPECPREP
dsymux	shellscript	starts DSYMUX
dcymux	shellscript	starts DCYMUX
ds1ux	shellscript	starts DS1
ds2ux	shellscript	starts DS2
dc1ux	shellscript	starts DC1
dc2ux	shellscript	starts DC2
/u/prog/daisy/doc/english/*	text	help files
/u/exp/stan/nmr/lists/sym/*	text	character tables
/u/exp/stan/nmr/lists/symnew/*	text	character tables

It is possible to store the executables anywhere on the disk if the corresponding shellscripts are modified accordingly. The character tables are part of the UXNMR program and are automatically stored in the correct directories.

9.10.3.1 References

For more information, please check the manual pages in the directory

/u/prog/daisy/doc/english

or refer to the literature published by the authors of the DAISY program package. See, for instance:
G. Haegele, M. Engelhardt and W. Boenigk *Simulation und automatisierte Analyse von Kernresonanzspektren,* Weinheim : VCH, 1987, ISBN 3-527-26550-3

9.11 MAXENT/MEMSYS5

The purpose of this program is to derive spectral features such as line positions and peak widths from NMR data by using probability theory, in this particular case the maximum entropy method.
Maximum entropy is a procedure for inferring positive distributions from limited data. Not only 1D spectra, but also multidimensional data and images can be processed.
The method enables the ”best“ result, meaning the most probable spectrum or image to be accompanied by realistic error bars, so that the reliability of any result can be assessed. MEMSYS5 provides quantification of error limits and gives the scientist increased confidence in the interpretation of the results.

For 1D NMR spectra, for example, the result is a deconvolved spectrum which represents the best fit of the experimental data.

Parameter set up and execution takes place from within XWIN-NMR, and the generated data are immediately accessible from there.

MEMSYS5 is a product of Maximum Entropy Solutions Limited (Cambridge, UK). It is *not* a standard component of the XWIN-NMR package and must be purchased as a separate option.

The following XWIN-NMR commands provide a user interface to the MEMSYS5

maximum entropy package.

9.11.1 Generate a Point Spread Function [psf]

The command psf generates a point spread function for a MEMSYS5 run with the command maxent. A point spread function is basically the lineshape of a single line or peak of a region in an NMR spectrum. A PSF can also be the sum of a number of lineshapes defining a more complex peak or peak group. Such a PSF must be generated with the command wpsf.

The point spread function is used for the deconvolution of an NMR spectrum. Only 1D-MEMSYS5 will work with a user-defined PSF. For 2D and 3D the parameters defining the PSF are set in the editor maxed.

When running psf the user is asked for some parameters which describe the output PSF (or lineshape). Lineshapes can either be symmetrical or asymmetrical. Each half of the lineshape (with respect to the highest point) can be designed individually so that Lorentz/Gauss/Winged contribution to the lineshape and half width at half height can differ.

You are asked for the following parameters.

1. **Number of points defining the lineshape (PSF)**
   This number should be less than or equal to the number of points in the region of the NMR spectrum you want to deconvolve. If the number of points in the PSF is greater than the number of points in the region of the NMR spectrum the number of PSF points is reduced for the deconvolution. The selected points are chosen with respect to the center of the PSF.

2. **Symmetrical or asymmetrical lineshape**
   There are two possible settings :

   symmetrical :
   - a symmetrical lineshape will be generated.
   - the program will ask for the following two values once

   asymmetrical :
   - an asymmetrical lineshape will be generated.
   - the program will ask for the following two values twice (one value for each half of the PSF).

3. **% Lorentz/Gauss/Winged**
   You have to enter the Gauss/Lorentz/Winged contribution to the lineshape.

With the following numbers these lineshapes are generated :

0 : 100% Gaussian
1 : 100% Lorentzian
2 : 100% Winged
0.6 : 60% Lorentz, 40% Gauss
1.4 : 60% Lorentz, 40% Winged

4. Half width at half height

Enter the half width at half-height of the lineshape here, because the lineshape is split into two halves !
When designing a PSF for a certain region of the NMR spectrum you should measure the half width of a peak in the spectrum in Hz using the mouse cursor. The selected peak should be a single peak not overlapping with other peaks and the peak should be representative for the region you want to deconvolve.

5. Write PSF or *1r* or both

You can store the lineshape in three different ways.

p : will write the lineshape into the file *mem.psf* for a MEMSYS5 run.
r : will write the lineshape to the file *1r* as if it would be a spectrum.
b : will write the lineshape into the file *mem.psf* and *1r*.

To check the generated PSF it is sometimes useful to compare it with the original NMR spectrum. If you want to do this, proceed as follows.

1. Switch to a new processed data set number.

2. Process the spectrum again, e.g. by typing fp.

3. Run psf and select the output option write 1r or both.
The PSF will now overwrite the processed data set. The size of this new spectrum is adjusted according to the number of points used for the PSF. SW and Hz/Pt will also be adjusted. The PSF will always be centered at 0 ppm.

4. Type 2s offset

5. Enter a value which corresponds to the region of interest of your experimental spectrum.

6. Switch back to the original data set via the *TASKS* menu.

7. Specify the PSF processed data set as the second data set using edc2.
8. Compare the experimental NMR spectrum and the PSF with the dual display function.

Important note :
Do not use the PSF processed data set number as second or third data set in edc2 for a maxed or maxent run. The sizes, SW and Hz/Pt in the status processing parameters no longer match with the original NMR spectrum !

9.11.2 Write *1r* file into point spread function [wpsf]

wpsf allows you to generate any lineshape form to be used with MEMSYS5. This could be a single line extracted from a 1D data set or the result of adding several PSF functions which were created using the psf command. The lineshape to be written as a PSF must exist as a 1D data set (an *1r* file). The *1r* file is copied into a PSF file when running wpsf.

9.11.3 Edit MEMSYS5 parameters [maxed]

maxed sets all relevant parameters for a maxent run.

Before setting up the parameters with maxed you need to define :

- a point spread function (PSF)
- the region of interest in the NMR spectrum (*Output->Define plot region* menu)
- a second and third data set for the output files (command edc2)

The two data sets will contain the following results :

- second data set : deconvolution result
- third data set : mock data

The *mock* data set is the multiplication of the deconvolution result with the PSF. The mock data are most useful for controlling the quality of the deconvolution (i.e. how similar are deconvolved and original data). The deconvolution result contains all single peaks after the deconvolution maxent run.

Check the settings of the following parameters.

- **Number of iteration**s
  Can be set to any number. But the default value 99 is usually sufficient to lead

to convergence (given that the PSF is realistic). If the number of iterations does not lead to convergence, the PSF is probably unrealistic.

- **Step results**
  If STEPRES is set to ‘yes’, output information is written to disk after each iteration.
- **Mock data**
  Set the flag to ‘yes’ if you want to write the mock data to the third data set.
- **The positive/negative toggle**
  There are two possible settings :

  - positive : only peaks with a positive amplitude will be deconvolved.
  - pos/neg : peaks with a positive and negative amplitude will be deconvolved.
- **SIGMA**
  SIGMA has a default value of 0. In this case, MEMSYS5 determines the statistical noise. To compare two PSF functions, SIGMA must be set to an estimated noise value. There are three possibilities to determine the noise value.

  - You can use the CSIGMA value calculated by MEMSYS5 at the end of a run (see maxres)

  - You can use the command noie to determine the noise value. Enter 1 as the order of difference.

  - You can use the sino command to determine the noise value. Select NOISF1, NOISF2, SIGF1 and SIGF2 in such a way that the values roughly correspond to the region you want to deconvolve with MEMSYS5.
- **F1P, F1, F2P, F2**
  The frequency limits reflect the area currently defined as the plot region.These values are only displayed for 1D data sets. On nD data sets the whole spectrum as you see it on the screen is used for the deconvolution. If you want to deconvolve only a part of the nD matrix, use the strip transform feature to get the region of interest.
- **PSFDEF**
  You can run 1D MEMSYS5 either with a user-defined PSF or with a parametric PSF. Use the command psf or wpsf to generate a user-defined PSF.
  A parametric PSF is defined by the parameters ASYM, PSFWI and PSFSH.

These parameters correspond to those the user is interactively asked for when running the psf command. See the chapter on psf for more details.

All other parameters are set to default values which have been found to lead to a reliable deconvolution. These parameters are used internally by the MEMSYS5 program and should be modified only by a user with detailed knowledge of the MEMSYS5 program. Please see the "User's Manual for Running 1D MaxEnt" for more details.

9.11.4 Start MEMSYS5 deconvolution [maxent]

There are two ways of starting a MEMSYS5 deconvolution.

- By typing maxent. In this case, the second and third data set, the plot region and the MEMSYS5 parameters must have been defined already (see above).
- By answering "y" to the question after you have saved the MEMSYS5 parameters in the editor program maxed.

At any time during a maxent run you can make use of the dual display function to inspect the progress of the iteration.

9.11.5 Show MEMSYS5 results [maxres]

MEMSYS5 writes a report for all iteration steps into the *mem.log* file which can be inspected at any time during or after a maxent run. This file contains useful information about the progress of the deconvolution. After MEMSYS5 has reached convergence, important values like *good* and *evidence* are stored in the *mem.log* file.

The following parameters are especially important.

- **Omega**
  Omega is the convergence criterion. Convergence is reached when Omega is 1. Checking Omega during a maxent run gives an indication of the progress of the iteration.
- **Evidence**
  Evidence tells you how reliable the result of the deconvolution is compared to an earlier run with different parameters or a different point spread function (PSF). The smallest value of Evidence indicates the most probable and most reliable (and therefore hopefully best) result.
- **CSIGMA**
  CSIGMA is the statistical noise value determined by MEMSYS5. It must be set

in maxed if you want the exact comparison of two runs using two different PSFs.

A more detailed evaluation of the MEMSYS5 result can be obtained by running xline.

9.12 Bayesian Analysis

9.12.1 Introduction

In the past years many alternative methods of data analysis have received increased attention in the field of Nuclear Magnetic Resonance (NMR) spectroscopy. Some of these methods try to avoid the disadvantages of the Fourier Transform (truncation artefacts, implicit periodicity,~...) and of the accompanying routine processing (apodization, filters, ...) by extracting all the information of interest directly from the acquired time domain data, without tampering with the data first. This is a very important principal difference because most NMR spectra do not reflect as accurately as possible what has actually been measured because of all the data manipulations that have taken place in order to arrive at the spectrum.

The Bayesian analysis module in XWIN-NMR is based on Bayes' Theorem from probability theory and not only allows to fit different time-domain models to the raw acquisition data, but also to reasonably answer such questions as: *Can a signal be detected in the data (and how well)?;* and *How many decaying sinusoids are there in the data?*, and in fact provides a *quantitative* statement of the intuitive rule, that of several models which fit the data equally well, the simplest model should be used.

The program is therefore capable of performing a fully automatic analysis of the data, including the determination of the most probable number of spectral components in the data along with all the corresponding parameters (amplitude/integral, frequency and decay rate/line width as well as phase) in a *black-box* fashion without the need of setting any parameters. The method proves to be very powerful and reliable, and data with very broad, overlapping signals together with sharp, closely spaced lines and shoulders that almost disappear in the noise, are easily handled.

The Bayesian analysis module in XWIN-NMR can perform all of the following tasks: parameter estimation, signal detection, and model selection . The user can control every detail of the analysis, or - more importantly - use the program in a black-box fashion, without having to specify any parameters.

9.12.2 How the Program Works

In parameter estimation the number of exponentially decaying sinusoids to be fit to the data is specified by the user, and the optimal parameters are then determined by finding the maximum of the probability surface. In signal detection the free induction decay is searched for positive evidence of damped sinusoids as a function of frequency and decay. These findings are then automatically passed to the parameter estimation routine for optimizing the parameters.

Initially, the original FID data are searched for positive evidence of frequencies. All maxima that are found with values greater than a certain threshold are immediately used for setting up a model of this many decaying sinusoids. If there are only positive maxima below the threshold, then only the highest BAYNADD maxima will be used to set up the initial model. This model is then optimized with respect to its parameters, and the model's probability is calculated. This probability is compared to the probability of the previous model (which initially is the "constant model"), and if its value is higher then the new model is accepted. The model FID is calculated and subtracted from the original data. This residual is then used as input to the next signal detection pass. If, on the other hand, the new model's probability is not greater than that of the previous model, then this model is rejected. If there were more positive evidence maxima in the previous signal detection pass, the next highest BAYNADD frequencies are tried in a new model. This cycle continues until either no more positive evidence maxima are found in a signal detection pass, or until none of the found evidence maxima would lead to a model with a higher probability than that of the previous model.

9.12.3 Files

The parameters for the Bayesian analysis are stored in the file *baypar* in the directory of the current data set. During the calculation the program will write a protocol file *bay__prot* (unless the parameter VERBOSE is set to *no*) and, after each iteration, a file *bayes.mod* which contains the model parameters at that point of the analysis. The file *bay__prot* can be viewed or printed with the commands vibay or libay , and the file *bayes.mod* can be used to continue a previous calculation with the last model parameters.

9.12.4 NMR Data Type

The Bayesian analysis package was originally developed to handle fids acquired with analog filters in *sequential* acquisition mode. Data of this type are usually generated by Bruker AM/AC and AMX/ARX type spectrometers (depending on

the selected acquisition parameters). Data aquired using digitals filters on Avance type instruments must be back-converted to analog type data with the convdta command before they can be treated with the Bayes algorithm. Data acquired in simultaneous acquisition mode (which is always true for Avance spectrometers if digital filters were used) must be converted to *sequential* mode data using the following processing steps:

1. Set the desired transform size SI.
2. Transform the data using the command ft.
3. Set the status parameter FT_mod to sequential mode *fqr* using the command *2s FT_mod*.
4. Execute an inverse transform using the command ift.
5. From these data, generate a new fid using the command genfid. This fid can be treated with the Bayesian analysis.

9.12.5 Commands

Activation of the menu field *Simulation->Bayesian Analysis* allows the choice between five commands, which can also be typed in:

bayed
The parameter editor is called to set or edit the parameters for a subsequent Bayesian analysis.

bayx
Start of the Bayesian analysis. The parameters previously defined with the commands bayed or bayedx will be used. If no Bayesian analysis parameters have yet been defined the program will assume default values for the parameters and try to analyze the data with repeated signal detection passes in a black-box fashion. An optional argument of *y* or *yes* can be specified with the commands bayx and bayedx which has the following effect: if the parameters BAYSTO2 or BAYSTO3 are set to *yes* , then results are stored in data set 2 or 3 (to be setup via the command edc2). However, the program will first check if the data set already exists and if so ask for permission to overwrite it. The argument *y* or *yes* explicitly gives permission to overwrite data set 2 or 3 and thus prevent the popup message from appearing.

bayedx
This is a combination of the commands bayed and bayedx . First the parameter editor is called to set or edit the Bayesian analysis parameters. Leaving the editor with

SAVE will cause the actual analysis to be started, using the parameters just set. Exiting the editor with ABORT will terminate immediately (without calculation), leaving the parameters unchanged. An optional argument of *y* or *yes* can be specified with this command. See command bayx for an description of its effect.

vibay
The protocol file created during the last calculation is output to the screen. If no protocol file exists, an error message will come up.

libay
This is the same as vibay, except output is sent to the device (or file) defined in the device parameter CURPRIN (to be set up vwith the command edo).

9.12.6 Parameters

While there are quite a few parameters with which the Bayesian analysis can be controlled, the module is set up such that it can operate in a black-box fashion without the user having to set any parameters at all. Reasonable default values have been assigned to all parameters.

When the parameter editor for the Bayesian analysis is started via the commands bayed or bayedx for the very first time for the current data set (i.e. Bayesian parameters do not yet exist for the current data set), then only a few parameters are brought into the editor. The top parameter, BAY_ANA, controls the extent of the Bayesian analysis to be performed. It is initially set to automatic, which means that the program will perform repeated Signal Detection passes, adding all signals found to the model and optimizing the corresponding parameters, until either the model's probability no longer increases or there is no more evidence for a signal. The setting of the parameter BAY_ANA determines which parameters are available for editing in order to control the analysis.

The individual parameters are:

BAY_ANA
BAY_ANA can be set to the values *automatic*, *sig_detection* , *mod_selection* and *par_estimation* . The first, *automatic*, is the default setting when the parameters are first set up and is identical to *mod_selection* except that default values are assumed for all parameters. If the user sets BAY_ANA to *mod_selection* all relevant parameters will be displayed in the parameter editor. With *sig_detection* the program will only perform the specified number of signal detection passes (BAYNPAS), optimizing the parameters for all added lines. *par_estimation* causes the program to simply optimize the parameters for the specified number of lines (BAYNLIN),

which must be listed in the file specified by the parameter PEAKLST .

BAYPHAS

This parameter defines the phase model to be used in the analysis: *no_phase* , *coherent* or *individual*.

BAYREST

This parameter specifies whether the calculation should start off using the results from a previous analysis (BAYREST=yes) or not. The results from a previous analysis are taken from the file *bayes.mod* in the current data set directory. During the calculation this file is updated after each completed signal detection pass.

TDeff

TDeff gets the number of points of the FID to be used in the analysis. The default (achieved by setting TDeff=0) is to use all data.

TDoff

The first TDoff points of the FID can be ignored in the Bayesian analysis. This is particularly useful for data sets with distorted baselines.

SI

The size of the resulting model FID (or other output, see BAYOUT) is set with this parameter.

BAYNSIG

It is often very advantageous to provide a noise sample for the Bayesian analysis, as this allows a more accurate estimation of the magnitude of the noise and thus helps greatly with deciding on the presence of more signals in signal detection. If BAYNSIG is set to a value greater than zero, the program will first look for a file *noise* in the same directory as the current data set FID (file *fid*). If there is no file noise , then the program will take the last BAYNSIG points of the FID and use them as a noise sample.

BAYOUT

This parameter defines the way in which the result of the analysis is output: as a model FID (BAYOUT=*FID*), as a spectrum obtained from the model FID via Fourier transform (BAYOUT=*spectrum*), as zero-width lines at the frequency positions and with heights approximately equal to the integral (BAYOUT=*lines*) or no data output at all (BAYOUT=*none*).

BAYAOPT

Signal detection is performed at different decay rates, starting with the values set for BAYAOPT, which should be set to the width of one of the narrower lines in the data. A default value is automatically calculated from the available resolution of

the data, but it tends to lead to too many noise peaks which need to be tested in the model selection process, thus making the calculations much slower.

VERBOSE

The parameter VERBOSE determines the amount of information written to the protocol file *bay_prot* during the analysis.

BAYSTO2

With this parameter the resulting model can be stored as the data set defined as data set 2 (set up via edc2), either after each new model has been optimized (BAYSTO2=*intermediate*) or only after the whole Bayesian analysis has been completed (BAYSTO2=*final*).

BAYSTO3

This parameter works exactly like the parameter BAYSTO2 , except that the residual FID , i.e. the difference between original FID and model FID , is stored in the data set defined as data set 3 (set up via edc2).

BAYCMD

If an intermediate model or residual is to be stored in data set 2 or 3, then a command can also be executed on this data set, e.g. ft or efp. Any command that can be typed in from the keyboard can be used here.

BAYVARP

This parameter only takes effect for the coherent-phase model and determines whether the phase parameters (defined by the processing parameters PHC0 and PHC1 in XWIN-NMR) should be held fixed at their specified values (BAYVARP=*no*), or whether they should be optimized together with the frequencies and decay rates.

BAYVARA

In the parameter optimization routine, the decay rates can be treated in three different ways: they can all be held fixed at the same value taken from the parameter BAYAOPT (BAYVARA=*no*), they can be forced to all have the same value which is to be optimized (BAYVARA=*unique*), or they can be allowed to be optimized independently (BAYVARA=*individual*). Note that the decay rates in the time domain are equivalent to the line widths in the frequency domain.

BAYAMAX, BAYAMIN

These two parameters allow for a range to be specified within which the decay rates should lie. Components for which the decay rates move outside this range are removed from the model during the analysis. Note, however, that due to the fact that the analysis is performed in the time domain, it is possible that certain compo-

nents are so strong that they keep appearing in the signal detection or are repeatedly encountered in the maximum search, so that an endless loop could occur if the range is set too narrow. The main use for these parameters is to allow for a faster (than through the model probability test) elimination of components that correspond to noise (very small decay rate) or to very broad baseline artefacts.

PEAKLST
This parameter specifies a peak list file to be used in parameter estimation. This file must have the format of the peak list file described in the chapter on 1D processing and can, for example, be set up with the peak picking commands.

BAYNADD, BAYELEV
After a signal detection pass the highest BAYNADD frequencies of all frequencies above the evidence threshold, BAYELEV, are added to a new model at once. If fewer than BAYNADD frequencies are found above the evidence threshold, then all those are added at once. Frequencies below the threshold BAYELEV are always added one at a time to check the new model's probability. Even if the frequencies are added one at a time, only BAYNADD frequencies may be added to a model (and accepted) in one signal detection pass. Once BAYNADD frequencies have been added to the model successfully, a new signal detection pass will be performed.

BAYNPAS
If BAY_ANA is set to *sig_detection* the maximum number of consecutive signal detection passes (with intermediate model optimization) is set by the parameter BAYNPAS .

9.12.7 Bayesian Analysis in AU Programs

The module for Bayesian analysis can also be called from within automation (AU) programs and will work exactly as described above. However, the Bayesian parameters in the file *baypar* are handled separately from the other XWIN-NMR parameters and can thus not be accessed via the AU commands fetchpar and storepar . The commands fetchbaypar and storebaypar must be used for the Bayesian parameters instead. The syntax is exactly the same as for the corresponding commands fetchpar and storepar (see the chapter on AU programs).

Example: storebaypar("BAYAOPT", 5.0)

The commands uselastpars and usecurpars in AU programs also work with the Bayesian analysis parameters, i.e. the command uselastpars causes the Bayesian parameters from the last data set to become active.

9.12.8 Bibliography

[1] Bayes, Rev. T., *Philos. Trans. R. Soc. London* **53**, 370 (1763); reprinted in *Biometrika* **45**, 293 (1958).

[2] Bretthorst, G. Larry, "Bayesian Analysis. I. Parameter Estimation Using Quadrature NMR Models", *J. Magn. Reson*. **88**, pp. 533-551 (1990).

[3] Bretthorst, G. Larry, "Bayesian Analysis. II. Model Selection", *J. Magn. Reson.* **88**, pp. 552-570 (1990).

[4] Bretthorst, G. Larry, "Bayesian Analysis. III. Applications to NMR Signal Detection, Model Selection and Parameter Estimation", *J. Magn. Reson.* **88**, pp. 571-595 (1990).

[5] Jaynes, E.T., in "Papers on Probability, Statistics and Statistical Physics" (R.D. Rosenkrantz *Ed*.), Reidel, Dordrecht, The Netherlands (1987).

[6] Jaynes, E.T., *Bayesian Spectrum and Chirp Analysis in* "Maximum- Entropy and Bayesian Spectral Analysis and Estimation Problems" (C.R. Smith and G.J. Erickson Eds.), p. 1, Reidel, Dordrecht, The Netherlands (1989).

[7] Jaynes, E.T., *How Does the Brain do Plausible Reasoning?* in "Maximum-Entropy and Bayesian Methods in Science and Engineering" (G.J. Erickson and C.R. Smith Eds.), Vol. **1**, pp. 1-24, Kluwer Academic, Dordrecht, The Netherlands (1988).

[8] Jeffreys, H., "Theory of Probability", Oxford Univ. Press, London (1939); Later editions 1948, 1961, reprinted in 1985.

[9] See: Garret, A.J.M., *Ockham's Razor* in "Maximum-Entropy and Bayesian Methods, Laramie, Wyoming, 1990" (W.T. Grandy, Jr. and L.H. Schick *Eds*., pp. 357-364, Kluwer Academic, Dordrecht, The Netherlands (1991). And: Thorburn, W.M., "The Myth of Occam's Razor", Mind 27, 345 (1918).

Chapter 10

The *1D Output* Menu

The *Output* menu (Figure 10.1) contains the plot commands and utilities to print or

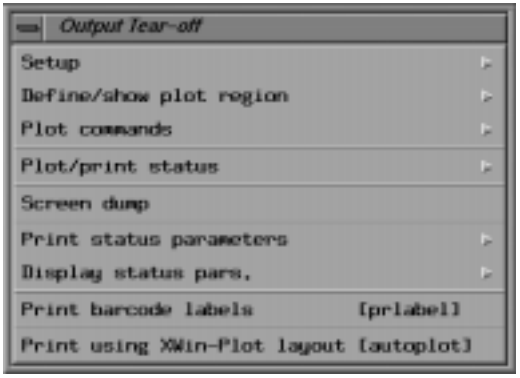


Figure 10.1 The *1D Output* menu

display parameters. Please note that XWIN-NMR provides two different plot systems: The parameter driven plot commands described in this chapter, and the interactive plot editor XWIN-PLOT described in its own manual. The command autoplot, displayed in Figure 10.1, plots the current data set based on a layout generated

with XWIN-PLOT. The layout can be selected using the LAYOUT parameter to be set with the edo command.

To use the parameter driven plot system, please proceed as follows:

1) Make all connected printers or plotters known to XWIN-NMR with cfpp
2) For pen plotters, specify the pen colors with edpp
3) Define a page layout with edg (normal spectrum plot), edgx (auto-expansions) or rpar (predefined layout), and a plot title with setti
4) Define with edo to which of the connected plotters the output should be sent (see also the setres command)
5) Preview the plot on screen with view (normal spectrum plot) or viewx (auto-expansions). Adjust the layout if required.
6) Output the plot with plot, plots, plotx, flplot.

The following sections will describe these steps in detail.

The setup commands are part of the *Output->Setup* submenu (Figure 10.2).

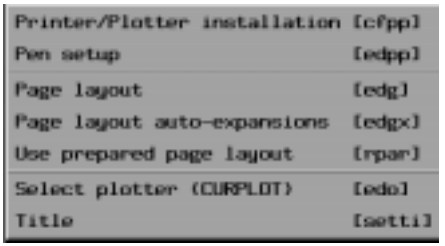


Figure 10.2 The *Output->Setup* submenu

Modifications of the above UNIX procedure in the Windows NT Version

1. cfpp is not supported; install a printer or plotter with the general Windows NT printer installation tool.

If you specify "FILE:" as port, a plot code file generator is installed, instead of a real printer.

2. edpp is not supported; use the Windows NT printer installation tool.

Linewidths may be specified within edg, edgx or edgw.

3. Definition of page layout and plot title as in UNIX.
4. Within edo all printers, installed with the Windows NT printer installation tool, may be selected. If you select a plot code file generator, a plot command results in a plot code file, not in printed paper.

   You may also select "Enhanced Metafile". In this case a plot command results in a Windows NT Enhanced Metafile (.emf), not in printed paper.

   You may also select "Clipboard". In this case a plot command results in an image in the Windows NT Clipboard.

   "Enhanced Metafile" and "Clipboard" are not allowed when using the plotx command.

5. The previewer for regular printers and plotters, including those ones with a "FILE:" port, is the same as in UNIX.

   If "Enhanced Metafile" or "Clipboard" is selected for plot output, the graphs are written into the Windows NT Clipboard and the corresponding viewer is called. In this case viewx is not permitted.

6. The plot commands are the same as in UNIX, but files, generated by a plot command are located in the directory, specified with the environment variable XWINNMRPLOTFILES. The default is $XWINNMRHOME/plotfiles.

   The filename starts with a number, representing year, month, day, hour, minute and second of the generation. The process ID of the plot interpreter follows after a ’-’ sign. For plot code files the printer name follows after a ’-’ sign; Enhanced Metafiles have a ".emf" extension instead. If a plot request generates more then one Enhanced Metafile, a ’-’ and a number are inserted between process ID and the extension.

10.1 Printer/plotter installation [cfpp]

XWIN-NMR can plot spectra on plotters with colored pens (pen plotters), laser printers, and ink jet printers. The latter two belong to the class of printer-plotters, i.e. they are used to print text as well as graphics. The devices are connected to a parallel (Centronics type) or to a RS232 interface of the computer. XWIN-NMR can

operate several devices on different interfaces simultaneously. The purpose of the command cfpp is to tell the program to which interface channel a printer or plotter is connected. cfpp must be repeated after installation of a new XWIN-NMR version.

cfpp (*configuration of plotters and printers*) opens a dialog window according to Figure 10.3. Select *Plotter installation* for pen plotter, *Printer-plotter installation*

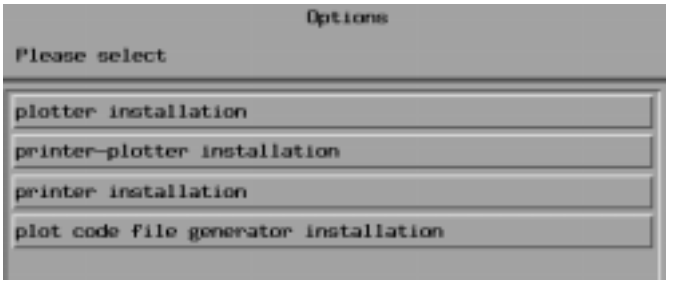


Figure 10.3 Initial cfpp dialog window

for laser and ink jet printers, and *Printer installation* for printers which are only able to output text, but no graphics. *Select Plot code file generator installation* if you want to define a special plotter type which outputs to a file rather than to a printing device. HPGL and PostScript file generators are available.

Plotter installation, Printer-plotter installation, Printer installation
A dialog window such as the one shown in Figure 10.4 appears. Except for listing all existing plotters, the superuser password is requested when you click on one of the items.

In XWIN-NMR, a plotter is uniquely defined by the following four specifications:

- The plotter type, e.g. *hpdj550c* (all supported plotters are listed at the end of this chapter).
- The individual name of the plotter assigned to it by the operator, distinguishing plotters of the same type from each other.
- The plotter class. This is only of interest if there are several plotters connected which the user may want to divide into classes. If a plot command is issued, a class instead of an individual name can be specified. The spooler of the operating system will then send the plot to the first free device of that class.

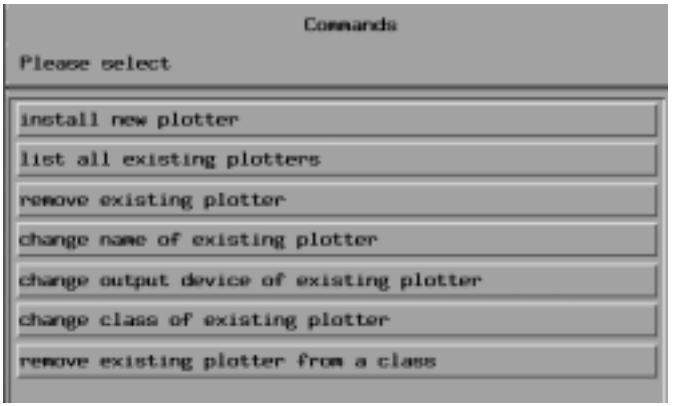


Figure 10.4 *Plotter installation* dialog window

- The parallel or serial interface to which the plotter is connected.

install new plotter
Click on *install new plotter*. XWIN-NMR will first list all supported plotter types in a dialog window. This information is stored in the directory

/u/conf /plotter/init/
/u/conf /printer/init/

depending on the device type. Choose the type matching the connected plotter. The program will then ask if it should be given an individual name by printing

Enter identification name (RET for no name), q=quit:

This is only necessary if several plotters of the same type are connected. Otherwise this question can be answered with Return so that the individual name of the plotter will be set equal to its type specifier, e.g. *hpdj550c*. The plotter will then always be referred to by that name. In the case of several plotters, the chosen name (max. 6 characters) will be appended to the plotter type by the program. If, for example, two plotters of type *hpdj550c* have been assigned the names 1 and 2, they will be referred to as *hpdj550c_1* and *hpdj550c_2*.

The next question

Do you want to specify a class (y/n):

allows you to specify the plotter class, which is most often answered with *n*.

Next, the interface to which the plotter is connected must be entered:

Please enter output device:

On AspectStations, enter *lp* for the parallel channel, or *tty1, tty2, tty3, tty4, tty17, tty18* to select the appropriate serial channel. On Silicon Graphics systems, enter *plp* for the parallel channel, or *ttyf1, ttyf2* to select the appropriate serial channel. The program will then make the necessary calls to the line printer scheduler of the operating system, which will make the necessary changes to the file */etc/inittab* and print a few messages on the monitor.

Next, *cfpp* will print the question

Configure like existing plotter? (y,n):

Type *y* if the new plotter is to be configured like a previously installed plotter of the same type, e.g. with the same paper feed options and pen positions. Otherwise, a default configuration is taken from the file

/u/conf/plotter/init/ < chosen plotter type >

delivered with the release software.

list all existing plotters

This is the only option of the table that does not require a superuser password. A list of all configured plotters and the channels they are connected to is given.

remove an existing plotter

After execution of this option, the plotter will no longer be accessed from XWIN-NMR.

change name of an existing plotter

The name of a plotter can be changed if required for some reason.

Change output device of existing plotter

Select this option after connecting the plotter to an output channel different from the channel specified during the first installation.

change class of existing plotter

Change the class assignment of a plotter

remove existing plotter from a class

Remove it.

After the installation of a new operating system version from a boot tape, the plotter installation must be repeated.

Creation of Plot Files

The plot commands eventually send their output directly to the plotter or printer. XWIN-NMR offers the possibility of storing the plot data in a file. At a later time this file can then be sent to a suitable plotter or printer, which could even be connected to a different computer. Select the type of output file from the table. Specify the paper correctly since the plot is continued on additional sheets of paper if it was chosen larger than one sheet (if the plot parameter CLIP is set to *no*). The generated plot files are stored in the user's home directory. XWIN-NMR creates a subdirectory *plot.code* there. The HPGL or PostScript files are stored in that directory and carry the name of the corresponding plotter type followed by a sequence number. For example, say, you want to write your plot into a file which is suitable for output on a HP7475A A3 plotter. Simply set the device parameter CURPLOT=hp_A3 (using the command edo) and execute the plot command. The generated file is stored in the subdirectory *plot.code* of your home directory with the file name hp_A3-1. The next plot is accordingly stored in the file hp_A3-2, and so on.

Supported Plotters and their Switch Settings
A plotter must be connected to the X32 using the same type of cable that is used to connect a terminal to a RS232 port, if not specified differently in the following tables for a particular plotter.

10.2 Pen setup [edpp]

edpp allows these parameters to be set for the current plotter, i.e. the plotter specified in the device parameter CURPLOT (see edo). If no valid current plotter is defined, or the parameters for a plotter other than the current one are to be edited, edpp will list the plotters configured with cfpp, from which one may be selected.

10.2.1 Paper feed options: Parameters PAPERFD, FAMOUNT

1. PAPERFD=*default*
Plotters that can change sheets between plots will insert a new sheet after completion of the plot. For plotters with endless paper the paper will be advanced by an amount derived from the length of the plot. The parameter FAMOUNT has no meaning in these cases.

2. PAPERFD=*no*
   After completion of the plot the paper will not be advanced or changed.
3. PAPERFD=*single*
   This option only works for plotters with endless paper. It ensures that the plotter will behave exactly like a single sheet plotter, with the length of the sheet being determined by the parameter FAMOUNT (in cm). This has the following effects: If a plot needs more room than given by FAMOUNT, the paper will be advanced by FAMOUNT. The plot will be finished on the next sheet if the plot parameter CLIP (see below) is set to *no* or clipped if CLIP=*yes*.
4. PAPERFD=*multiple*
   This option also is only effective for plotters with endless paper. The plot will be drawn correctly, even if it is longer than FAMOUNT. After completion, that paper will be advanced to the next multiple of FAMOUNT.

10.2.2 Pen positions

The various objects of a plot, i.e. spectrum, axes, integral, parameters and title, can be plotted in different colors. The corresponding colors are specified by special plot parameters (using the command edg). The program needs to know which color is to be found in which pen position of the plotter. This information is entered by the operator via the parameters BLA_POS, BLU__POS, etc. (=position of the black, blue, etc. pen).

Position 0 means that a pen of this color is not available. If the operator has accidentally entered a non-existent color for an object to be plotted (specified position 0 for this color), the object will still be plotted in a different color, as long as there is an available pen.

10.2.3 Other pen properties

Some plotters allow for different pens to be used at the same time, but the types have to be used at different speeds to obtain the best possible plot quality. For this reason, XWIN-NMR offers the possibility of specifying additional pen properties for different pen types (and colors) by use of the parameters BLA_PAR, BLU_PAR, ... If these parameters do not contain an entry, the default parameters of the respective plotter (see manufacturer's manual) are used.

The pen parameters are entered in the form 60s 4a 7f 1.2t (s=speed, a=acceleration, f=force, t=thickness). The numbers in front of the letters are set by the user as

desired. The units of the parameter t are mm. t influences the resolution of the spectra on the paper. For laser printers it determines the line thickness. The units of the other parameters vary between plotters and have to be taken from the manufacturer's manual. It is not necessary to specify all four parameters, but if more than one are specified they must be separated by a blank.

10.3 Page layout [edg]

The command edg displays a dialog window with a number of sub-pages, containing parameters to be set up by the user. These parameters determine the position, color, scaling, etc. of the objects to be plotted. XWIN-NMR recognizes the following 1D plot objects: *Spectrum, Integral, Numeric Integral Values, X-axis, Y-axis, Peak List, Plot Title* and *Parameter List*. Figure 10.5 shows a typical basic page layout. It consists of the following 3 windows, which may overlap if desired:

1. Window 1, the *Spectral Window*, contains *Spectrum, Integral, Numeric Integral Values, X-axis, Y-axis* and *Peak List.* The x and y coordinates of its *lower left corner* are given by the parameters SXLLEFT, SYLLEFT (cm). The window width and height are given by the parameters CX and SHEI (cm).

2. Window 2, the *Title Window*, contains the *Plot Title.* The x and y coordinates of its *lower left corner* are given by the parameters TXLLEFT, TYLLEFT (cm). The window width and height are given by the parameters TWIDTH and THEI (cm).

3. Window 3, the *Parameter Window*, contains the *Parameter List.* The x and y coordinates of its *upper left corner* are given by the parameters PXULEFT, PYULEFT (cm). The width is given by PWIDTH (cm). The height of the parameter window is determined by the number of spectral parameters to be plotted, so that, only the width need be specified. For the same reason, its position is given by the upper left instead of the lower left corner.

In order to plot a certain plot object, use edg to set the corresponding object parameters to *yes*: SPECT (spectrum), TITLE (title), XAXIS (x axis), YAXIS (y axis), INTEG (integral), PARAM (parameter list), ILABELS (numeric integral values), and PLABELS (peak list). If one of these parameters is set to *no*, the correspoding object is omitted from the plot. In order to set up the layout parameters for a plot object, click on the corresponding edit entry:
EDSPECT, EDTITLE, EDAXIS, EDINTEG, EDPARAM, EDPLABL.

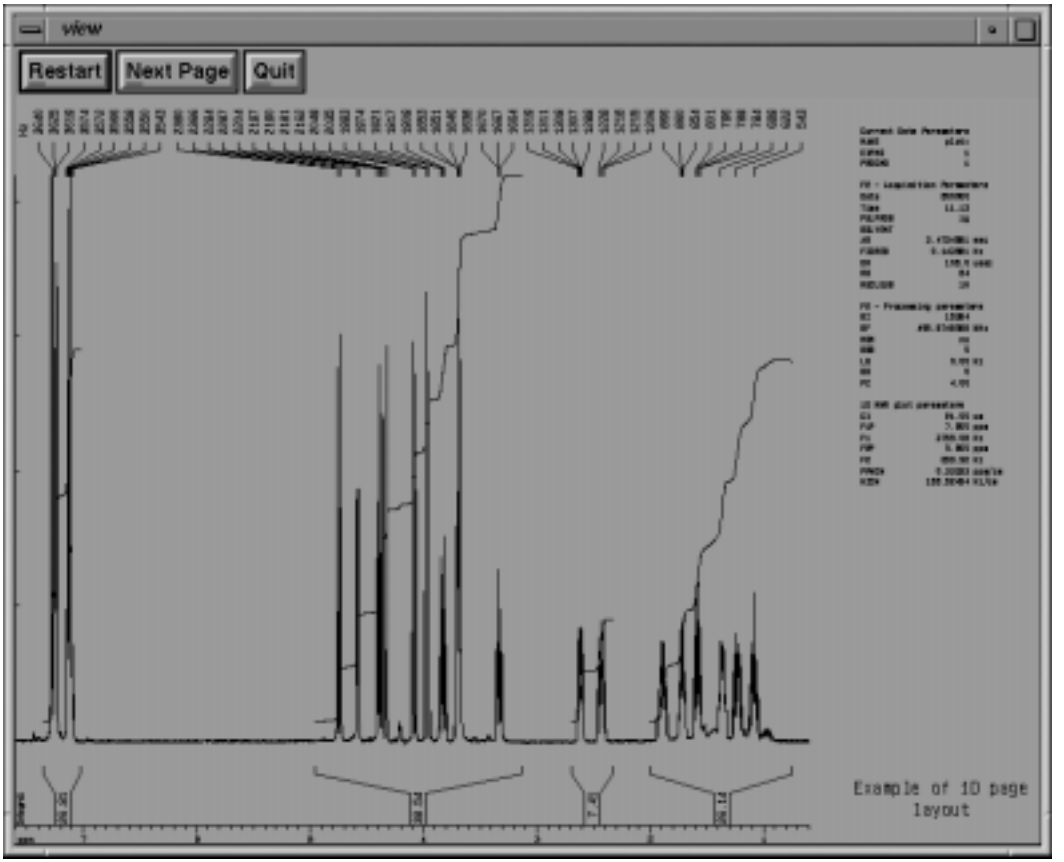


Figure 10.5 Basic 1D page layout

Please noted that all plot parameters contained in the edg dialog window pages may also be typed in on the keyboard using lower case characters, e.g. cy, followed by Return, or cy 20, followed by Return.

10.3.1 The spectral window

This window is subdivided into four, non-overlapping windows plottet from bottom to top:

1. *X axis* with height XHEI (cm)
2. *Numeric Integral Values* with height ILHEI (cm)
3. *Spectrum* and *Integral* with height DHEI (cm)
4. *Peak List* with height PLHEI (cm)

If the sum of these heights exceeds the specified height of the spectral window SHEI, the data window, i.e. DHEI, will be adjusted (shrunk) accordingly. The common width of these windows is determined by the parameter CX (cm).

10.3.1.1 The x axis

The x-axis is plotted as a horizontal line from the upper left corner of the x-axis window, the scale marks show towards the bottom. The tic marks are located in the lower part of the x-axis window. The window height XHEI must be set sufficiently large to hold the axis and its labels. Other parameters are:

XTICLEN (mm): tic length of scale marks.
XUNIT: physical unit of x-axis scaling (Hertz or ppm).
XTICDIS: distance between scale tics in the unit given in XUNIT. Only values with a mantissa of 1, 2, 5 or 25 are allowed to ensure reasonable axis labeling. Examples: 1.0, 0.1, 0.25, 50.0, 2.0. Not allowed: 3.0, 2.2, 5.25, etc.
XCHAR (mm): character height for axis labels.
XCOLOR: color in which the x-axis, scale tics, and labels will be plotted (according to the color/pen position assignment command edpp).

10.3.1.2 The y axis

A y axis can only be plotted with certain limitations: the axis will be plotted on the left edge of the spectral window with scale tics showing to the right. Labeling of the y axis has not been provided. The following parameters determine the plotting of the y-axis: YTICLEN, YTICDIS, YCOLOR. They have the same meaning as the corresponding parameters for the x axis.

10.3.1.3 The numeric integral values

The window for the integral values and the plot of their intervals lies directly above the x axis window. The operator can control the plot with the following

parameters:

ILCHAR (mm) character height for the digits of the integral values.
ILDIGIT number of digits with which to plot the greatest integral value. The other values will be plotted with leading blanks.
ILCOLOR color for integral values

The integral values appear in the plot between two lines marking the corresponding integration interval. These lines will either be cut off at the upper end of the integral values window (of height ILHEI), or they will be continued from there to the zero line of the spectrum, depending on the parameter ILONG (*yes*=continue, *no*=clip).

XWIN-NMR provides the capability of generating integral plots on which the numeric integral values are scaled relative to the previous plot in order to be able to compare the values of different spectra. While relative scaling of the integral traces on a plot is enabled by setting the plot parameter IHEI=-1, relative scaling of the numeric values is achieved by setting the processing parameter INTSCL=-1. In detail, INTSCL can be set as follows:

INTSCL > 0
The numeric values are scaled individually for each spectrum plotted. The user can influence the absolute value by adjusting the first line in the *intrng* file (see description).

INTSCL = 0
The integrals on a plot will obtain the same numeric values as the user defined in the interactive integration routine on the screen.

INTSCL = 1
Scaling is performed relatively to the last spectrum plotted. XWIN-NMR maintains the scaling factor in the file

/u/prog/ < version >/curdir/ < user > /scale.

10.3.1.4 The spectrum

As has already been mentioned, the spectrum will be plotted in the data window of height DHEI within the spectral window. The data window is immediately above the integral window. The zero line of the spectrum (with intensities of value zero) is placed SZERO cm above the lower edge of the data window (and thus SZERO cm above the upper edge of the integral window).

The horizontal scaling of the spectrum is controlled by the parameters F1 and F2 or F1P and F2P. These parameters set the range of the spectrum to be plotted if the parameter LIMITS is set to fixed. These parameters are best set with the *Define plot region* commands of the *Output* menu after adjusting the desired region on the display, or explicitly with the parameter editor.

F1 (Hz), F1P (ppm)
These parameters determine the left (low field) limit of spectral region to be plotted. If this parameter is changed, then the right plot limit F2/F2P is automatically adjusted so that the plot resolution (Hz/cm) and the width of the spectral window (CX) remain unchanged. Therefore a change of F1/F1P means a shift of the plot region.

F2 (Hz), F2P (ppm)
These parameters determine the right (high field) limit of spectral region to be plotted. A change in this parameter leaves the width of the spectral window CX and the left plot limit F1/F1P unchanged, while the scaling HZCM/PPMCM (Hz or ppm per cm) is adjusted accordingly. Therefore, a change of F2/F2P yields an expansion or compression of the plot region.

PPMCM (ppm/cm), HZCM (Hz/cm)
These parameters determine the horizontal resolution of a plot. If changed, the plot region F1/F2 is left constant and the width of the spectral window CX is adjusted according to the expression HZCM = (F 1- F 2)/CX.

The *vertical scaling* of the spectrum is controlled by the parameters PSCAL and CY. With the processing parameter PSCAL the user defines a position in the spectrum which then serves as a reference for the vertical scaling. CY can be positive, zero, or negative. The following paragraphs discuss these values.

CY > 0
If CY is greater than zero, then the intensity at the reference position will be plotted with height CY cm. All other intensities will be scaled relative to that position. PSCAL can take the values *global, preg, ireg, pireg, sreg,* and *noise*:

1. *global*
The highest intensity of the entire spectrum serves as the reference.

2. *preg*
The highest intensity within the plot region (i.e. the spectral window determined by F1/F2) serves as the reference.

3. *ireg*
The highest intensity in those spectral regions which are specified in the text file *reg* (in the current processed data directory), serves as the reference. This text file can either be created manually (by use of the command edmisc), or by entering XWIN-NMR's integration mode: Define the integration regions, and store them with the command Save as 'reg' of the *File* menu. If no such file exists, *global* is assumed.

4. *pireg*
As with *ireg*, the highest intensity within the spectral regions stored in the file intrng serves as the reference. Only those regions of the file which lie within the plot range, i.e. within the spectral area F1/F2, will be considered. If no such file exists, *preg* is assumed.

5. *sreg*
As in the case of *ireg*, the highest intensity of a set of spectral regions stored in a text file will serve as the reference. However, these regions will not be taken from the file *reg*. Instead, these regions need to be entered into a file whose name is stored in the processing parameter SREGLST. This file is located in the directory

/u/exp/stan/nmr/lists/scl/

and must contain the desired spectral regions in the same format as *reg*. *sreg* makes it possible to define spectral regions dependent on solvents.
SREGLST is automatically set to the file name NUCLEUS.SOLVENT by the automation command rjx/rjxn, for example SREGLST=*13C.Acetic* if NUCLEUS=*13C* and SOLVENT=*Acetic*. For a large number of nucleus/ solvent combinations, the XWIN-NMR release tape contains the corresponding region files in the directory

/u/exp/stan/nmr/lists/scl.exam/.

The command expinstall copies them to the directory *scl*.

6. *psreg*
This is identical to *sreg*, but only those regions in the file which lie within the plot region defined by F1/F2 will be included. *psreg* is an analogue to *sreg*, just as *pireg* is to *ireg*.

7. *noise*
Instead of the intensity at a certain position of the spectrum, the intensity height of noise is plotted with the height CY cm.

ASSFAC

If required, the second largest peak can be used for the vertical scaling of a plot rather than the biggest peak. This might be desirable if the biggest peak is of no interest for the user and would cause the peaks of interest to be plotted too small. The processing parameter ASSFAC accounts for this case:
If ASSFAC > 1, the second largest peak will be used as reference for plot scaling if

$$h2 < hmax/ASSFAC,$$

where h2 is the intensity of the second biggest and hmax that of biggest peak. If this condition is not fulfilled, the biggest peak is taken as reference.

If ASSFAC < -1, two plots will be generated on two sheets of paper if

$$h2 < hmax=abs(ASSFAC).$$

The first plot is performed with the second biggest peak as reference, the second plot with the biggest peak. However, the second plot is omitted in the case of 2D projections, or if it was issued with the commands plots or plotx.

If ASSFAC has any other value, the biggest peak serves as reference.

The search for the second biggest peak is not performed in the vicinity of the biggest peak. The size of this interval must be specified by the processing parameter ASSWID in Hertz.

CY=0
If CY is zero, the spectrum will be plotted with exactly the same vertical scaling as the operator has set up interactively on the display and stored with one of the *Define region* commands of the *Output* menu.

CY=-N
Scale a spectrum on the plot relative to the spectrum last plotted with CY>=0, and apply the factor N. For example, if CY=-1, the spectrum will be plotted using the scaling of the last plot with CY>=0 and you can visually compare the plots. If CY=-10, the spectrum will be 10-fold enlarged with respect to the last plot with CY>=0. This can be useful for difference spectroscopy. The program stores the scaling factor of a plot with CY>= 0 in the parameter file

/u/prog/ < version > /curdir/ < userloginid > /scale

as soon as the plot command is issued. The next plot request will cause the scaling factor to be taken from this file if CY<0.

Color

The color in which a spectrum is to be plotted is specified by the parameter SCOLOR. The operator is given a choice of several colors. In order for the spectrum to actually appear in this color, if a pen plotter is used, the plotter pens must be in their corresponding holders and the assignment of colors to pen positions must be correct (see command edpp).

10.3.1.5 The Integral

Like the spectrum, the integral is plotted inside the spectral window. The zero line of the integral is defined relative to the zero line of the spectrum by placing it IZERO cm above this line. Since the spectrum's zero line is drawn SZERO cm above the lower edge of the data window, the integral's zero line is plotted SZERO cm above this lower edge. The color of the integral is determined by the parameter ICOLOR. The vertical scaling is done by the parameter IHEI (cm), which affects the integral in the same way as CY affects the spectrum (cf. the previous section).

IHEI
The biggest integral contained in the file *intrng* is plotted with height IHEI cm. If one prefers to use the second largest integral for scaling, the parameter ASSFACI must be set appopriately, if

$$h2 < hmax/ASSFACI$$

where h2 is the value of the second largest and hmax that of the largest integral, the second largest is taken, otherwise the largest.

IHEI=0
The integral is plotted with the height set up by the operator on the screen.

IHEI=-1
The integral is plotted such that it may be compared with the previous integral plot. The vertical scaling factor is taken from the file

/u/prog/ < version > /curdir/ < userloginid > /scale,

where the last scaling factor with IHEI >= 0 was stored.

10.3.1.6 Peak lists

The top window within the spectral window contains a list of peak positions, with vertical lines simplifying their identification in the spectrum. The listed maxima and minima are identical to those output by the command pp, if they all fit into the spectral window of width CX. If this is not the case, then the available room will

first be shared by the highest intensity peaks, with the restriction that the horizontal distance between a label and the corresponding peak not be greater than ca. 3 cm. Therefore, the reason for unlabeled lines is usually that there is not enough room available. The labeling is controlled by the following parameters:

PLHEI
Height of the peak list window (in cm).

PLCHAR
Height of characters used (in mm).

PLDIGIT
Total maximum number of digits to be used for one peak label. This allows you to set the number of significant digits for output. PLDIGIT includes the decimal point and the sign.

PLMRK
This parameter can take values *yes* and *no*. In the former case, all intensities found by the peak search will be marked with a little vertical line. It appears as a plus sign, if it did not fit into the peak list due tp space reasons.

PLCOLOR
Color of peak list characters.

PLUNIT
Units to be used for peak list (Hz or ppm).

PLMUL
This parameter, with the values *yes* or *no*, is provided for additional labels specifying the multiplicity of a line. An analysis of multiplicities is prerequisite.

10.3.2 The Title Window

Any text stored in a text file may be output in this region of the plot. The name of the text file must be specified in the plot parameter TITNAM. The following rules apply:

- A name starting with a slash (/) is interpreted as a full path name to the title file. Example: TITNAM=*/u/guest/title1*
- If TITNAM starts with a Dollar sign ($), then XWIN-NMR expects the name of an environment variable immediately after the $. The current value of that environment variable is then substituted for the title file name, perhaps fol-

lowed by additional path specifications.
Example:
TITNAM=$XYZ/title2
would cause the title text to be taken from the file */u/guest/title2* if the environment variable is set to XYZ=*/u/guest*. If the environment variable is undefined, no title will be plotted.
A special case is TITNAM=*$STAN/....* Then the title text is taken from the file */u/exp/<ExpLoc>/nmr/...*
where *ExpLoc* is selected by the setres command (or from th menu *Display->Options->User interface*). However, if this one is undefined, stan will be substituted for it.
Example:
TITNAM=*$STAN/titledir/title3* with ExpLoc=*guest* is resolved to */u/exp/guest/nmr/titledir/title3*.

- If the name does not start with one of the above special characters, then the specified file name or path is searched for in the current data directory.
  Example:
  TITNAM=*title4*
  is resolved to
  DU/data/USER/nmr/NAME/EXPNO/pdata/PROCNO/title4,
  where DU, USER, NAME, EXPNO, PROCNO refers to the parameters describing the current data set.
- If the name starts with the characters dot dot slash (../), then the current data set directory is also the starting point, but for each occurrence of ../ Uxnmr moves up one step in the path name.Example:
  TITNAM=*../title4*
  is resolved to
  DU/data/USER/nmr/NAME/EXPNO/pdata/title4, and
  TITNAM=*../../info*
  is resolved to
  DU/data/USER/nmr/NAME/EXPNO/info.
  With the latter example it is thus possible to print out the *info* file which was set up with the command edinfo and which is stored hierarchically together with the acquisition data.

If TITNAM is not defined at all, XWIN-NMR automatically substitutes the name *title* for it, so that the parameter TITNAM only needs to be defined if the title text is to be taken from a file other than title. The command setti is used to set up this

file *title* in the current data set directory. This command calls the system's text editor. If other files are to be used as title files, then these have to be edited by directly calling the system editor.

As has already been described in the beginning of this chapter, the position of the title window is determined by the parameters TXLLEFT and TYLLEFT (all in cm), with the following exception. If the plot parameters LIMITS and ADJUST (see below for their discussion) are set to region and CX, respectively, the total length of the plot is not known in advance. In order to avoid overlapping of spectrum and title, the following parameters are effective:

TPOS now determines the position of the title window. Two values are selectable: *top* causes the title window to be positioned above the spectral window. *right* causes it to be appended to the right of the spectral window.

TOFFSET determines the distance (in cm) between the title window and the left, or top edge of the spectral window (for TPOS = *top* or *right*, respectively).

The size of the title window is specified by TWIDTH and THEI in cm, as described earlier. The title is plotted inside with a white margin of width TMARGIN cm around it. The text is rotated by +90 or -90 degrees if the parameter TROT was set to one of these values instead of 0. The character size is determined by TCHAR in mm. The characters may also be tilted with TSLANT specifying the tangent of the angle with the vertical. TFONT selects the character set to be used, which may be selected from Table 10.1.

If TMODE=*a* (auto wrap), the plot software will divide a continuous text into individual lines automatically. A Return in the text will always cause a new line. The user may force a new line anywhere in the text (even within a word) by inserting the character !. This character must be defined as the separation character in the parameter TSEP. Any special character may serve for this purpose. The color of the text is specified by TCOLOR.

If TMODE=*o*, a long text line will not automatically be continued on a new line if there is not enough room, but clipped instead. Also, the separation character (TSEP) has no effect.

The position of the text within the title window may be controlled via the parameter TALIGN. The following values may be selected:

TFONT	Font
0	ISO international
1	ISO BRITISH
2	ANSI ASCII
3	ISO SWEDISH
4	JIS ASCII
5	ISO SPANISH
6	ISO GERMAN
7	ISO FRENCH
8	ISO_NORWEGIAN

Table 10.1 Fonts available for plot title

- TALIGN=3
  The text is written left adjusted, starting at the upper left corner of the title window.
- TALIGN=2
  The text is written left adjusted, but vertically centered in the middle of the title window.
- TALIGN=1
  The text is written left adjusted, with the last line starting in the lower left corner.
- TALIGN=9, 8, 7
  Analogous to 3, 2, 1, the text being right adjusted.
- TALIGN=6, 5, 4
  Also analogous to 3, 2, 1, but centering each line of text horizontally, i.e. leaving the same distance to the left and right end of the title window. 5 will cause horizontal and vertical centering, positioning the text at the middle of the title window.

If the text, considering all these parameters, does not fit into the title window, the program will decrease the character size. This is done until a minimum (still readable) size is reached. Then the text is clipped at the edges of the title window.

10.3.3 The Parameter Window

In the parameter window (with the upper left corner being defined by PXULEFT and PYULEFT in cm), acquisition, processing and plot parameters are output. The operator may specify which parameters to plot in the order and format he desires. In order to do this, he must edit the text files (format files) in the directory */u/exp/stan/nmr/form/* (Table 10.2).

acqu.l/normpl	for acquisition parameters (F2 dimension)
acqu2.l/normpl	for acquisition parameters (F1 dimension)
proc.l/normpl	for processing parameters (F2 dimension)
proc2.l/normpl	for processing parameters (F1 dimension)
plot.l/normpl	for plot parameters (F1 dimension)
plot2.l/normpl	for plot parameters (F2 dimension)

Table 10.2 Format files for parameter plot

The format files that have come with your system may serve as examples. If a parameter is to be removed, its entry must be deleted up to and including the corresponding END. For adding parameters, it is best to use the existing entries as guidelines. The parameters appear on the plot in the same order as listed in the format file. Before modifying any of the standard files, it is recommended to make a copy.

If only those acquisition parameters that were really used in the pulse program are to be plotted, all acquisition parameters must be placed after the CONDITIONAL: statement at the end of the format file, as is the case for the acquisition parameter set up commands as and ased. If CONDITIONAL: is the last statement of the format file, only those parameters that occur before it are plotted.

Should a parameter plot be based on a format file generated by yourself, the name of it must be specified in the parameter PFORMAT with edo, replacing *normpl* there.

The parameters are plotted in the color PCOLOR and with character size PCHAR (mm), starting at the upper left corner of the parameter window. The width of the parameter window is PWIDTH cm. If chosen too small, the program automatically decreases the character size, but only to a minimal, still readable

size. If a parameter line is still too long, all characters outside the window are clipped.

The whole parameter window is rotated about the upper left corner by 0 or -90 degrees, if the parameter PROT is set to one of these values instead of 0.

10.3.3.1 Miscellaneous topics

Taking the plot limits from a region file
As described above, the parameters F1 and F2 determine the plot limits if the parameter LIMITS is set to *fixed*. If it is set to *region*, the plot limits are taken from the text file *reg*, which must be stored in the current processed data directory. The file format of *reg* equals that of a region file used for integration (see The *File* menu, command edmisc). The left plot limit is the biggest frequency value listed in *reg*, the right plot limit the smallest one.
If LIMITS = *region* is used, several other parameters (specifying the x-scaling of the spectrum) need to be considered:
ADJUST can have values *scale* and *CX*. In the former case the x scaling is calculated from the plot region (taken from the file *reg*) and the width CX of the spectral window. In the latter case, the plot length is adjusted to fit the plot resolution as set by the user with the parameter SCALEP (in ppm/cm) or SCALEM (in Hz/cm). The parameter window is plotted unrotated left of the spectral window independent of the setting of the parameters PXULEFT, PYULEFT, PROT. The position of the title window is determined by the parameters TPOS, TOFFSET.

The file reg can be obtained in different ways:

- With edmisc.
- By reading it in from a prepared file using rmisc.
- By generating a *intrng* file by manually integrating the spectrum in integration mode and storing the integration regions, or by generating *intrng* using the command abs. If no *reg* file exists but *intrng* exists, the latter file is copied to *reg*.
- If neither *reg* nor *intrng* exist, *intrng* is automatically generated through an internal call of the command abs.

Rotation and reflection
The whole plot is rotated by 90, 180 or 270 degrees if the parameter WROT is set

to one of these values instead of 0. Some plotters allow the reflection of the plot along a vertical line through the middle (e.g. for transparencies, cf. manufacturer's manual). This may then be activated by setting the parameter WMIR to *yes* instead of *no*.

Suppress plot if no peaks found (conditional plot)
For automated acquisition and processing it is sometimes useful if an already issued plot command can suppress the plot when no lines can be found in the plot region. To accomplish this, the parameter SUPPRESS must be set to yes instead of no. The detection of peaks is analogous to that of the processing command pp.

Automatic continuation on additional paper sheets
The length of a plot is physically limited by the paper size for single sheet plotters, or by a manufacturer's setting for plotters with endless paper. If the parameter CLIP is set to *yes*, all plots will be clipped at those limits, even if plot parameters, like CX or CY specify a larger plot. If it is set to *no*, however, the plot will always be finished according to the parameters, if necessary on several sheets of paper. This applies to the x direction as well as to the y direction and is especially important for plots showing sufficient resolution and small paper size. The program will draw marks at the edges so that the single sheets may easily be joined together for the whole plot.

Plotters with cutter accessory
Various types of plotters with endless paper can automatically cut off the paper. For these plotters the parameter CUTP has been introduced. If set to *yes*, the paper is cut off automatically after completion of a plot.

10.4 Page layout for auto-expansions [edgx]

XWIN-NMR allows you to generate plots containing a number of selected plot regions which are automatically expanded in the vertical direction to fill the available space, optionally with an overview spectrum (Figure 10.6). Reasonable automatic plots may thus be obtained from samples with small signals and integrals in the presence of large ones.

Auto-expansions are set up with the command edgx, which is analogous to the command edg. Parameterscontained in the edgx dialog window can also be entered from the keyboard by preceding their name with **px**, e.g. px shei, followed by

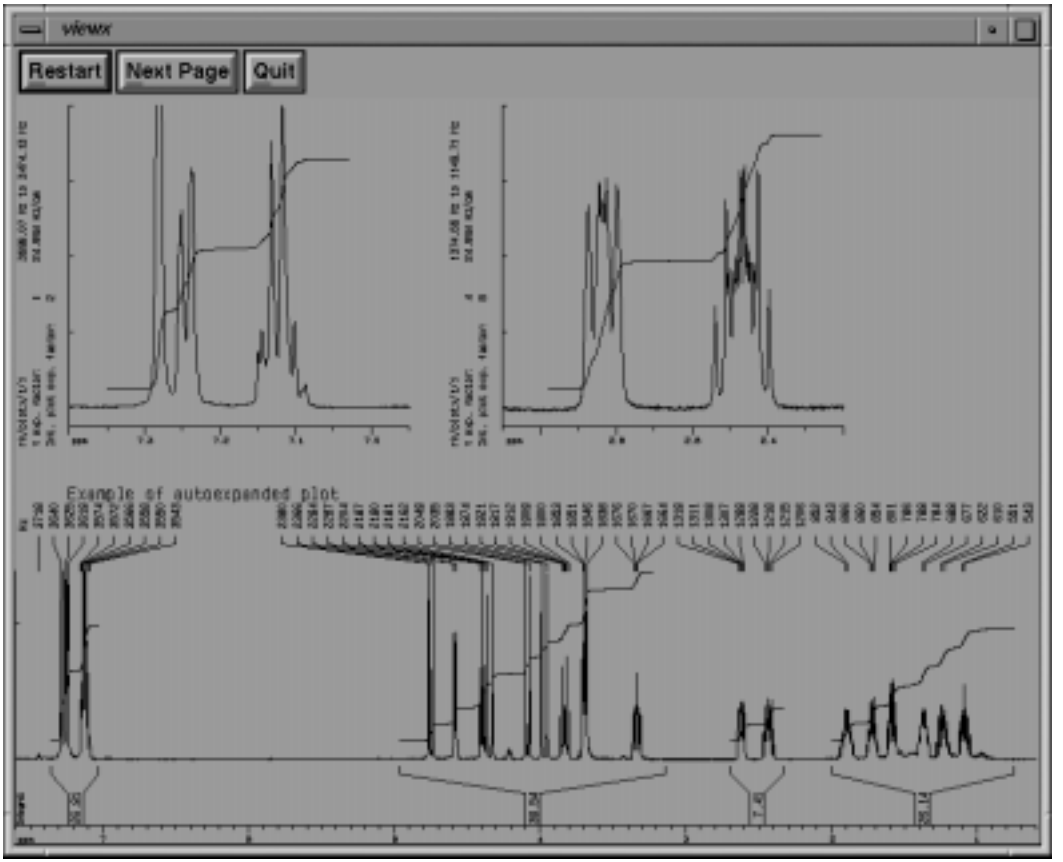


Figure 10.6 Example of auto-expanded plot regions

Return, or px shei 5, followed by Return.

The spectral regions to be auto-expanded are taken from the text file *reg*, the format of which is described in the chapter *The File Menu*. This file is stored in the same directory as the spectrum, and may either be set up manually with edmisc read as a previously set up file with the command rmisc, or generated in manual integration mode. If no *reg* file exists in the current data set directory, but the file *intrng* does, then the *reg* file is created automatically by copying *intrng*. If *intrng* also does not exist, an error message is issued.

10.4.1 The plot limits

The plot limits for normal plots are defined using edg, for plot expansions they must be set in the file *reg* as described above. There are two special parameters, namely EXTF1 and EXTF2, or EXTF1P and EXTF2P respectively, which are specified in Hertz or ppm. These parameters ascertain that only those regions listed in the *reg* file, which are at least partially contained in the interval EXTF1-EXTF2, are plotted.

10.4.2 Only plot those regions containing signals

The parameter SUPPRESS controls whether only those regions which contain at least one signal (searched for with pp) are plotted.

10.4.3 Horizontal scaling

The spectral regions listed in the file *reg* are plotted with the horizontal scaling specified in the parameters SCALEP (ppm/cm) or SCALEH (Hz/cm). The lengths of the region plots are adjusted accordingly.

10.4.4 Automatic vertical scaling

The vertical scaling of the region plots and their integrals is determined automatically and depends on the room available, i.e. on the difference DHEI - SZERO, which is the room above the spectrum's zero line (DHEI - IZERO for integrals). In edgx the parameters used for normal plots, CY and IHEI, are not available. The vertical scaling is always an integer multiple of the scaling of the overview spectrum (the *overview* spectrum can be plotted on the same page along with the auto-expansions, as will be discussed later). For the scaling of a region, XWIN-NMR looks at the intensity of a reference signal in this region. Only those signals are considered, which are smaller than the plot reference signal of the overview spectrum. Except for this restriction, the reference signal of a region is the one with maximum intensity therein, if the processing parameter ASSFACX is in the range 1 to -1. Otherwise, if

- ASSFACX > 1
  If h2 < hmax/ASSFACX
  then the second largest signal becomes the reference signal. If not, the largest signal becomes the reference signal.

- ASSFACX < -1
  Every region is plotted twice, once with the largest and once with the second largest signal as a reference.

The second largest signal is only searched for outside an interval around the largest signal. This interval is to be specified by the processing parameter ASSWID (in Hertz).

The vertical scaling factor of a region is assigned the greatest integer with which the intensity of the reference signal can be multiplied without exceeding the size of the plot's data window. The room available is given by the difference DHEISZERO (DHEI-IZERO for integrals). These parameters must be set with edgx.

The scaling factor can never become smaller than one, even if DHEI was set too small. On the other hand, this factor will only become so large that the noise will not exceed 3 cm.

The scaling of the integrals is determined by the largest integral value in this region which is not greater than the integral plot reference signal of the overview spectrum. A parameter ASSFACI which exists for the overview spectrum is not available for the integrals of the regions.

10.4.5 Parameter Lists

Auto-expanded regions cannot contain parameter lists. However, XWIN-NMR plots the text of Table 10.3 (rotated by 90 degrees) at the left of each expansion:

user/name/expno/procno/		dddd.dd Hz to dddd.dd Hz
Y exp. factor:	ddd	dd.ddd Hz/cm changed !
Int. plot exp. factor	ddd	

Table 10.3 Text plotted along with an auto-expansion

Character size and color can be specified with the parameters PCHAR and PCOLOR, respectively. The first line gives the data set description and the plotted frequency range. The second line shows the applied vertical expansion factor (integer, referring to the overview spectrum's scaling) which allows for the comparison of intensities in different regions. The horizontal scaling factor *changed!* only appears if the program had to change the value set by the user with

edgx because the region would not have fit onto the sheet otherwise. The third line shows the vertical expansion factors of integrals and is omitted if no integrals were to be plotted.

10.4.6 Plot Layout

There are four possible layouts for auto-expanded plots. The three including the overview spectrum are shown schematically in Figure 10.7. For normal plots the position of the spectral window on the paper is specified by the parameters SXLLEFT and SYLLEFT. As these parameters are not available for expansion plots, the parameter LAYOUT takes their place and can take on the following values (Figure 10.7):

- *separat*
  In this case only the regions listed in the file *reg* are plotted. The plots start at the bottom left corner of the sheet and are plotted one after the other. The following rule applies to single-sheet plotters and endless-paper plotters with PAPERFD=*single*: if the next region completely fits onto the same sheet, it will be plotted there, otherwise a paper feed is performed. If the plot of a single region already exceeds the size of the paper, the horizontal scaling is increased by an integer factor. On endless-paper plotters operating with PAPERFD=*multiple*, all partial plots are done in sequence. After the last plot, so much paper is advanced that the length of the total plot, that is the paper consumption of all region plots together, becomes a multiple of FAMOUNT.
- *top-of-one*
  In this case, the overview spectrum as well as all the regions specified in the file *reg* are plotted. First the overview spectrum is plotted according to the layout parameters set with edg. The regions are plotted above the overview spectrum. The sign of the parameter ASSFAC, which specifies the reference signal for scaling the overview spectrum, is ignored. If there is no signal in the region of the overview spectrum and the parameter SUPPRES=*yes*, the overview plot is omitted and the region plots are done as for LAYOUT=*separat*. The parameters CLIP, MIRROR and ROTATION for the overview spectrum are ignored. The three main windows of the overview spectrum (i.e. spectral, title and parameter window) are positioned on the paper independent of the values of the parameters LIMITS and ADJUST, as if LIMITS=*reg* and ADJUST=*CX*. If the room needed for the region plots exceeds the length of the overview spectrum, then the next regions are put to the right of the ones already plotted, again up to the length of the overview spectrum. The overview spec-

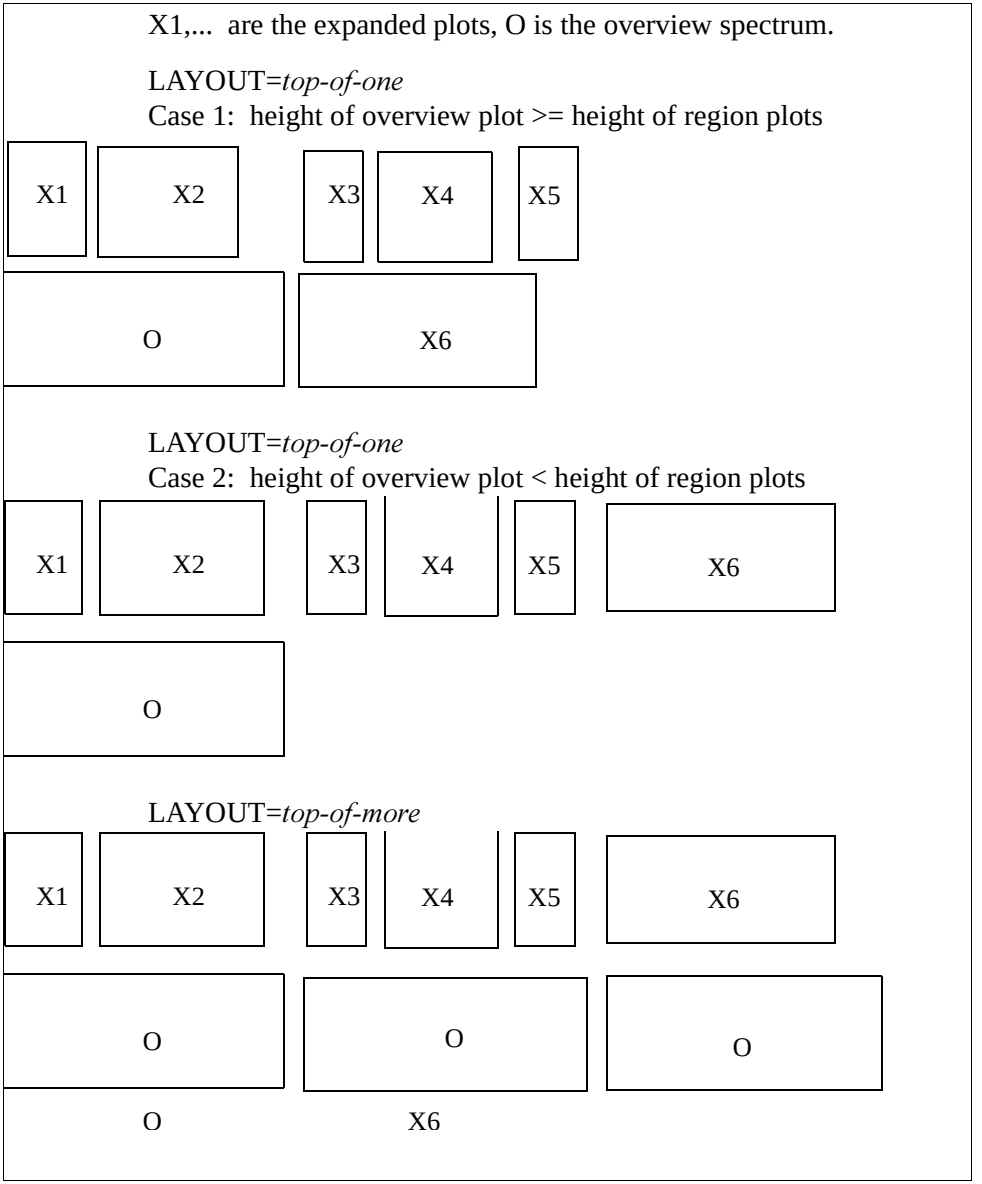


Figure 10.7 The three layouts of auto-expansion plots with overview spectrum

trum thus acts like a page. If there are still more regions to be plotted, they are put to the right of or underneath the ones plotted thus far, depending on the room they need vertically. Paper feed is done as for LAYOUT=*separat*.

- *top-of-more*
  Like *top-of-one*, except that all regions are plotted next to each other and the overview spectrum is plotted underneath each group of plots. Thus, each sheet of a single-sheet plotter has the overview spectrum.

10.4.7 Further differences between plot and plotx

The parameters CLIP, MIRROR and ROTATION are not available for expansion plots.

10.5 Using a prepared page layout [rpar]

XWIN-NMR is delivered with many prepared parameter sets suitable for different experiments. They also include plot parameter, i.e. layouts suitable to plot these experiments. The command rpar allows you to copy a selected plot layout to your current data set, thereby overwriting the current plot parameters. After rpar, you can call edg or edgx to apply modifications. rpar is described in the chapter *The File Menu.*

10.6 Define/show plot region

As described in the previous sections, the frequency limits of a plot are defined by the parameters F1/F2 (in Hz) or F1P/F2P (in ppm). They can be set in edg, or by typing f1, f2, f1p, f2p on the keyboard. An alternate, frequently used way to define the plot region is the interactive setting from the display. Expand the spectrum to show the region you want to plot, then select one of the following commands from this submenu.

10.6.1 Retain CX. Auto-adjust Hz/cm

F1, F2, F1P, F2P will be set according to the display limits. The plot length CX as defined by the user previously will be retained, and the frequency resolution (Hz/cm, ppm/cm) of the plot will be adjusted accordingly (new resolution=(F1-F2)/CX). The new resolution is printed on the status line.

10.6.2 Retain Hz/cm. Auto-adjust CX

F1, F2, F1P, F2P will be set according to the display limits. The frequency resolution (Hz/cm, ppm/cm) as defined by the user previously will be retained. The plot length CX of the plot will be adjusted accordingly (new CX=(F1-F2)/resolution). The new CX is printed on the status line.

10.6.3 Retain CX and Hz/cm. Auto-adjust F2

F1 and F1P will be set according to the display limits. The frequency resolution (Hz/cm, ppm/cm) and the plot length CX as defined by the user previously will be retained. The right plot limit F2 of the plot will be adjusted accordingly (new F2=F1-resolution*CX). The new F2 is printed on the status line.

10.6.4 Show plot region

Invoking this command will set the display such that it shows the range defined by the plot limits F1, F2, F1P, F2P.

10.7 Plot Commands

Before sending a plot to a pen plotter or printer, it is advisable to preview it using the command view or viewx (described in the chapter *The Windows Menu*).

10.7.1 Plot [plot]

The command plot outputs the layout set up with edg or rpar on the current plotter CURPLOT, to be defined with edo. CURPLOT does not play a role if the *Plotter* system variable is set, which has priority over CURPLOT (use the command setres of the chapter *The Display Menu*). As soon as the layout has been plotted, the next paper sheet is fed into the plotter.

10.7.2 Plot to queue [plots]

plots (*plot suspend*) outputs the layout into a queue rather than sending it to the plotter. This allows you to set up more plots of the same or a different data set, and send those to the queue. All plots contained in the queue are plotted on the same sheet of paper by the command flplot. Complicated layouts may thus be created. plots can also be included in AU programs (from where you can also set plot

parameters with the function *storeplpar*). Layouts with several titles, expansions etc. can be realized this way.

A plot queue is stored in the directory

/u/prog/<XWIN-NMR version>/curdir/<user>/mgplot/.

The queue consists of the parameter files *meta.1*, *meta.2*, ... , corresponding to the number of plots command given, and the text files *info.1*, *info.2*, ... , containing the data set names and other information.

10.7.3 Output queue on separate sheet [flplot]

flplot outputs the contents of a queue filled by one or more plots commands, to the current plotter.

10.7.4 Remove all plots from queue [rmplot]

All plots issued with the plots command are deleted, if they were not sent yet to the plotter with flplot.

10.7.5 Plot expansions [plotx]

The command plotx outputs auto-expansions set up with edgx (and the corresponding overview spectrum set up with edg, if enabled) on the current plotter.

10.8 Plot/print status

The commands plot, plotx, and flplot let the operating system's *lp* spooler handle the plotter or printer devices. *lp* actually sends the data bytes out to the parallel or serial channel. Each plot gets a *plot request id number* assigned, which is printed in an information window when the plot command is given (unless the message is disabled due to the setting of the *PlotMsg* system variable, see command setres). When a data file is plotted, a copy of it (realized via a link) is used stored in the directory

/DU/data/USER/plotfiles/<ProcessID>/,

created by the plot command. This ensures that you can further process the data file, even if the plot is not yet complete.

10.8.1 Show status [uxlpstat]

uxlpstat lets you examine the *lp* status. Particularly, you can see by the *request ids* which plots are still waiting for execution. For more information on the lp spooler, please refer to the documentation of the operating system (e.g. use the Unix command man lp). uxlpstat corresponds to the Unix command lpstat.

10.8.2 Cancel plot/print request [uxcancel]

uxcancel allows you to cancel a plot request by specifying its id number. If plotting is already in progress, you may have to reset your plotter/printer afterwards. uxcancel corresponds to the Unix command cancel.

10.9 Screen dump

This command should only be used on AspectStation computers. On Silicon Graphics systems, we recommend the *snapshot* program. *snapshot* allows you to make screen dumps of any portion of the monitor, and store it in a so-called *.rgb* file. Programs are available (also from public domain or shareware, such as *convert* and *xv*) to convert *.rgb* to any other bitmap format (such as *.bmp*, *.ps*, *.epsi*, *.tiff*, *.gif*, *.xwd*, etc.).

The screen dump command puts the bitmap in a file of your home directory with the extension *.lj* (LaserJet II format) or *.ps* (PostScript format), according to your choice. You may either dump the entire XWIN-NMR window, or the data area only.

10.10 Print status parameters

XWIN-NMR maintains two sets of acquisition and processing parameters for any data set. The first set contains the parameters set up with the commands eda and edp, the second one contains the so-called status parameters. A status parameter has the same name as its corresponding set up parameter, e.g. NS or PHC0. It is written into a status parameter file upon completion of an operation. For example, at the end of an acquisition the last scan acquired is stored as parameter NS in the acquisition status parameter file of the current data set. NS need not be equal to the set up parameter NS; you might have entered NS=16 with eda, but halted acquisition after 8 scans. This is the reason for the two sets. Table 10.4 shows the file names of the status parameter files. They are stored in the acquisition or processed

acqus	acquisition status parameter file
acqu2s	acquisition status parameter file for 2nd dimension (if 2D parameter set)
procs	processing status parameter file
proc2s	processing status parameter file for 2nd dimension (if 2D parameter set)

Table 10.4 Status parameter file names

data directory of the current data set.

The command lpa and lpp print the acquisition and processing status parameters on the current printer CURPRIN (to be set up with edo). In addition, lpg and lpgx print the plot parameters, i.e. the frequency range of a plot and the resolution Hz/ppm. lpc lists the data set parameters DU, USER, NAME, EXPNO, PROCNO. lp is identical to the sequence lpc, lpa, lpp, lpg. CURPRIN must contain a valid printer name made known to XWIN-NMR with cfpp. The printer name must be preceded by a $ sign, otherwise the name is interpreted as a file name in the current processed data directory, and the parameter output is stored there.

A special command is lppl. It stores a parameter list called *param.txt* in the current processed data directory. From this file, the parameter list of a plot is generated. For this reason, lppl is called by any plot command that is to output parameters.

The format of a parameter printout is taken from a *format file*. It contains the parameters to be included in the list, and their numerical format and physical units. Format files are stored in the directory */u/exp/stan/nmr/form/*. Table 10.5 shows the

lpc	*curd.l/normlp*
lpa	*acqu.l/normlp*
lpp	*proc.l/normlp*
lpg	*plot.l/normlp*
lpgx	*plotx.l/normlp*

Table 10.5 Format files for the lp commands

file names valid for the lp commands. For 2D and 3D data sets, the file names are the same, but the files for the second or third dimension are contained in the subdirectories *acqu2.l*, *proc2.l*, *plot2.l*, *acqu3.l*, *proc3.l*. The lp commands do not necessarily use the *normlp* format files: You may setup your own ones, and replace *normlp* in the parameter LFORMAT by the name of your file using the edo command.

10.11 Display status parameters

The commands dp, dpc, dpa, dpp, dpg, dpgx are analogous to the lp commands described in the previous section. Their output goes to screen, however, rather than to a printer or to a file. The format files for these command are listed in Table 10.5. You can specify own format files in the parameter DFORMAT, to be set up

dpc	*curd.l/normdp*
dpa	*acqu.l/normdp*
dpp	*proc.l/normdp*
dpg	*plot.l/normdp*
dpgx	*plotx.l/normdp*

Table 10.6 Format files for the dp commands

with edo.

10.12 Print barcode labels [prlabel]

The command prlabel prints barcode labels on a thermo-printer of type HCP-60. This printer is connected to a free RS 232 channel of the computer. The interface configuration is demanded by the command cfbacs and stored in the file

/u/conf/instr/barcode_prn

if a barcode reader exists. The printer must be configured according to Table 10.7.

This corresponds to the following settings:

	S1	S2	S3	S4
	12345678		1234	12345678
on	** *		**	
off	** ** *		* *	********

Table 10.7 Bar code printer configuration

9600 baud (S2)
7 data bits (S1:7-8)
2 stop bits (S1: 3-4)
No parity (S1:5-6)
automatic formfeed before label printing (S3:1)
printing speed max. 500 Hz (S3:2)
no XON/XOFF protocol (S3:3)

After entering prlabel the display shows a table which is headed by *define new series*. During the barcode operation XWIN-NMR keeps a protocol of which barcode ID numbers have already been measured in the file

/u/conf/instr/label_info.

Another measurement with the same number is then impossible, preventing already acquired data from being overwritten by further measurements. It is also impossible to create new labels for these numbers when selecting the first entry, *none*, from the table. In this case, only numbers not used so far are allowed for printing. If a certain range of numbers (e.g. 1-10000) is to be released again, because the corresponding data sets have already been archived or are no longer needed, the respective number range must be selected from the table.

Now prlabel shows the tables of the user login IDs, the solvents and the experiments. After selecting the desired entry, the ID number of the label to be printed is asked for, and XWIN-NMR will offer the next available number. Then the number of labels to be printed may be specified. If this number *n* is larger than 1, then *n* labels will be printed with increasing label ID number, but the same user ID, the same solvent and the same experiment. As soon as the labels have been printed, the user table appears again, and labels for a different user/solvent/experiment combination may be generated. prlabel is terminated via the ABORT field of the table.

10.13 Supported printers/plotters

Table 10.8 presents a list of the plotters supported by XWIN-NMR and/or XWIN-PLOT.

The tables following Table 10.8 inform you about the switch settings of a number of printers/plotters required for them to work correctly. The XWIN-NMR plotter type specified in the header of these tables is the name to be set in edo (parameter CURPLOT) to send to output to this plotter.

For the HPLJ5L, you can select either *hplj5l* or *hplj5lr* in cfpp. The first mode will generate ouput with 600 dpi, the second mode with 300 dpi resolution. The standard memory equipment of a HPLJ5L is 1MB. When using 600 dpi mode, more memory could be required for certain types of plots (especially for contour plots). This depends on the number of vectors generated.

Type	Format	XWIN-NMR	XWIN-PLOT	
+ = The printer/plotter was tested, ? = The printer/plotter will probably work but was not tested and is not supported by Bruker				
Canon BJ200	A4 ?		?	
Canon BJ330	14”el	+	+	
Canon BJC800	A3	+	+	
Canon BJC4550	A3	+	+	1)
GRAPHTEC FP6302	A3	+		
GRAPHTEC FP7200	A2	+		
GRAPHTEC GP1002	A0	+		
GRAPHTEC GP1002R	36”el	+		
GRAPHTEC MP3200	A3	+		
GRAPHTEC MP3200	A3	+		
GRAPHTEC MP4200	A3	+		
GRAPHTEC wx4731	12”el	+		
HP 7475A	A3	+		
HP 7550A	A3	+		
HP 7570A	A1	+		
HP 7575A	A1	+		
HP 7576A	A0	+		
HP 7580B	A0	+		
HP 7596A	36”el	+		
HP DraftPro Plus	A1	+		
HP LaserJet IId	A4		?	
HP LaserJet IIp	A4		?	
HP LaserJet 3P	A4	+	+	
HP LaserJet 4/4L/4ML	A4	+	+	
HP LaserJet 5L/5P/MP	A4	+	+	

Table 10.8 Supported printers plotters

Type	Format	XWIN-NMR	XWIN-PLOT	
HP LaserJet 4V	A3	+	+	
HP Color LaserJet	A3	+	+	
HP DeskJet 500	A4	+	+	
HP DeskJet 510/520	A4	+	+	
HP DeskJet 550C/560C	A4	+	+	
HP DeskJet 660C	A4	+	+	
HP DeskJet 690C	A4	+	+	2)
HP DeskJet 850C	A4	+	+	
HP DeskJet 870C	A4	?	?	3)
HP DeskJet 1200C	A4		+	
HP DesignJet 750C	A0	+	A1	
dito. +Postscript	A0	+	+	
HP PaintJet XL300	A3		+	
Kyocera F5000/F5500	A3	+	+	
LaserImage 1035	A4	+	+	
LaserPro Gold	A4	+		
Lexmark 4079 plus	A3	+	+	#
Nofatec RG04	A4	+	?	
ZETA 8	12”el	+		
ZETA 912	12”el	+		
ZETA 924	24”el	+		
ZETA 936	36”el	+		

Table 10.8 Supported printers plotters

XWIN-NMR plotter type: bj330

SW1:	01: on	SW2:	01: on	SW3:	01: off
	02: on		02: off		02: off
	03: on		03: off		03: on
	04: off		04: on		04: off
	05: off		05: off		05: off
	06: off		06: off		06: off
	07: on		07: off		07: off
	08: off		08: off		08: off
			09: off		
			10: off		

This is the configuration for the ASPECT X32; SW3 is on the serial interface board BJIF-3020. This board is not used for communication with an ASPECTstation.

In this case SW1:03 is to be set to off and the Centronics printer cable (part no. HZ03318) is to be used.

The pen thickness is realized in steps of 0.14 mm (1/180 ") with the minimum of 0.07 mm (1/360 ").

All other pen parameters have no meaning and are ignored.

Table 10.9 Canon BJ-330; bubble jet printer; 14" endless paper

XWIN-NMR plotter type: bjc800	
Configuration	original factory settings ("NVRAM INITIALIZE")
Interfacing	parallel
	When using a seriell parallel converter, set it to 38 400 baud. When using the newer Wiesemann & Theis Interface 82008, S2 is to be set to "on"; all other settings are the same as in the general part.

The logical XWIN-NMR colors are realized in the following way:

black	black ink
blue	CMY blue
violet	CMY blue
magenta	CMY magenta
red	CMY red
orange	CMY red
yellow	CMY yellow
green	CMY green
turquoise	CMY cyan
brown	CMY black

The pen thickness is realized in steps of 0.14 mm (1/180 ") with the minimum of 0.07 mm (1/360 ").

All other pen parameters have no meaning and are ignored.

Table 10.10 Canon BJC-800; A3 single sheet feeding bubble jet printer

XWIN-NMR plotter type: hp7475a	
B1:	0
B2:	1
B3:	0
B4:	1
A3/A4	depending on paper
MET	
S1:	0
S2:	0

Table 10.11 HEWLETT-PACKARD 7475A; A3 single shee

XWIN-NMR plotter type: hp7550A	
Configuration Dialogue:	SERIAL
MONITOR MODE:	OFF
DATA FLOW:	REMOTE; STAND ALONE
BYPASS:	OFF
HANDSHAKE:	XON/XOFF; DIRECT
DUPLEX:	HALF
PARITY:	8 BITS; OFF
BAUD RATE:	9600

Table 10.12 HEWLETT-PACKARD 7550A; A3 single sheet feede

XWIN-NMR plotter type: hpdj500			
A:	1: up	B:	1: down
	2: down		2: up
	3: up		3: down
	4: down		4: down
	5: down		5: down
	6: up		6: down
	7: down		7: down
	8: down		8: down
Interfacing:		serial or parallel	
The pen thickness is realized in steps of 0.17 mm (1/150 ") with the minimum of 0.085 mm (1/300 ").			
All other pen parameters have no meaning and are ignored.			

Table 10.13 HEWLETT-PACKARD DeskJet 500; A4 single sheet feeding printer

XWIN-NMR plotter type: hpdj550c/660c

A:	1: up	B:	1: down
	2: down		2: down
	3: up		3: down
	4: down		4: down
	5: down		5: down
	6: up		6: down
	7: down		7: down
	8: down		8: down

Interfacing: serial or parallel

The logical XWIN-NMR colors are realized in the following way:

black	black ink
blue	CMY blue
violet	CMY blue
magenta	CMY magenta
red	CMY red
orange	CMY red
yellow	CMY yellow
green	CMY green
turquoise	CMY cyan
brown	CMY black

The pen thickness is realized in steps of 0.17 mm (1/150 ") with the minimum of 0.085 mm (1/300 ").

All other pen parameters have no meaning and are ignored.

Table 10.14 HEWLETT-PACKARD DeskJet 550C; A4 single sheet feeding printer

XWIN-NMR plotter type: hpdprop		
Plotter Setup:		
Grapics Language		HP-GL (7595A)
Memory		off
Serial	Baud Rate	9600
	Handshake	both
	Parity	none
Page Format:		
Expand		on
Plot Scaling		01:01
Plot Management:		
File Timout		off

Table 10.15 HEWLETT-PACKARD DraftPro Plus; A1 single sheet

XWIN-NMR plotter type: hplj3p	
SIZE	A4
I/O	SERIAL
BAUDRATE	9600
ROBUST XON	OFF
DTRPOLAR	HI
PAGEPRO.	A4
RAM size:	3072K bytes (minimum total)
Interfacing:	parallel possible, too

Table 10.16 HEWLETT-PACKARD LaserJet IIIP; A4 single sheet feeding

XWIN-NMR plotter type: hplj4l (hplj3p is possible, too)	
PAPER SIZE:	A4
IMAGE ADAPT:	AUTO
PAGE PROTECT:	AUTO
AUTOCONTINUE:	OFF
ECONOMODE:	OFF
RAM size:	1024K bytes
Interfacing:	parallel

Table 10.17 HEWLETT-PACKARD LaserJet 4L; A4 single sheet feeding Laser Printer

XWIN-NMR plotter type: kyf5000	
H1(RS232C baud rate)	96
H2(RS232C data bits)	8
H3(RS232C stop bit)	1
H4(RS232C parity bit)	0
H5(RS232C protocol)	1
H9(F.F. time out)	0
P1(Default Emulation)	6
G0(KCGL option)	4

Table 10.18 KYOCERA F-5000; A4/A3 single sheet feeding Laser Printer

XWIN-NMR plotter type: kyf5000		
Current interface	RS232C	
RS232C baud rate	H1	96
RS232C data bits	H2	08
RS232C stop bits	H3	01
RS232C parity	H4	00
Protocol type	H5	03
F.F. timeout	H9	00
Default Emulation	P1	06
KCGL option	G0	04

Table 10.19 KYOCERA F-5500; A4/A3 single sheet feeding Laser Printer

XWIN-NMR plotter type: ps300
For printer settings see printer manual
Adobe and PostScript are registered trademarks of
Adobe Systems Incorporated in the U.S. and other countries

Table 10.20 Adobe PostScript compatible printer with 300 dpi

XWIN-NMR plotter type: wx4731										
	Interface PC2601					or		Interface PC2609		
	(1 means OPEN or ¦↑)									
S1(F):	1: 0	S2(E):	1: 0	S3(D):	1: 1		SW1:	1: on	SW2:	on
	2: 0		2: 0		2: 1			2: on		on
	3: 0		3: 0		3: 1			3: on		off
	4: 1		4: 0		4: 1			4: off		on
	5: 1		5: 1		5: 1			5: off		on
	6: 0		6: 0		6: 0			6: on		off
	7: 0		7: 0		7: 1			7: on		off
	8: 1		8: 0		8: 1			8: off		on

Table 10.21 GRAPHTEC WX4731; 12“roll pap

Chapter 11

The *2D Output* Menu

The *Output* menu (Figure 11.1) contains the plot commands and utilities to print

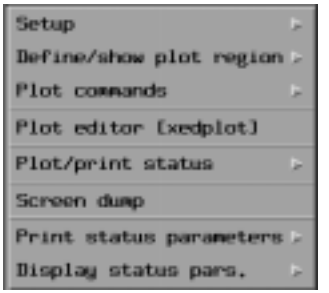


Figure 11.1 The *2D Output* menu

or display parameters. For plotting, the following steps are required:

1) Make all connected printers or plotters known to XWIN-NMR with cfpp
2) For pen plotters, specify the pen colors with edpp
3) Define a page layout with edg (contour plot), edgw (stacked plot) or rpar (predefined layout), and a plot title with setti
4) Define with edo to which of the connected plotters the output should be sent

(see also the setres command)
5) Preview the plot on screen with view (contour plot), or vieww (stacked plot). Adjust the layout if required.
6) Output the plot with plot, plots, flplot, plotw

The following sections describe these steps in detail. We assume, however, that you are already familiar with 1D spectrum plotting.

The set up commands are part of the *Output->Setup* submenu (Figure 11.2).

Figure 11.2 The *Output->Setup* submenu

11.1 Printer/plotter installation [cfpp]

This command is described in the chapter *The 1D Output Menu*.

11.2 Pen setup [edpp]

This command is described in the chapter *The 1D Output Menu*.

11.3 Page layout contour plot [edg]

If the current data set is a 2D spectrum, the command edg displays the layout parameters for a contour plot (Figure 11.3).

11.3.1 Components of a Contour Plot

Like a 1D spectrum plot, a 2D contour plot also consists of three spectral windows which may be positioned independently. They are (again): the spectral window, the title window and the parameter window. The latter two will not be discussed in this chapter, since they are completely identical to those of a 1D spectrum plot and have already been described in that chapter.

11.3.2 The Spectral Window

The spectral window includes the following plot objects

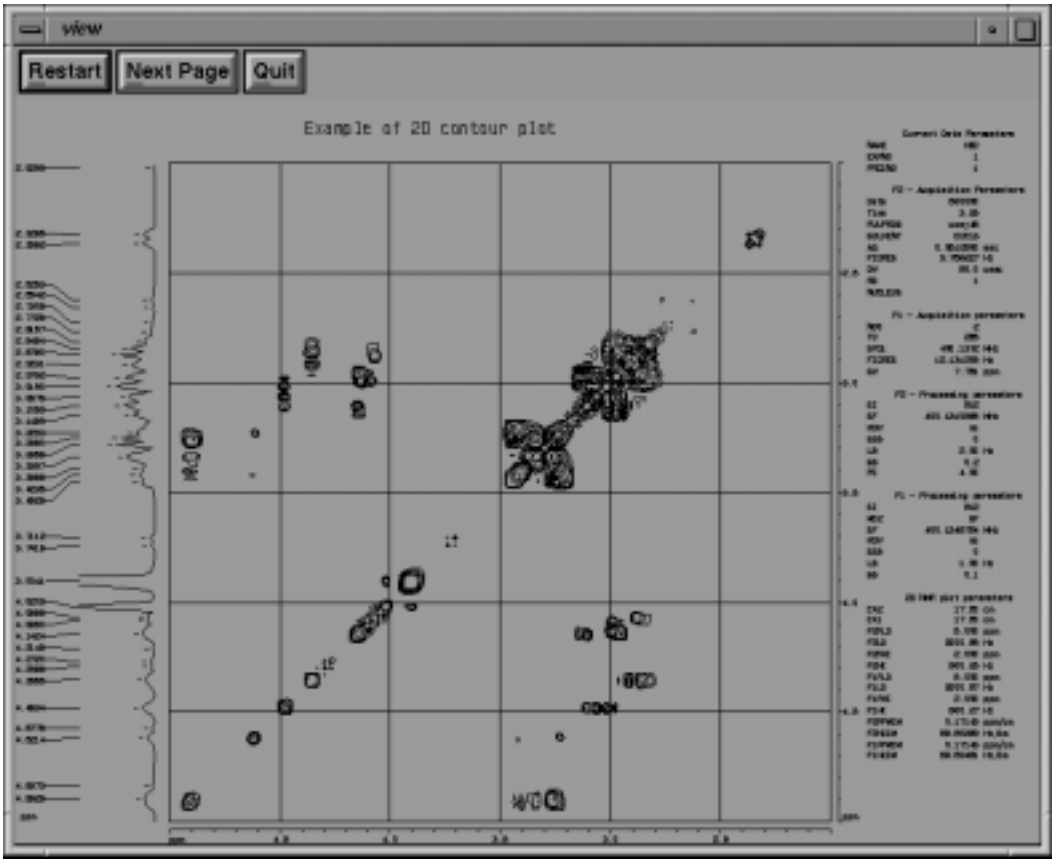


Figure 11.3 Example of a contour plot

- contours
- axes
- coordinate grid
- 1D spectra along both coordinates
- peak list of the 1D spectra .

It is divided into 7, non-overlapping windows, for

1. the contours
2. the F1 axis (vertical)
3. the F2 axis (horizontal)
4. a 1D spectrum along F1
5. peak labels for that spectrum
6. a 1D spectrum along F2
7. peak labels for that spectrum.

The sizes of these windows are determined by the following parameters:

CX1/CX2	size of contour window along F1 and F2 in cm
X1HEI/X2HEI	width (height) of F1 (F2) axis window in cm
PF1HEI/PF2HEI	width (height) of 1D spectrum window along F1 (F2)
PL1HEI/PL2HEI	width (height) of 1D peak list windows

In order to enable or disable a plot object, select *yes* or *no* for the follwoing parameters displayed on the first page of the edg window:

CONTPLO	contours
CONTADD	contours of a second data set
TITLE	title
X1AXIS, X2AXIS	axes
PARAM	parameter window
PROJ1, PROJ2	1D spectra or projections along F1, F2
FRAME	frame around contour window and around integration regions
PLABEL1, PLABEL2	peak labeling of 1D spectra or projections along F1, F2

If a plot object is enabled, you may open a new page within edg to edit the object's parameters by clicking on the corresponding edit button:

EDCONTP	contour parameters
EDCONTA	parameters for additional contour plot
EDAXIS	axis parameters
EDPROJ1, EDPROJ2	spectrum or projections along F1, F2
EDTITLE	title parameters
EDPARAM	parameter window definitions
EDFRAME	parameters for frame and integration regions

11.3.3 Axes

The axes along F1 and F2 are plotted inside the axes windows at a distance X1ZERO and X2ZERO (cm) from the contour window. Scale tics, units, color and labeling are specified through the parameters

X1TICL, X1UNIT, X1TICD, X1CHAR, X1COLOR
and
X2TICL, X2UNIT, X2TICD, X2CHAR, X2COLOR.

Their meanings are analogous to those of the corresponding parameters for the x axis of a 1D spectrum plot.

For contour plots it is sometimes useful to overlay the plot with a grid of vertical or horizontal (or both) lines. A set of horizontal lines is obtained if the parameter X1GRID is set to *yes* (instead of *no*). They correspond to the labeled scale tics on the F1 axis. Likewise, a set of vertical lines is obtained with X2GRID=*yes*. If an axis has been suppressed (see parameters X1AXIS and X2AXIS below), the corresponding set of lines will not be plotted either. The color of a set of lines always equals that of the corresponding axis.

11.3.4 Contours

In order to plot the contours of a 2D spectrum, the program must perform a contour search which may require significant time particularly for large spectra. For this reason XWIN-NMR offers the possibility for a faster generation of an overview plot of spectra whose size in one or the other dimension exceed 512 points: they can be plotted with reduced resolution of 512 points by utilizing the file dsp (of format 512*512 points) in the current data directory that is created for displaying the spectrum, rather than the actual *2rr* spectrum file. The operator selects this feature by setting the parameter REDRES (reduced resolution) to *yes* (the default value is *no*).

The contours of a 2D spectrum are plotted in the contour window of height CX1 and width CX2, as has already been mentioned. The spectral region is determined by the parameters F1(2)LO and F1(2)HI, and F1P(2)LO and F1(2)PHI. They may also be set interactively on the display by defining the desired region from the *Output->Define region* submenu.

F1LO (Hz), F1PLO (ppm)
Left (low field) limit of spectral region in F1. Changing this parameter causes the

right plot limit F1HI (F1PHI) to be adjusted accordingly, so that the plot resolution (Hz/cm) and the height of the contour window CX1 remain unchanged (the plot region is shifted along F1).

F1HI (Hz), F1PHI (ppm)
Right (high field) limit of the spectral region to be plotted. Changing this parameter causes the height of the contour window CX1 and the left plot limit F1LO (F1PLO) to be kept constant, while the scaling F1HZCM, F1PPMCM (Hz or ppm per cm) is adjusted accordingly (plot region is expanded along F1).

F1HZCM (Hz/cm), F1PPMCM (ppm/cm)
These parameters specify the plot resolution along the F1 direction. Any change causes the height of the contour window CX1 to be adjusted while keeping the plot region F1LO/F1HI constant.

An analogous set of parameters exists for the F2 direction:
F2LO, F2PLO
F2HI, F2PHI, and
F2HZCM, F2PPMCM.

Before plotting, the intensity levels on which to search for the contours must be defined. This is done either interactively on the display, or with the edlev command. Automatic calculation of the levels can be performed by the command levcalc that generates 6 levels. It is mainly used in AU programs.

The contours are either plotted in different colors (when the parameter CPLIN is set to color) or with different line styles (if CPLIN=line types). In the latter case the parameter CPLCOL determines the color common to all contours. If contours of negative and positive intensity are to be plotted, the parameter CPDIFCL should be set to *yes* (instead of *no*). Then the colors (line styles) of the positive contours are taken from the parameter CPCOLPA (CPLTYPP) which may hold up to 10 colors (line styles). The first color (line style) is assigned to the lowest level. The colors or line styles are repeated in a cycle if more levels than colors (line styles) must be plotted. Note that the assignment of colors to plotter pen positions must have been done correctly with edpp for pen plotters.

The parameter CPCOLNA (CPLTYPN) is responsible for the plotting of negative intensities, just as CPCOLPA (CPLTYPP) is for positive levels.

If CPDIFCL is set to *no*, but negative levels are to be plotted nonetheless, the colors (line styles) contained in CPCOLPA (CPLTYPP) are transferred to both

positive and negative contours, where the first color (line style) is assigned to the lowest negative level. CPCOLNA (CPLTYPN) will be ignored.

If line styles are to be used instead of colors, they are entered into the parameters CPLTYPP and CPLTYPN as *A.B*, where *A* is the code for the line style and *B* is the length of a dash or pattern in decimeters (1 dm = 10 cm). *A* may be one of the following:
0 = continuous line (*B* has no effect)
1 = dotted line (*B* is the distance between dots)
2 = short dash (*B* is the distance between the starting point of one dash and the starting point of the next).

3 = long dash (see above)
4 = dot + dash (B is the distance between dots)
5 = long dash + short dash (B is the distance between the starting points of the long dashes)
6 = long dash + short dash + short dash (B is the length of one pattern repetition)

Example: The value 2.05 (i.e. A=2, B=0.05) causes a short-dashed line with a repetition length of 0.05 dm = 0.5 cm to be drawn.

The contour window will be framed if FRAME = *yes* (instead of *no*). The color of the frame is determined by the parameter FRCOLOR.

11.3.5 Simultaneous plot of a second spectrum

Inside the contour window, the contours of two 2D data sets may be plotted side by side (CONTADD=*yes*): the current data set in the upper left triangle and a second data set in the lower right triangle. The second data set is specified by the parameters:

CADDU	disk partition
CADUSER	user (owner)
CADNAME	name
CADEXP	experiment number EXPNO
CADPROC	processed data number PROCNO

As for the current data set, the intensity levels for the second data set may be set interactively, or with edlev. The corresponding colors or line styles are specified with the parameters
CALIN, CADIFCL, CALCOL, CALTYPP, CALTYPN, CACOLPA and

CACOLNA,
which have the same meanings as the previously described parameters
CPLIN, CPDIFCL, CPLCOL, CPLTYPP, CPLTYPN, CPCOLPA and CPCOLNA.

The additional spectrum may also be plotted with reduced resolution by setting the parameter REDARES to *yes* (default value *no*).

11.3.6 Plotting integration regions

As described in the chapter *2D Data Processing Tutorial*, 2D spectra are integrated by interactively defining the integration regions and storing them in a file. The integrals are then calculated with the processing command int2d. For their identification in the spectrum they may be plotted as rectangles (labeled with the corresponding number) either in the contour plot or on a separate sheet (e.g. transparency). For this the parameter FRINT must be set to *yes* instead of *no*. The color is the same as that of the contour window frame (FRCOLOR), and the label of the integration region is plotted in size FRCHAR mm. The characters may be tilted by an angle (with the vertical), the tangent of which is given by FRSLANT. These parameters are accessible through EDFRAME.

11.3.7 Plotting 1D spectra or projections along F1 and F2

Along the dimensions F1 and F2 of a contour plot, 1D spectra or projections of the 2D spectrum (parameters PROJ1 and PROJ2=*yes*) may be plotted. Click on EDPROJ1/2 to edit the projection parameters. Select the type of projection from PF1EXT and PF2EXT. The complete definition of the 1D data set must be given with the parameters
PF1DU, PF1USER, PF1NAME, PF1EXP, PF1PROC
PF2DU, PF2USER, PF2NAME, PF2EXP, PF2PROC,
i.e. disk partition, user, name, experiment number and processed data number.

The distance of the spectrum's zero line from the contour window is PF1ZERO (PF2ZERO) cm and the color is given by PF1COL (PF2COL).

The same spectral region will be plotted (in the corresponding direction) as was defined for the 2D spectrum. In order for the signal positions in contour plot and 1D spectra to agree with the axis labels, the 1D spectra must be calibrated to the same reference signal as the 2D spectrum.

The vertical scaling is controlled by the parameter PF1CY (PF2CY for horizontal

scaling), which must be greater than zero, and ensures that a reference intensity is plotted in the given size in cm. This reference is chosen through the processing parameter PSCAL of the corresponding 1D spectrum (y scaling of 1D spectra) or, for projections, of the corresponding dimension. PF2CY=1 causes both projections to be scaled equally.

11.3.8 Peak list

Plotting the peaks of 1D spectra plotted along the F1 and F2 directions of a contour plot is analogous to that of 1D plots and is controlled by the parameters
PL1HEI, PL1CHAR, PL1DIG, PL1MRK, PL1COL, PL1UNIT, PL1MUL
PL2HEI, PL2CHAR, PL2DIG, PL2MRK, PL2COL, PL2UNIT, PL2MUL
for the F1 and F2 directions, respectively. They correspond to the parameters
PLHEI, PLCHAR, PLDIGIT, PLMRK, PLCOLOR, PLUNIT and PLMUL
as described in the chapter *The 1D Output Menu*.

11.3.9 Rotation and reflection of a 2D plot

These geometric operations are controlled by the parameters WROT and WMIR, as described in the chapter for 1D plots.

11.3.10 Automatic plotting on several sheets of paper

If the parameter CLIP is set to *no*, and CX1 or CX2 have values greater than the paper size, the plot is continued on additional sheets in both directions. This way, even on small plotters, a 2D plot may be output in any desired resolution and size.

11.3.11 Plotters with endless paper

Please refer to the chapter *The 1D Output Menu*.

11.4 Page layout stacked plot [edgw]

The command edgw is used to define the layout of *stacked plots*. Figure 11.4 shows an example, consisting of a sequence of 1D spectra. They are extracted from the *2rr* file (real part) of a 2D type data set. Remember that a such a file is created by one of the transform commands xf2, xf1, xfb, xtrf. For this reason, stacked plots may be obtained from 2D spectra, but also from a series of 1D spec-

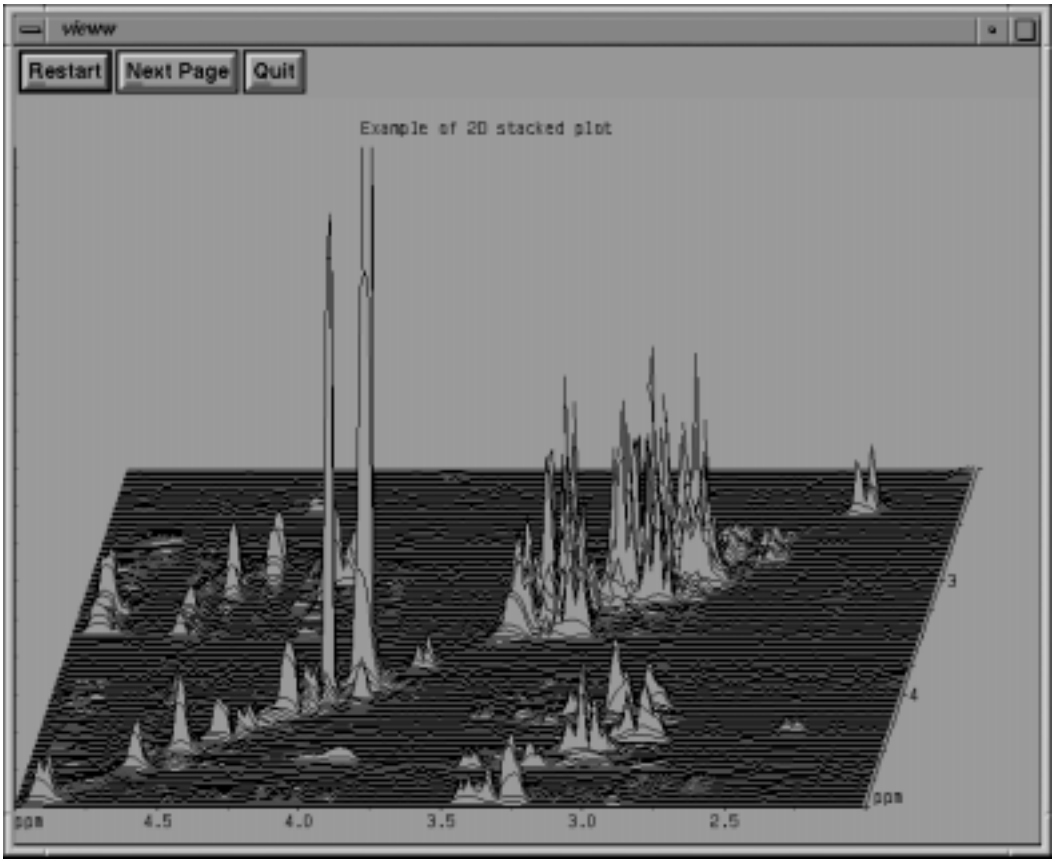


Figure 11.4 Example of a stacked whitewash plot

tra whose fids were acquired into a *ser* file by means of a suitable pulse program, such as a T1 or T2 experiment. They are normally transformed by the xf2 command.

The 1D spectra in a stacked plot are governed by the following 1D plot parameters:
SXLLEFT, SYLLEFT, SHEI, CX, F1/F1P, F2/F2P, HZCM/PPCM,
DHEI=SHEI-XAXIS*XHEI,SZERO, CY>=0, SCOLOR, XAXIS, XHEI,
XTICLEN, XUNIT, XTICDIS, XCHAR, XCOLOR, YAXIS, YTICLEN,

YTICDIS, YCOLOR.
Please refer to the chapter *The 1D Output Menu* for their description. 1D parameters not mentioned in this list do not play a role for stacked plots.

The following parameters define the stacked layout.

STACKED=*yes/no*
Draw/do not draw the spectra of a stacked plot

SWID
Width/cm of whole window

DWID=SWID-ZAXIS*ZHEI
Shows how much space is left for the stacked data

STKZ1,2
Plot region limits in the stacked (z) dimension. The numbers to be specified here depend on the chosen z axis units (parameter ZUNIT). Assume the z axis units should correspond to the values of a VDLIST (variable delay list of a T1 experiment). Set the parameter ZLIST=VDLIST. If the z axis should be labelled with seconds, set ZUNIT to *sec*. The program transforms the units in the VDLIST automatically to the units specified in ZUNIT. Then set STKZ1 and STKZ2 to the VDLIST range to be plotted, e.g. 0.3 - 5 sec. STKZ1 is the spectrum drawn first at the bottom of the page, STKZ2 the last one. STKINC must be set to 0 for this application.

STKXZ
This is the x offset of two spectra on the page. It must be specified in cm per z unit according to the parameter ZUNIT.

STKYZ
This is the y offset of two spectra on the page. It must be specified in cm per z unit according to the parameter ZUNIT. It must have the same sign as STKZ2 minus STKZ1.

STKINC
This parameter defines the row increment in the z dimension, i.e. it selects which rows of the *2rr* file are to be plotted, starting at the first row. If the value is 0, individual rows may be specified in the array parameter ZARR if ZLIST=0. If 0 and ZLIST > 0, an application of the type above (STKZ1,2) is to be plotted. If STKINC is -1, the row index is given by ZUNIT, and ZAXIS must be disabled.

ZARR
IF STKINC=0 and ZLIST=0, you may specify the list of rows to be plotted. In all other cases, ZARR is calculated from the other parameters.

ZLIST
Select list type: ZARR, VTLIST, VDLIST, VPLIST, VCLIST. If ZARR is selected, the row array ZARR must be set manually.

STKSCAL
This parameter defines the y scaling (spectrum height).
global=the global maximum of the spectrum will be plotted with height CY
preg=the maximum of the plot region will be plotted with height CY
pscal=check the processing parameter PSCAL: if PSCAL=*preg*, perform accordingly, otherwise assume *global*.

WHIWA
Enable/disable whitewashing (whitewash=do not plot those parts of a spectrum row hidden behind previously plotted rows).

ZAXIS, ZHEI, ZTICLEN, ZUNIT, ZTICDIS, ZCHAR, ZCOLOR
These parameters define how the z axis is plotted (they are analogous to the x axis parameters described earlier). Please remember that ZAXIS must be disabled (=*no*) if STKINC=-1. If STKINC>0 and ZUNIT=sec, the numbers along the axis are calculated from the acquisition parameters. This allows one plot reasonable z units if the second dimension of a *2rr* file is e.g. obtained from an LC run.

11.5 Use prepared page layout [rpar]

Please refer to the chapter *The 1D Output Menu*.

11.6 Plot commands

11.6.1 Plot [plot]

The command plot outputs the contour plot layout set up with edg or rpar on the current plotter CURPLOT, to be defined with edo. CURPLOT does not play a role if the *Plotter* system variable is set, which has priority over CURPLOT (use the command setres of the chapter *The Display Menu*). As soon as the layout has been

plotted, the next paper sheet is fed into the plotter.

11.6.2 Plot to queue [plots]

plots (*plot suspend*) outputs the contour layout into a queue rather than sending it to the plotter. Please refer to the chapter *The 1D Output Menu* for more details about plots.

11.6.3 Output queue on separate sheet [flplot]

flplot outputs the contents of a queue filled by one or more plots commands, to the current plotter.

11.6.4 Remove all plots from queue [rmplot]

All plots issued with the plots command are deleted, if they were not sent yet to the plotter with flplot.

11.6.5 Stacked plot [plotw]

The command plotw outputs the stacked plot layout set up with edgw or rpar on the current plotter CURPLOT defined with edo. CURPLOT does not play a role if the *Plotter* system variable is set, which has priority over CURPLOT (use the command setres of the chapter *The Display Menu*). As soon as the layout has been plotted, the next paper sheet is fed into the plotter.

11.7 Other commands

The remaining commands in the *2D Output Menu* are identical to the corresponding commands in the *1D Output Menu*. Please refer to that chapter.

Chapter 12

The *Display* Menu

Figure 12.1 shows the *Display* menu. The following sections describe the various

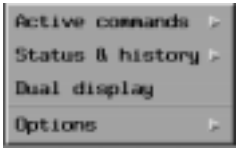


Figure 12.1 The *Display* menu

menu entries.

12.1 Active Commands

In the *Active Commands* submenu the following commands are available:

Show
Show and allow for killing
Show and update table online

The same commands may be invoked from the keyboard by typing show cmd, kill,

or follow cmd, respectively.

XWIN-NMR is a multi-tasking program, i.e. several commands may be in progress at the same time. For example, during automation with a sample changer acquisition is performed on a data set 3. While acquisition is in progress the previously measured data set 2 is processed (i.e. transformed, phase corrected, etc.), and at the same time the previously processed data set 1 is plotted. In addition a user is setting up the experiments for the next samples.

In general multi-tasking is handled in the following way. When you enter commands either from the keyboard or from a menu, they are queued and applied to the current data subsequently. For example, typing em (Return), ft (Return), apk (Return) will execute these commands in the specified sequence on the current (1D) data set. However, when you apply a command to the current data set, then change to a different data set (which will become your new current data set) and type in a command to process it, the command will be executed simultaneously with the first command, provided the latter has not terminated in the meantime. For example, select a 2D data set and type xfb to start a 2D Fourier transform. Then, while this tranform is executing, select another 2D data set and type xfb again. This will cause the second transform to be executed in parallel with the first one. The 2D processing module is started twice by the XWIN-NMR command processor as two Unix processes, each copy of the module working on a different data set.

The purpose of the Show command is to display a table of all commands found in execution or scheduled for execution. Figure 12.2 presents the table as it would

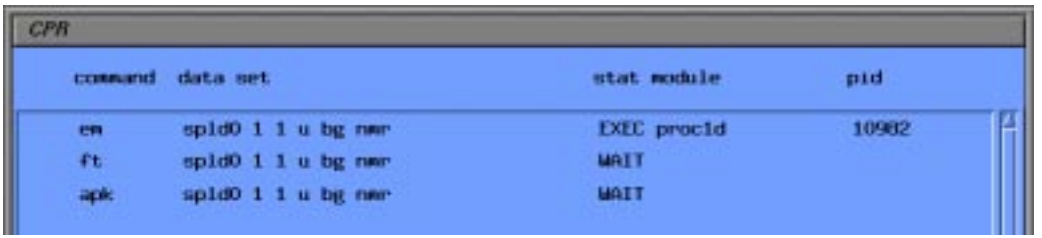


Figure 12.2 Table of active commands

look after entering the command sequence em, ft, apk. It displays the command

names and the data sets on which the commands operate. Furthermore, the execution status EXEC indicates which commands are already in progress. The status WAIT indicates that a command must wait until a previous command has finished. In our example, the three commands were typed while the same data set was on screen. Therefore they are queued, i.e. ft must wait until em terminates, and apk must wait until ft terminates. There are two more columns in the table giving the name of the XWIN-NMR module containing the program code for the command, and the Unix process id number assigned to the module after it was started. Figure 12.3 shows an example with two simultaneously executing 2D transforms.

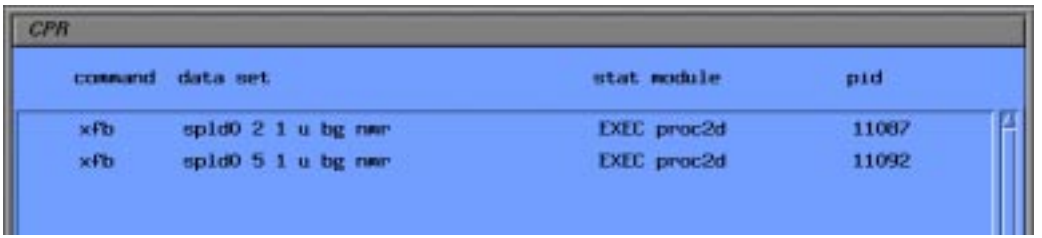


Figure 12.3 Simultaneously active commands

The Show and allow for killing command displays the same type of tables (equivalent to the kill keyboard command). The difference from Show is that you may now click on a command, forcing it to terminate immediately. The data set on which the command was operating will be left in an incomplete state. You should only kill data processing commands or automation programs this way. To stop data acquisition, the special commands halt or stop should be used. Particularly halt will abort acquisition not before the current scan is complete, ensuring correct fid data be stored on disk. If acquisition hangs for some reason and does not respond to stop or halt, you may try to clear it with kill.

The Show and update table online displays the same table like Show, but updates it according to the progress of the commands until you close the table. In our example (sequence em, ft, apk), XWIN-NMR will delete the entry for em from the table as soon as em is finished, and change the status of ft from WAIT to EXEC. When all commands have terminated, the table will be empty.

12.2 Status & History

In the *Status & History* submenu the following commands are available:

On for all data sets (default setting)
On for current data set only
On for selected data sets
Off
Enable 'history' file
Disable 'history' file (default setting)

The same commands may be invoked by typing the following commands on the keyboard:

status all (default setting)
status auto
status cmd
status off
hist on
hist off (default setting)

The first four commands control the way messages are displayed in the bottom status line of the XWIN-NMR window. Every XWIN-NMR processing command displays a message when it is started or terminated. Many commands with longer execution times report special information while they are running. For example, the 2D transform command monitors the rows or columns being processed. On for all data sets is the default setting. XWIN-NMR tries to output all messages of all currently active commands, even those executing in background on data sets other than the current data set visible on screen. In this case not all messages may be readable, because they are sent asynchronously from various modules to the status line and may overlap. On for current data set only disables the messages sent from background commands, and displays only those generated by commands operating on the currently visible data set. On for selected data sets displays a table of active commands (if any) and the associated data sets. You may select one, and XWIN-NMR will disable the status messages for all other ones. Off disables the status line completely, and you will never see any messages appear.

XWIN-NMR is capable of monitoring all acquisition, processing and plotting commands together with their start and termination times and possible error messages in a text file

/u/prog/<XWIN-NMR version>/curdir/<user>/history

By default history is disabled. When enabling it with Enable 'history' file please be aware that the file grows with time (in fact each time you execute a command some lines are appended). It is your responsibility to delete it from time to time.

12.3 Dual Display

The command Dual display (equivalent to the keyboard command dual) switches XWIN-NMR into a special mode where two spectra are displayed above each other for comparison. You may scale the spectra individually, shift them horizontally against each other, and display and store on disk their difference or sum. Figure 12.4 shows the window layout of dual display mode.

The first spectrum displayed in dual mode is the spectrum on screen when invoking the command Dual display. The second spectrum must be defined as the so-called *second data set* by entering the command edc2. A dialog window will be opened where you may enter the name and other characterisitics of the second spectrum. If no second data set is defined, or the data set specified in edc2 does not exist, Dual display will print a message and allow you to reenter edc2 in order to specify a valid data set.

All scaling and shifting commands located below the label *dual* in the left hand button panel apply to both spectra on screen. The buttons underneath the label *spectrum1* allow you to shift spectrum 1 exactly to the bottom or to the center of the screen, or to any other position by moving the mouse. The buttons below the label *spectrum2* provide the same functions for spectrum 2. In addition, you may change its vertical scaling and its horizontal position relative to spectrum 1. Both values are monitored in the *Info* window opened when entering dual mode. The buttons *sum* and *diff* cause the sum or difference of the two spectra to be displayed. While you are in one of these mode, you can save the sum or difference on disk. Select the command *Save & return* from the *File* menu, or from the pop up window which appears when you click the *return* command button. The result will be stored under the data set specifications of the so-called *third data set* which may, like the *second data set*, be defined in the edc2 dialog window. It is automatically called if the third data set already exists, and you decide not to overwrite it with the sum or difference. The *undo* command button terminates sum or difference mode, and returns to dual display.

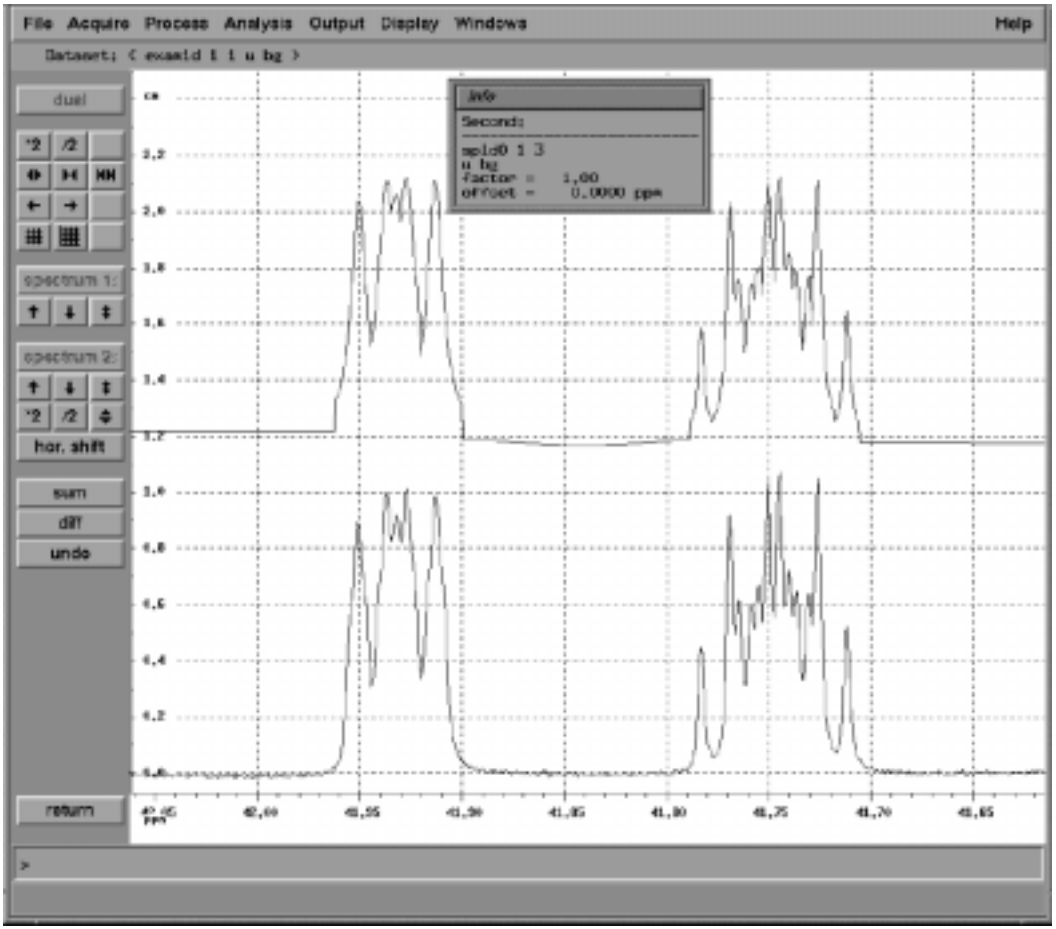


Figure 12.4 Dual display of an acquired (lower trace) and a simulated spectrum

When you enter dual display mode, the two data traces appearing on the screen correspond to the real parts (files *1r*) of the spectra. In order to switch to the corresponding imaginary parts (files *1i*), open the *Display* menu and click on the commands Imaginary spectrum 1 or Imaginary spectrum 2. Likewise, you may put the fids on screen by invoking the commands Fid 1 or Fid 2 from the *Display* menu.

The *Display* menu offers additional advanced features which help you to quickly

toggle through a number of spectra while you are in dual display mode. Assume the spectrum specifications only differ either in their EXPNO or in their PROCNO. Then you can load the spectrum corresponding to the next or previous EXPNO or PROCNO by just clicking on one of the commands

Incr. EXPNO of 2nd
Decr. EXPNO of 2nd
Incr. PROCNO of 2nd
Decr. PROCNO of 2nd

If you have another series of data differing from the previous series not only in EXPNO or PROCNO, but also in its name, user and disk location, you may define the data as *third data set* with edc2. Then you may rapidly switch between the second and third data set in dual display mode by invoking the commands 2nd data set and 3rd data set from the *Display* menu. To toggle through the spectrum series of the 3rd data set you may apply the commands

Incr. EXPNO of 3rd
Decr. EXPNO of 3rd
Incr. PROCNO of 3rd
Decr. PROCNO of 3rd

In order to leave dual display mode, select the command Return from the *File* menu, or from the menu which pops up when the return button is cliccked.

12.4 Options

The *Options* submenu contains a number of commands influencing the data display on screen. If the current data set is a 1D data set, XWIN-NMR displays the file *1r* of the processed data directory of the current data set. This file contains the real spectrum if a Fourier transform was already executed. If *1r* does not exist, the file *fid* from the acquisition data directory is displayed. If neither *1r* or *fid* exist, the current data set does not contain any data, but only parameter files. This is the situation if either the data set is a new one, created by the New command from the *File* menu, or if all data files were deleted using one of the commands dela or delp from the *File* menu. Let's assume both files *1r* and *fid* are present. Then the command Fid will switch the screen to show the fid, and the command Imaginary spectrum will change the display to the imaginary part of the transformed spectrum stored in the file *1i*. Clicking on Real spectrum will bring you back to the normal spectrum

display.

The command Separ. even/odd points changes the display such that the left half of the screen shows the data points with even numbers 0,2,4,... , and the right half the odd data points 1,3,5,... . This mode can be useful if an fid is on the screen, where the even points have been acquired by the first detector channel, the odd points by the second one in quadrature mode. The command Undo separation cancels the channel separation.

By default, XWIN-NMR connects the data points of a spectrum with a straight line in order to generate a continuous curve on the screen. Somtimes it is convenient to observe the data points themselves. Clicking on the command Dots/solid lines will turn off vector drawing between the data points, leaving only the data points. Re-invoke the same command to toggle back to the solid line display.

The toggle command Y axis on/off allows you to enable or disable the display of a y axis. If enabled, the command Y axis, absolute/cm units lets you toggle between two types of axis units. Absolute units show the intensities on a relative scale. They allow you to compare the intensities of different spectra. For example, if you would (under the same experimental conditions) acquire two fids, the first one with 8 scans and the second one with 16 scans, the second signal will be twice the intensity of the first one. The absolute units will reflect this fact. In a second example, when you multiply the whole spectrum with 0.5 (type dc 0.5, then mulc) you will see the intensity change by looking at the y axis in absolute mode. Setting the y axis to *cm* units allows you to the view the spectrum in the same way it would occur on a plot. The so-called *plot reference peak* (often the biggest peak in the spectrum) will be plotted with a height of CY cm. The plot parameter CY can be set from the plot parameter editor edg, or by entering cy on the keyboard. If cm scaling of the y axis is selected, the axis is labelled such that the intensity of this peak will get the value of CY.

12.5 User Interface

This command (which may also be invoked by typing in setres) lets you tailor a number of user interface attributes. A window according to Figure 12.5 is opened, which allows you to adjust the various settings. After having changed one or several user interface items by clicking on the corresponding radiobutton, you must click on the *Apply* button in order to make the modifications effective. If you want to close the window, click on the *File* entry of its top bar and choose Exit. If you

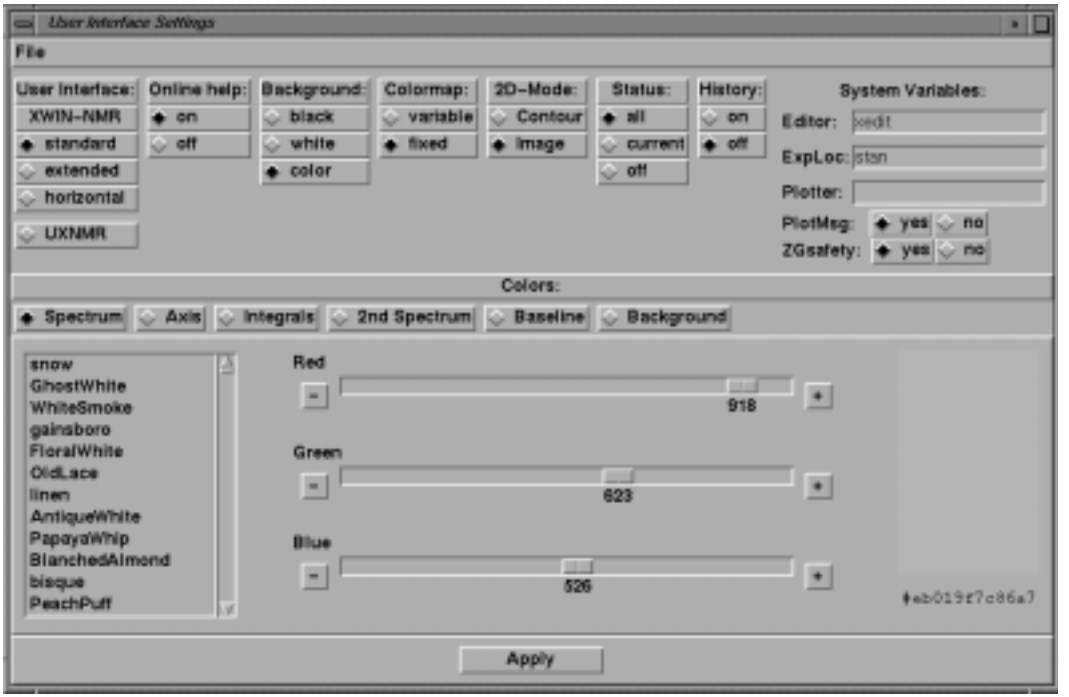


Figure 12.5 User interface settings

want to restore the default settings, choose default settings from the *File* entry, and then click on the *Apply* button. All settings are stored on disk in the file *.xwinnmr/resources* in the user's home directory.

The available user interface items are discussed below.

User interface
XWIN-NMR *standard*, *extended*, *horizontal*, or UXNMR. The *standard* layout displays the quick access buttons at the left side in vertical order. 1D and 2D modes are very similar, with corrsponding buttons in nearly the same positions. The *extended* mode displays additional quick access buttons. These functions are also available in standard mode, but you must must open the *Display->Options* menu or the *Output* menu to have access to them. The *horizontal* mode displays all quick access buttons in horizontal order, including all buttons of the *extended* mode. The

old style UXNMR interface is also provided. Please note that it will stay as it is, without further developments.

Online help
On or *off*. By default, XWIN-NMR displays help messages in the status line when you move the cursor over the command buttons. You may disable the messages by choosing *off*.

Background
Black, *white* or *color*. You may define whether the background of XWIN-NMR's data area (the area where the spectrum is drawn) is set to black, white, or to a color. If you decide for *color*, you must enable the radiobutton *Background* in the *colors* section of the window. Then you can adjust the desired color using the sliders, or select a color from the table with a double click.

Colormap
Variable or *fixed*. This is not important for Silicon Graphics and other machines which provide several color look up tables. Select *variable* if a second color intensive application is running parallel to XWIN-NMR. This ensures that, when moving the cursor into an application's window, its own color table is loaded and thus colors are displayed.

2D mode
Contour or *Image*. 2D data sets may be displayed in form of a contour map, a colored map (image) of intensities, or an oblique view. *Contour* and *image* are the most frequently used modes, and you may switch between them at any time by clicking on the respective command button. The purpose of the *2D mode* selection within the *User Interface Settings* window is to define in which mode a 2D spectrum should come up when you start a new XWIN-NMR session and select a 2D data set.

Status
Define how messages in XWIN-NMR's bottom status line are printed. The three radio buttons *all*, *current, off* are equivalent to the commands status all, status auto, status off , respectively, described earlier in this chapter. The advantage of the radio buttons versus the command is that the buttons allow you to view the current setting.

hist on
hist off (default setting)

History
If enabled, all XWIN-NMR given by the user are written in the file

/u/prog/<XWIN-NMR version>/curdir/<user>/history.

The radio buttons *on* and *off* are equivalent to the commands hist on and hist off, respectively, described earlier in this chapter.

Editor
Certain XWIN-NMR commands call a text editor, e.g. for writing pulse or AU programs. The default editor is *xedit*, an editor of the X Windows system. You may change this entry and specify your preferred editor, e.g. SGI's *jot*. If you decide for an editor requiring an *xterm* window such as the traditional Unix *vi*, you must specify it in the form *xterm -e vi*. Please note that XWIN-NMR can only start an editor if it is located in a directory which is part of the shell's search path.

ExpLoc
Standard experiment files, such as pulse programs and many other files, are searched for by XWIN-NMR in the directory */u/exp/stan/nmr/*. If you change the ExpLoc field, e.g. to *stan2*, XWIN-NMR will search these files in */u/exp/stan2/nmr/.* This feature allows you to maintain several such directories.

Plotter
By default, this field is empty, and XWIN-NMR plot commands send their output to the device defined by the parameter CURPLOT, set by edo. If you enter the name of a plotter configured with cfpp into this field, this plotter will gain priority over what is specified in CURPLOT. Setting CURPLOT is no longer necessary. However, the flexibility to send plots of different data sets to different plotters (if connected) is lost.

PlotMsg
A plot command given in XWIN-NMR is passed on to the operating system's spooler where it gets assigned an id number. The number is printed in an information window. If this message disturbs you, select *no* instead of *yes*, and it will be suppressed.

ZGsafety
If set to *off*, the zg acqusition command overwrites existing acquisition data (files *fid* or *ser*) with the new measurement. This is potentially dangerous if you type zg inadvertently, because you may loose important data. If you select *on* instead of *off*, zg will ask you whether the data can be deleted if existing.

Spectrum, Axis, Integrals, 2nd Spectrum, Baseline, Background
Click on one of these items to define the graphical object whose color you want to adjust. Use the sliders to create your own color by mixing the Red, Green, and Blue components, or double-click on a pre-defined color in the table. Click on *Apply* to make the chosen color effective. *Integrals* denotes the integral traces in manual integration mode. *2nd spectrum* is spectrum 2 in dual display mode. *baseline* is the baseline trace which may be real-time adjusted in manual baseline mode. *Background* was already described above. Please note that these colors do not influence plot colors, which must be defined separately when setting up the plot layout.

Chapter 13

The *Windows* Menu

The *Windows* menu contains XWIN-NMR commands which generate new windows

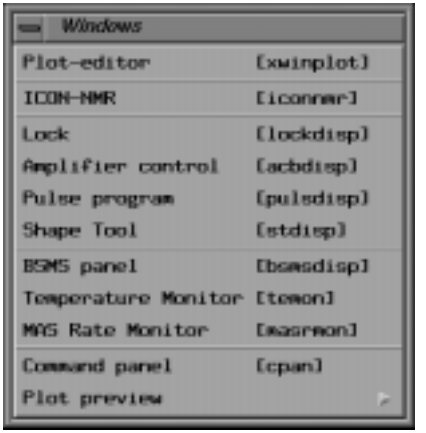


Figure 13.1 The *Windows* menu

independent of the main XWIN-NMR window. They may remain on screen simultaneously with other windows of this type, and with the main XWIN-NMR window. Since XWIN-NMR is a multi-tasking program, the commands are executed in paral-

lel. For example, if a data acquisition is in progress, and the lock and the amplifier control windows are open, the contents of all three windows are refreshed at the same time, and the user may at any time move the mouse into one of the windows and execute a command.

13.1 Command panel [cpan]

The command cpan (in the *Windows* menu) allows you to display a command button panel whose contents and layout may be tailored according to your requirements. The command panel consists of an arbitrary number of buttons that you may label with your own text. You must set up a *panel file* where you specify the button labels, the corresponding XWIN-NMR command, and a help text. The name of the panel file is *default*. Its must be located in the subdirectory

.xwinnmr-<hostname>/cmdpanels/

of your home directory. Whenever you invoke cpan, the command button panel is displayed according to the definitions in *default*. If you click on a button, the corresponding XWIN-NMR command is executed. The directory *.xwinnmr-<hostname>/cmdpanels/* may contain additional panel files describing different panel layouts. The button *Load* at the bottom of a panel will display a listbox with all these files. If you select one (with a single mouse click, followed by a click on the OK button, or by a double click), the current layout will be replaced by the new one. It is also possible to have more than one panel open at the same time by invoking cpan several times. The structure of a panel file is illustrated by the example: of Table 13.1, the corresponding panel is shown in Figure 13.2. Empty lines are ignored. The line after TITLE will be displayed as a panel title. BUTTON starts a button entry. NEWROW forces the button to be placed in the next row, SAMEROW in the same row as the previous button.
The next line is the XWIN-NMR command, followed by the desired button label and a help text to be displayed at the bottom line of the panel when the cursor moves over the button. The buttons defined for a row will be centered in this row during display. All buttons will get the same size, calculated from the longest button label string. The keyword END indicates the end of a panel file.

If you want to start up cpan with a panel other than *default*, proceed as follows:
Write an XWIN-NMR macro, e.g. by entering the command edmac cpan1. A text editor window will be opened. Insert the line

```
TITLE
Command Panel Example

BUTTON
NEWROW
edc
new
Define new data set

BUTTON
SAMEROW
eda
eda
Acquisition parameter setup

BUTTON
NEWROW
zg
zg
start acquisition

BUTTON
SAMEROW
lb 0.3
LB
set line broadening to 0.3Hz
END
```

Table 13.1 Panel file structure

xwish -f /u/exp/stan/nmr/tcl/cmdpanel <panel file name>.

Figure 13.2 Panel resulting from example panel file

Specify your desired panel file and terminate the text editor. You have created a new XWIN-NMR command cpan1 which will come up with the desired panel.

13.2 Plot preview

The plot preview commands send the plot output into a screen window (cf. the figures of 1D and 2D plots in the chapters *The 1D/2D Output Menu*). You may change plot parameters and observe the effect of the modification. The plot layout may thus be tailored until it is satisfactory.

The following preparations are necessary to enable the use of the preview commands:

- The plotter configuration command cfpp must have been executed. This task is normally accomplished immediately after XWIN-NMR installation.
- The current plotter CURPLOT must be defined (command edo). This is important because the preview commands display the output exactly as it would come out on this type of plotter, with the only difference that the absolute size is scaled down to the window size, thereby retaining the proportions.
- The plot layout must be set up using the commands edg, edgx, or edgw, depending on whether spectrum plots (1D or 2D), automatically expanded 1D plots, or stacked (whitewash) plots are desired, respectively.

This means, you set up a preview exactly the same way as a normal plot. Then,

instead of executing a plot command (plot, plotx, plotw) you start up the corresponding preview command view, viewx, or vieww. XWIN-NMR will open a window containing the desired plot. This will be a spectrum plot or a contour plot in the case of view, depending on the data set, an autoexpanded 1D plot in the case of viewx, or a stacked (whitewash) plot in the case of vieww. You may resize the preview window, and XWIN-NMR will redraw the plot according to the new window size. However, the plot will be restricted to an area within this window having the same size ratio as the paper of the selected plotter CURPLOT. Usually, setting up a plot layout is an iterative process. You must modify plot parameters several times before the layout appears satisfactory. It is not necessary to close the preview window if you want to change a parameter. The easiest way is to type in the new value on XWIN-NMR's command line, e.g. cy 20. Or you enter edg, edg, or edgw to change several parameters, and exit from the parameter editor to save the new values. In order to preview the effect of the changes, click on the *Restart* button located in the top bar of the preview window, and the plot will be redrawn. Particularly, in the case of 2D plots, you may have to wait a few seconds, until the contour search is finished. During this time, the cursor shape changes to a clock.

In order to terminate the preview window, click on its *Quit* button. A preview window always belongs to the data set from which the preview command was given. If you leave a preview window for a particular data set open and change the current data set, you may execute another preview command. The result is that two preview windows will be open at the same time, containing the plot of the two data sets. This procedure may be continued even with more data sets, although it might not be very useful.

The plot parameter CLIP (type clip on the keyboard or set it in edg) may take on the two values *yes* or *no*. If set to *yes*, which is the default, all parts of a plot not fitting on the paper will be clipped. If set to *no*, those parts will be drawn on additional pages. Assume, for example, the coordinates of the plot title are set in edg such that the title would lie outside the paper. If CLIP is set to *yes*, the plot would not contain the title text, and you may wonder why. During previewing a plot, we therefore recommend to set CLIP to *no*. An additonal plot page is generated containing the title. By clicking on the button *Next Page*, you may toggle through all additional pages generated, and you can check whether parts of the plot appear on extra pages due to improper setting of layout parameters. If the last page is reached, you must click on *Restart* to get the first page back.

A final preview command remains to be discussed, viewmg. It displays plots gen-

erated with the command plots. Remember that plots does not output a plot directly to the plotter, but stores it in a queue. This is useful as you might want to add more plots to the same sheet of paper using plots. The command flplot (flush queued plot) would output all queued plots to the printer or plotter, while viewmg draws everything in the preview window.

Chapter 14

The *Help* Menu

XWIN-NMR includes the complete software manual as online documentation. If

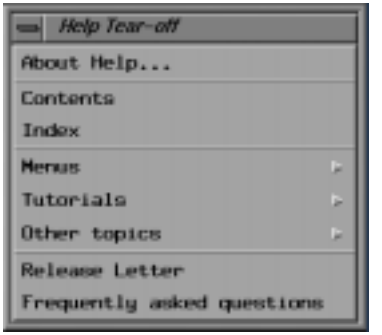


Figure 14.1 The *Help* menu

you select an item from the *Help* menu, the corresponding document file is displayed. Document files are stored in the directories

$XWINNMRHOME/prog/docu/english/xwinacqu/
$XWINNMRHOME/prog/docu/english/xwinproc/,

containing the chapters about data acquisition and processing/viewing, respectively. The file name extensions of the document files is *.pdf*. Their storage format is of type *Adobe Acrobat PDF*.

As soon as you click on a menu item of the *Help* menu, XWIN-NMR will display the corresponding document file using the *Adobe Acrobat Reader* software, or the public domain tool *xpf* of Derek B. Noonbourg. Please note that for AspectStation systems and IRIX versions < 5.3 only the latter program is available. The viewers are installed from the XWIN-NMR release media at installation time of XWIN-NMR.

The *Index* and *Contents* entries of the *Help* menu cause the display of a keyword index and a table of contents. Both are organized as *hypertext*. You may click on an index entry such as *ft command* (click on the page number!), or on a chapter or section number in the table of contents. The document file containing the selected item will be displayed, with the correct page placed on screen.

Apart from the XWIN-NMR manual, the *Help->Other Topics* submenu provides a number of other NMR related manuals.

You may enter the command help on the keyboard. In this case the contents of the XWIN-NMR manual will be opened, from where you can switch to a desired chapter via hypertext click.

Any document may be opened from within XWIN-NMR by entering the following command on the keyboard:

xhelp <document path name>.

The path name is absolute if it begins with a "/" character. Otherwise the specified path is appended to the initial path *XWINNMRHOME/prog/docu/english/*.

Chapter 15

File Formats

15.1 Parameter Files

The parameter files *acqu**, *proc**, and *meta** containing acquisition, processing, and plot parameters, respectively, are text files (see Figure 15.1). Their format corresponds to the so-called JCAMP-DX standard. It allows the inclusion of user specific parameters, which are marked by the character sequence *##$*. For this reason, all parameters XWIN-NMR parameters in the file are preceded by this sequence.

15.2 Acquisition Data

The files *fid* and *ser* contain one dimensional or multi-dimensional acquisition data, respectively. They consist of a sequence of 32 bit integer numbers in binary format. On workstations with different microprocessors an integer number may be represented by a sequence of 4 bytes ordered differently. There are two common modes, *little endian* and *big endian*. The latter one is typical for MIPS and SPARC microprocessors, the former one for Intel x86 and Bruker X32. XWIN-NMR stores the byte order corresponding to the acquisition data in the acquisition status parameter BYTORDA (in the file *acqus*), which may take on the value *little* or *big*. This allows XWIN-NMR (or other software packages) to convert the data to the correct byte order, if the endian mode of the computer where the data are to be proc-

```
##TITLE= Parameter file,  XWIN-NMR version 1.0
##JCAMPDX= 5.0
##DATATYPE=  Parameter Values
##ORIGIN=XWIN-NMR
##OWNER= jos
$$ Mon Jul 18 16:30:18 1994
$$ File: /u/data/guest/nmr/IQUIN128/1/acqu
##$AQ_mod= 1
##$D= (0..31)
3e-06 2 0.071 0.002 0.015 0.001 0.0005 1.5 10 0.7 0 0 0 0 0 0 0 0 0 0
##$DR= 12
##$LOCNUC= <2H>
##$LOCSHFT= no
##$NS= 8
##$NUC1= <1H>
##$PULPROG= <zg>
##$PW= 4.5
##$RO= 20
##$SW_h= 826.719576719577
##$TD= 131072
##$TE= 303
##END=
```

Figure 15.1 Section of a an *acqu* parameter file

essed is different from that of the acquisition computer.

Within an fid, the data points coming from channels A and B alternate for quadrature detection data. A 1D *fid* file contains a single fid with TD(F2) points, which is the time domain size TD stored in the acquisition status parameter file *acqus*. A *ser* file contains TD(F1) 1D fids, which is the parameter TD in the file *acqu2s*. Each 1D fid in a *ser* file start at a 1024 byte block boundary, even if its size is not a multiple of 1024 bytes (corresponding to 256 data points).

15.3 1D spectra

Applying a 1D processing command to a 1D fid or reading a 1D slice from a 2D spectrum generates *processed* data. They are stored in two files *1r* and *1i*, corre-

sponding to real and imaginary part. Like in 1D fids, the data points are stored as a sequence of 32 bit integers. Their byte ordering is given by the parameter BYTORDP, which may be read from the processing status parameter file *procs*.

15.4 JCAMP-DX format

1D acquisition and processed data (files *fid, 1r, 1i*) may also be stored in text files based on the JCAMP-DX format. The initial part of such a file is similar to Figure 15.1 and contains the parameters, followed by the data section with the fid or spectrum encoded according to the option chosen when calling the JCAMP conversion command tojdx. Fore detailed information about the JCAMP format, please refer to the following literature.

1. JCAMP-DX: A Standard Form for Exchange of Infrared Spectra in Computer in Readable Form.
Robert S. McDonald and Paul A. Wilks, JR.
Applied Spectroscopy 42, Number 1, 1988

2. Generic JCAMP-DX, Version 5.0 Draft 1.0 February 28, 1991
Robert S. McDonald, JCAMP-DX subcommitte, 9 Woodside Drive, Burnt Hills, NY 12027, USA

3. JCAMP-DX NMR Standard
Dr. A. N. Davies, ISAS, Institut für Spektrochemie, Dortmund, Germany (private comm.)

15.5 2D spectra

Like 1D spectra and acquisition data, 2D processed data are stored as 32 bit integers. The spectrum real part is contained in the file *2rr*, the imaginary parts in the files *2ii*, *2ri*, *2ir*. The latter two are only present if the spectrum is phase sensitive. All files are stored in the so-called submatrix format. The submatrix dimensions are given by the status parameters XDIM contained in the files *procs* and *proc2s*. XDIM is calculated by automatically (depending on the available computer memory) so as to optimize the fourier transform time. If the entire data set fits in memory, XDIM(F1) will be 1, and a row-wise ordering results.

On disk, a complete submatrix is stored before the next submatrix starts. The order of the data points within one submatrix is the same as the order of the submatrices

within one data set, first F2 (the acquisition direction), and then F1.

Figure 15.2 shows the file structure of a processed 2D data file with the

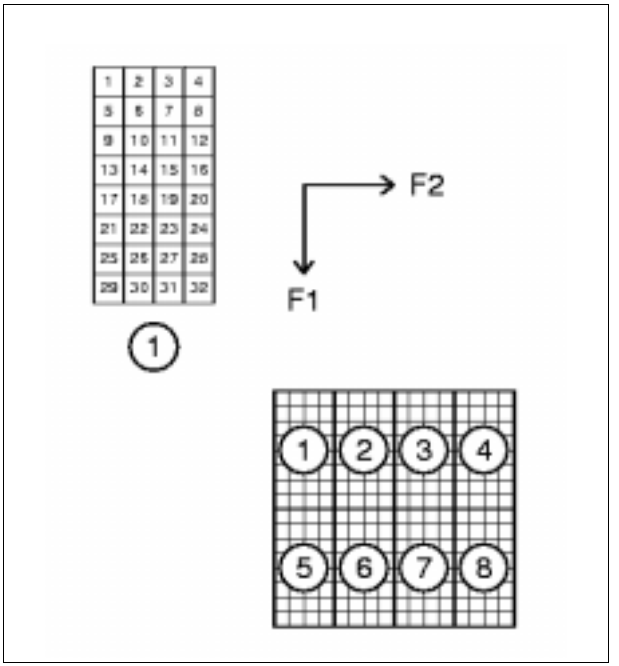


Figure 15.2 Submatrix format

parameters SI(F2)=16 points, SI(F1)=16, XDIM(F2)4, XDIM(F1)=8.The upper left part of the figure shows the sequence of the individual data points of submatrix 1, the lower right part shows the sequence of the submatrices in the entire spectrum.

15.6 3D Spectra

For 3D processed data, the spectrum real part is contained in the file *3rrr*, the imaginary parts are generated according to the description of the command tf1. The files *3rrr*, *3irr*, ... are stored in the so-called subcube format, a generalization of the 2D submatrix format. The subcube dimensions are given by the status parameters XDIM contained in the files *procs*, *proc2s* and *proc3s*. XDIM is calcu-

lated by automatically (depending on the available computer memory) so as to optimize the fourier transform time. XDIM is always a power of two.

On disk, a complete subcube is stored before the next subcube starts. The order of the data points within one subcube is the same as the order of the subcubes within one data set, first F3 (the acquisition direction), then F2, and finally F1.

Figure 15.3 shows the storage order of a processed 3D data file with the

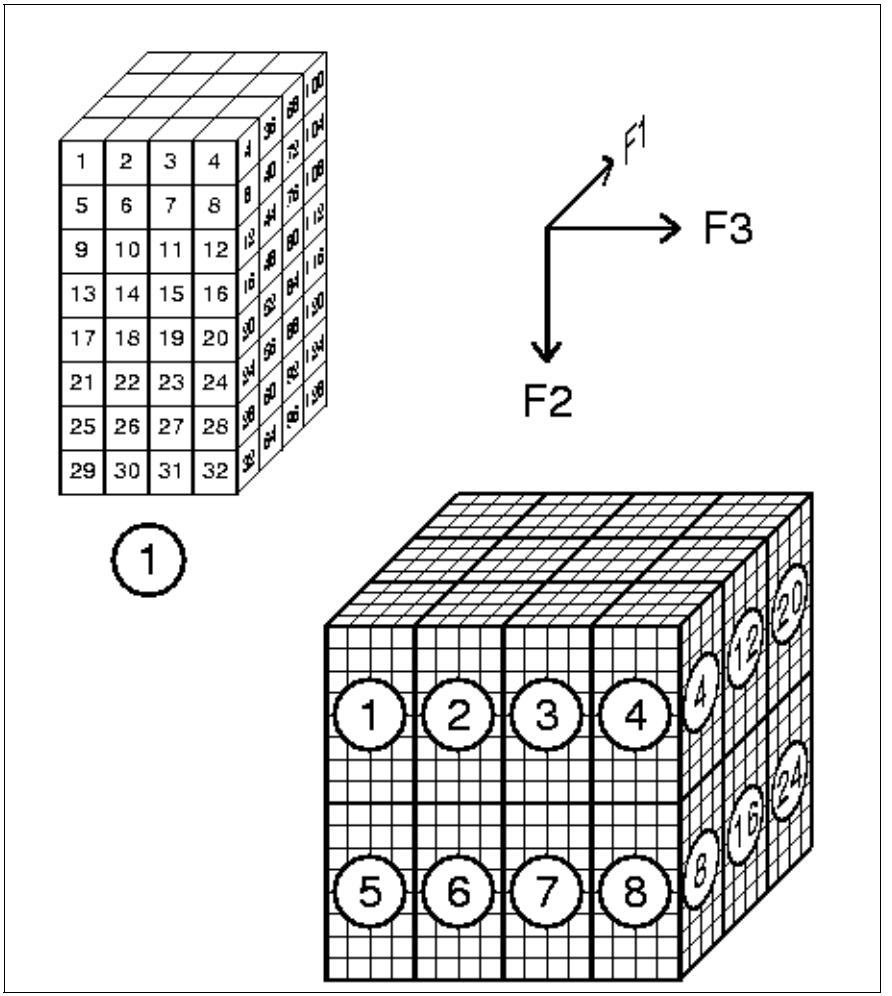


Figure 15.3 Subcube format

parameters SI(F3)=16 points, SI(F2)=16, SI(F1)=16, XDIM(F3)=4, XDIM(F2)=8, XDIM(F1)=4. The upper left part of the figure shows the sequence of the individual data points of subcube 1, the lower right part shows the sequence of the subcubes in the entire spectrum.

15.7 Pulse Shapes

15.7.1 XWIN-NMR 2.0 and later

Pulse shapes are stored in the directory *XWINNMRHOME/exp/stan/nmr/lists/wave/* in ASCII format conforming to JCAMP-DX. Table 15.1 shows a section of a shape file. The data points represent the amplitude and phase values.

15.7.2 XWIN-NMR versions earlier than 2.0

Pulse shapes are stored in the directory *XWINNMRHOME/exp/stan/nmr/lists/wave/*. There are two types of data formats, a binary one required by XWIN-NMR for the execution of shaped pulses, generated by the program *shape*, and a text format which can be read into *shape* by the command ar, and stored back in binary format by w for use with XWIN-NMR.

Text Format

The text format is particularly useful if you want to generate own shapes by a suitable program. Figure 15.4 shows an example. The file starts with the text RFVERSION_F, then the pulse shape follows. The first column contains the amplitudes, the second one the corresponding phases. If the phases are omitted, the following rule applies: the phases are set to 0 if the amplitudes are positive, and to 180 degrees if negative. Phases are specified in degrees in the range 0-360 (floating point numbers are legal). A file must not contain more than 32768 lines. Amplitudes and phases are separated by a space, tabulator, or comma.

Relative power values may range from 0-100. If a value in the file exceeds 100, it will be rescaled to 100 during execution. All other points will be rescaled accordingly. Values less than 100 will not be rescaled.

Binary Format

The binary file format of shape files is as follows:

```
##TITLE= /u//exp/stan/nmr/lists/wave/CosSinc
##JCAMP-DX= 5.00 $$ Bruker JCAMP library
##DATA TYPE= Shape Data
##ORIGIN= Bruker Analytik GmbH
##OWNER= <guest>
##DATE= 96/09/21
##TIME= 12:11:57
##MINX= 0.000000e+00
##MAXX= 9.956761e+01
##MINY= -9.552316e+01
##MAXY= 1.794392e+02
##$SHAPE_EXMODE= None
##$SHAPE_TOTROT= 0.000000e+00
##$SHAPE_BWFAC= 0.000000e+00
##$SHAPE_INTEGFAC= 2.052226e-03
##$SHAPE_MODE= 4
##NPOINTS= 256
##XYPOINTS= (XY..XY)
0.000000e+00, -1.869338e-01
7.815741e-01, 1.794392e+02
1.522001e+00, 1.790653e+02
2.671471e+00, 1.783176e+02
2.985925e+00, 1.779437e+02
3.074292e+00, 1.775699e+02
2.907826e+00, 1.771960e+02
2.468165e+00, 1.768221e+02
1.748423e+00, 1.764483e+02
1.689626e+01, -8.412024e+00
....................., .....................
##END=
```

Table 15.1 JCAMP-DX shape file format

- **Header**: consists of 256 32 bit float words.
  1. word 1 contains file type information:
     371242.0 (= GRADVERSION_F) or 18906.0 (= RFVERSION_F)
  2. word 2 contains # of amplitudes (=amps) stored

```
RFVERSION_F
10  0
20  90
30  180
40  270
50  360
```

Figure 15.4 old ASCII shape file format

3. word 3 contains # of phases (=phas) stored
4. word 4 to word 256 are unused

- **Data**: consists of (amps + phas) 32 bit float words.
  1. *RFVERSION*: pairs of phases and ampitudes: phases (in degrees) from 0.0 to 360.0 and amplitudes from 0.0 to 100.0 %. Phases are always stored as first value followed by the corresponding amplitude.
  2. *GRADVERSION*: only amplitudes are stored (values from 0.0 up to 100.0 %). Phases allowed are 0.0 and 180.0 degrees. If the phase is 180.0 degrees, the stored amplitude is negated. No phases are stored in the file.

The maximum size of amplitudes (for GRADVERSION) or pairs of amplitudes and phases (=RFVERSION) is 32K (or 32768 points/pairs).

15.8 Other files

Many other files play a role in XWIN-NMR, such as integral range files, peak lists etc. They are described in the chapter *The File Menu*, or in conjunction with the command generating them.