# FELIX 2002

## User Guide

MARCH 2002

*All updated documentation (**User Guide**, **Tutorials**, and FELIX **Command Language Reference**) for the latest release of FELIX is available at the Accelrys website documentation library:*

**http://www.accelrys.com/doc/life/index.html**


Accelrys Inc.
9685 Scranton Rd.
San Diego, CA
92121-3752

Tel: 858 799 5000
Fax: 858 799 5100

# Copyright (U.S. version)



### Restricted Rights Legend



### Trademark Acknowledgments



## Permission to Reprint, Acknowledgments, and References

Accelrys usually grants permission to republish or reprint material copyrighted by Accelrys, provided that requests are first received in writing and that the required copyright credit line is used. For information published in documentation, the format is “Reprinted with permission from *Document-name*, *Month Year*, Accelrys Inc., San Diego.” For example:

> Reprinted with permission from *FELIX 2002 User Guide*, March 2002, Accelrys Inc., San Diego.

Requests should be submitted to Accelrys Scientific Support, either through electronic mail to **support@accelrys.com** or in writing to:

Accelrys Scientific Support and Customer Service
9685 Scranton Road
San Diego, CA 92121-3752

To print photographs or files of computational results (figures and / or data) obtained using Accelrys software, acknowledge the source in a format similar to this:

> Computational results obtained using software programs from Accelrys Inc.—dynamics calculations were done with the Discover® program, using the CFF91 forcefield, ab initio calculations were done with the DMol program, and graphical displays were printed out from the Cerius$^2$ molecular modeling system.

To reference a Accelrys publication in another publication, no author should be specified and Accelrys *Inc.* should be considered the publisher. For example:

> *FELIX 2002 User Guide*, March 2002. San Diego: Accelrys Inc., 2000.

# How To Use This Guide

The Felix 2002 *User Guide* is a basic guide to working with the FELIX software.

There are two ways to access the online version of this book:

- at Accelrys's website:

  http://www.accelrys.com/doc/life/index.html

- FELIX installation CD

Any updates or corrections will be posted to the web site, making the information at www.accelrys.com the most current.

> **Note:** If you are prompted for a username and password at the Accelrys documentation website, use the following:
>
> Username: **science**
>
> Password: **faster**

An index and table of contents are provided You can also use the Accelrys Site Search at:

http://www.accelrys.com/search.html

Select **All Documentation** in the **Search Area** list. Enter term(s) in the **Search** field, then click **Search**.

While viewing the document online, you can use the searching capabilities of your browser to locate information.

You can print individual chapters of these documents via your browser's printing capability.

For information about the FELIX command language or to access the tutorials, please see the online *FCL Command Language Reference* and *FELIX Tutorials* books.

## Using this book

The general information contained in the main body of this manual include the information described in “Topics covered” below.

The FELIX 2002 *User Guide* also contains several appendices. These provide additional information that might make installing and running FELIX easier.

- Appendix , *References*, contains complete references for citations made in the text.
- Appendix , *Keyboard Shortcuts and Accelerator Keys*, lists keyboard shortcuts for the commands in the interface.
- *Appendix , FELIX Startup*, gives helpful hints on starting FELIX the first time.
- *Appendix , Data Files*, list important file formats.

## Who should use this book

This book is intended as a basic guide to FELIX 2002 for both novice and advanced users of the program. Novice users will also want to look at the *FELIX Tutorials*, while advanced users may want to also consult the *FELIX Command Language Reference*.

*Topics covered*

The FELIX 2002 *User Guide* discusses the basic use of FELIX, including:

- Working with the interface
- 1D, 2D, and ND processing of NMR data
- Visualization of data
- Theory of NMR processing
- Hot-key reference
- File formats
- FELIX startup
- Data transfer and conversion

*Things to be familiar with*

You will probably want to familiarize yourself with a few things before working with FELIX 2002:

- The windowing software on your workstation
- Use of the mouse on your workstation
- Basic DOS commands
- The FELIX menu interface

*Workstation requirements*

Before you begin, be certain that you have these things available on your workstation:

- An installed and licensed copy of FELIX 2002
- A directory in which you can create subdirectories and files

## What does FELIX do?

FELIX is an interactive program for processing, displaying, and analyzing data acquired on nuclear magnetic resonance (NMR) spectrometers.

A complete NMR data processing and analysis program, FELIX provides you with tools for efficiently transforming NMR data of almost any dimensionality and for processing, displaying, storing, and retrieving the resulting spectral information.

FELIX is flexible and efficient to use. It can run either as a menu-driven graphical interface or as a concise and powerful command-driven program (via the FELIX Command Language: FCL). In addition, the FELIX macro processor enables you to automate lengthy and complex processing procedures (for example, routine or ND data processing). FCL is powerful enough to permit you to create your own menus and user interface or to customize the existing menus.

The quantity and variety of data that FELIX handles, ranging from peak integrals to assignment names of ND peaks, demands powerful data storage and management features, which are provided by the FELIX database. The database is accessible from many FELIX functions (e.g., the peak pickers and assignment interface), from the command line, or from within macros. The tools provided by the database allow you to quickly store data temporarily or perma-

nently in files, to display the data as lists, and to edit the spectral information using a table interface. The database also sorts data into lists according to user-defined criteria and can compare lists for similarities or differences.

## Feature list

FELIX's major functions include general features for NMR spectral processing and basic analysis, and specialized module features for biomacromolecular resonance assignment and receptor-ligand binding analysis. You may need to obtain ND, Assign, or Autoscreen license privileges to access the special features of FELIX.

### General features

- Platform-independent data file transfer between machines without file conversion.
- Direct reading of native spectrometer FID files from different vendors (Bruker, Varian, JEOL).
- Zero-filling of data sets.
- Linear prediction of the first and last points of an FID, with or without root reflection.
- Linear prediction of last points using mirror image methodology.
- DC offset correction.
- Window functions: exponential, sinebell, sinebell squared, skewed sinebell, skewed sinebell squared, Gaussian, trapezoidal, Kaiser, and convolution difference.
- Fourier transforms: complex fast Fourier transform (FFT), Bruker FFT, inverse FFT, real FFT and digitally oversampled FFT for Bruker data.
- Generation of complex data from real data for phasing using Hilbert transform.
- Baseline correction with automatic and manual baseline point selection; also cubic spline and polynomial baseline correction.
- Solvent suppression using time domain convolution, LP-SVD, and polynomial fitting.

- Spectrum phasing: automatic, real-time, and manual.
- Data buffer stack for easy storage and retrieval of free induction decays (FID's), spectra, and other plots, allowing comparison
- and point-by-point arithmetic operations between pairs of data buffers.
- Integration of the full spectrum of segments and integral values displayed on plot (and stored in database).
- Automated 1D line fitting for obtaining accurate integrals of noisy or poorly resolved data.
- Automatic or manual peak picking of 1D and 2D data; labeling of picked peaks with axis units.
- Enhanced peak picking using example peaks.
- 2D peak fitting and peak modeling.
- J-coupling extraction for 2D DQF and E-COSY spectra.
- J-coupling extraction based on heteronuclear E-COSY, FIDS, FIDS-E-COSY, and DQ-ZQ methods.
- Easy page setup, print preview and printing.
- Display features that include spectrum expansion, real plots, imaginary plots, and real/imaginary plots for 1D data or 1D slices of 2D data.
- Menu access for rapid, customized 1-D processing.
- Matrix storage of two-dimensional (2D) data for easy access to t1 and t2 data vectors.
- Supplied macros for simplified processing of states, TPPI, states-TPPI, and N/P 2D data.
- Enhanced contour plot for accurate and fast representation of peak intensity in two dimensions.
- Intensity plot for fast 2D data display of positive and negative peaks.
- Volume integration in two dimensions.
- Database tools for storing and correlating peak assignments.

- Tile plot to display isolated sets of overlapping peaks, simplifying identification of related peaks.
- Correlated cursors to permit accurate comparison of peak positions in several graphics frames at one time.
- Flexible frame connection to analyze multiple nD spectra concurrently.
- Keypad navigation within plots.
- Lists that allow you to sort and compare information (e.g., cross peak data from the database).
- Matrix compression to reduce 2D data set storage requirements, with minimal loss of spectral information.
- Importing processed data from other processing / analysis software: NMRCompass, NMRPipe, Bruker, and Varian.
- Flexible restraint generation tools for NOE-distance, NOE-volumes, 3J-dihedral, and ambiguous NOE-distance and NOE-volume categories-either in Discover / DG-II or X-PLOR format.
- Relaxation-time analysis for 2D heteronuclear data.
- Table interface to the database.
- Platform-independent data file transfer between machines without file conversion.
- Direct reading of native spectrometer FID files from different vendors (Bruker, Varian, JEOL).

## Special features for ND processing

The ND license allows you to access the following capabilities in addition to the general features in FELIX 2002.

- 3D transformation macros for states, TPPI, states-TPPI
- and N / P data.
- Plane transformation for 3D states, TPPI, states-TPPI
- and N / P data.
- 4D transformation macros for states, TPPI, states-TPPI
- and N / P data.

- Distributed processing for 3D and 4D transformation.
- Rapid "bundle mode" access to matrix vectors.
- Convenient display and analysis of 2D planes in 3D and 4D matrices from any direction.
- Keypad navigation between planes.
- Slider control for plane selection.
- Convenient access to 1D vectors from 3D and 4D matrices.
- 3D / 4D peak picking and volume integration.
- Enhanced peak picking using example peaks.
- 3D / 4D cross peak filtering.
- 3D / 4D cross peak modeling.
- Matrix compression to reduce 3D / 4D data set storage requirements, with minimal loss of spectral information.

## Assign module features

- Comprehensive features to organize the assignment project in a database.
- Define up to 12 spectra in one experiment.
- Overlay multiple contour plots in real time.
- Overlay multiple peak entries on contour plots.
- Tile and strip plots from frequency clipboard, spin systems (patterns), or prototype patterns.
- Display frequency clipboard or frequencies of spin systems (patterns) or prototype patterns on plots.
- Automated routines for detecting spin systems via systematic search in:
- 2D TOCSY, COSY, and / or NOESY spectra
- 3D homonuclear spectrum (e.g., 3D TOCSY-NOESY)
- 3D 15N HSQC (or HMQC)-TOCSY spectrum
- 2D 15N-1H HSQC and 3D 15N HSQC-TOCSY spectra

- 3D HNCO, HNCA, and HN(CO)CA spectra
- 3D CBCANH and CBCA(CO)NH spectra
- 2D 15N-1H HSQC and 3D CBCANH and CBCA(CO)NH spectra
- 3D HNCO, CBCANH, and CBCACO(N)H spectra
- 3D HNHA, CBCANH, and CBCA(CO)NH spectra
- 4D HNCAHA and HACA(CO)NH spectra
- 3D HCCH-TOCSY spectrum
- 3D H(CC-TOCSY)(CO)NH spectrum
- Automated routine for detecting spin systems via simulated annealing in 2D TOCSY, and 2D COSY and / or 2D 13C-1H HSQC spectra.
- User-tailorable semi-automated routine to exploit virtually any combination of heteronuclear double and triple resonance experiment to detect spin systems.
- Fuzzy algebra-based procedures for verifying new patterns.
- Library-based identification of patterns and frequencies using all-atom matching or the C -C combined chemical shift expectation value method.
- Sequential connectivity detection routine based on:
- 2D NOESY spectrum
- 3D homonuclear NOESY spectrum (e.g., 3D NOE-NOE)
- 3D 15N HSQC (or HMQC)-NOESY spectrum.
- Triple resonance spin systems.
- Rule-based approach to make sequence-specific assignments.
- Simulated annealing-based approach to make sequence-specific assignments.
- Tools to visually inspect and manually override the results of automated methods in all stages of the assignment procedure.
- Point-and-click manual assignment of frequencies or peaks.

- Automated peak assignment of up to 4D spectra based on assigned patterns for NOE and / or COSY type spectra; generating and storing ambiguous (multiple possible) assignments.
- Chemical shift index calculation based on H , C , and C chemical shift libraries.
- Tool for generating reports about the assignment of patterns.

## Autoscreen module features

- Comprehensive features to organize a 1D or 2D SAR by NMR-like project in a database.
- Unlimited number of 2D 15N-HSQC spectra can be processed, plotted, and reviewed.
- Unlimited number of 1D spectra can be processed, plotted, and reviewed.
- Automated processing of spectra based on parameters from a control spectrum.
- Novel algorithm for automated phasing of ND spectra.
- New algorithm for automated baseline correction.
- Innovative scoring algorithms that provide reliable peak-matching and -identification in situations where peaks in a reference spectrum disappear or additional peaks appear in spectra of protein-ligand complexes.
- Use of peak shape, including both peak widths and heights, for reliable peak matching and scoring.
- New progressive peak tracing (PROPET) algorithm that provides more reliable peak matching by taking advantage of titration spectra.
- Automated peak matching and scoring of an unlimited number of test spectra.
- Tools for overlaying multiple contour plots in real time and displaying peak displacements.
- Tools for defining and using a selected subset of peaks (region of interest) for scoring.

- Tools for generating reports about interesting spectra (high-affinity ligands) and interesting peaks (binding subsites) and for Kd fitting.
- Connection to Insight II to display molecular structure of ligand and protein.
- Automated coloring of peak displacements on protein surface displayed in Insight II.
- Exporting scores to Cerius2 study tables for further analysis using QSAR tools.
- Clustering experiments and peaks for identifying individual binding subsites in a protein.

## Starting FELIX 2002

Before reading this section you must have successfully installed FELIX.

You can start FELIX either by double clicking the FELIX 2002 icon on your desktop, or by clicking the **Start** button on the task bar, then selecting **Programs/Accelrys Felix 2002/Felix 2002**.

When FELIX is started, it reads a ***felixrc.ini*** file in the intallation directory. This file defines the paths that FELIX searches to find macros, menus, data, and other files. If necessary you can edit this file to customize it.

The FELIX product includes the macros and menus that are required for the FELIX program to start and run. These macros, menus, schema, and any other files that are essential for running FELIX are placed in *runtime directories*, as shown in Figure 1.

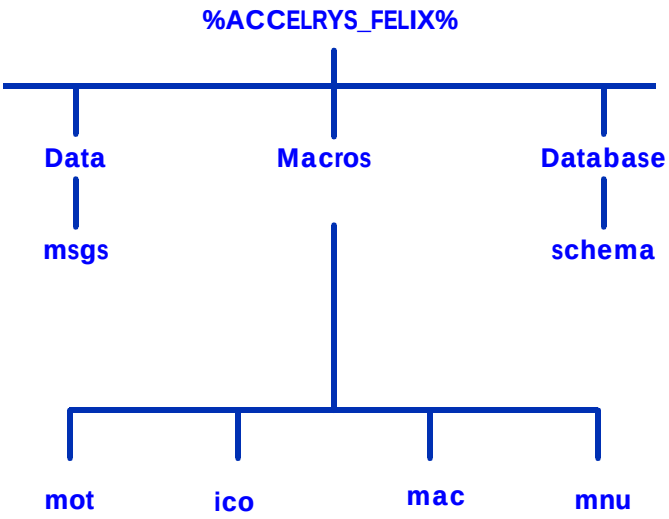


**Figure 1. FELIX runtime directories**

**Note:** ACCELRYS_FELIX is the environment variable created while FELIX is installed on your computer. It points to the folder where the FELIX executable files are located. By default the installation folder is **C:\Program Files\Accelrys\Felix 2002**.

For further details on starting FELIX and for tutorials on using it, please see *FELIX Tutorials*.

If you experience difficulties in running FELIX, please refer to the troubleshooting section in the Insight *System Guide*.

## Related books

You can find additional information about FELIX 2002, general molecular modeling, structure determination, and NMR data analysis in several other online books:

- *FCL Command Language Reference*—provides information for advanced FELIX users about using FCL and the command mode of FELIX.
- *FELIX Tutorials*—Contains step-by-step examples to help you learn to use FELIX.
- *NMRchitect*—Describes the theory of NMR data analysis and how to use the NMRchitect software to analyze NMR data in the Insight II environment.
- *Insight II*—Describes the Insight II general molecular modeling program environment.
- *System Guide*—Provides step-by-step instructions for installing and administering Insight II products in your operating environment.

## Typographical conventions

Unless otherwise noted in the text, this book uses these typographical conventions:

- Names of pulldowns, menu items, command names, and other things in the FELIX interface are presented in **bold** type. For example:

  Select the **Assign** pulldown.

- FELIX output and file samples are presented in a `courier` font. If the example indicates something you must type, it is given in **`bold courier`** font. For example:

  ```
  mat yuin.mat
  ```

- In referring to the menu items that are used when running FELIX through its menu interface, this guide uses the format **Pulldown/ Command**, since you use the mouse to select the pulldown first, before the command name appears. Where there is more than one cascading pulldown to access before the command name appears, the pulldowns are simply given in the order that you select them.

.

# 1 Theory

## Processing

### Linear prediction

Linear prediction estimates the value of a point based on the values of adjacent points. This method can be used to replace corrupted values in an FID. For example, in NMR experiments with fast relaxation rates, data collection must begin almost immediately following an RF pulse, even before the receiver and preamplifier can fully recover from ringing and saturation caused by the RF pulse. Consequently, several of the first datapoint values can be corrupted by instrumentation-induced noise. The effect of this noise on the spectrum can be profound because it often amounts to large fractions of the highest-amplitude signal values. Therefore the spectrum can improve noticeably when these corrupted point values are replaced with values estimated by linear prediction, where the estimated values are based on subsequent points of greater integrity.

A second application of linear prediction to NMR data is to extend an FID. This is useful for experiments in which data collection ceased before the signals completely decayed; that is, the FID is truncated. Here, the data values of the FID are used to estimate new data values that are appended to the end of the FID. Often the FID can be extended 20% or more by this means.

The **Process/Transform/Linear Predict First** menu item uses linear prediction to replace data values at the beginning of the FID, while the **Process/Transform / Linear Predict Last** menu item uses linear prediction to replace data values at the end of the FID and can also be used to extend the FID. The mathematical underpinnings of the linear prediction algorithm are discussed next, so that in the ensuing descriptions of the menu items, the settings in the control panels may be related to the mechanics of the algorithm.

Forward prediction is based on the assertion that the value of the $n^{th}$ data point $X_n$ is determined from the values of $K$ antecedent points:

$$X_n = \sum_{k=1}^{K} a_k X_{n-k} \qquad \textbf{Eq. 1}$$

where the $a_k$ are linear prediction (LP) coefficients. These LP coefficients are determined from $n$-$k$ values of $X_n$, cast in matrix notation as:

$$X\vec{a} = \vec{x} \qquad \textbf{Eq. 2}$$

In general, the matrix X is over-determined; therefore, the solution to this equation is found by least-squares analysis, which is based on a singular value decomposition in order to ensure a solution when $X^TX$ is not invertible:

$$\vec{a} = \nu\Lambda^{-1}\upsilon^T\vec{x} \qquad \textbf{Eq. 3}$$

The matrices ν, Λ, and υ are obtained from the singular value decomposition of the matrix X. The matrix Λ is a diagonal matrix of singular values of the columns of ν and υ containing the right and left singular vectors.

With knowledge of the maximum number of signals in the FID, the number of singular values in the matrix Λ can be reduced by truncating the matrices ν, $\Lambda^{-1}$, and $\upsilon^T$. The result is reduced noise in the predicted points (Makhoul 1978, Kumaresan and Tufts 1982, Barkhuijsen et al. 1985).

To assure that the predicted data points are well behaved; i.e., decaying, the technique of “root reflection” is often used. In this procedure, one constructs a complex polynomial defined by the LP coefficients. Then the roots are determined from this polynomial. For a noiseless FID, the roots should lie within the unit circle, which, of course, does not always hold for real data. To enforce proper behaviors, one reflects all the roots into the unit circle and reconstructs the polynomial from which a new set of LP coefficients are extracted (see e.g., Zhu and Bax 1993).

For backward linear prediction, you start with an analogous linear equation, as in Eq. 1:

$$X_n = \sum_{k=1}^{k} b_k X_{n+k} \qquad \textbf{Eq. 4}$$

The mathematics of determining $b_k$ is entirely similar to that used for forward prediction. However, you must pay attention when using root reflection that the roots in this case are reflected outside the unit circle.

Lastly, as pointed out by Zhu and Bax (1993), it is sometimes advantageous to perform both backward and forward predictions to obtain averaged LP coefficients. This procedure tends to be very good at suppressing noise and artifacts.

As mentioned earlier, linear prediction is accessed using the **Process/Transform/Linear Predict First** or **Last** menu items. Choosing any of these menu items displays a control panel. The following three controls, which determine how the LP calculation is performed are set in the control panels: **Points to use**, **Number of coefficients**, and **Number of peaks**. **Points to use** defines the number of points used to calculate the LP coefficients $a_k$ or $b_k$. This corresponds to the value *n* in Eq. 1. The **Number of coefficients** corresponds to the value *n-k*. And **Number of peaks** is used to truncate the matrices ν, Λ, and υ (Eq. 3).

## Automatic phase correction

FELIX 2002 has several alternative autophasing methods for 1D spectra and one method for 2D and 3D spectra. The basic method and the method based on peak integration are used exclusively for 1D spectra. The method described in this section is based on PAMPAS (Dzakula 2000) or APSL (Heuer 1991) and is commonly used for 1D, 2D, and 3D spectra.

For a 1D spectrum in the workspace or a selected dimension of an ND spectrum, FELIX divides the spectrum into 10-40 segments of equal widths along the phasing dimension and tries to find a best

sample peak per segment. The phase error of each individual sample peak is detected using the PAMPAS or APSL algorithm, and then the global zero- and first-order phase errors (represented by FELIX symbols **phase0** and **phase1**) are obtained by linear regression analysis.

If solvent peaks or other noise peaks exist in the spectrum, it is important to exclude them from the phase detection process. FELIX allows you to define up to 10 excluded areas (using the **aph exclude** command). An excluded region is specified by the starting and ending datapoint numbers, and the dimension if it is an ND spectrum (which can be different from the phasing dimension). Any peaks that fall in the specified range along this dimension are not used as sample peaks.

For an ND spectrum, FELIX searches all the vectors along the phasing dimension and tries to find a best sample peak for each segment, without considering peaks in the excluded areas. Peaks are selected based on their width (must be wider than a user-defined threshold), height (the higher the better), and symmetry (the more symmetric the better). For a 1D spectrum, sample peaks are searched similarly in the only vector for each segment.

If more than two sample peaks are found for the whole spectrum, the phase errors are detected from each of them. Next, the global phase parameters (**phase0** and **phase1**) are obtained by linear regression analysis. The phase errors detected are between 0° and 180°, which do not reflect the actual phase differences between them. The possible π-jumps between the individual phase errors are automatically compensated within the expected range of phase1. The default range of **phase1** is approximately -720° to 720°. You can change the range of **phase1** when using the **aph** command. See *FELIX Command Language Reference* for more details about using this command.

# Assign

## Basis of the assignment strategy

One approach to making assignments, developed by Wüthrich and coworkers (Wüthrich 1986), is the “sequential assignment strategy”,

which obtains sequence-specific resonance assignments (see Figure 2).

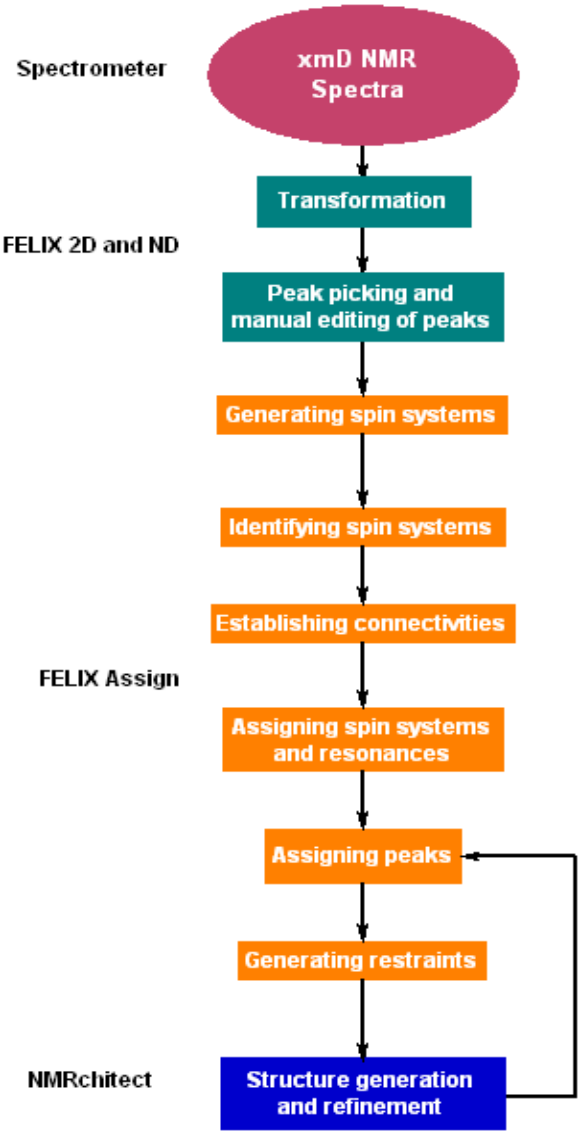


**Figure 2 Sequential Assignment Strategy**

The first step in this procedure is the delineation of individual spin systems, based on homonuclear 2D and / or 3D spectra or on heteronuclear spectra. If homonuclear 2D spectra are utilized, this spin-system delineation can be based on COSY type spectra (e.g., DQF-COSY or P-COSY). A *spin system* is understood to be a group of spins connected by scalar or through bond spin-spin couplings. This means that, for homonuclear spectra of a peptide or protein, one amino acid residue can contain one or more spin systems. However, due to the peptide bond (which yields a structure in which no non-labile hydrogens are three bonds apart in neighbor residues) spin systems do not comprise spins from more than one residue. Therefore, in a COSY-type spectrum, cross peaks are expected between scalarly coupled spins within individual residues.

For spin-system delineation, the TOCSY or HOHAHA type spectra can also be used, since the cross peaks arise only within individual spin systems. Ideally, TOCSY contains more cross peaks for the same spin system than COSY; thus, in complicated overlapping situations it is advantageous to combine the information from both types of spectra.

For spin-system delineation or detection it is possible to use two different methods: systematic searching (when all the spin systems in the spectrum are collected through systematically searching the spectrum for well-aligned peaks) and simulated annealing.

The complete delineation of spin systems is facilitated by the fact that the chemical shifts of the protons are in well-defined regions depending on structure (primary or secondary). The expectation values have been extracted for different protons in amino acid residues, based on analysis of assigned proteins (Gross 1988). You would expect to find most $H_N$-$H_\alpha$ cross peaks at 6-12 ppm and 3-6 ppm. The chemical shift information and the pattern of cross peaks is necessary, but not sufficient by itself, for identification of spin systems with specific residue types, since neither all the chemical shifts nor all the cross peak patterns are unique.

To reduce ambiguities, additional homonuclear experiments can be performed (either NOESY or changing the solvent, pH, or temperature). In more difficult situations, heteronuclear experiments may also be added, if possible.

After spin-system delineation is complete, typically some spin systems remain that are not identified uniquely. Also, unique residue

type identification is not always enough, since most biopolymers contain several copies of most residue types. These facts then necessitate the second step of sequence-specific resonance assignment, i.e., matching spin systems with specific residues. This is done through identification of NOE connectivities between spin systems corresponding to neighbor residues in the sequence.

Rigorous analysis of X-ray structures of globular proteins (Billeter et al. 1982) showed that several inter-residue distances are almost always small enough to yield cross peaks in a NOESY spectrum. These are, in particular: $d_{\alpha N}(i,i+1)$, $d_{\beta N}(i,i+1)$, and $d_{NN}(i,i+1)$. Usually $d_{\alpha N}(i,i+1)$ and $d_{\beta N}(i,i+1)$ are not short enough. Therefore, these distances can be used to tie together stretches of spin systems by analyzing the NOE interactions between protons of *different* spin systems. If such stretches can be matched with stretches of residue sequence, sequence-specific assignment is achieved with a high level of confidence.

Since NMR data tend to be incomplete or ill-resolved, a systematic search of perfect assignments yields only partial results. One has to tolerate a significant amount of imperfection and use loose criteria in order to find a probable assignment. Even then, an exhaustive inspection of all possible solutions might become impossible within reasonable computation times. Optimization methods such as simulated annealing then provide a powerful alternative; all constraints are used to define an energy for the system, which is minimized according to careful schemes. No general tolerance has to be adjusted. The minimizer spontaneously allows imperfections where (and only where) no better solution can be found with the available data (just as it does if peaks are missing from a TOCSY pattern, or if NOEs or even residues are missing from a sequential assignment) (Morelle et al. 1994a, 1994b).

## 2D/3D homonuclear assignment strategy

The following steps were used by Eccles et al. (1991) for sequence-specific assignment of proteins based on homonuclear 2D spectra (DQF-COSY, TOCSY, and NOESY):

- Peak-pick all spectra.

- Identify $^1$H spin systems of individual amino acid residues using systematic searching (see Figure 3). Starting from the HN-$H_\alpha$ cross peaks, the spin systems were disentangled based on DQF-COSY and TOCSY patterns and on the chemical shifts of key protons (chemical shift database). Each amino acid residue was represented by a single spin system, except for the aromatic side chains, asparagine and glutamine, which contain two separate spin systems. The substeps were:

  a. Start with the $H_N$-$H_\alpha$ cross peak in DQF-COSY.

  b. Identify all cross peaks belonging to this spin system using TOCSY (the alignment of COSY and TOCSY spectra may be rather poor--0.03 ppm).

  c. Check all resulting groups against a database of 20 amino acid residues and score them.

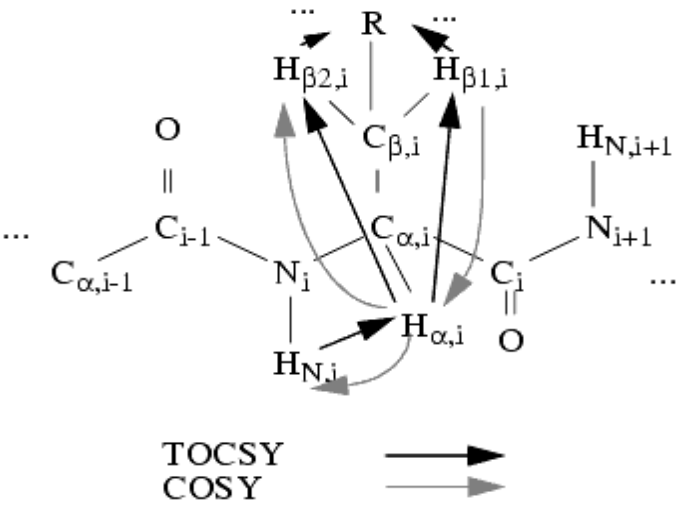


**Figure 3 Homonuclear spin systems**

- Assign individual spin systems to amino acid residues in the sequence. Using the sequential NOEs, polypeptide segments of variable length were identified and matched against the amino acid sequence of the molecule. This step can be automated, since

all (except one, the Xxx-Pro) sequential NOEs ($d_{NN}$, $d_{\alpha N}$, and $d_{\beta N}$) involve at least one amide proton.

a. A search can be carried out along the $\omega_2$ direction at the $H_\alpha$, $H_\beta$,... resonances of each spin system (with an allowed uncertainty of 0.01 ppm) to find the corresponding $H_N$ resonance of the sequential neighbor spin system. This can yield a list of NOE connectivities between the alpha, beta, and amide protons of each spin system and the amide proton of other spin systems.

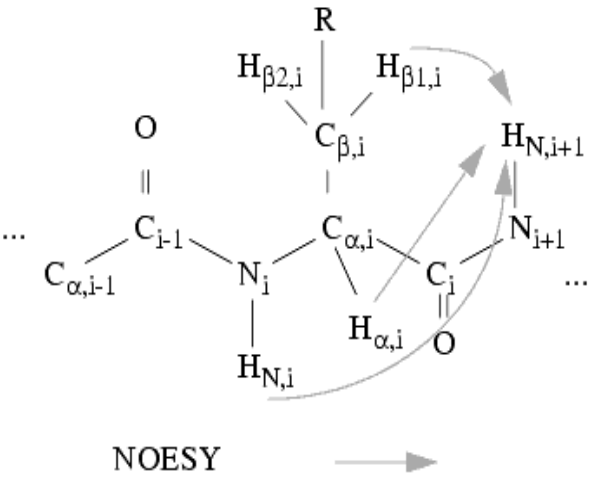


**Figure 4 Sequential connections between spin systems**

b. Uninterrupted pathways can be searched for, using the sequential NOEs and the amino acid sequence.

♦ These steps result in an assigned resonance list, which can be transferred to NOE peaks. This assigned NOE peak list can then be refined by using structure refinement and repeating the identifying and assigning steps several times

For spin-system delineation you can use (besides 2D spectra) homonuclear 3D spectra such as the 3D TOCSY-TOCSY spectrum (Kleywegt et al. 1993). The spin system can be connected sequentially via 3D TOCSY-NOE or 3D NOE-NOE spectra.

## 3D/4D heteronuclear assignment strategy

The more reliable and more efficient assignment strategy is based on higher dimensional heteronuclear spectra of singly or doubly labelled proteins or nucleic acids (some examples include Redfield et al. 1994, Hansen et al. 1994, Freund et al. 1994, Clubb et al. 1994, Anglister et al. 1994). Here, several methods of assigning singly or doubly labelled proteins are briefly described. The number of experiments that have been carried out on nucleic acids is still quite limited; for more detailed information on these types of experiments please see Farmer et al. 1993, Sklenar et al. 1993, Marino et al. 1994, Legault et al. 1994.

One of the first methods described in the literature used several combinations of double- and triple-resonance experiments to unravel the assignment for much bigger proteins than was possible earlier with homonuclear methods (Grzesiek et al. 1992). The steps involved can be summarized as follows:

- Measure HNCO, HN(CO)CA, HCACO, HNCA, NOESY-($^{15}$N)HMQC, TOCSY-($^{15}$N)HMQC, and HCCH-COSY for $^{15}$N and $^{13}$C doubly labelled proteins.
- Pick all peaks in all spectra.
  a. Some peaks could have been missing from some spectra. For example, if water presaturation was used to measure HNCA (which might have been necessary) the $H_N$ protons, which were in fast exchange, would not give a signal when compared to HNCO.
  b. The typical peak halfwidth is 45-55 Hz for $H_\alpha$ and 35-45 Hz for $C_\alpha$, therefore the peak positions were prone to ambiguities.
  c. Some spectra could have given extra peaks. For example, HNCA gave extra peaks for $H_{N,i}$-$N_i$-$C_{\alpha,i-1}$, but this can be used in conjunction with HN(CO)CA. Special care was taken in automated assignment.
- Look through the HNCO list and find all $H_{N,i}$-$N_i$-$C(O)_{i-1}$ correlations.
- Compare the above list with HN(CO)CA, which also yielded (for each resonance triplet $H_{N,i}$-$N_i$-$C(O)_{i-1}$) the $C_{\alpha,i-1}$ resonance. The

extra peaks of HNCA could have also been used instead of or in addition to the above at this step to obtain the same information (that is, if it were known which ones were the *extra* peaks).

- Using the list of $C_\alpha$ and CO frequencies, search through the HCACO and find the possible corresponding $H_{\alpha,i-1}$ resonance (or the set of such resonances).
- Search the HNCA, the TOCSY-($^{15}$N)-HMQC, and the NOESY-($^{15}$N)-HMQC spectra for $H_{N,i}$-$N_i$-$C(O)_{i-1}$) resonance pairs from earlier assignments to the corresponding $C_{\alpha,i-1}$ (in HNCA) $H_{\alpha,i-1}$ (in TOCSY-NOESY) pair. This can give multiple possibilities due to severe overlap in the H / C region. The differences and similarities in the TOCSY-($^{15}$N)-HMQC and NOESY-($^{15}$N)-HMQC spectra could help with the overlap. For example, if there is a $d_{NN}$ connectivity (because of helical structure), NOESY gives rise to signals for both $H_{N,i-1}$-$N_{i-1}$-$H_{N,i}$, $H_{N,i-1}$-$N_i$-$H_{N,i}$ and $H_{N,i-1}$-$N_{i-1}$-$H_{\alpha,i_{-1}}$, while TOCSY results in $H_{N,i-1}$-$N_{i-1}$-$H_{\alpha,i-1}$.
- Obtain the corresponding H / C pairs for the $H_\beta$ / $C_\beta$ pair via HXCH-COSY.

For smaller proteins you can use fewer triple-resonance experiments and still achieve good sequential assignment of the backbone. In particular, you can use a combination of two 3D triple-resonance experiments (Grzesiek and Bax 1992), which correlates the $C_\beta$, $C_\alpha$, N, and $H_N$ resonances of the i$^{th}$ and i-1$^{th}$ residues using CBCANH or HNCACB and CACB(CO)NH experiments. Then the residue types can be identified using a database of $C_\alpha$ and $C_\beta$ chemical shifts (Grzesiek and Bax 1993).

You can also choose to apply 4D methods using these steps. Although they are usually limited by spectrometer time, the information content of two 4D spectra can be very large. For example, you can use an HNNCAHA or HACANHN and HACA(CO)NHN experiment and, by theoretically analyzing only these two spectra, obtain a list of $H_{\alpha i}$, $C_{\alpha i}$, $N_i$, $H_{N,i}$, $H_{\alpha,i-1}$, and $C_{\alpha,i-1}$ resonances. From this list you can then construct a full sequential assignment of the backbone (Boucher et al. 1992a, 1992b, Constantine et al. 1993).

## Assign building blocks

The Assign module includes several building blocks: the database, the visualization and analysis tools, and the automated algorithms.

The database consists of the data structures, which are built on the Felix database management system. Using this Assign tool, you can keep all necessary information in a compact binary file, from which the specific entities (i.e., tables) can be exported in ASCII format. Also, FELIX provides spreadsheet capability, allowing you to view and, in certain cases, edit the tables.

The Assign database is structured in several layers. First you define a project entity, which is a *container* entity (i.e., it is at the top level and contains all information about the lower-level entities). The project entity, in which the assignment is carried out, then contains spectrum parameters (e.g., plot appearance, peaks, and spectra type description); spin-system information (patterns and frequencies, together with properties); sequence information (if any); the library of chemical shifts of residues; and various entities needed in the assignment procedure.

Table 1 lists the most important entities and symbols used in an Assign project:

**Table 1 Important Assign project entities and symbols**

| Entity | Variable name | Schema | Function |
|---|---|---|---|
| project | proent | kbexper.sch | Stores defined experiment names, types, plotting parameters |
| prototype patterns | rprent | rpr.sch | Set of frequencies that may have encoded frequency types |
| patterns | rpaent | rpa.sch | Pointers for each pattern-to-resonance list, neighbor item, and residue type probability item |
| resonance list | reg:reslis[1] | rrl.sch | Pointers to resonances for a particular pattern |
| resonances or frequencies | rreent | rre.sch | Contains a generic shift, a set of specific shifts and possible names for a given resonance for each item |
| neighbors | neient | rnl.sch | Possible neighbor patterns |

[1]No variables defined

**Table 1 Important Assign project entities and symbols (Continued)**

| Entity | Variable name | Schema | Function |
|---|---|---|---|
| residue type probabilities | typent | rprob.sch | Possible neighbor patterns for each given pattern with probabilities |
| atoms (with pseudoatoms) | reg:rasn [1] | rasn.sch | Real atom names and pseudoatoms of the current molecule (no coordinates) |
| sequence | reg:fulseq [1] | fulseq.sch | Sequence of the molecule |
| residue types | reg:rseq[1] | rseq.sch | Molecule residue types |
| clipboard or resonance buffer | reg:resbuf[1] | rrb.sch | Set of frequencies in the clipboard with generic types. |

[1]No variables defined

Visualization and analysis tools are provided by the graphical user interface (GUI). Functionality in the Assign module is organized into several menus. Menu items dealing with repetitive visualization tasks (e.g., changing plot type, redefining plot limits, activating plots, drawing peaks) are easily accessible at all times through a menu connected to the right mouse button. Menu items specific to Assign include tools for panning the spectrum with the mouse, tiling patterns in different spectra, overlaying contour plots, overlaying peak boxes on top of different spectra, easy access to strip plots, displaying frequency lists from different sources on different spectra.

When working with multiple 3D / 4D data sets at once, quick access to different spectra as well as coordinated navigation through 2D planes of different views of different spectra is crucial for successful assignment. Therefore up to four different views from four different spectra can be connected, letting you define which axes from which views are to be changed together.

Many automated algorithms that facilitate the sequential assignment strategy are found within the **Collect Prototype Patterns**, **Frequency Clipboard/Compare Frequencies**, **Residue Type/Score Residue Type**, **Residue Type/Match Residue Type**, **Neighbor/Find Neighbor Via...**, **Sequential/Systematic Search**, S**equential/ Simulated Annealing**, and **Peak Assign/Autoassign Peaks** menu items.

The prototype-pattern detection menu items serve as spin-system detection tools. FELIX currently has several methods for automatically detecting spin systems:

The 2D homonuclear systematic search algorithm performs the steps shown in Figure 5.

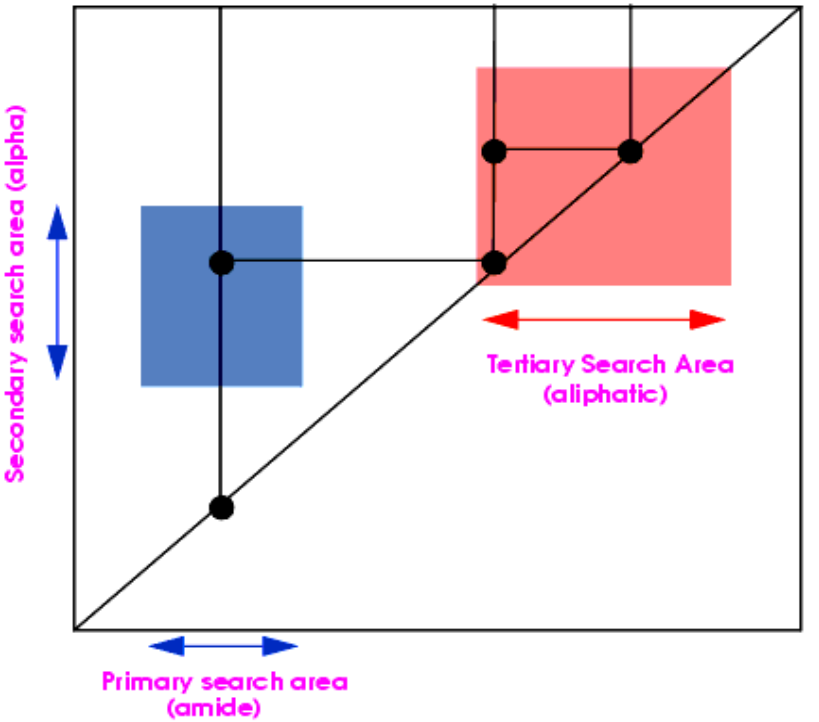


**Figure 5 Homonuclear 2D systematic search**

- Depending on the method, collect the cross peaks in the defined seed area (in COSY, TOCSY, or both).
- Depending on the method, collect the cross peaks in the expansion area (in COSY, TOCSY, NOESY, or all).
- For every peak in the seed area, use the coordinates as the first two frequencies of a potential prototype pattern.
- Loop through all expansion peaks. If a peak is found for which exactly one frequency occurs in the new prototype pattern and

the other does not, then this frequency is a candidate. The program scores and removes the candidate if the constraints are not satisfied.

- Sort the candidates by score, keeping the best ones.
- Start over at the beginning of the loop, unless: (a) there were no suitable candidates, (b) the prototype pattern now contains the maximum allowed frequencies, or (c) the maximum number of loops has been reached.
- Apply filters, if any. With filtering, prototype-pattern detection is made selective. That is, you can decide how many frequencies a valid prototype pattern should contain in various regions. Also, by defining the minimum and maximum number of frequencies that may be contained in a new prototype pattern, you can selectively detect spin systems (e.g., glycines and AMXs).

The 3D homonuclear spin-system detection algorithm can be used on a [J,J]-, a [J,NOE]-, or a [NOE,J]-type experiment and performs the steps in Figure 6.

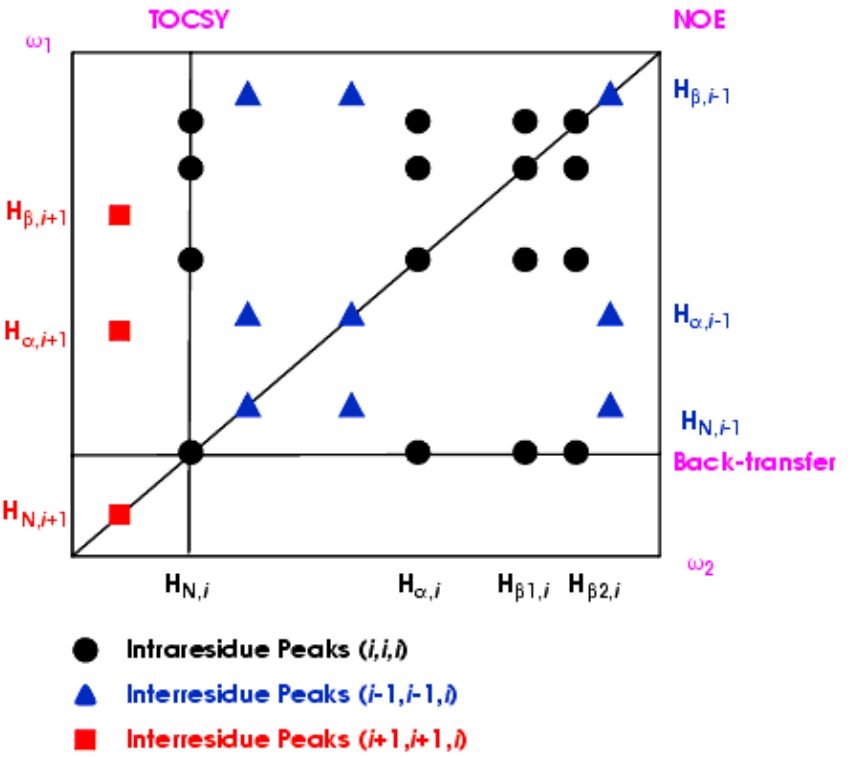


**Figure 6 Typical components in the 2D $\omega_1$-$\omega_2$ plane of an AMX residue in a 3D TOCSY-NOE spectrum**

- Collect the cross peaks in the defined seed area. The pseudodiagonal peaks are left out of this set.
- Collect the cross peaks in the expansion area. The pseudodiagonal peaks are left out of this set.
- Use user-specified coordinates as the first two frequencies of a potential prototype pattern for every peak in the seed area.
- Loop through all expansion peaks. If the algorithm finds a peak in which exactly two frequencies occur in the new prototype pattern and the third frequency does not, this frequency is a candidate if its coordinate was set to be used. The algorithm scores and removes the candidate if the constraints are not satisfied.
- Sort the candidates by score, keeping the best ones.

- Start over at the top of the loop, unless: (a) there were no suitable candidates, (b) the prototype pattern now contains the maximum allowed frequencies, or (c) the maximum number of loops was reached.
- Apply filters, if any. With filtering, prototype pattern detection is made selective. That is, you can decide how many frequencies a valid prototype pattern should contain in different regions. Also, by defining the minimum and maximum number of frequencies that can be contained in a new prototype pattern, you can selectively detect spin systems (e.g., glycines and AMXs).

The 2D simulated annealing detection algorithm uses these steps:

- Use only half of the TOCSY or COSY spectra, for the sake of simplicity, no matter what the peak-picking file contains. The algorithm uses the half beneath the diagonal, except for the fingerprint region ($\omega_1$>6 ppm, $\omega_2$<6 ppm), for which the upper part is used to avoid the water signal. Then the selected peaks are reflected about the diagonal to yield complete symmetrical spectra.
- Attempt to assign peaks to a requested amino acid spin system, rather than recognizing patterns in the spectra. This system is defined from the usual shifts and correlations library as a list of requested TOCSY cross peaks. To reduce the space to be searched, the homonuclear TOCSY cross peaks are sorted according to frequency in several overlapping regions, each corresponding to one correlation (or a few equivalent ones) of a specific residue type. If the COSY spectrum is available, the TOCSY peaks are matched against it to further reduce the number of suitable peaks in each region. In addition, a natural-abundance $^{1}H$-$^{13}C$ HSQC spectrum can be recorded to identify possible related carbon frequencies for each proton frequency and to use the powerful carbon chemical shift ranges as an additional constraint (Medvedeva et al. 1993). Each proton frequency is matched to a few HSQC cross peaks (one for each overlapping proton spin). The carbon chemical shift is very specific to amino acid type and spin, which allows you to reduce the number of possible regions for the corresponding TOCSY peak.
- Iteratively fill the pattern with these TOCSY peaks, using an adapted simulated annealing routine. The algorithm computes an energy at each step, which takes into account the relative

alignment of the peaks in the motif. The program randomly chooses one cross peak among the unsatisfactory ones (empty, or misaligned peaks) and tries a permutation between the original peak and one from the adequate region in the TOCSY pool. Depending on the energy variation, the former peak might be released into the pool and the new one stored into the pattern, Eventually, the system converges to a pattern of consistently aligned peaks. A maximum number of missing peaks is tolerated, according to the specified amino acid type. Whenever the algorithm does not find enough aligned peaks, the pattern contains some empty (null) peaks. If the pattern ends up completely empty, it is not stored as a prototype pattern.

Assign implements several heteronuclear spin-system detection methods, which can be based on double- and triple-resonance experiments.

It is possible to use a $^{15}N$ separated TOCSY experiment to collect spin systems starting on $H_N$ and N frequencies of the pseudodiagonal peaks (i.e., $^{15}N$-$H_N$-$H_N$ peaks). The algorithm is similar to the 3D homonuclear one, but you need to specify the dimension that the root frequency (i.e., the $H_N$) is in, and then all the frequencies are collected into spin systems that have peaks with the same $H_N$ and N frequencies. A method for collecting spin systems starting with 2D peaks is also implemented. For this method, you need to provide a $^{15}N$-$^{1}H$ HSQC and a $^{15}N$-HSQC-TOCSY spectra (certainly they can be HMQC, too).

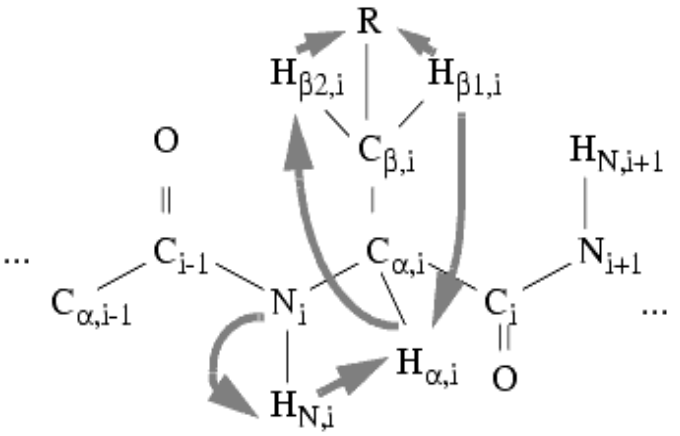


**Figure 7 Spin systems in $^{15}$N-edited TOCSY experiments**

Triple-resonance based spin-system detection algorithms are also provided in Assign. Particularly, you can use a combination of three 3D spectra–the HNCO, HN(CO)CA, and HNCA–and collect inter-residue spin systems or prototype patterns containing $H_{N,i}$-$N_i$-$C_{\alpha i}$-$C_{\alpha i-1}$-$C_{i-1}$ frequencies. Then during promotion to patterns, an automated routine can find the sequential neighboring systems and store the results as neighbor probabilities.

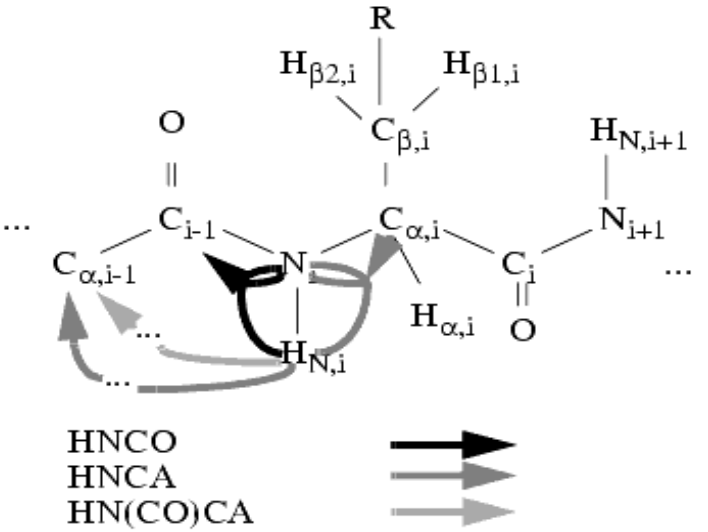


**Figure 8 Spin systems in a typical triple-resonance 2D experiment**

Similarly, you can use two triple-resonance 3D experiments–a CBCA(CO)NHN and a CBCANHN or HNNCACB–to collect prototype patterns containing $H_{Ni}$-$N_i$-$C_{\alpha i}$-$C_{\beta i}$-$C_{\alpha i-1}$-$C_{\beta i-1}$frequencies, and a similar promotion can be carried out. Now you can also use the $^{15}N$-$^{1}H$ HSQC spectrum to help collect the spin systems–mixing the information from heteronuclear double- and triple-resonance spectra.

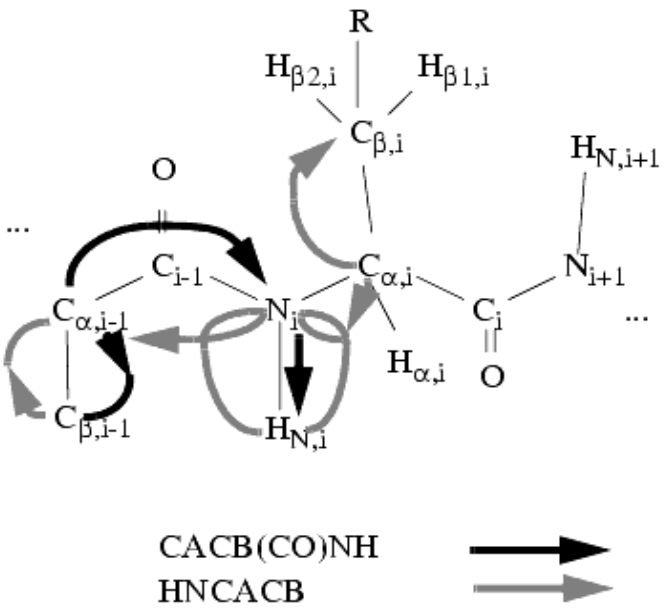


**Figure 9 Spin-system detection in 3D triple-resonance spectra**

In a similar method to that used in the previous example, you can collect spin systems in three triple-resonance experiments: HNCO, CBCANH, and CBCACO(N)H. This method results in spin systems containing: $H_{Ni}$-$N_i$-$C_{\alpha i}$-$C_{\beta i}$-$C_{\alpha i-1}$-$C_{\beta i-1}$-$C_{i-1}$.

It is possible to use an experiment to specifically find glycine spin systems (Wittekind 1993). A spin-system detection method is implemented in Assign for this, using the HNHA, CBCANH, and CBCA(CO)NH spectra.

Finally, you may use two 4D triple-resonance experiments–an HACA(CO)NHN and HACANHN–to automatically collect prototype patterns containing $H_{Ni}$-$N_i$-$C_{\alpha i}$-$H_{\alpha \iota}$-$C_{\alpha i-1}$-$H_{\alpha i-1}$ frequencies.

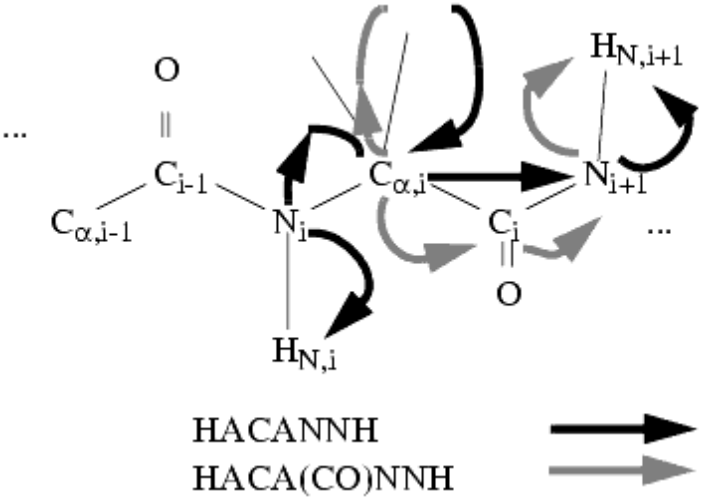


**Figure 10 Spin-system detection in 4D triple-resonance spectra**

A visual tool is provided to allow you to quickly find the previously detected spin systems in multidimensional spectra and to verify them visually.

The **Frequency Clipboard/Compare Frequencies** menu item checks a frequency list against a set of patterns or prototype patterns. The algorithm finds how many “fuzzy similarities” (Kleywegt et al. 1989) are present between the current collection of frequencies and the frequencies in each pattern or prototype pattern. You can use this to decide whether a new frequency list constitutes a new pattern.

The residue-type matching and scoring of patterns helps in assigning frequencies in patterns to specific residues. For each frequency in a pattern, the match-type action calculates a delta value based on the library expectation values and standard deviations collected for each residue type (delta = [actual_shift_of_freq - expectation_value]/standard_deviation). The score-type action automatically matches patterns and residue types. The algorithm tries to find a matching frequency in the pattern. The best-matching frequency may lie no more than a given number of standard deviations away

from the expectation value. If at least a certain number of atoms can be matched, a score is computed.

The **Neighbor/Find Neighbor Via...** menu items serve to find potential neighbor patterns through a 2D NOESY, a 3D [J,NOE], a 3D [NOE,NOE], or a $^{15}$N-resolved NOESY spectrum. Therefore, this is the first step in sequence-specific assignment of patterns.

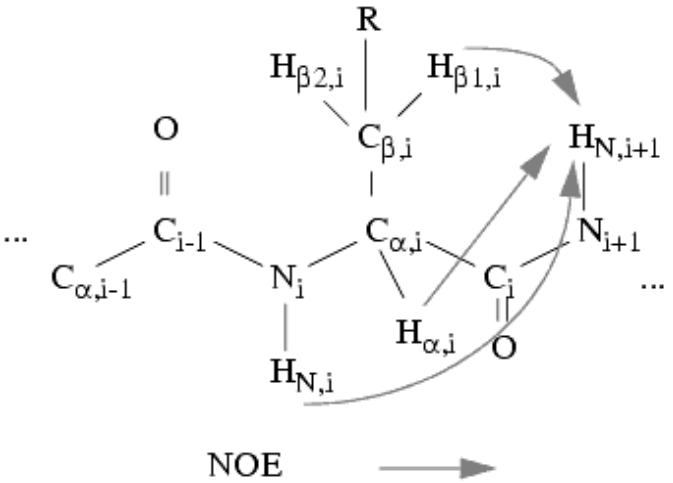


**Figure 11 Neighboring spin-system detection based on NOE**

The algorithm consists of these steps:

- Loop through all NOE cross peaks. If one of the peak coordinates occurs in the selected set of frequencies of patterns considered and the other does not, the algorithm stores the latter as a candidate neighbor frequency.
- Store the candidates and search to ascertain which pattern contains the candidate frequency as root frequency. If such a pattern is found, this is a candidate neighbor pattern.
- Sort the candidate neighbor patterns and store them together with their scores.

FELIX implements a triple-resonance-neighbor-finding algorithm in the **Neighbor/Find Neighbor Via 3D/4D** menu item. This action works on a triple-resonance experiment (e.g., HN(CO)CA or

CBCA(CO)NHN) which contains cross peaks connecting neighboring residues. The algorithms for each pattern look through the spectrum and try to find peaks that connect this pattern to, for example, the *i*-1$^{th}$ residue. Then the sequential connectivity is set for the pattern that had the candidate frequency.

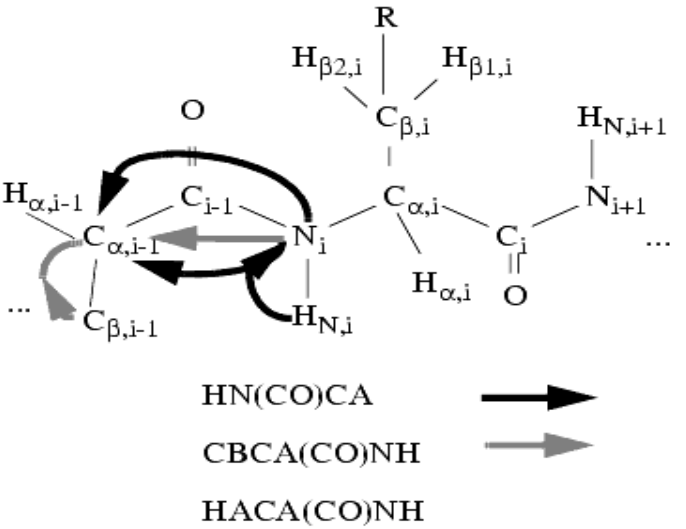


**Figure 12 Neighboring spin-system detection based on triple- resonance spectra**

The **Sequential/Systematic Search** and **Sequential/Simulated Annealing** menu items find potential matches of patterns with the sequence of an unbranched biopolymer. This can be done using systematic search methodology through a recursive matching procedure or using simulated annealing. In the latter case, the program tries to assign a set of patterns to the primary sequence (or a requested segment of it) of the molecule using optimization techniques rather than systematic searching. Each step in the simulated annealing routine is a permutation between the pool of patterns and a residue in the sequence (initially filled with empty dummy residues and, optionally, some previously assigned residues) or between two stretches of residues tentatively assigned in the primary sequence. The energy is computed from previously determined residue-fit scores and neighbor information. Eventually, the results converge to a solution of low energy, which is unique in well

defined parts of the sequence. However, when the data are inadequate, several runs of the program might yield locally different solutions.

The **Peak Assign/Autoassign Peaks** menu item automatically assigns J or NOE peaks based on a set of assigned patterns, library, and molecule information. The patterns must have spectrum-specific shifts set for their resonances. The algorithm works as follows:

- Each peak's frequencies are compared against the assigned resonances of patterns.
- If candidate assignments are found, then the NOE connectivities are filtered against the provided distance cutoff (rejection cutoff), and the J connectivities are filtered against the J-step cutoff.
- If multiple candidate assignments remain:

  a. If one of the possible assignments has a distance connected to it that is shorter than the unambiguous cutoff and none of the others do, then this assignment is stored as a single assignment.

  b. If more than one of the possible assignments has a shorter distance than the unambiguous cutoff, or none do, and multiple assignment is requested, then these are stored as multiple possible assignments.
- If a single candidate assignment remains, it is stored as a single assignment.
- NOE distance-filtering is omitted if only one candidate assignment is present and a flag is set to not enforce distance cutoffs. This latter option should be used with caution, since it can lead to misassignment. However, it prevents you from including ambiguous structure information in the assignment if there is no good structural model yet.

# Autoscreen module

## Introduction

NMR techniques for characterization of protein-ligand interactions are being widely used in rational drug design. As an example, the SAR by NMR strategy, devised by the Fesik group at Abbott Laboratories (Shuker et al. 1996; Hajduk et al. 1997), consists of the following general steps:

1. First, a library of small molecules is screened for binding to an $^{15}$N-labeled protein. If a molecule binds to the protein, it alters the local chemical environment and thereby cause changes in the chemical shifts of nuclei in the protein's binding site. Such changes are detected in HSQC spectra acquired in the presence and absence of added ligand.
2. Once initial ligands are identified, analogs are screened and binding constants are obtained in order to optimize the interactions with the protein.
3. Next, the 3D structure of the protein complexed with the ligands is obtained, and linked compounds are synthesized based on this structural information.

The Autoscreen module of FELIX automatically characterizes the protein-ligand interactions by identifying the changes between the reference and test spectra. With Autoscreen you can organize the project, process and score the 2D test spectra, and identify high-affinity ligands. For the SAR by NMR steps described above, Autoscreen offers help for the first two steps.

The core of Autoscreen consists a series of novel algorithms for tracing the shifting of peaks between the control and test spectra. In addition to the automatic peak tracing (APT) for high-throughput analysis, Autoscreen also provides an option for progressive peak tracing (PROPT) which is ideal for semi-automatic analysis of a series of titration spectra. In such a case, the shifting of the peaks is more reliably traced step-wise between the neighboring spectrum pairs following the titration curve.

**Note:** Although Autoscreen provides automatic processing and interactive analysis for 1D spectra, the discussions below apply only to 2D spectra.

## Autoscreen scoring strategy

A crucial phase in using the Autoscreen module is scoring of the test spectra against the control spectrum. Follow the steps below:

1. Pick peaks in the test spectrum.
2. Match test peaks to control peaks based on chemical shifts and peak shapes, and then score the experiment based on the displacements between the matched peak pairs.
3. Add penalties for the unmatched control and test peaks to the score.

The underlying principles are explained in the following sections.

### Basics

The prerequisites for a test peak to be matched to a control peak are Eq. 5-Eq. 7.

$$\Delta H \leq max\Delta H \qquad \textbf{Eq. 5}$$

$$\Delta N \leq max\Delta N \qquad \textbf{Eq. 6}$$

$$S \geq minS \qquad \textbf{Eq. 7}$$

where $\Delta H$ is the absolute displacement of the $^{1}H$ chemical shift between the test peak and the control peak, $max\Delta$H the upper limit of this $^{1}H$ displacement, $\Delta N$ the absolute displacement of the $^{15}N$ chemical shift, $max\Delta N$ the upper limit of the $^{15}N$ displacement, $S$ the shape similarity between the test peak and the control peak, and $minS$ the lower limit of the shape similarity.

The test in Eq. 7 is applied when you choose to consider the shape of peaks. You can choose to define the shape similarity $S$ as the product of the similarities of the peak widths and intensity:

$$S = S_{\omega_H} \times S_{\omega_N} \times S_I \qquad \textbf{Eq. 8}$$

where $S\omega_H$, the similarity of peak widths along the $^1$H dimension, is calculated based on the width of the control peak $\omega_H^C$ and the width of the test peak $\omega_H^T$ along the $^1$H dimension:

$$S_{\omega_H} = \frac{\omega_H^C}{\omega_H^T} \qquad \textbf{Eq. 9}$$

$$if\ S_{\omega_H} > 1, S_{\omega_H} = \frac{1}{S_{\omega_H}}$$

Analogously, $S_{\omega_H}$, the similarity of peak widths along the $^{15}$N dimension, is calculated based on the width of the control peak $\omega_N^C$ and the width of the test peak $\omega_N^T$ along the $^{15}$N dimension:

$$S_{\omega_N} = \frac{\omega_N^C}{\omega_N^T} \qquad \textbf{Eq. 10}$$

$$if\ S_{\omega_N} > 1, S_{\omega_N} = \frac{1}{S_{\omega_N}}$$

$S_I$, the similarity of peak intensities, is calculated based on the peak heights of the control peak $I^C$ and of the test peak $I^T$:

$$S_I = \frac{I^C}{I^T} \qquad \textbf{Eq. 11}$$

$$if\quad S_I > 1, S_I = \frac{1}{S_I}$$

If a test peak is matched to a control peak, its contribution to the score of the experiment is:

$$c = \frac{\alpha\Delta H + \beta\Delta N}{S} \qquad \textbf{Eq. 12}$$

where $\alpha$ and $\beta$ are the weights of the $^1$H and $^{15}$N chemical shifts, respectively.

A control peak that does not match any test peak contributes a penalty *c'* (default is 0.6) to the score. The score of an experiment is

hence the sum of the contributions of all matched and unmatched control peaks:

$$c = \sum c + \sum c' \qquad \textbf{Eq. 13}$$

Sometimes some of the test peaks do not match any control peak. Although most unmatched test peaks are noise, any that are legitimate peaks can, if you choose to, contribute to the score such that:

$$c = \sum c + \sum c' + \sum c'' \qquad \textbf{Eq. 14}$$

where $c''$ is the penalty for each unmatched test peak.

Normally, the contribution of peaks with too-small displacements are ignored, namely:

$$\text{If } \Delta H < min\Delta H \text{ and } \Delta N < min\Delta N \text{ then } c = 0 \qquad \textbf{Eq. 15}$$

where $min\Delta H$ and $min\Delta N$ are the lower limits of the $^{1}H$ and $^{15}N$ displacements, respectively.

In practice you can change the weight of some control peaks to hide or highlight their contributions. In this way the reported score becomes:

$$c^* = \sum Wc + \sum Wc' + \sum c'' \qquad \textbf{Eq. 16}$$

where $W$ is the weight of each control peak. Since test peaks are automatically picked on the fly, you cannot specify weights for them. Furthermore, although they contribute to the score, unmatched test peaks are not stored in the database as are other peaks.

Except in the simplest case, Eq. 5-Eq. 7 do not guarantee an unambiguous mapping of all test peaks to control peaks. Based on the assumptions that only a small portion of the peaks are displaced and

that most peaks retain their locations and shapes in the test spectrum, Autoscreen assumes that the mapping that has a global-minimum value *C* as defined in Eq. 13 is the true mapping.

## Intelligent peak picking in test spectra

Peak picking in test spectra is done on the fly during scoring; i.e., you have no chance to optimize the picked peaks, hence the quality of the picked peaks is one of the key factors for successful scoring. Although most of the peak-picking parameters should remain identical to those used for peak picking in the control spectrum, the threshold may need to be adjusted for differences in sample concentrations.

Autoscreen first determines a peak-picking area, which is the minimum rectangle that includes all the control peaks plus margins of $max\Delta H$ and $max\Delta N$ in the D1 and D2 dimensions, respectively.

Autoscreen provides three ways of defining the threshold for peak picking: **Automatic**, **Control**, and **Define**. If the default setting (**Automatic**) is selected, an initial threshold is automatically calculated, and then the threshold is raised or lowered according to the following rules for each test spectrum:

1. If the number of test peaks is fewer than 1.1 times the number of control peaks, lower the threshold.
2. Otherwise, for each control peak, the test peaks that meet the criteria of Eq. 5-Eq. 7 are taken as candidate matches to the control peak. If over 10% of control peaks do not have a candidate match, lower the threshold.
3. Otherwise, if the number of test peaks is greater than 1.5 times the number of control peaks and each control peak has more than 3.0 candidate matches on average, raise the threshold.

No more than seven adjustment loops are performed.

If **Control** or **Define** is selected for the threshold method, the same threshold as that used for the control spectrum or a user-defined threshold is applied to all test spectra, respectively, and no automatic adjustment of thresholds is done.

## Mapping control peaks to test peaks

For each control peak, the contributions of those test peaks that meet the criteria of Eq. 5-Eq. 7 are calculated using Eq. 12. These test peaks are sorted in ascending order of *c*, and the first four are retained as candidate matches. At this stage, similarities in peak heights ($S_I$) are not used in Eq. 8, since the intensity level may be different between the control and test spectra.

To normalize the intensity level of the test spectrum, an intensity ratio IR is calculated as follows:

$$IR = \frac{\sum I^C}{\sum I^T} \qquad \textbf{Eq. 17}$$

where $\Sigma I^C$ is the sum of peak heights for all control peaks that have at least one candidate match, and $\Sigma I^T$ is the sum of peak heights of the first candidate matches to all control peaks. The peak heights of the test peaks are then normalized by multiplying by *IR*.

Next, the list of candidate matches is updated by including the peak intensity in

$I^C$ is the sum of peak heights for all control peaks that have at least one candidate match, and $I^T$ is the sum of peak heights of the first candidate matches to all control peaks. The peak heights of the test peaks are then normalized by multiplying by *IR*.

Next, the list of candidate matches is updated by including the peak intensity in Eq. 8.

The remaining task is to choose one or no test peak for each control peak, so that the sum of contributions (Eq. 13) is minimal. Some control peaks may not match any test peak, and such unmatched control peaks also contribute to the score.

Autoscreen provides two alternative methods for searching the mapping to find a global minimum for Eq. 13.

The first is a depth-first tree-search method. This deterministic method, in principle, should always find the true minimum, but in practice it may be too slow for complex spectra. Several heuristic

methods are used to enhance its efficiency, so that, for normal spectra, satisfactory results are obtained in a few seconds. By default, the total CPU time is limited to 10 seconds although you can change this.

Another method is simulated annealing, a stochastic method that efficiently lowers the score (or energy) to or close to the global minimum. This method is recommended for very complicated spectra where tree searching does not give satisfactory results in a reasonable amount of CPU time.

## Unmatched peaks

There are many possible reasons for a control peak not to match any test peak. For example, if the upper limits for peak matching are set to too low a value, a peak that is shifted very far will not find its match. More often, two well separated peaks in the control spectrum become overlapped in the test spectrum. When this happens, only one test peak is picked, leading to one of the control peaks' being unmatched.

To unravel the overlapping test peaks in the latter case, Autoscreen fits such unmatched control peaks to the test spectrum using a local optimization method (see "Peaks / Optimize" on page 239 in Chapter 4, *Processing, Visualization, and Analysis Interface (1D/2D/ND)*. All unmatched control peaks, together with those neighboring control peaks that lie closer than four times their peak width to the unmatched control peaks, are fitted to the test spectrum with only the peak centers optimized. This process is iterated up to three times or until the change in the penalty is less than 5%. Next, the widths of these peaks are optimized for one iteration.

For each optimized peak, its equivalent peak height in the test spectrum is normalized using the intensity ratio, as for other test peaks. Then the optimized peak is matched to the original control peak to see if the conditions in Eq. 5-Eq. 7 are met.If this is successful, the contribution is calculated using Eq. 12. Otherwise, the control peak remains unmatched and its penalty is added to the score according to Eq. 13.

The fitting of unmatched control peaks to the test peaks is done only when the percentage of unmatched control peaks is below a certain threshold (default = 40%). If too many control peaks are not matched, this usually means that the spectrum is corrupted, and the

fitting is not done. A corrupt spectrum usually has a score significantly higher than a normal spectrum, because of the contribution of unmatched control and test peaks.

Test peaks that are not matched to a control peak are usually noise, but they can also be real peaks corresponding to unwanted control peaks (e.g., the side-chain amide peaks are usually excluded from the control peaks), or to a control peak that was not noticed in the control spectrum because of peak overlap or other reasons. A legitimate test peak that was not observed in the control spectrum but that is picked in the test spectrum should contribute to the score.

Therefore, from all the unmatched test peaks, Autoscreen identifies legitimate test peaks based on their chemical shifts and peak shapes. If you choose to use the chemical-shift constraint, then only those peaks that are displaced less than $max\Delta H$ and $max\Delta N$ (in the $^{1}H$ and $^{15}N$ dimensions, respectively) with respect to their closest control peaks are considered. Next, the peak widths and heights of all the matched test peaks are analyzed statistically. At this point, an unmatched test peak is considered legitimate if its peak widths ($\omega_H$ and $\omega_N$) and peak height (I) meet the following criteria:

$$\bar{\omega}_H - n\sigma_H \le \omega_H \le \bar{\omega}_H + n\sigma_H \qquad \textbf{Eq. 18}$$

$$\bar{\omega}_N - n\sigma_N \le \omega_N \le \bar{\omega}_N + n\sigma_N \qquad \textbf{Eq. 19}$$

$$\bar{I} - n\sigma_I \le I \le \bar{I} + n\sigma_I \qquad \textbf{Eq. 20}$$

where $\bar{\omega}_H$, $\bar{\omega}_N$, and $\bar{I}$ are the average values of the peak width along the $^{1}H$ dimension, the peak width along the $^{15}N$ dimension, and the peak height, respectively; $\sigma_H$, $\sigma_N$, and $\sigma_I$ are the standard deviations of the peak width along the $^{1}H$ dimension, the peak width along the $^{15}N$ dimension, and the peak height, respectively; and $n$ is a user-defined coefficient (default value is 2.0).

By default, each unmatched test peak contributes a penalty of 0.20 to the score, though you can always change this parameter. This penalty is set smaller than that for an unmatched control peak, because you usually have a better chance to refine the control peaks based on the assignment. The test peaks, on the other hand, are automatically picked so are usually less reliable.

Presently, unmatched test peaks are only reported to you in the text window and in a log file, but not saved in the database like other peaks.

## Table of scoring parameters

The **Autoscreen/Setup Scoring** menu item allows you to change or verify the parameters used for scoring. Table 2 correlates these parameter names with the symbols and terms mentioned in this chapter, for your convenience.

For more details about these parameters, please refer to “Processing and scoring menu items” on page 320 in Chapter 6, *Autoscreen User Interface*.

### Table 2 Meanings of Autoscreen scoring parameters

| Parameter | Options | Default | Symbol or meaning |
|---|---|---|---|
| ***Peak Displacement Limits (ppm)*** | | | |
| | **Control Peak Table** | **Use One, xpk:peaks** | Use the same set of control peaks for all test spectra, or prompt for different set for each test spectrum. And the peak table entity name when **Use One** is selected. |
| | **Test Peaks** | **Auto Pick, xpk:tpeaks** | Automatically pick test peaks, or prompt for a set of pre-picked test peaks for each test spectrum. And where to save the test peaks when **Auto Pick** is selected |
| | **D1 Minimum** | **0.02** | $min\Delta H$ in Eq. 15 |
| | **D1 Maximum** | **0.13** | $max\Delta H$ in Eq. 5 |
| | **D2 Minimum** | **0.2** | $min\Delta N$ in Eq. 15 |
| | **D2 Maximum** | **0.8** | $max\Delta N$ in Eq. 6 |
| ***Scale Factors*** | | | |
| | **D1** | **1.0** | $\alpha$ in Eq. 12 |
| | **D2** | **0.2** | $\beta$ in Eq. 12 |
| **Threshold Methods** | | **Automatic** | Method for determining peak-picking threshold |
| **Use Peak Widths** | | **Matching only** | Match peak width and/or include it in the score. |

**Table 2 Meanings of Autoscreen scoring parameters (Continued)**

| Parameter | Options | Default | Symbol or meaning |
|---|---|---|---|
| **Use Peak Heights** | | **Matching only** | Match peak intensity and/or include it in the score. |
| **Minimum Shape Similarity** | | **0.05** | *minS* in Eq. 7 |
| **Search Method** | | **Tree Search** | Method to search for the global minimum. Options are **Tree Search** and **Simulated Annealing**. |
| **CPU Time Limit** | | **10** | Maximum CPU time (seconds) to spend searching the global minimum. Used only when **Tree Search** is selected. |
| ***Unmatched Control Peaks*** | | | |
| | **Fit to Test** | **Yes** | Fit unmatched control peaks to test spectrum. |
| | **Maximum** | **40** | Upper limit of unmatched control peaks, above which the spectrum is regarded as corrupt and no fitting is done. |
| | **Penalty** | **0.6** | Penalty for an unmatched control peak (*c'* in Eq. 13) |
| ***Unmatched Test Peaks*** | | | |
| | **Selection** | **Close to Control Peaks** | Method for identifying unmatched test peaks. |
| | **Num.of RMSD** | **2** | *n* in Eq. 18-Eq. 20 |
| | **Penalty** | **0.2** | Penalty for an unmatched test peak (*c"* in Eq. 16) |

# 2 Tasks

## Task: Importing data

FELIX 2002 can read seven specific data formats:

- Old FELIX format
- New FELIX format
- ASCII format
- FELIX for Windows FID format
- Bruker FID or SER format (from AMX and newer models; parameter files must be in ASCII format)
- Varian FID format
- JEOL ALPHA and LAMBDA.

All other spectrometer data must be converted to one of the first three formats before FELIX can access it. The new FELIX format is preferred. Unlike the old format the new format can accurately and consistently move data between systems with different byte ordering. The ASCII format is generally used only when no other method of data conversion is available.

### Transferring data

First, ensure that FELIX can access the spectrometer data on disk. One option is to mount the spectrometer data disks via a network connection so that they are accessible to the workstation running FELIX. If this is not possible, transfer the data from the spectrometer computer to the FELIX workstation. This is usually accomplished by using ftp to transfer the data via Ethernet.

**Caution:** All of the data filters require that the spectrometer data files be kept in their native form. For example, the X32 data filter for Bruker data expects to see the same file and directory structure that exists on the spectrometer. The Bruker "data file" is actually a directory containing subdirectories with experimental and processed data files. When the spectrometer data is transferred to the FELIX workstation, this entire directory structure must remain unchanged.

One way to assure that the directory structure remains unchanged is to first make a .tar file of the desired data directories on the spectrometer. This .tar file can then be transferred to the FELIX workstation using ftp in binary mode. Once the data have been transferred, you can untar the data to use it in FELIX.

**Caution:** Transfer the data in binary mode to avoid corrupting the data.

## Converting processed data

Use a conversion filter to convert processed data directly to the FELIX matrix format. Use the **File/Convert/Matrix** command to perform this task. Currently, you can convert NMRPipe, NMRCompass, Bruker 2rr and 3rrr files, and Varian data. The resulting FELIX matrix file can then be directly accessed by the FELIX program.

## Converting Varian spectra

You can import processed Varian spectral data (that is, phase files) into FELIX. These phase files can be for 2D or 3D data sets. Using a macro called sv2dpf, you create a parameter (.par) file corresponding to 2D processed data sets. Similarly, using a macro called sv3dpf, you create a parameter (*.par*) file corresponding to 3D processed data sets. Each of these macros is provided below, and both *must* be run by VNMR (that is, Varian processing software). These two macros are located in the FELIX gifts\VARIAN\matrix\ directory and must be transferred to the VNMR macro library directory before they can be run by VNMR.

When you prompt FELIX to import a Varian spectrum and provide the filename for that spectrum, FELIX searches for a corresponding

.par file and converts the data in that file into FELIX matrix format. If FELIX does not find a .par file with the same root name as the spectrum filename that you specified, FELIX prompts you to enter the spectral parameters manually.

If you are importing a 2D spectrum, you must rename the phase file to a name that corresponds to the root name of the .par file. For example, if you use the sv2dpf macro to create a file called test.par, you must rename the phase file to test.

### *The sv2dpf macro*

```
"sv2dpf - save 2d phasefile"
"usage: sv2dpf (basename)"

" $# is the number of input arguments. It must be greater than zero. "

if ($# < 1) then
write ('error','usage:sv2d(filename)')
return
endif

" If the file already exists, delete it. "

exists ($1,'file') : $e
if $e then
rm($1)
endif

"Flush the phasefile completely from memory"

write ('line3', 'saving raw data to disk')
trace='f1' dcon flush
trace='f2' dcon flush

" Create the text file with suffix .par containing parameters of the data"

$parfile=$1+'.par'
write ('line3', 'saving parameters to disk')
write ('reset', $parfile)
write ('file', $parfile, '%d %d %d',ni,np,0)
write ('file', $parfile, '%10.1f %10.1f %10.1f',sw1,sw,0)
if (tn=dn) then
$frq1=sfrq
else
$frq1=dfrq
endif
write ('file', $parfile, '%10.1f %10.1f %10.1f', $frq1, sfrq, 0)

" We are finished writing the parameters"

write ('line3', 'Data written to %s and %s', $1, $parfile)
```

### *The sv3dpf macro*

You use the sv3dpf macro to create phase files corresponding to different planes (f1f2, f2f3, f3f1) in a 3D data set. You generate the .par file corresponding to each plane by using the sv2dpf macro (described above). The sv3dpf macro is shown below:

```
"sv3dpf - save 3d phasefile"
" usage: sv3dpf(basefilename,orientation)"
"  where basefilename is the basic filename; to that will be appended"
"          '.orientation' to indicate the orientation and"
"          '.xxx' to indicate the index"
"        orientation is 'f1f2','f1f3', or 'f2f3'"

if ($#<2) then write('error','improper arguments') return endif
if ($2='f1f2') or ($2='all') then "save f1f2 data"
$i=1
repeat
select('f1f2',$i)
trace='f1' dcon flush
trace='f2' dcon flush
format($i,0,0):n1 length(n1):$len
if ($len=2) then n1='0'+n1 endif
if ($len=1) then n1='00'+n1 endif
copy(curexp+'/datdir/phasefile',$1+'.f1f2.'+n1)
$i=$i+1
until $i>(fn/2)
endif

if ($2='f1f3') or ($2='all') then "save f1f3 data"
$i=1
repeat
select('f1f3',$i)
trace='f1' dcon flush
trace='f3' dcon flush
format($i,0,0):n1 length(n1):$len
if ($len=2) then n1='0'+n1 endif
if ($len=1) then n1='00'+n1 endif
copy(curexp+'/datdir/phasefile',$1+'.f1f3.'+n1)
$i=$i+1
until $i>(fn2/2)
endif
if ($2='f2f3') or ($2='all') then "save f2f3 data"
$i=1
repeat
select('f2f3',$i)
trace='f2' dcon flush
trace='f3' dcon flush
format($i,0,0):n1 length(n1):$len
if ($len=2) then n1='0'+n1 endif
if ($len=1) then n1='00'+n1 endif
copy(curexp+'/datdir/phasefile',$1+'.f2f3.'+n1)
$i=$i+1
until $i>(fn1/2)
endif
```

### *The .par file*

As described above, each macro creates a parameter (*.par*) file. The name of this file must correspond to the name of the spectrum that you want to import. The table below lists the contents of the .par file.

| Line number | Description |
|---|---|
| 1 | Size of data rows, columns, and tiers. |
| 2 | Sweep width along rows, columns, and tiers. |
| 3 | Spectrometer frequency along rows, columns, and tiers. |
| 4 | Flag to define spectral zooming and the range of ppm along the tier dimension for a 3D data set. |

You need not have a fourth line in the .par file to import a Varian spectrum. However, you can add this line if FELIX displays flags that prompt you to define spectral zooming or the range of ppm along the tier dimension for 3D data.

The format of the .par file is shown below. According to VNMR convention, the order of the values in the first three lines of the .par file varies, depending on which set of planes is being read. That is:

1. For sv3d-generated f1/f3 planes (where f2 is the tier dimension):

| | f3 | f1 | f2 |
|---|---|---|---|
| 1st line: | fn/2 | fn1/2 | fn2/2 |
| 2nd line: | sw | sw1 | sw2 |
| 3rd line: | sfrq | (dfrq) | (dfrq2) |

2. For sv3d-generated f2/f3 planes (where f1 is the tier dimension):

| | f3 | f2 | f1 |
|---|---|---|---|
| 1st line: | fn/2 | fn2/2 | fn1/2 |
| 2nd line: | sw | sw2 | sw1 |
| 3rd line: | sfrq | (dfrq2) | (dfrq) |

3. For sv3d-generated f1/f2 planes (where f3 is the tier dimension):

| | f2 | f1 | f3 |
|---|---|---|---|
| 1st line: | fn2/2 | fn1/2 | fn/2 |
| 2nd line: | sw2 | sw1 | sw |
| 3rd line: | (dfrq2) | (dfrq) | sfrq |

A space is used as a delimiter in all the examples.

**Caution:** Because only a few pulse sequences use the second and third channel for the f1 and f2, respectively, the location of dfrq and dfrq2 will rarely be in the correct position. Therefore, you will need to add the correct spectrometer frequency. You can use the **axis** parameter in VNMR to determine the correct spectrometer frequency for all three axes

## Task: Modifying the quick-access (context) menu

FELIX includes a quick-access (also called *context*) menu that is invoked by the secondary mouse button. This menu is intended to make it easy to customize the FELIX interface. (It is possible to rewrite the whole user interface; however, just change this menu and macro and make your favorite commands easily accessible.) The necessary changes are described here.

Locate *mouser.mot* in the **macros\mot** folder under your installation directory. (See Starting Felix, How to Use This Guide in this book for more details about the installation directory). This macro defines the items to be displayed in the context menu, and the callback macros (and in some cases, parameters) when the items are clicked.

```
mouse.mot
item "Draw Frequency"         D NULL NULL "ex drawfreqs 1" NULL NULL
item "Add Frequencies"        A NULL NULL "ex drawfreqs 2" NULL NULL
item "Clear Frequencies"      F NULL NULL "ex drawfreqs 3" NULL NULL
Separator Separator           NULL NULL NULL NULL NULL NULL
item "Correlated Cursors"     C NULL NULL "ex cursorval 2" NULL NULL
Separator Separator           NULL NULL NULL NULL NULL NULL
item "Pick One Peak"          O NULL NULL "ex pickone" NULL NULL
item "Edit One Peak           "E NULL NULL "ex editpeak" NULL NULL
item "Merge One Multiplet     "M NULL NULL "ex mergeone" NULL NULL
item "Remove One Peak"        n NULL NULL "ex removeone" NULL NULL
```

The format of this file follows that for the general FELIX menu files. For example, the first line displays a 'Draw Frequency' item in the context menu, and when it is clicked FELIX executes a macro drawfreqs with the first parameters as 1. For more details about the for-

mat, see “Changing the menubar interface” on page 50, Chapter 5, *Menus and Control Panels* in *FELIX Command Language Reference.*

# Task: Working with 1D data

FELIX provides a comprehensive set of tools for processing, displaying, and analyzing NMR data. This section outlines ways to use the menu interface for processing and analyzing one-dimensional (1D) NMR data. For information on converting data files into a format that can be read by FELIX, please see “Task: Importing data” on page 75, and Appendix , *Data Files.*

## Reading data files

FELIX reads two distinct file formats of its own. You must tell FELIX what format your data is in for it to be interpreted correctly.

- The ‘old’ FELIX data format (the format that is read using the **re** command) is the format that most of the current public-domain and commercial reformatting programs generate. Although this data format works perfectly well on any given computer, data of this type cannot always be transferred between different types of computers due to byte-order differences.
- The new FELIX format was designed to provide increased portability for FELIX files (use the **rn** command to read these files). Files written in the new format can be transferred via ETHERNET without conversion.The built-in FELIX data filters create all new-format files.
- FELIX can also read native spectrometer data formats via a set of new commands:

  **rf** for reading FELIX for Windows FID files

  **rb** for reading Bruker FID or SER files

  **rv** for reading Varian FED files

  **rj** for reading JEOL ALPHA .nmf files and LAMBDA .nmfid files.

To read a data file, use the **File/Open** command. You can specify the data as either one-dimensional or *N*-dimensional, and as either Bruker, Varian, FELIX For Windows, JEOL, or FELIX old or new format.

If you define the data type as one-dimensional, the data file is explicitly closed before it is read, to reset the record pointer to the first record. When you are first working with *N*-dimensional data, FELIX reads the first record from the serial data file. Each subsequent reading (during the same FELIX session) increments the record pointer by one and reads the next data record.

*Sample task:*

As an example of reading in a sample data file, copy the 1D data file sample.dat from the **%ACCELRYS_FELIX%\tutorial** directory to your working directory.

Once you have copied the sample data file to your working directory, try reading it into FELIX using the **File/Open** command.

## Saving data

Sometimes, you might want to save the data to disk as a permanent record. For example, you can save a fully processed spectrum to a file so you can display it quickly without re-transformation. To do this, select the **File/Save As** command. Specify a file name and the data are saved in FELIX new format. If the file already exists, FELIX prompts you to **quit** or **overwrite** the file.

## Displaying 1D spectra

The most common way of manipulating displays is to use the **View** pulldown. **View** contains a set of commands designed to manipulate the display of 1D data. For example, to redraw the current workspace, select the **View/Plot** command.

By default, when FELIX initially reads and displays a new file, it draws all the data points (the entire spectral width). To display or work with an expanded region of the spectrum, you must select the expansion limits.

## Changing 1D limits

Use the **View/Limits/Set Limits** command to choose spectrum limits in real time. FELIX displays a crosshair cursor. Move this cursor to the region you want to expand, push and hold down the left button on your mouse, drag out the region to be expanded, and then release the mouse button.

Alternatively, you can use the icons—particularly the **Zoom** icon—to execute this task.

- To draw all the points in the spectrum, select the **View/Limits/Full Limits** command. All subsequent draws of your data will display the entire spectrum.
- To redisplay the last plot drawn, select the **View/Limits/Last Limits** command.
- You can use the keypad to navigate in a zoomed region—the **Keypad Help** icon shows all the options.
- The **View/Limits** pullright displays a menu with numerous additional choices for expanding your spectrum.

## Adjusting plot parameters

Use control panels to tailor the appearance of your data plots. FELIX shows the current settings for the plot appearance; you can change these settings. Access the control panels from the **Preference/Plot Parameters** command; or click the **Plot Parameters** icon.

- Access the PLOT PARAMETERS - BASIC control panel by selecting the **Preference/Plot Parameters** command or the **Plot Parameters** icon. FELIX displays the settings that control the appearance of the plotted data points. You can choose the plot color, the data scaling, the **Plot Type** (**Lines** or **Points**), and the **Plot Mode** (**Real**, **Imaginary**, or **Real/Imag**). Also, use the buttons to access successive control panels where you can modify **Scale**, **Tick**, **Place**, and **Stack** parameters.
- Click the **Scale** button to open the PLOT PARAMETERS - SCALE control panel. Here you can control the appearance of the non-data portions of the plot; that is, the axis units, the box around the

plot, the grid spacing, the annotations, the 2D contour level display, and the picked peaks.

- Click the **Place** button to open the PLOT PARAMETERS - PLACE control panel. Here you control the position and size of the plot within the FELIX window.

## 1D data buffers

FELIX allows and enhances processing and analysis of one-dimensional and *N*-dimensional spectra through the availability of additional addressable memory locations. These storage spaces are called buffers and may be used to temporarily save data. The data stored in the buffers may be as simple as a single data value or as complex as a protein spectrum.

The FELIX buffers are addressed by number: buffer 1, buffer 2, … buffer *n*. The size and number of buffers available is determined by the amount of memory configured for this use. Use the **Preference/Memory** command to change the program's memory allocation. For example, if four 1D buffers are defined, each having a buffer size of 16384 complex points, enough space can be reserved to let you work with the current data plus four buffers for data containing up to 16384 points.

Buffers are accessed with the **Tools/Buffers** command. For example, to store the current information to a buffer, select the **Tools/Buffers/Store Work to Buffer** command and enter the buffer number in the control panel. To visualize this information, you must change the stack depth to include the buffers that you want to visualize. The **Tools/Buffers** command contains many additional controls that allow you to manipulate the contents of the buffers.

## Adjusting stack and buffer display

The stacks and buffers are generally used for displaying multiple spectra simultaneously. The stack represents the range of buffers selected for display. Change the various stack parameters (**Depth**, **Order**, and **Overlap**) in the PLOT PARAMETERS - BASIC control panel. Access it by selecting the **Preference/Plot Parameters** command or the **Plot Parameters** icon.

- Use the **View/Plot** command to show all the data in the stack. This function draws the workspace first, followed by each item in the stack, from top to bottom or from bottom to top, based on the **Stack Order** parameter.
- Use the stack to display several spectra along the same axis. By default, the spectra are scaled equally to use the space available along the y-axis. Use the **Stack Overlap** parameter to set the display to no overlap (**0**), full overlap (**1**), or any fraction thereof. Use the **Stack Depth** parameter to control how many buffers in addition to the workspace are displayed when data are redrawn.
- The **Absolute Intensity** parameter also affects the appearance of the stack display. Use this parameter to set the scale factor for all data in the window to the same value, so that data intensities can be compared directly. When this parameter is set to **No**, all data in the stack are scaled to fill the available space. If this parameter is set to **Yes**, all data are scaled to the last spectrum drawn in the workspace. This switch is useful for displaying difference spectra, where the full spectrum and the difference can be compared on the same scale.

## Axes and referencing

FELIX can display a spectral axis with several types of labels. The default axis label is **points**, because reliable point positions do not change when the values for the spectrum width and spectrometer frequency change. Change axis with the **Preference/Plot Parameters** command in the **Scale** control panel. Choices for the **Axis Units** parameter are **None**, **Points**, **Hertz**, **PPM**, **Seconds**, and **1/cm**.

To reference a one-dimensional spectrum, select the **Preference/Reference** command, which asks you for the referencing information using the REFERENCE 1D DATA control panel. The **Spectral Frequency** and **Spectral Width** parameters must be set to the values of the spectrometer frequency in MHz and the spectrum width in Hz.

Enter the **Reference Point** value either by typing it in the box on the control panel or by clicking the **Cursor** button in the control panel. This prompts FELIX to display a vertical cursor; here you move the cursor to the desired reference point on the plot and then click the left mouse button.

At this time FELIX displays the REFERENCE 1D DATA control panel. Now enter the reference value in the appropriate box; for axis units of Hertz enter a reference value in the **Reference Hertz** box, for ppm units enter a reference value in the **Reference PPM** box. When you are finished, click **OK** to close the control panel and redisplay the data with the selected axis units.

## Finding data values in spectra

Use the **Measure/Cursor Position** command to obtain point numbers, ppm values, and corresponding data values. When you select the **Measure/Cursor Position** command, FELIX displays a vertical cursor, which updates the current axis position and data value when you move it with the mouse. The axis position is in axis-based units; if your axis is in points it tracks in points, if your axis is set to ppm it tracks in ppm. To quit the cursor-tracking mode, press <Esc>. The data value shown is the actual data value stored at that location in the workspace.

## Correlated cursors

When you display data in several frames at once, you can use correlated cursors multiple crosshair cursors) that track in more than one frame. Use the mouse to point to the same data-point values in all the frames simultaneously.

For example, if there are three frames, three crosshair cursors track the mouse motion. The center of each cursor falls on the same data point value in each frame. To activate correlated cursors, select the **Measure/Correlated Cursors** command. To find the cursor position and data-point value, click the mouse button. To quit the correlated-cursor mode, press <Esc>.

## Spectrum separation

You can calculate the separation between any two spectrum features in a display with the **Measure/Distance/Separation** command. Use a crosshair cursor to select two locations on the display. FELIX reports the separation in points, ppm, and Hertz. To quit this mode, press <Esc>.

## Zero-filling and removing DC offset

Zero-filling is a common manipulation performed on time-domain data. Select the **Process/Zero Fill** command to increase the number of points in the transformed spectrum, and thereby increase the spectrum's apparent digital resolution.

To remove any DC offset that may have occurred during data acquisition, raw FIDs are usually corrected prior to Fourier transformation, To do this, select the **Process/DC Offset** command. Select regular **BC** (baseline correction) or **DBC** (for Bruker oversampled data) and set a **Baseline Correction Fraction**. Use the **Baseline Correction Fraction** command to specify the fraction of the FID, starting from the right side, to be averaged to eliminate the DC offset.

## Linear prediction

Linear prediction estimates the value of a point based on the values of adjacent points; they can be used to replace corrupted values in an FID. Often, several of the first data-point values can be corrupted by instrumentation-induced artifacts. Therefore the spectrum can be noticeably improved when these corrupted point values are replaced with values estimated by linear prediction, where the estimated values are based on subsequent points—points of greater integrity.

A second application of linear prediction to NMR data is to extend an FID. This is useful for experiments in which the data collection stopped before the signals completely decayed; that is, the FID is truncated. Here, the data values of the FID are used to estimate new data values that are appended to the end of the FID.

### Process/Linear Predict First

Use the **Process/Linear Predict First** command to use linear prediction to replace data values at the beginning of the FID. FELIX displays a control panel; specify the following parameters for the prediction:

- **Points to use** (the points used to calculate the coefficients)

- **Number of Coefficients** (specify the number of LP coefficients that are calculated)
- **Number of Peaks** (for compatibility with existing macros, but is not used in the current LP implementation)
- **Points to use** (define the number of points used to calculate the LP coefficients)

The time taken to complete an LP calculation is determined in large part by the number of data points that are used to determine the LP coefficients. In general, the quality of the LP calculation also increases with the number of data points used in the calculation.

Therefore, you must pick a value for the **Points to use** that is large enough to produce accurate predicted data values but that does not unduly lengthen the time taken by the LP calculation. The value for the **Number of Coefficients** is generally set to be one-quarter to one-third of the **Points to use**.

## Process/Linear Predict Last

Use the **Process/Linear Predict Last** command to use linear prediction to replace data values at the end of the FID. You can also use it to extend the FID.

Conventional linear prediction works well when the data are being extended by a small fraction of the number of points. However, when the number of data points is being extended by a large fraction (e.g., doubling the number of points), the predicted data can contain signals with increasing amplitude due to the effect of noise on the predicted LP coefficients. The net effect can be an FID with *increasing* (as opposed to *decreasing*) amplitude as a function of time.

To prevent this you may use **Root Reflection** when predicting data in FELIX. **Root Reflection** ensures that the calculated frequency components decay as a function of time and thus more accurately reflect the correct physical nature of an FID. Using **Root Reflection** increases the time needed to perform the LP calculation, but the predicted points are more representative of a true FID. **Root Reflection** is essentially *required* when predicting large numbers of data points, to avoid having noise components with increasing amplitude dominate the predicted FID at longer time values.

## LP calculation methods

FELIX allows great flexibility in how the LP calculation is performed. By default, the **Process/Linear Predict Last** command is used for extending the data, but you may also specify exactly which points you would like to use in the LP calculation and which points you would like to predict.

You specify the method used to perform the LP calculation. The options include **Forward**, **Backward**, **Forward-Backward**, and **Mirror Image**.

- The **Forward** and **Backward** methods of linear prediction were used in previous versions of FELIX.
- The **Forward-Backward** technique (Zhu and Bax 1992) performs both a **Forward** and a **Backward** calculation on the data. FELIX then averages the resulting LP coefficients to produce a more accurate set of LP coefficients than either the **Forward** or **Backward** method can produce alone. The **Forward-Backward** method is very robust and generally applicable, and does not require any prior knowledge of the FID.
- **Mirror Image** linear prediction uses predicted data at negative time values in the calculation. This technique allows more LP coefficients to be calculated and hence is useful for severely truncated data.

  When you use the **Mirror Image** technique, you can increase the **Number of Coefficients** to between one-half and two-thirds the value of the **Points to use**.

  The **Mirror Image** technique requires prior knowledge of the phase of the data and nondecaying signals. Because of these restrictions, the **Mirror Image** technique is used primarily for severely truncated indirect dimensions of *N*-dimensional data sets when there is a need to calculate more LP coefficients than would be possible with the **Forward-Backward** method. The **Mirror Image** method includes options for data collected with no sampling delay and data collected with a one-half dwell time sampling delay.

## Solvent signal suppression

FELIX offers three methods for reducing the intensity of strong solvent signals: a linear-prediction-based algorithm, a convolution-based method, and a polynomial-based method. All three methods are accessible by using one of the options in the **Process/Solvent Suppression** command.

### Linear-prediction-based solvent reduction

To access linear-prediction-based solvent reduction, select **Linear Prediction** from the **Method** popup in the control panel that FELIX displays when you select the **Process/Solvent Suppression** command. This technique uses the LP algorithm to estimate and remove contributions from the most intense components in the spectrum. This technique works well when the intensity of the solvent signal to be removed is much greater than the other signals that are present.

### Convolution-based solvent reduction

To access convolution-based solvent reduction, select **Time-Domain Convolution** from the **Method** popup in the control panel that FELIX displays when you select the **Process/Solvent Suppression** command. In this technique, FELIX performs a convolution to first identity the lowest-frequency components that are present. Then, FELIX subtracts these components from the data. This technique is very useful if the solvent signals to remove are present at the carrier frequency.

### Polynomial-based solvent reduction

To access polynomial-based solvent reduction, select **Polynomial** from the **Method** popup in the control panel that appears when you select the **Process/Solvent Suppression** command. In this technique, FELIX fits a polynomial to the data. Then, FELIX subtracts the resulting function from the time domain data. This technique works best when the solvent resonance is close to zero frequency.

## Viewing and applying window functions

Time domain NMR data can be multiplied by window functions that perform digital filtering for the purpose of reducing noise or increasing spectral resolution. For example, the noise level in 1D NMR data can be attenuated by multiplying the FID by an exponential window function.

FELIX offers two methods for selecting window function parameters: you can enter the parameter values directly or you can adjust them interactively. The **Process/Window Function** command allows you to select a window function and adjust its parameters interactively while FELIX displays plots of both the window function and the product of the FID and window function. FELIX can also display the spectrum rather than the FID and window function product, while you adjust the window function parameters. This function is extremely useful in determining which window function is appropriate for your data.

You can also explicitly specify a window function and its parameters with this command. When you know exactly what window function you want to use, as well as its parameters, this action lets you apply it quickly and precisely.

### Window function descriptions

**Sinebell**, **Sinebell squared**, **Skewed sinebell**, and **Skewed sinebell squared** windows are a useful (but potentially dangerous) family of apodization functions. These sinebell windows can be shaped in many different ways, depending on their size, phase, and skew.

- The basic sinebell window illustrates the sine function from zero to π, mapped onto the number of points in the FID. If a spectrum is to be integrated following application of a sinebell window, a phase shift of 90⌋ is the only honest sinebell to use, because the first point of the FID is not changed.
- The sine squared window has the advantage that it approaches zero smoothly, whereas the sinebell does not. Sinebell squared windows shifted 90⌋ are useful for avoiding truncation effects when zero filling badly truncated data.

- The skewed sinebells are primarily useful for adjusting the line shapes of magnitude and power spectra when phase information has been lost.

## Exponential linebroadening

*Exponential linebroadening* is the most commonly used window function. This function starts out equal to one at the first point and decays at the rate of the specified FID. Because the value of the first point in the FID is not changed, exponential windows do not alter integral intensities and are a good window to use if integrals are to be measured. Exponential windows allow you to trade line width for signal to noise but preserve the Lorentzian line shape.

> **Caution:** If you use an exponential window to reduce noise, be aware that the lines in the spectrum still have Lorentzian shapes but no longer have natural line widths. For more detailed information, please refer to the **em** command in the FELIX *Command Language Reference Guide*.

| | |
|---|---|
| lbroad | (slider) Adjusts the line-broadening parameter for the exponential. |

## Gaussian linebroadening

*Gaussian linebroadening* is another popular window that changes not only the line width, but also the line shape. Gauss / Lorentz multiplication modifies the value of the first point of the FID and hence the value of the integral. Gauss / Lorentz is commonly used for resolution enhancement and changes the line shape to be partly Gaussian. Gaussian lines have narrow tails and yield a nicer-looking spectrum. This window is appropriate for cosmetic resolution enhancement, but the line widths and line shapes are no longer natural. Gaussian multiplication also alters the integral of spectral lines and differentially reduces the integral of broad lines with respect to narrow lines.

> Although a spectrum with Gaussian line shapes looks great, use caution if you attempt to integrate it, since the integral is affected by the line shape. For more detailed information, please refer to the **gm** command in the FELIX *Command Language Reference Guide*.

| | |
|---|---|
| lbroad | (slider) Adjusts the line broadening parameter for the exponential. |
| gbroad | (slider) Adjusts the Gaussian parameter for the exponential. |

### Kaiser

The *Kaiser* window is a function by Hamming (1989). This window is useful for apodizing data that are truncated. For more detailed information, please refer to the **kw** command in the FELIX *Command Language Reference Guide*.

| | |
|---|---|
| wsize | (slider) Adjusts the number of data points for the window function. |
| alpha | (slider) Adjusts the alpha parameter of the Kaiser window. |

### Trapezoidal

The *Trapezoidal* function multiplies the data in the workspace by a window that rises from zero at the first point to one at <p1>, is equal to one from <p1> to <p2>, and falls to zero from <p2> to <p3>. For more detailed information, please refer to the **tm** command in the FELIX *Command Language Reference Guide*.

| | |
|---|---|
| p1 | (slider) Adjusts the first point of the trapezoid. |
| p2 | (slider) Adjusts the second point of the trapezoid. |
| p3 | (slider) Adjusts the third point of the trapezoid. |

## Fourier transforms

The advantages of applying a pulse to the nuclear spins and collecting the resulting transient—advantages that range from improving the signal-to-noise ratio to making possible forbidden detection of multiple quantum coherence—have made this the predominant NMR technique. The Fourier integral transform is central to modern

NMR data processing because it transforms the transient from the time domain to the frequency domain, thereby yielding a spectrum.

Accordingly, FELIX provides a battery of Fourier integral transforms that are accessed by selecting the **Process/Transform** command, which opens a control panel containing a list of the integral transforms that FELIX can perform on the data in the work space. Five Fourier transforms and a Hilbert transform are available. The most commonly used transform method are listed below.

- The **Complex FFT** method applies a complex Fourier transform to the data in the work space. For this transform, the data must be true complex data, characterized by simultaneous sample and conversion of the real and imaginary signals. This is the most common of the Fourier transform methods and is used for most data.
- The **Bruker FFT** method performs a complex Fourier transform on complex data that is unique to some Bruker spectrometers. These spectrometers cannot sample and convert the real and imaginary signals simultaneously; instead, they collect the real and imaginary signals alternately. This is reflected in the acqus file when the AQ_mod parameter is set to 2. If your data were collected in this mode, you must use the **Bruker FFT** method.
- The **Oversampled FFT** method performs a complex Fourier transform on digitally oversampled data collected on Bruker DMX series and newer spectrometers. If your data were collected using digital oversampling, you should use this method for the transform.

## Phasing

After Fourier transformation, a spectrum frequently appears to be out of phase; that is, the resonance lines appear to be a mixture of absorptive and dispersive shapes. This is due to several factors, including finite pulse lengths, acquisition delays, and analog filter response. NMR spectra can be phase-corrected after transformation by multiplying each datapoint value pair by a phase factor.

FELIX has three ways to apply phase correction to a spectrum. To access these, select the **Process/Phase Correction** menu item. The automatic phase-correction algorithms include the PAMPAS and

APSL methods for spectra with non-split peaks, such as decoupled $^{13}C$ and DEPT spectra, an algorithm based on peak integration for general in-phase 1D spectra, and a basic algorithm intended for common proton spectra. In addition to the automatic phase-correction routine, FELIX provides an interactive real-time phase routine that is easy to use. FELIX also allows you to enter the zero- and first-order phase parameters explicitly.

## Correcting baseline distortions

An NMR spectrum can exhibit substantial baseline distortions caused by non-ideal experimental conditions. Such distortions can interfere with analysis of the spectrum (for example peak picking and integral calculation), so they must be minimized. Fortunately, most baseline distortions can be minimized easily, by first identifying a set of points on the spectrum that are free of peaks. These points are called *baseline points*. Next, a smoothly varying function is fitted to the set of baseline points. This function is expected to closely approximate the baseline distortion. Finally, at each point, the value of the smoothly varying function is subtracted from the data value in the work space, thereby removing the baseline distortion from the spectrum.

With the **Process/Baseline Correction** command, you can either add and delete baseline points or you can use different options to correct the baseline.

## Baseline point entities and files

FELIX uses an integrated database for storing spectrum information, including the identities of selected baseline points. By default, the name of the baseline point entity is bas:baseline; however you can change this name. In fact, if you want to retain the baseline points stored in the current entity while you pick a new set of baseline points, you *must* change the entity name.

- To change the name of the current baseline point entity, select the **Preference/Table** command. To change the name, enter the name of a new or existing baseline point entity and click **OK**.
- To view the contents of the baseline point entity, select the name of the entity in the **Edit/Table** command. This creates a table dis-

play of the baseline point entity contents in a separate window. You may add or delete baseline points on the spreadsheet.

- To save a baseline point entity in an ASCII file, select the **File/Export** command. FELIX prompts you to specify an output file name, as well as the name of the baseline entity to save. The ASCII file is written to the current directory or to the FELIX text directory, depending on how your directory structure is configured.
- To read a baseline point ASCII file may into a baseline point entity, select the **File/Import** command. FELIX prompts you to specify an input file name as well as an entity name.

## Spectrum display for baseline correction

When you are correcting the baselines of spectra, you may need to test several sets of baseline correction points and functions before a spectrum can be satisfactorily corrected. This is especially true if it is difficult to define baseline points due to spectral crowding.

Therefore, before applying any baseline-correction function, select the **File/Save As** command and save your spectrum. Later, if you are not satisfied with the results of any given baseline-correction function and its application to your data, select the **File/Open** command to read your saved spectral data in again.

## Adding and deleting baseline points

Select the **Process/Baseline Correction** command to open a control panel containing the available choices for adding and deleting baseline points. First you have to toggle **on** the **Baseline Points** radio button.

### To define baseline points

To define the baseline points, select the **Auto Pick Points** or **Auto Pick Points w/FLATT** option from the popup. FELIX generates a list of baseline points. FELIX displays markers for each baseline point picked in the spectrum at the bottom of the current spectrum.

### To add baseline points singly

To add baseline points singly, use the **Pick Points Via Cursor** option. Move the crosshair cursor to baseline points, and click them with the left mouse button. To quit this mode, click outside of the spectrum. If you want, you may add baseline points explicitly using point numbers with the **Manual Pick Points** option.

### To modify the list of baseline points

If you make a mistake while selecting individual baseline points or if you want to modify the current list of baseline points, you can delete a region of points. Select the **Delete Points in Region** option in the **Process/Baseline Correction** command's control panel to create a small crosshair cursor. Then drag out a region of baseline points to delete.

### To delete all baseline points

To delete *all* the baseline points, select the **Delete All Points** option. This deletes the current baseline points entity from the database: it requires confirmation via a dialog box.

## Applying baseline-correction functions

Once the baseline points are defined, you can choose a baseline-correction algorithm.

### Cubic spline algorithm

The cubic spline algorithm generates a baseline that passes exactly through each baseline point. To apply it, set the **Baseline Correction** option to **Cubic Spline** in the control panel for the **Process/Baseline Correction** command, A cubic spline may yield a kinked baseline if the defined baseline data points are close together and noisy.

### Polynomial baseline correction

The polynomial baseline correction algorithm generates smoother baseline correction functions from baseline points. To apply this baseline correction, set the **Baseline Correction** option to **Polynomial** in the control panel for the **Process/Baseline Correction** com-

mand. The polynomial correction differs from the cubic spline correction algorithm in that the baseline does not necessarily pass exactly through each baseline point, but a best fit is calculated.

## Real-time baseline correction

The FELIX real-time baseline correction feature lets you adjust the coefficients of a polynomial baseline function while displaying the resulting baseline function superimposed on the baseline-corrected spectrum. To access the real-time baseline correction feature, set the **Baseline Correction** option to **Real-Time Polynomial** in the control panel for the **Process/Baseline Correction** command.

## Automatic baseline flattening

The baseline-correction methods described above require that a set of baseline points already exist. In contrast, the three baseline correction methods discussed below do not need pre-defined baseline points. Instead, they alone determine what constitutes the baseline.

## Automatic baseline correction

One of the baseline-correction functions supported by FELIX that does not require explicit baseline points is applied by setting the **Baseline Correction** option to **Automatic w/abl** in the control panel for the **Process/Baseline Correction** command. FELIX selects noise points and performs a baseline correction for each point. The method involves the DC convolution of baseline points with a moving average, while applying a straight-line correction to non-baseline intervals (Dietrich et al. 1991).

## FLATT baseline-correction

FELIX also performs the FLATT baseline-correction algorithm (Güntert and Wüthrich 1992), which finds baseline segments in the spectrum and uses a linear least-squares solution to fit a truncated Fourier series to these points. To use FLATT, set the **Baseline Correction** option to **Automatic w/FLATT** in the control panel for the **Process/Baseline Correction** command.

### FaceLift baseline correction

The third baseline-correction function that does not require explicit baseline points is based on FaceLift, a technique introduced by Chylla and Markley (1993). The FaceLift algorithm automatically identifies base points from peak signals. To use FaceLift, set the **Baseline Correction** option to **Automatic w/FaceLift** in the control panel for the **Process/Baseline Correction** command.

## Miscellaneous work tools

FELIX contains several menu items that affect frequency domain spectra in the workspace. Although most of these controls are directly related to the transformation of multidimensional spectra, several of them affect the processing of one-dimensional data. To access these tools, select the **Tools** pulldown.

# Task: 1D peak picking and integration

Picking 1D peaks or resonances within FELIX is performed using the menu items within the **Peaks** pulldown.

## 1D peak entities and files

FELIX uses an integrated database for storing spectrum information, including the identities of picked peaks. By default, the name of the 1D peaks entity is **pic:1d_picks**; however, you can change this name. In fact, if you want to retain the 1D picked peaks stored in the current entity while you pick a different set of peaks, you *must* change the entity name.

To change or view the name of the current 1D peaks entity, select the **Preference/Pick Parameters** command. FELIX displays a control panel with the name of the current 1D peaks entity displayed in a the **Peak Pick Table** box. To change the name, enter the name of a new or existing 1D peaks entity and click **OK**.

To view the contents of the 1D peaks entity, select the **Edit/Peaks** command. FELIX creates a spreadsheet display of the 1D peaks

entity contents. You may add or delete 1D peaks on the spreadsheet, which also changes the information in the database.

To save a 1D peaks entity in an ASCII file, select the **File/Export/Peaks** command. FELIX prompts you to specify an output file name, as well as the name of the entity to save. FELIX then writes the ASCII file to the current directory or the text subdirectory, depending on how the FELIX directory structure is configured.

To read a1D peaks ASCII file into a 1D peaks entity, select the **File/Import/Peaks** command. FELIX prompts you to specify an input file name as well as an entity name.

## Working with picked peaks

You must define the threshold value before picking peaks. To do this, select the **Preference/Pick Parameters** command, set the **Threshold Value** option to **Cursor**, and click **OK**. Move the cursor so that the horizontal half-crosshair is located at the level of the smallest peak you want to pick, and then click the left mouse button. FELIX displays a dialog box with the newly set threshold. To accept it, click **OK**.

### Selecting peaks

To automatically select peaks in the current display, select the **Peaks/Pick All** command.

To select peaks in a sub region of your display, select the **Peaks/Pick Region** command. This creates a small crosshair cursor that you may drag to select a picking region.

To select one peak at a time, use the **Peaks/Pick One** command.

Use the **Preference/Pick Parameters** command to use a control panel listing those parameters that affect 1D peak picking. You can inspect and set threshold values here. To control the displayed names for picked peaks, use the **Peak Pick Units** parameter. All peaks are displayed in **Points**, **PPM**, **Hertz**, or **None**. To display peak names on the peaks, set the **Peak Pick Units** parameter to **Assignment**.

Three different peak-selection modes are provided. They are:

- Positive peaks only

- Negative peaks only
- Positive and negative peaks together

You can also vary the style of the peak markers, specifying arrowheads only, lines only, or lines with arrowheads.

## Deleting picked peaks

There are three options for deleting picked peaks.

- Use the **Peaks/Remove All** command to delete the 1D pick entity and the peak table. This option removes all picked resonance information.
- Use the **Peaks/Remove Region** command to delete peaks in a region. Select the region by dragging the small crosshair cursor.
- You can delete peaks one-by-one by selecting the **Peaks/Remove One** command and clicking the peaks.

## 1D line fitting

FELIX offers a powerful line-fitting interface for deconvolution of complex spectra into individual peaks that are described by an analytic function of intensity, linewidth, and frequency. These functions allow precise integration of peaks individually.

To access the interface and display utilities, select the **Peaks/Optimize** command. This gives you access to the 1D line fitting function and other functions that display the actual data, the synthetic data, or residual data—either separately or all together in an overlay.

To fit a spectrum, select the **Optimize** option in the control panel associated with the **Peaks/Optimize** command.

> **Remember:** The spectrum's baseline must be flat in order for you to obtain meaningful optimization results. So, before you calculate integrals, be sure to perform baseline correction on your spectrum.

# Integration

Peak integration of a spectrum gives information about the relative number of spin species. Accurate integration is an important part of 1D data analysis. The integration options are accessed with the **Measure/Integral/Volume** command.

> **Remember:** The spectrum's baseline must be flat in order for you to obtain an accurate integral. So, before you calculate integrals, be sure to perform baseline correction on your spectrum.

FELIX allows you to integrate the entire spectrum either as a single integral or in shorter segments. To integrate the entire spectrum, select the **View/Draw Integrals** command. However, if current segments are defined, the integrals for each segments are displayed separately by default. You can select the **Measure/Integral/Volume** command to add, remove segments, change the display parameters, or to normalize the integral values. Options are displayed in the control panel upon selecting **Measure/Integral/Volume**.

## Segment entities and integral files

FELIX uses an integrated database for storing spectrum information, including the identities of selected integral segments. By default, the name of the integral segment entity is **seg:segments**; however, you can change this name. In fact, if you want to retain the integral segments stored in the current entity while you pick a new set of integral segments, you *must* change the entity name.

To change the name of the current integral segment entity, type a name under **Use Database Entity** option in the control panel belonging to **Measure/Integral/Volume**. Or, select the **Preference/Table** command. If you want to change the name, set the **Table Type** parameter to **Integral Segments**, enter the name of a new or existing integral segment entity, and click **OK**.

To view the contents of the integral segment entity, select the **Edit/Table** command and choose the segment entity. This creates a spreadsheet display of the integral segment entity contents. You may add or delete integral segments on the spreadsheet.

At times you may want to save an integral segment entity in an ASCII file. Select the **File**/**Export/Table** command. FELIX prompts you to specify the name of the entity (table) to save, as well as for an output file name. The ASCII file will be written to the directory selected in the file selection control panel.

An integral segment ASCII file may be read into an integral segment entity by selecting the **File**/**Import/Table** command. FELIX prompts you to specify an input file name and an entity name.

## Spectrum display for integrals

When integrating spectra, it is often necessary to try several permutations of integral segments and normalization before a spectrum can be integrated satisfactorily. This is especially true if it is difficult to define segments or baseline points due to crowding. Thus, FELIX provides keypad navigation tools to quickly change displayed regions, as well as the **Adjust** option in the control panel belonging to the **Measure/Integral/Volume** command. Use it to change display type or manipulate integrals in real time.

While changing display type, you have the following options:

- **Entire Spectrum** shows only the full spectrum
- **Segments, no values** shows integrals only, with no integral values
- **Segments, values above** displays integral values above the integrals
- **Segments, values below** displays the integral values below
- **None** turns off the display of integrals

## Defining and deleting integral segments

To define integral segments, select the **Add Segment** option in the control panel belonging to the **Measure/Integral/Volume** command. Integral segments are added by dragging out a segment region with the cursor. When you select a valid region within your spectrum, you can add additional segments without reselecting the **Measure/Integral/Volume** command. To exit this mode, press the <Esc> key.

If you make a mistake while selecting individual segments or if you want to modify the current list of segments, you may delete a small subregion of segments graphically. First, select the **Remove** option in the control panel belonging to the **Measure/Integral/Volume** command. Use the small crosshair cursor to drag out a region of segments to delete.

To delete all segments, select the **Remove All** option in the same control panel. This action deletes the current integral segments entity from the database and requires confirmation via a dialog box.

## Adjusting integral slope and bias

For some spectra, it is impossible to accurately define baseline points. Use the **Adjust** option in the control panel belonging to the **Measure/Integral/Volume** command to adjust the slope and bias parameters in real time using sliders. If the baseline is significantly distorted, even adjusting the slope and bias may not be able to generate correct integral shapes.

> **Caution:** By adjusting the slope and bias, you are able to dial an integral value to anything you want. Thus, use these adjustments cautiously.

- The **Slope** slider is used to adjust the integral slope parameters between -1 and 1.
- The **Bias** slider is used to adjust the integral bias parameters between -1 and 1.
- The **Overlap** slider is used to adjust the vertical location of the integral curves.

## Integral normalization

FELIX enables you to normalize the integral of any segment of the spectrum to an arbitrary value. Four different normalization options are available via the **Measure/Scalar/Normalize** command. After normalization, the volume element in the integral segment entity is updated to the normalized value.

- **By Item Number of Segment** creates a list box where you must graphically select the segment to normalize based on its beginning and ending point. You must specify a normalization value for this segment.

- **By Data Point Limits** prompts you within a control panel for a low and high point to define a normalization range, as well as the normalization value.
- **Select Segment via Cursor** generates a small crosshair cursor that lets you drag to select the segment to normalize.
- **Raw Absolute Integrals** stores and displays each segment's integral as its raw intensity with no normalization at all.

# Task: Processing 2D data

This section describes the general procedures for processing 2D data. For information on converting your data files into a format that FELIX can read, please see "Task: Importing data", and Appendix D, *Data Files*.

## General processing steps

Multidimensional data are processed and stored by FELIX using matrix files. Matrix files are designed to allow easy access to and manipulation of the individual vectors that compose the data. In fact, virtually all *N*-dimensional processing is a repeated process of loading a vector from the matrix into the workspace, processing that vector, and then storing that vector back in the matrix. Therefore, you must build a matrix file that will be used to hold the *N*-dimensional data.

Once a matrix file has been created, the individual FIDs that make up the FELIX data file are read in one by one, processed, and saved to the matrix file. Subsequent dimensions are processed by loading each vector in the given dimension one by one from the matrix into the workspace, processing each vector in turn, then storing the processed vectors back in the matrix file. This process is repeated for each dimension of the matrix.

In practice, these steps are generally carried out using macros. Macros are files that contain a series of instructions used to process the data. FELIX includes a very flexible macro language (FCL), which allows you to control all aspects of how the data are processed. You can use your own macros to process the data or to use the EZ mac-

ros. The EZ macros are a series of predefined processing macros designed for the most common types of multidimensional data. Generally, using your own custom macros is preferred, but the EZ macros will often suffice.

## Processing the D1 dimension using macros

The macro below shows an example of how to process the D1 dimension of a 2D data set. This sample macro is appropriate for either States or TPPI data. (The line numbers are for reference only and are not included in the actual file.)

```
1   c**simpled1.mac
2   ;
3   def datfil 'my2dfile.dat'
4   def matrix 'my2dfile.mat'
5   def d1zfil 2048
6   def d2zfil 2048
7   def numd1 512
8   ;
9   cmx
10  bld &matrix 2 &d1zfil &d2zfil 0 y
11  mat &matrix w
12  ;
13  def temph0 -63.7941
14  def temph1 -130.5324
15  ;
16  def datsiz &d1zfil
17  def datype 1
18  set 1
19  ss 2048 60
20  stb 1
21  ;
22  ; D1 processing
23  ;
24  ty  D1(t2) Processing.
25  ty ---------------------
26  ;
27  cl
28  for row 1 &numd1
29   esc out
30   if &out ne 0 quit
31   rn &datfil
32   if &status ne 0 eof
33   def phase0 &temph0
34   def phase1 &temph1
35   def datype 1
```

```
36   bc 0.2
37   lpf 32 16 8 1
38   cnv 0 48
39   mwb 1
40   ft
41   ph
42   red
43   sto 0 &row
44   ty Row=&row
45  next
46 eof:
47  if &status ne 0 then
48   def status 0
49   ty End-of-file on record &row
50  eif
51  ty  D1(t2) transform completed.
52  ty --------------------------------
53 quit:
54  cmx
55  def status 0
56 ret
57  end
```

### Lines 3–7:

These lines define a series of symbols whose values determine how the data processing is to be performed. You do not need to define these symbols separately in the beginning of the macro. You can choose to enter processing-specific parameters on the individual command lines. However, grouping important symbol definitions at the beginning of a macro tends to remind you which parameter values may need to changed for different processing sessions.

### Lines 9–12:

The **cmx** command is used to close any open matrix files. Then the **bld** command is used to create an empty matrix file of the appropriate size. The zero in the **bld** command line indicates that the matrix to be created will be real. Most multidimensional data processing is done with real matrix files. The **mat** command is then given to open the matrix that was just created with write access.

### Lines 13–14:

These two lines define a pair of symbols (**temph0** and **temph1**) which are used to store the zero and first-order phasing parameters.

You need to store the desired phasing parameters in temporary symbols because when the input data files are read in later, the **phase0** and **phase1** symbols will be overwritten by the phasing values that were stored in the input data file.

**Lines 16–20:**

This section of the macro illustrates how to set up an apodization function in buffer 1. Later in the processing-loop section of the macro, each FID is multiplied by the contents of buffer 1. This is a more efficient method of doing the apodization than recalculating the appropriate apodization function for each FID. Lines 16 and 17 define the **datsiz** symbol as the appropriate number of complex data points and the **datype** symbol as one that indicates complex data. The **set** command then sets all the real values in the workspace to 1. The appropriate apodization function is then performed and the result is stored in buffer 1.

**Lines 24–25:**

These lines print out a message to the text window that the D1 processing loop is about to begin.

**Line 27:**

The **cl** command is used to close any open data file. This ensures that a subsequent read command (**re** or **rn**) will read the first record of the data file.

**Lines 28–45:**

This section of the macro forms the main processing loop. This loop is executed once for each data file in the input data set.

**Lines 29–30:**

These two lines of the macro show how the **esc** command can be used to allow you to interrupt a macro during execution. The **esc** command monitors keyboard input for the escape character. When the <Esc> key is pressed the user-defined symbol (here, the **out** symbol) is set to one. An **if** statement transfers control out of the loop when the **out** symbol is no longer equal to zero.

**Lines 31–32:**

These two lines read the next data set from the input data file and check the status of the read command. The **re** command is used to read old-format data and the **rn** command is used to read new-format data. If the read command is successful, the status symbol is set to zero. If the read command is not successful because, for example, you tried to read more data sets than were present in the input data file, then the status symbol is set to a non-zero value. If the read is not successful, then control is transferred out of the loop.

**Lines 33–34:**

These two lines make sure that the **phase0** and **phase1** symbols, which are used by the subsequent phasing command (**ph**), are set correctly based on the saved phasing parameters (**temph0** and **temph1**). This is necessary (as mentioned above) because, when a read command is given (**re** or **rn**), the **phase0** and **phase1** symbol values are overwritten with the phasing values that are stored in the input data file.

**Line 35:**

This line sets the **datype** symbol to one which represents complex data. This is required so that subsequent processing operations operate properly.

**Lines 36–41:**

These are the main processing steps. The data are baseline corrected, the first point is corrected, and solvent is removed, apodized, ft'd, and phased. You would customize this section of the macro to correctly process your data. Apodization is performed by multiplying by buffer 1. Buffer 1 contains the apodization function that was set up earlier in the macro.

**Lines 42–44:**

The data are then reduced so that only the real part of the FID is retained. These real data are then stored in the matrix. A type command (**ty**) is used to print out the current row so that you can monitor the progress of the macro.

**Lines 46–50:**

If a read command should fail (such as by trying to read more input data sets than are present in the input data file) then control transfers to this section of the macro. A message is printed telling you which record triggered the failed read.

**Lines 53–57:**

If the macro finishes normally or you exit the macro early, then control transfers to this section. A **cmx** command closes the open matrix. The status symbol is set to zero, indicating a normal exit, and control is returned to the menu interface.

## Processing the D2 dimension using macros

The macro below shows an example of how to process the D2 dimension of a 2D data set. This macro is appropriate for States data. This example assumes that 512 FIDs (256 complex points in D2) were collected in the D2(t1) dimension and that a real matrix was used with 1024 × 1024 points. (The line numbers are for reference only and are not included in the actual file.)

```
1   c**simpled2.mac
2   cmx
3   def matrix 'my2dfile.mat'
4   mat &matrix w
5   ;
6   ty D2(t1) Processing.
7   ty ---------------------
8   ;
9   for col 1 &d1size
10    esc out
11    if &out ne 0 quit
12    loa &col 0
13    def datype 1
14    def datsiz 512
15    zf 1024
16    ss 256 90
17    ft
18    ph
19    red
20    sto &col 0
21    ty Col=&col
22  nex
23 quit:
```

```
24  ty D2(t1) transform completed.
25  ty ----------------------------
26  cmx
27 ret
28  end
```

**Lines 2-3:**

The **cmx** command is used to close any open matrix files. Then the **mat** command is used to open the appropriate matrix file for writing.

**Lines 9–22:**

This section forms the main processing loop. Since this macro operates on the D2 dimension, a column must be processed for each D1 point value. The **for** loop increments the symbol **col** from 1 to the value of the symbol **d1size**. When a matrix is opened, the value of the symbols **d1size**, **d2size**, etc. are set to the number of points in the corresponding dimension. Thus, the symbol **d1size** is automatically set to the number of points in the D1 dimension and is therefore a logical choice to use as the maximum value of the **for** loop.

**Lines 10–11:**

These two lines show how the **esc** command can be used to allow you to interrupt a macro during execution. (Such checking slows down the processing, so you may prefer not to use it.) The **esc** command monitors keyboard input for the escape character. When the <Esc> key is pressed the user-defined symbol (in this case the **out** symbol) is set to one. An **if** statement transfers control out of the loop when the **out** symbol is no longer equal to zero.

**Line 12:**

The **loa** command is used to load the next column into the workspace for processing.

**Lines 13–15:**

Since this is States data, the **datype** symbol is set to 1, indicating complex data. Now that the data are complex, the data size (**datsiz** symbol) must be reduced by half, because there are only half as many complex points compared to the number of points in the vector when it was first read in as real. If the data size is not reduced by half, then the second half of the workspace will contain invalid data.

The data size is then doubled by zero-filling, so that the second half of the FID will contain zeros. The data size in complex points (**datsiz**) is now equal to the number of points in the D2 dimension of the matrix.

**Lines 16–18:**

The data are then apodized, ft'd, and phased. The apodization is over 256 complex points. This is due to the fact that the 512 FIDs collected in the D2 dimension correspond to 256 complex points in D2. Because this is States data, a complex fit is performed.

**Lines 19–21:**

The data are then reduced so that only the real part of the FID is retained. The data size (**datsiz** parameter) remains unchanged and is equal to the number of points in the D2 dimension of the matrix. These real data are then stored in the matrix. A type command (**ty**) is used to print out the current column so that you can monitor the progress of the macro.

**Lines 23–28:**

If the macro finishes normally or you exit the macro early, then control transfers to this section. A **cmx** command is given to close the open matrix. Control is returned to the menu interface.

## Processing 2D data with the supplied macros

The supplied precoded processing macros (in the **Process** pull-down) provide an alternative to writing your own processing macros. These macros are designed to process the most common kinds of 2D data. These macros do not provide the same flexibility inherent in writing your own custom macros, but they do offer the capability to quickly process 2D data collected with most of the typical acquisition schemes.

Access the precoded 2D processing macros with the **Process/2D Data Processing** command. FELIX displays a file-selection control panel. Here, you can select the data type: FELIX old and new data (.dat) file, FELIX matrix file (.mat), Bruker fid or ser file, Varian fid file or JEOL Alpha or Lambda file, and the filename.

If FELIX can access the data, it opens a second control panel with the header parameters. The acquisition parameters can be valid only if

the appropriate files are present (for Varian the procpar file and for Bruker the acqus and the acqu2s file. For Bruker the program also attempts to read the pdata / 1 / procs and proc2s files.) In this control panel, you can update the header parameters if they show incorrect values.

When you select **D1 dimension** for processing FELIX displays a menu of choices which define how the D1 dimension is to be processed. For more information on each of the various processing options see Chapter 4, *Processing, Visualization, and Analysis Interface (1D/2D/ND)*. When you click **OK**, the macro first builds a real matrix of the size that you specified with the **Dimension 1 Size** and **Dimension 2 Size** parameters. The macro then reads in each of the FIDs from the specified input file, processes the data according to the options you selected, and saves the processed vector to the matrix. The macro completes when each of the FIDs, as specified by the **D2 Parameters Data Size** header parameter, has been processed.

When the D1 macro finishes processing the data, you will have a FELIX matrix file (normally with a *.mat* extension) with each of the D1(t2) vectors processed. At this point you can process the D2(t1) vectors by again selecting the **Macro/2D Data Processing** command and choosing the matrix from the previous processing.

Then, FELIX displays a menu of choices, which in this case defines how the D2 dimension will be processed. When you click **OK** to begin processing the D2 dimension, the macro opens the matrix you have selected, reads each of the D2 vectors into the workspace, processes each vector according to the options you selected, saves the processed vector back to the matrix, and closes the matrix. When the D2 processing macro is finished, you will have a FELIX matrix file where all of the D1 and D2 vectors have been processed.

## Checking/examining the data as they are being processed

As part of the transformation process, before proceeding to the next stage in the transformation process, remember to check the data and verify that everything is correct.

Whether you choose to process your data using the precoded processing macros or your own custom macros, the process of transforming your data in the D1 dimension is basically the same. The macro generally first builds a new real matrix to hold the data, reads

in each FID from the input data file into the workspace, processes each FID in turn, and stores the result to the matrix.

1. The first check is to make sure that the raw data are correct or that the spectrometer-format data was correctly converted to a FELIX old- or new-format data file. To do this you can use the **File/Open** command to examine the individual FIDs that make up the input data file. By changing the **Dimension** parameter between **1D** and **ND**, you can control whether this action always reads in the first FID in the series or that subsequent reads load in successive FIDs from the input data file.
2. Once you have verified that the input data file is correct, you should experiment with processing the first FID in the series to determine the optimum processing parameters for the D1(t2) dimension. Only after you have determined a good set of starting parameters to use for the D1 dimension should you proceed to processing the entire D1 dimension using a matrix.

> **Note:** At this point in the processing, you will have a FELIX matrix file where all the D1 vectors have been processed but none of the D2 vectors have been processed.

3. Examine the matrix at this point to make sure that the D1 processing is acceptable. Open the matrix file, extract 1D slices (using the **View/Draw 1D Slices** command), and then examine the individual 1D vectors using commands such as **View/Limits/Manual Limits**. The **View/Draw 1D Slices** command is especially helpful because it allows you, through the use of a slider, to scan through the individual D1 vectors in order, to determine if the data needs to be reprocessed.
4. If you decide that the basic processing operations such as apodization and solvent suppression were performed correctly, then you generally do not have to fully reprocess the D1 dimension. It is possible to rephase or baseline flatten the existing data.

- To rephase the D1 dimension you could use the **Process/Phase Correct Matrix** command.
- To baseline flatten a matrix that has already been transformed in the D1 dimension, you could use one of the **Process/Baseline Correct Matrix** commands.

5. You can write your own macro to rephase or baseline correct the data in the D1 dimension. Once you have determined that the D1 dimension has been processed correctly and have reprocessed the data if necessary, you should look at some of the D2 vectors. Select some peaks in the first transformed row that you expect to have strong cross peaks and note the point numbers. Alternatively, you could click the **Vertical 1D Slice** icon to select an initial column vector to view, then use the left and right arrows on the keyboard to scan through a series of column vectors noting the D1 point numbers for a few columns which show a strong interferogram.
6. It is often desirable to load these selected columns into the workspace using the **View/Limits/Manual Limits** command and then save them to FELIX data files. Use these saved column vectors to determine the optimum processing parameters for the D2 dimension. Once you have decided on the optimum processing parameters for the D2 dimension, you can move on to processing all the D2 vectors using your own macros or the precoded processing macros.

> **Note:** Whether you process the D2 dimension with your own macros or the precoded macros, the basic process is the same. The macro opens the appropriate matrix, reads each column vector into the workspace one by one, processes the vector, saves the vector back to the matrix, and closes the matrix file.

7. After you have transformed the D2 dimension, you need to examine the individual D2 vectors by either selecting the **View/Limits/Manual Limits** command or clicking the **Vertical 1D Slice** icon. If the D2 vectors need to be rephased or baseline corrected, you can do this with your own macros or the precoded macros in the **Macro** pulldown as discussed above for the D1 dimension.

## Task: Analysis of relaxation data

Throughout this documentation, we use the terms $R_1$ and $R_2$ for the relaxation rates, which are simply the inverse of the relaxation times $T_1$ and $T_2$.

We assume that $R_1$ and $R_2$ relaxation data and heteronuclear NOE were measured as a series of 2D HSQC (or equivalent) spectra (Skelton et al. 1993). You must pick peaks in one of the relaxation spectra or in an equivalent HSQC. Peak assignments are helpful, but not necessary.

## Evaluate peak heights or peak volumes

The first step in relaxation analysis is to evaluate peak heights or peak volumes in a series of 2D spectra using the **Measure/Relaxation/Measure Heights/Volumes** command. In practice, peak heights have proved more reliable and yield better signal to noise ratios (Skelton et al. 1993). FELIX automatically centers the peak boxes (if they were picked in another spectrum and are slightly off) and then determines the peak heights or volumes in the relaxation spectra.

For peak-volume determination, the raw volumes are optimized by fitting a 2D Gaussian function to the spectra and deriving volumes from the fit. In the first spectrum of the series, peak widths are optimized, then peak amplitudes, and finally both are optimized simultaneously. In subsequent spectra, centers and widths are left as they are, and only amplitudes are adjusted. For peak-height determination when peak boxes overlap, the peak box is iteratively shrunk by one point in each dimension until no other peak box overlaps it, to prevent cross talk from any overlapping peak maxima.

Thus volumes and heights are accurate within the limits of the methodology even for overlapped peaks, and the procedure requires no interaction on your part. Peak heights or volumes are stored into a regular FELIX volume table, but the last row holds relaxation delays instead of mixing times.

## Estimate signal-to-noise ratio

The next step is to estimate the signal-to-noise ratio from one or more duplicate spectra, using the **Measure/Relaxation/Signal/Noise Ratio** command. The average standard deviations obtained for these spectra are interpolated to the other points in the series as appropriate. The time courses of peak heights / volumes that you obtain in this way can be visualized and plotted by:

Selecting the **Measure/Relaxation/View Timecourse via Cursor** command and then selecting a peak with the mouse

or

Selecting the **Measure/Relaxation/View Timecourse via Item** command and entering a peak number

If the time series is already fitted to an appropriate exponential function, FELIX displays this function, and reports the relaxation rate in the text window.

FELIX stores duplicate peak volumes and heights in their own volume table. This is essentially a copy of the original volume table, with the appropriate column overwritten with values from the duplicate spectrum. FELIX records uncertainties in the last row of this table.

## Fit data to exponential function

After obtaining peak heights and volumes and their errors, you can fit these time series to an appropriate exponential function. Use the **Measure/Relaxation/Fit R1/R2/NOE** command. For $R_1$ data, this is a general exponential function:

$$y = a_0 + a_1 e^{a_2 x} \qquad \textbf{Eq. 21}$$

where $R_1 = -a_2$.

For $R_2$ data, where in theory the values decay to zero, the simpler equation:

$$y = a_0 e^{a_1 x} \qquad \textbf{Eq. 22}$$

where $R_2 = -a_1$ may be appropriate. In practice, it is often found that the more general Eq. 21 yields statistically better fits (Skelton et al. 1993). FELIX tries both fits and retains the coefficients for the fit with the lower $\chi^2$ value.

## Analysis of heteronuclear NOE data

Heteronuclear NOE data are analyzed in a single step. Select the **Measure/Relaxation/Fit R1/R2/NOE** command. After obtaining the filenames for the spectra obtained with and without $^1$H saturation, FELIX again centers the peak boxes (see above) and then proceeds to determine peak heights or volumes as desired. Next, FELIX analyzes two duplicate spectra to derive the uncertainty of the measurement. The NOE is the ratio of the volumes obtained with and without $^1$H saturation.

## Preparing input data files for Modelfree

In addition to generating $R_1$, $R_2$, and NOE values, the relaxation analysis tools provided in FELIX let you prepare input data files for the Modelfree program of A. G. Palmer (accessed via the **Measure/Relaxation/Modelfree Input** command). Modelfree can be downloaded from Columbia University's website at http://cpmcnet.columbia.edu/dept/gsas/biochem/labs/palmer

For further details on analyzing relaxation data with the Modelfree program, see Mandel et al. (1993).

# Task: Working with assignment databases

The assignment work starts with defining a project. It is highly advisable to define only one project within one database file. The project then contains all the information needed in the course of the work, such as spectrum definitions, last display parameters, and entities necessary to store the assignment information.

The definition of the project starts with reading in an Insight II or X-PLOR coordinate file of the molecule (or complex) being studied. This coordinate file does not necessarily need to be the true structure, especially since at the beginning of the assignment the final structure is not usually known. As the assignment progresses and refined structures become available, the new coordinates can be read into the project.

The crucial part of residue identification is the library. The library is an ASCII file, where the mean chemical shifts and standard deviations are listed for the usual residues. You can change these values and can also add non-standard residues to the end of the file. This procedure is illustrated later. This file is then read into the project and used in spin-system identification.

You can define the matrices that are used in the assignment. These matrices should be referenced in ppm and are in the standard Felix format. In adding the experiments, it is important to define a reasonable tolerance for each axis, since the automated spin-system detection routines rely heavily on them. This tolerance should mirror the uncertainty of the peak positions in each dimension (in ppm). Later on you can delete unused experiments from the database and add new ones. For example, you can start with J spectra (TOCSY, DQF-COSY, HSQC-TOCSY, or CBCA(CO)NH) to do the sequence-specific resonance assignment. Once that is done, you can delete the J spectra and define NOE-type experiments (e.g., NOESY or HSQC-NOESY) to do peak assignment and restraint generation.

# Task: Adding modified residues to the Assign database

If you work with non-standard amino acids or nucleic acids, you can add these new residue types to the database, by simply editing two files in the **data/felix/asglib/** folder under the installation folder (See “Starting FELIX 2002” on page 34 in the preface, *How To Use This Guide* in this book for more details about the installation directories). One file contains the residue definition (atoms, median chemical shifts, standard deviation, and topology); the other contains the alias names.

## Editing the residue-definition file

The residue-definition file is called pd.rdb or rna.rdb. For example, to add an aminobutyric acid residue, you should go to the end of the pd.rdb file, which looks like:

```
!
!
! ROOM FOR MORE RESIDUE TYPES ...
!
```

1. **Each line that starts with an exclamation mark is a comment. So first you add the formula as a comment:**

```
>   !
    ! ABU (aminobutyric acid)
    !
    ! H - N     H
    !     |     |
    ! H - Ca - Cb - Cg - H3
    !     |     |
    ! O = C     H
    !
```

2. **The next part of the file is the residue name:**

```
RESIDU ABU X
```

   where X is just a one-letter shortcut for the residue name.

3. **Now you need to enter the atoms:**

   First enter the keyword RESATM, then the atom type, atom name (in Insight notation), a number (this field is not active yet), a mean chemical shift, a standard deviation, and finally, if there is more than one atom (e.g., a pseudoatom) in that group, enter the number of atoms.

   This section of the file should now look something like:

```
RESATM H HN    1   8.16    0.60
RESATM H HA    2   4.30    0.20
RESATM H HB1   3   3.30    0.20
RESATM H HB2   3   3.30    0.20
RESATM H HG*   4   1.10    0.10   3
RESATM N N     8   126.0   5.0
RESATM C CA    10  51.0    5.0
RESATM C C     9   150.0   50.0
RESATM O O     15  0.0     999.0
RESATM C CB    12  26.0    5.0
RESATM C CG    12  17.0    5.0
```

4. **Next is another comment line that contains the atoms used to help construct the neighbor matrix:**

```
!       HN HA HB1 HB2 HG* N CA C O CB CG
```

5. **The next part of the file is the connectivity matrix:**

   Each element shows how many bonds are between the constituent atoms (groups).

```
CONECT 0
CONECT 3  0
CONECT 4  3  0
CONECT 4  3  2  0
CONECT 5  4  3  3  0
CONECT 1  2  3  3  4  0
CONECT 2  1  2  2  3  1  0
CONECT 3  2  3  3  4  2  1  0
CONECT 4  3  4  4  5  3  2  1  0
CONECT 3  2  1  1  2  2  1  2  3  0
CONECT 4  3  2  2  1  3  2  3  4  1  0
```

6. **Finally a keyword finishes the residue definition:**

```
ENDRES
```

### Editing the alias file

The other file to edit is the biosym.alias file.

7. **Append a line to the alias file which contains the names of that residue for all the charged/terminal forms:**

```
ABU ABUN ABUC ABUn
```

After these two files are edited, Assign can deal with the new residue type.

## Task: Peak picking

Usually the first step in assignment is peak picking. FELIX has two types of peak pickers: the regular one and one that uses example

peaks to distinguish genuine peaks. Depending on the data, it is probably advisable to try several rounds of peak picking using both methods, and then use whichever filtering functions (such as symmetrizing, deleting the diagonal, and merging multiplets) you prefer. Good peak sets are crucial for the success of the analysis procedure.

After picking peaks, you may want to use peak optimization to better define the peak centers, widths, and / or volumes. To do this you need to first measure the peak volumes. Sometimes the peak optimizer can merge close peaks; therefore, it is advisable to measure the peak volumes interactively and save the peak entity in a backup file within the database. By defining small regions where the optimizer should work and examining the results, you can see if the peaks get misplaced too much or merged unnecessarily. Then you can retrieve the saved copy and omit optimization for those peaks.

## Task: Using connected frames to navigate among multiple spectra within Assign

In assignment work you often need to analyze multiple spectra simultaneously. To help with this, FELIX allows you to connect multiple frames so that the definition of plot limits in one frame automatically triggers the same changes in the other connected frames.

To connect frames, use the **Preference/Frame Connection** menu item. To illustrate the process, assume you have a CBCANH and a CBCA(CO)NH experiment. The CBCANH was transformed so that D1 is $^{13}C$, D2 is $^{15}N$, and D3 is $^{1}H$. Assume that CBCA(CO)NH was transformed so that D1 is $^{1}H$, D2 is $^{13}C$, and D3 is $^{15}N$.

Also assume that you want to see $^{1}H$-$^{13}C$ slices, and that CBCANH is the experiment in **Frame 1** and CBCA(CO)NH in **Frame 2**. In **Frame 1** you would select the CBCANH spectrum from the **Experiment** table and in **Frame 2** you would select the CBCA(CO)NH spectrum. After going back to **Frame 1** and selecting the **Preference/Frame Connection** menu item you would do the following:

1. In the control panel, leave the **First Frame** set to **1** and the **Second Frame** as **2**.

2. For the connection method, click the **D1-D2-D3<=>D1-D2-D3** radio button and then click **OK**.

Now, if you zoom in on a region in the CBCANH spectrum (**Frame 1**), then the same plot limits are defined in the CBCA(CO)NH spectrum (**Frame 2**). Similarly, if you step onto another plane (for example, in the CBCA(CO)NH spectrum (**Frame 2**), then a plane at the same $^{15}$N frequency is displayed in the CBCANH spectrum (**Frame 1**).

Certainly, if you have 2D, 3D, and 4D spectra to connect, you can use this interface to define quite sophisticated schemes.

The following is a theoretical example of how to connect a 2D spectrum with 3D spectra:

Assume you have a 2D $^{15}$N-$^{1}$H HSQC spectrum with $^{15}$N as D2 and $^{1}$H as D1. Assume also that you have a $^{15}$N TOCSY and $^{15}$N NOE spectra, where D1 is the amide $^{1}$H, D2 the full $^{1}$H, and D3 the $^{15}$N dimension. You want to look through the amide peaks in the HSQC spectrum and then see the corresponding slices in the TOCSY and NOESY spectra.

1. First, select the HSQC spectrum from the **Experiment** table for **Frame 1** and the TOCSY and the NOE spectra, respectively, for **Frame 2** and **Frame 3**.
2. In the first control panel leave **First Frame** set to **1** and **Second Frame** set to **2**. This connects the HSQC spectrum with the TOCSY spectrum. Select the **Custom** option and click **OK**.
3. In the next control panel, set:

| First Frame 1 | Second Frame 1 |
|---|---|
| D1 | D1 |
| D2 | Null |

4. Toggle **Define Jump** to **on** and for **Jump Direction** set:

| First Frame 1 | Second Frame 1 |
|---|---|
| D2 | D3 |
| Null | Null |

5. Click **OK**.

6. Next, connect the HSQC spectrum with the NOE spectrum with the first control panel (which should appear again). Leave the **First Frame** set to **1** and the **Second Frame** to **2**. Select the **Custom** option and click **OK**.

7. In the next control panel, set:

| First Frame 1 | Second Frame 1 |
|---|---|
| D1 | D1 |
| D2 | Null |

8. Toggle **Define Jump** to **on** and for **Jump Direction** set:

| First Frame 1 | Second Frame 1 |
|---|---|
| D2 | D3 |
| Null | Null |

9. Click **OK**.

   Linking all these spectra assures that plot navigation in the HSQC spectrum (e.g., zooming in on a peak) translates to the same movement in the HSQC-TOCSY and HSQC-NOE spectra. If you press the period <.> on the keyboard while in the HSQC frame you will be able to use the cursor to select a new $^{15}N$ position (e.g., clicking on a HSQC peak), which then translates to new planes also in the TOCSY and NOE spectra.

   If in addition you want to navigate the TOCSY and NOE spectra together, you need to set that up by using the **Preference/Frame Connection** menu item again.

10. Make **Frame 2** current. Select the **Preference/Frame Connection** menu item. In the first control panel, leave **First Frame** and **Second Frame** set to **2** and **3**, respectively. Select **D1-D2-D3<=>D1- D2-D3** as the method and click **OK**. After the control panel appears again, click **Cancel**. This way you have connected all three axes of the TOCSY and NOE spectra.

# Spin systems

The key step in sequential assignment is spin-system detection. In FELIX spin systems are detected in three ways:

### Automated spin-system detection

The first type of spin system results from automated spin-system detection. These are called *prototype patterns* or *protos*. This system can be intraresidue or can contain spins from neighboring residues, depending on the spectrum and method by which the data were collected.

### Frequency clipboard

The second type of spin system is the *frequency clipboard*, which is a unique list of frequencies that originates from a proto or a concatenation of several protos, or which are hand picked from a spectrum. The clipboard is usually used where you would delete spurious resonances from the spin system or add missed resonances manually.

### Pattern

The third type of spin-system detection is the so-called *pattern*. Patterns can be scored against a library to find a type probability score or they can have sequential neighbors. The frequencies in the pattern may also then be assigned to atoms (or atom groups) in the molecule. Patterns can be edited, but with much more limited functionality.

At each stage, the spin system can be visualized by spawning tile plots or strip plots or by drawing lines along the frequencies. Visual interaction is an important element in assignment, since automated routines are not 100% reliable.

## Task: Manual spin-system collection

You can collect spin systems manually; that is, by picking peaks in a spectrum and promoting the chemical shifts to the clipboard. You would typically do this if there were unresolved or missed spin systems left after automated spin-system detection. Then you would go

through the spectrum (or spectra) and use the controls in **Assign/ Frequency Clipboard** to collect the peak positions as frequencies in the clipboard. After each spin system is collected this way, you then need to copy it to **patterns**, then clear the current spin system and start over with the next spin system.

## Task: Automated spin-system detection

The automated prototype pattern-detection routines work on one or several peak-picked spectra, depending on the method selected. You should optimize the values of several parameters by trial-and-error. To find as complete a set of spin systems as possible, several iterations are necessary. Guidelines for selecting prototype pattern parameters are given below.

### Tolerances

One of the most important variables in automated spin-system detection and neighbor-finding methods is the tolerance. A spectrum-specific tolerance for each experiment needs to be entered at the beginning. This number usually represents how well the peaks are lined up in each dimension up for any given frequency, that is, how much the peak centers differ along a particular dimension arising from the same spin system. As a rule of thumb, this number is roughly equivalent to the digital resolution in ppm (e.g., first try the ppm equivalent of one point, which then will use this number as a boundary--each peak whose center is within the target plus/minus the tolerance). This tolerance is used to automatically assign peaks to frequencies.

Another tolerance is also required during automated spin-system detection, which is sometimes needed to decide whether a peak belongs to a spin system. If you use the homonuclear method with one spectrum, this tolerance is defined by the resolution of the dimension having the worse digital resolution in ppm, usually the F1. If you use multiple spectra, then this tolerance should be defined based on a comparison of the chemical shifts of the peaks belonging to the same connectivity in different spectra.For example, if you measure the position of a couple of the “same” peaks in the NOESY and TOCSY spectra, the difference defines the tolerance. Certainly, if the automated methods do not give sufficient results, the first thing you should try changing is the tolerance.

It may of benefit to reference one base spectrum and also newly introduced chemical shift ranges in other spectra, according to scientific theory, and reference the other spectra so that known peaks match optimally. This can significantly increase the success of automated spin-system detection and peak-assignment routines.

## Two-dimensional systematic search method

The 2D systematic search method can be used on different combinations of COSY, TOCSY, and NOESY type spectra. In the first iteration for a protein, the spin systems are usually collected starting from a $H_\alpha$–$H_\beta$ region, and you must exclude spin systems having frequencies in the aromatic / amide region. Similarly, if you want to search for aromatic side chains, you must use the aromatic region as a seed and exclude the aliphatic region by filtering.

Between iterations, you can selectively delete prototype patterns that are clearly not right, thus releasing peaks, or you can delete all of them. Also, it is useful to note that after you assign or define several reliable patterns, then those patterns' frequencies can be copied back to prototype patterns, thus preventing you from re-detecting them in the following iterations.

## Two-dimensional simulated annealing method

This search should begin with the longest spin systems. Since the algorithm tries to fit peaks into a defined motif, it does not take care of possible additional correlated frequencies, which means that an AMX portion of a long spin system could be assigned to a four-spin system. Initially, the program should be run on the whole residue set of the primary sequence (which automatically takes the above priorities into account) and on the patterns examined with the usual interactive tools. Then it should be rerun on specific missing amino-acid types. To compensate for the limited number of iterations in simulated annealing, the process should be run for several loops (typically 6), from among which the program will retain the best results. One loop of the program for the whole sequence of a 53-residue protein requires about 10 min of computation time on an R4000 Silicon Graphics Indigo workstation. For aromatic residues, this method assigns only the AMX subsystems; therefore, the aromatic resonances should be found with the systematic search method and added through the clipboard.

## Double-resonance methods

Double-resonance spin-system detection is implemented in two different ways: one method which finds spin systems starting from the backbone, and one which finds spins in existing spin systems and extends them.

The first method works on $^{15}$N-$^{1}$H HSQC-TOCSY and $^{15}$N-$^{1}$H HMQC-TOCSY spectra. For this method, the spectrum should contain pseudodiagonal peaks–that is, HN-HN-N peaks–from which the spin systems will be collected along the sidechain. If the pseudodiagonal is not well resolved or if peaks are missing, you can use the $^{15}$N-$^{1}$H HSQC spectrum to help–the program tries to find new frequencies in the 3D from each peak in the 2D spectrum and stores them as spin systems. However, if the $^{15}$N-$^{1}$H HSQC spectrum is not well resolved, this method is less effective.

In the second spin-system detection method, you can use double resonance experiments to extend already existing spin systems--usually these are results of a triple-resonance spin-system detection run. Here, the purpose of the detection is to expand spin systems containing only backbone frequencies, to include sidechain information. Currently one method is implemented in FELIX which uses a 3D HCCH-TOCSY spectrum to achieve this. You can expand spin systems in the prototype pattern stage or in the pattern stage.

## Triple-resonance methods

A variety of triple resonance measurements can be used for making sequence-specific assignments. Several approaches are implemented, which have some common elements: usually you must first peak-pick several triple-resonance spectra and find out what the uncertainty is between the peak positions within each spectrum for a couple of resonances. This will be the spectrum-specific tolerance for each axis. Also, you need to find out, simultaneously inspecting the spectra, how well the peaks belonging to the same resonances can be overlaid. For example, you need to know, at a given X residue, what are the differences between the peak frequencies along the $H_N$ and $^{15}$N axis, and $C_\alpha$ in the HNCO and HNCA and HN(CO)CA spectra ($H_{N,x}$-$N_x$-$C_{x-1}$, $H_{N,x}$-$N_x$-$C_{\alpha,x-1}$, and $H_{N,x}$-$N_x$-$C_{\alpha,x-1}$, respectively). Those will be the interspectrum tolerances for the respective axis.

After that you can run one of the spin-system detection routines starting with, for example, a tolerance that is a little less than these inter-spectrum tolerances, and then telling the program to iterate and increase the tolerance at each iteration step. Depending on the spectrum quality, you can automatically get 50-90% of the theoretical spin systems. These spin systems may not be perfect; therefore, you need to inspect them before promotion and discard or alter any inadequate spin systems. Also, you can manually add missed spin systems to the list of prototype patterns. There is an advantage, since usually triple resonance experiments are measured as (at least) pairs, so if you add spin systems at this stage, you can manually add frequencies belonging to this residue as well as to the neighboring residue. As long as you let the program know what those frequencies are, then Assign will establish these connectivities in the promotion stage.

## User-definable automated spin-system detection

In addition to the predefined combinations of experiments given here, you can design your own spin-system detection methods. Typically you would design your own protocol if you have a good-sensitivity 2D or 3D double- or triple-resonance spectrum (e.g., $^{15}$N HSQC or HNCO), known as a “seed” spectrum, together with a couple of double- or triple-resonance 3D “secondary” spectra (e.g., HNCA and HN(CO)CA). The typical procedure is to start from the seed spectrum with each peak as a trial peak and try to collect connected resonances in the secondary spectra. For example, if you want to detect spin systems using a combination of HNCA and HN(CO)CA experiments, you can use the following procedure (assuming that the order of dimensions is $H_N$, N, $C_\alpha$):

1. First, select the **Assign/Collect Prototype Patterns/User Settable** menu item and set the **Number** of steps to **1**, since the seed peak selection is considered as the zeroth step. Select the HN(CO)CA spectrum as the seed spectrum (Step # 0) and the HNCA spectrum as the first step.

2. Since you need to match the HN and N frequencies in the two spectra for the corresponding peaks in the second control panel (Experiment **hncoca** in Step 0), you select the following for the **Match to** parameters:

| Match to | hnca | null | ... | null |
|---|---|---|---|---|
| hncoca D1 | D1 | No | ... | No |
| hncoca D2 | D2 | No | --- | No |
| hncoca D3 | | No | --- | No |

3. For the **D1 Type** select **HN**, for **D2** Type select **N**, and for **D3 Type** select **null**. The typical tolerances would be comparable to the differences between the chemical shifts of the same $H_N$-N-$C_\alpha$ peak in the two spectra, for example, 0.02, 0.15, and 0.15 ppm for **D1 Tol**, **D2 Tol**, and **D3 Tol**, respectively. You can perform selective detection using chemical shift ranges to reflect that, or you can use the full spectrum using -1 for all **Range** variables. Typically, several trials are needed from each peak, to find a prototype pattern.

   The next set of variables reflects this, that is, if with the given tolerances not enough frequencies (**Minimum Freqs** in **Proto**) are collected from a particular peak, then new iterations are carried out (up to **Number of Iterations** times) increasing the tolerances by the **Tolerance Factor**. You can also direct the program to delete peaks that may belong to already detected spin systems (**Remove First** is then **True**). For this, you would select **3** for the **Number of Iterations** and **1.2** for the factor by which the tolerances are multiplied after each unsuccessful trial (you need 4 frequencies).

4. In the next control panel you must decide how many peaks to store and which frequencies to store. Here, we need to use the D3 frequency (the $^{13}C$ dimension) (**Use D3**) and you need to store the frequencies for two peaks (**Store 2**). The **Type** variable is active if you select the **All** option for **Store** (it is useful in, for example, a 15N-HSQC-TOCSY experiment, because the program would not assign the resonances along the TOCSY line but would call them HXs). You set the first peak's frequency to store as $C_\alpha$ (this peak in the HNCA spectrum is usually more intense: #1 $C_\alpha$ and **Attr Larger**). The second peak's D3 frequency then results in the $C_{\alpha,i-1}$ frequency (#2 $C_\alpha$(-1) and **Attr Smaller**). Besides the magnitude of the intensity, you can use the sign of the intensity as a distinction (as for the HNCACB spectrum, where the $H_N$-N-$C_\alpha$ and $H_N$-N-$C_\beta$ peaks have opposite signs) or use a distinctive chemical shift range.

This procedure makes it possible to collect spin systems automatically in the user-defined combination of spectra.

## Task: Semiautomated spin-system detection

You can collect spin systems using a semiautomated method. Here, you use the cursor to select a peak in one spectrum, and the program then tries to extend this trial spin system in the spectra that you connected to it.

To illustrate: assume you have a 2D $^{15}$N-$^{1}$H HSQC spectrum with $^{15}$N as D2 and $^{1}$H as D1. Moreover assume that you have a $^{15}$N TOCSY, where D1 is the amide $^{1}$H, D2 the full $^{1}$H, and D3 the $^{15}$N dimension. You want to select an amide peak in the HSQC spectrum and then let the program collect frequencies in the corresponding slice of the 15N TOCSY spectrum.

1. First you select the **Assign/Collect Prototype Patterns/ Semiautomated Setup** menu item and set the **Number of frames/steps** to **2**, putting the HSQC in **Frame 1** and the HSQC-TOCSY in **Frame 2**.

2. In the second control panel, select the following for the **Connect to** parameters:

| Connect to | #2 | #3 | #4 |
|---|---|---|---|
| Frame 1 D1 | D1 | No | No |
| Frame 2 D2 | No | No | No |

3. Now set the **Slice Position** in **Frame 1** parameter to **D2** and the **Sliceplane** in #2 to **Along D3**. That means that you want to use the D2 coordinate of the HSQC spectrum to select a plane in HSQC-TOCSY along D3.

4. Next you must specify how to match the different frequencies. Here, you need to match the D1 of the HSQC to the D1 of the TOCSY, and the D2 of the HSQC to the D3 of the TOCSY spectrum ($H_N$ to $H_N$ and $^{15}$N to $^{15}$N). You also need to specify the spin types you expect in this first spectrum, together with search tolerances: $H_N$ with 0.02 and N with 0.1.

5. The number of iterations can be set with a tolerance factor, which specifies that the spin-system collection is be tried that many

times, increasing the tolerances that many times (you can set it to 3 and 1.4, for example).

6. In the third control panel you would set the **Orientation** parameter D1-D2 to D3: 118 ppm. You would set the **Connect to** parameters as follows:

| Connect to | #2 | #3 | #4 |
|---|---|---|---|
| Frame 2 D1 | D1 | No | No |
| Frame 2 D2 | No | No | No |
| Frame 2 D3 | No | No | No |

7. Now set the **Use** parameter to **D2** since you want to collect new frequencies along D2 of the HSQC-TOCSY spectrum (with D1 and D3 defined by the HSQC) and **Store All** with type $H_{\Xi}$.

This setup then would make it possible to zoom in on a peak in HSQC and on a strip in HSQC-TOCSY. Moreover, if you want the $^{15}N$ position from HSQC transferred to 3D spectra, you need to press the < **.** > key to activate the hidden **Jump** function. This function gives you a cursor: clicking the required peak (or $^{15}N$ position) with this cursor triggers a jump to a new slice. Finally, if you find a peak in the HSQC spectrum where you want a new spin system to be collected, you need to select the **Assign/Collect Prototype Patterns/Semiautomated Collect** menu item (for which the hotkey is **=**) and click that peak with the crosshair cursor. If the program can find connected peaks in the HSQC-TOCSY spectrum, it collects the spin system and shows it in the prototype pattern table.

## Task: Spin-system extension

Most of the automated methods in the Assign module collect spin systems containing backbone atoms. To automatically extend the spin systems to include atoms from the sidechain, you may want to run the routines accessed from the **Assign/Collect Prototype Patterns/Extend Prototype Patterns** menu item.

To illustrate this method, assume that the spin systems were collected so that the prototype patterns contain the $H_N$, N, $C_\alpha$, and $C_\beta$ shifts. Then you can use a 3D HCCH-TOCSY spectrum to automatically find the corresponding $H_\alpha$, $H_\beta$ frequencies. In the **Extend Prototype Pattern Using HCCH-TOCSY** control panel, you would

select appropriate tolerances for the search along the $^{13}C$ and $^{1}H$ dimensions (taking into account how well the $C_\alpha$ and $C_\beta$ resonances align between the spectra they were detected from (e.g., CBCANH) and this spectrum). You have to set how many times the extension will be attempted from each prototype pattern (**Number of iterations**) and how much to increase the tolerances each time (**Tolerance factor**). You also have to define the primary and secondary search dimensions for protons, as well as the carbon dimension. The primary proton dimension is where the heteronuclear transfer was done, and the secondary dimension is where the TOCSY was done.

You can use the extend option to add $H_\alpha$,i-1 and $H_\beta$,i-1 frequencies to spin systems containing $H_N$ and N frequencies using the HBHA(CO)NH spectrum (**Extend Prototype Pattern Along One Axis**).

## Task: Spin-system promotion

As soon as reliable spin systems are detected, you can promote them to patterns. This can be done in two ways.

One way is to copy the protos one by one, using the clipboard. This is the preferred method for spin systems detected in homonuclear or heteronuclear double-resonance spectra, since it is wise to visually inspect and correct the results of automated methods. After inspecting and correcting the frequency clipboard you can verify that the new “pattern” is unique by using the fuzzy algebra comparison control panel (**Assign/Frequency Clipboard/Compare Frequencies** menu item).

The other way, which might be more dangerous if you did not verify the prototype patterns, is to copy all prototype patterns to patterns directly. If you choose this method, you must delete false prototype patterns before copying. This method is preferred for prototype patterns resulting from triple-resonance heteronuclear spectra, since the neighbor information contained in protos detected in triple resonance spectra can be preserved this way.

## Task: Spin-system identification

The next step in peak assignment is to identify the spin systems–the patterns resulting from an automated or manual search. This can be

done in Assign based on all-atom chemical shifts contained in the database, by using either a simple scoring algorithm or by using a probability distribution for $C_\alpha$ / $C_\beta$ chemical shifts (Grzesiek and Bax 1993). The latter method gives better scores, since the proton chemical shifts are not so well dispersed for different residue types, but this method can only be used if labeled proteins are accessible.

The all-atom method gives scores that are not very distinguishable from each other, but, combined with the sequential probability scores, it still gives a good starting point for the sequential assignment-generation step. To help with the assignment, it is advisable to manually unset scores based on a DQF spectrum of the very unlikely residue type probabilities.

## Task: Establishing connectivities

This is the next step in the sequential assignment strategy. Usually this step uses the NOE effect as its basis, but in triple-resonance spectra, this connectivity shows up in J peaks, therefore the J-peak-based method is more reliable. Nevertheless, because of the difficulties encountered during labelling, and because a significant portion of the connectivities do not show up due to spectroscopic reasons, the NOE-based methods are still important.

The Assign module includes several menu items that deal with the sequential connection of patterns. It is advisable to have a possibly full set of patterns when you start to find sequential connectivities. Also, it is important that the spectrum-specific shifts be set correctly for the NOE spectrum that is to be used to find the sequentials. The root frequency for each pattern also should reflect the real (average) $H_N$ frequencies of the NOE spectrum for the homonuclear neighbor-finding methods.

You can use the **Assign/Neighbor/Find Neighbor Via 2D NOE** or **3D NOE** menu item to connect the aromatic sidechains with the aliphatic sidechains. You usually end up with prototype patterns having the aliphatic and aromatic part of the same residue as two different entries. Therefore, you promote them to separate patterns. Then you must find which aliphatic sidechain has several contacts to the $H_\alpha$'s (of Tyrs or Phes) which should be the root frequency for that particular pattern. If you find the correct connection (possibly

visually inspecting by tiling) then you must merge the two patterns (you can use the **Assign/Frequency Clipboard/Copy Pattern To Clipboard** and then the **Assign/Frequency Clipboard/Copy Clipboard To Pattern** menu items). Lastly, you must delete the purely aromatic sidechain pattern(s).

## Task: Sequential assignment

The **Assign/Sequential/Systematic Search** menu item can be used to match the sequence on a set of patterns that have at least residue-type probabilities and sequential probabilities assigned. The algorithm is flexible, therefore several strategies can be followed. One approach is to generate all possible assignments for the entire molecule (**Min length of assigned stretches** = full sequence). Here, the assumption is that the best scores will result from the correct assignment, but the **Min neighbor prob score** variable should be set to **0**, since there can be missing sequentials. This could mean that the number of possible assignments will be very large (Kleywegt et al 1993),but restricting the **Max #** of assignments to generate variable to a small number (100-1000) can give a usable result.

The other approach would be to assign shorter stretches (20-30 residues), still keeping the neighbor probability score comparatively small (e.g., 0.1). This method has the disadvantage that you cannot assume that the first solution is the correct one; therefore, you must check all high-scoring possibilities.

The third approach was named “iterative assignment by consensus” (Kleywegt et al 1993), which means that assignment generation starts with restricting the neighbor probability score to high values and letting the **Min length of assigned stretches** be relatively small.

Thus, well-connected stretches are found first, then the neighbor score is gradually relaxed and the minimum length is increased, and longer, well defined stretches are assigned. This procedure continues while no new assignments can be obtained. Then you might try to specifically assign the remaining stretches. At all stages the consensus means that those residues are assigned which are conserved in a majority of generated assignments.

The **Assign/Sequential/Simulated Annealing** menu item can be used on any set of homonuclear or heteronuclear patterns. It uses

only the type and neighbors scores, obtained by any method, to find the sequence-specific assignment. Optionally, previous assignments are loaded and respected. The amino acid type and / or residue number are considered assigned for a pattern, if they are consistent over all frequencies of the pattern (unique or specified assignments).

After careful inspection of the patterns and scoring of types and neighbors, the process might be run on the full sequence. Then you might inspect the result, modify it using the **Pattern Assign** functions, perhaps try another run, and identify some satisfactory parts from the scores listed. You should then discard the ambiguous assignments and rerun the program with the correct residues used as anchor points. If several such iterative processes fail to unambiguously determine the complete assignment, then some additional information should be input, such as more accurate scoring or some new patterns.

The results are stored as assignment pointers for all frequencies of the patterns (and set as the current specified frequencies). There should not be any residue named “null” in the molecule, or its assignment will be discarded.

Optionally, some parameters of the simulated annealing might be adjusted (scaled by a factor of 0.1 to 10) according to the complexity of the problem:

Initial temperature, number of iterations: if most parts of the sequence are well defined, these parameters can be decreased to speed up the program.

Sequential / Individual factor: weight is accorded to the neighbor information, relative to the spin-system fit scores.

## Task: Resonance assignment

When you assign particular resonances or frequencies in the patterns, you need to use the Insight II atom names if you plan to use NMR_Refine to generate or refine structures. If you make the assignments through the control panels, the atom names are automatically correct. The usual form of the so-called nmrspec is: 1:RESIDUENAME_RESIDUENUMBER:ATOMNAME(NUMBER) (e.g., 1:VAL_4:HN). If you need to use pseudoatoms, then the spec-

ification is: 1:RESIDUENAME_RESIDUENUMBER:ATOM-NAME(NUMBER)*.

For example, one of the methyl groups in valine would be named 1:VAL_4:HG1*, which encompasses atoms 1:VAL_4:HG11, 1:VAL_4:HG12, and 1:VAL_4:HG13. The methyl group in alanine would be named 1_ALA_23:HB*, which encompasses methyl protons 1:ALA_23:HB1, 1:ALA_23:HB2, and 1:ALA_23:HB3.

## Task: Peak assignment

After the resonance assignment is finished, you can try to assign your peaks (usually in an NOE spectrum). It is very important to have spectrum-specific shifts for all patterns that are as good as possible, since otherwise peak assignment can be very ambiguous. You can adjust the spectrum-specific shifts manually or automatically. Preferably you would do it both ways: first automatically, then manually for what was missed. After this is done you can make auto-assignments.

You can approach the automated assignment in different ways: you can use a linear chain as a model and only assign intraresidue and sequence peaks, as well as peaks having unique frequency assignments (that is, peaks that have only one possible assignment in each dimension), or you can use a homology or a low-resolution starting model (for example, from X-ray crystallography) and assign based on those distances. Or you can make ambiguous assignments and then later use those assignments as overlap restraints in SA or rMD protocols.

After a set of assignments is generated and (possibly manually) verified, structures are generated based on them. You can refine your assignments based on the new model or based on despots (where restraints are highly violated or the structure is bad).

## Task: Restraint generation

A crucial step in generating NOE distance restraints is to define suitable scalar peaks, that is, those peaks for which the assigned atoms

are in a rigid part of the molecule and which are associated with well defined distances. These peaks are often referred to as “reference peaks” and are used to calibrate the conversion of peak intensities (volumes) to distances. As a rule of thumb, it is always safe to use good-intensity, clean, non-overlapped peaks as scalars. One category would be $H_{\beta 1}$ to $H_{\beta 2}$ methylene peaks in the same residue. You can also use an intraresidue $H_N$-$H_\alpha$ peak as a scalar, since the variability of this distance is small across different secondary structures. If you use either single mixing time (**Single tm**) volumes or volume buildups through fitting (**Fit First N tm**) for calculation of the restraints (**Calculation Method**), you need to have scalar peaks for which the distance is roughly equal (e.g., use only methylenes or only intraresidue $H_N$-$H_\alpha$ peaks).

On the other hand, if you use an empirical fit of volumes versus distances (**Calculation Method** is **Empirical Fit**), then you typically need several types of peaks–representing some short, some medium, and some longer distances. FELIX then fits an empirical function through the volume / distance pairs, and the volume is converted using this empirical fit.

If you use a 2D NOESY spectrum, you may want to control which peaks' volumes (**Symmetry Selection**) are converted to distance restraints. **Use All** converts all peaks (and then only restraints are generated for the first occurrence of the symmetric pair), **Select Regions** uses different sides of the diagonal in certain regions of the spectrum, and **Use Weaker** uses only the weaker peak of a symmetry-related pair.

Converting the calculated distance to a restraint can happen in many different ways (**Method**): **Exact Distance**, **S-M-W Bins**, V**dW-Exact**, **Percentage**.

- The most dangerous method is using the exact distance as a restraint (**Exact Distance**)–in this case the lower and upper bounds become that calculated distance.
- The most popular method is to carry out a categorization into the strong, medium, and weak categories (**S-M-W Bins**) and to store the corresponding bounds in the restraint.
- The third method is to set the lower bound to the van der Waals distance and the upper bound to the calculated distance (**VdW-Exact**).

- In the fourth method (**Percentage**), you specify a percentage by which the calculated distance will be decreased and increased to yield the lower and upper bounds.

There is always a danger in doing a totally automated conversion--there can be overlapped peaks where the measured volumes do not represent the real volumes. To avoid over-restraining when using such peaks, FELIX can handle this type of peak differently, if you use the **Partial Overlap** option to check how much the area of integration is overlapped for each peak. You can then define a threshold (**Area Threshold**) by which the algorithm can skip to generate restraints from those overlapped peaks (**Discard**) or you can use a different method to generate bounds from calculated distances, (e.g., **Use as Qual**, which generates only qualitative upper-bound restraints).

## Task: Checking and redefining restraints

After assigning NOE peaks and generating restraints, you would typically run structure calculations either by using distance geometry (the DGII command within Insight II's NMR_Refine module) or simulated annealing (the **MD_Schedule** command within Insight II's NMR_Refine module).

Usually the assignment / restraint generation and structure calculation are done in an iterative way. That means that, after generating a set of structures, you need to analyze the structures and find the so-called hotspots (e.g., where many restraints are violated).

Those hotspots can result from misassignment or overlapped peaks. For misassignment, you use a list to reassign (or unassign) those peaks. You can use a simple ASCII file which just has the two (or three or four, depending on dimensionality) names in a row separated by a blank. You can also create a list within Insight II using this procedure:

1. First load the restraints on all the refined molecules in Insight II using the NMR_Refine modules' **Restraints/Read molname*** command.
2. Execute the NMR_Refine modules' **Distance/List** command. On the resulting output file run, the provided **numvioltofelix** script

redirects the output to another file. This is the file you can use in the **Filename** parameter.

3. Proceed similar to the previously described **Manual Assign Singly** action, but instead of selecting the peaks with the cursor, the peaks automatically will be centered in your current frame and the corresponding control panel will appear.

Hotspots can also be due to erroneous restraints. If so, the **NOE Distance Redefine** option in the **Measure/DISCOVER Restraints** menu can help to loosen, tighten, or delete the restraints showing the highest violations (or the most violations within a family). To do this you must:

1. Load the restraints on all the refined molecules in Insight II using the NMR_Refine modules' **Restraints/Read <*molname**>** command.
2. Execute the NMR_Refine modules' **Distance/List** command. On the resulting output file run, the provided **numvioltofelix** script redirects the output to another file.
3. Use the **Measure/DISCOVER Restraints/NOE Distance Redefine** command on this file. You should specify the **Restraint** entity that you want to work with (usually the **msi:noe_dist**) and the **Buildup Rate Calculation Method**. The program brings up a violation table, through which you can zoom in on each peak for which the defined restraint was violated and can report the calculated distance. The table also contains the restrained values and the violation statistics.
4. From the violation table you can redefine the restraints, delete restraints, or delete assignments.

# 3 Introduction to the Menu Interface

## About this chapter

This chapter is intended to provide a broad overview of the capabilities found within FELIX 2002. The program has evolved in response to a changing and growing NMR data-processing environment; therefore, we strongly recommend that you review this chapter after installing the software.

## About FELIX 2002

FELIX is an interactive program for processing, displaying, and analyzing data acquired on nuclear magnetic resonance spectrometers. To enhance the program's utility without compromising its flexibility, FELIX includes two separate user interfaces.

- The command-line interface (FELIX Command Language, or FCL) provides experienced users with direct access to the FELIX command statements.
- The interactive menu-driven interface. The menu interface reduces the need to memorize a large list of FELIX commands and provides access to the data-processing and analysis utilities using intuitive descriptions and a mouse-driven cursor.

This guide focuses on the interactive menu-driven interface. Advanced users should consult the FELIX *Command Language Reference Guide* for more in-depth information about FELIX capabilities.

## The software license

Each software release is distributed with FLEXlm license management. Please see the Accelrys *Products System Guide* for more detailed information.

## The program modules

FELIX 2002 contains the 2D and ND spectral processing module, the Assign module for biomolecular resonance assignment, and the Autoscreen module for analysis of receptor-ligand binding spectra. Each module is enabled through license control. If the ND module is enabled, then you can process and analyze 3D and 4D data, and the corresponding commands are accessible through the menu interface.

## Getting started

> These instructions presume that you have successfully installed FELIX on your computer and are ready to begin.

To start the program, double click the **Felix** icon on your desktop, or select **Start > Programs > Accelrys Felix 2002 > Felix 2002**.

## Completing a session

To exit FELIX and return to your operating system, select the **File/Exit** command. If changes have been made to the FELIX database (which contains spectrum information such as cross peaks, baseline points, etc.), a dialog box appears prompting you to **Save** or **Discard** the changes before exiting. You may also click the **Return** button to continue working in FELIX. If no changes have been made to the

database during the current session, you may exit FELIX without supplying confirmation.

# The initialization macro

At startup, FELIX tries to execute a macro called *init.mac* to initialize the program. (See "Macros" for more details on macros and how to use them in FELIX.) This file is located in the macros\mac folder under the FELIX installation directory. Once it finds the macro, FELIX executes it and displays the menu interface.

Please see Appendix C, *FELIX Startup*, for more information on the init.mac file.

## Some FELIX initialization options

FELIX allows you to modify the operating environment by setting several reserved symbols in the initialization macro, init.mac. The symbol *blkwht*, switches the black and white entries in the color map: when *blkwht* is set to 1, the background of the graphical interface is white. The symbol *objmem* sets a ceiling for the amount of memory that FELIX can use to store graphics objects. By setting *objmem* to a value (in bytes) below the available free system memory, you can prevent FELIX from exhausting the available memory (and consequently crashing) when it makes a graphics object. Instead, when FELIX reaches the ceiling, it closes the object and warns you.

FELIX reads these options only when it starts. You must restart FELIX if you change these options while FELIX is running.

# Workspaces

FELIX provides several different types of workspaces for processing and analyzing data.

## 1D workspace (work)

The 1D workspace (work) is the most important workspace, since most of the commands issued within FELIX directly operate on data in this space.

## Additional 1D workspaces (buffers)

The second type of workspace (buffers) is used when you want to save data for later analysis. For example, if you want to compare two 1D spectra acquired at different temperatures, additional data-storage areas (buffers) are useful. You can also store apodization functions (windows) in the buffers. It is faster to calculate the window function once and save it in a buffer than it is to recalculate the function every time it is used.

> **Note:** For a more detailed description of work and buffers and their most common uses, see Chapter 4, *Processing, Visualization, and Analysis Interface (1D/2D/ND)*

## N-dimensional matrix workspace (matrix)

The matrix type of workspace is used to process multidimensional data—data containing more than one time or frequency dimension. Data in the matrix workspace may be displayed and analyzed using FELIX commands and menus similar to those used when analyzing 1D data.

For a complete discussion of multidimensional matrix manipulation and transformation, see Chapter 4, *Processing, Visualization, and Analysis Interface (1D/2D/ND).*

# Memory allocation

FELIX lets you define the size of the 1D workspace and the number of 1D buffers. Before working with experimental data, you first should configure the program's memory for your specific application. The default memory allocation is set in the init.mac macro that

executes automatically at program startup. The allocation can be changed by selecting the **Preference/Memory** command.

If you try to exceed the allocated workspace size at any point during your data workup, or if you try to access nonallocated buffers, FELIX displays an error message. In some instances (for example, reading a large data file), the program automatically prompts you to reconfigure FELIX's memory. In other instances (storing data to a non-configured buffer), you must reconfigure your memory by selecting the **Preference/Memory** command.

> **Caution:** When memory is reallocated, all information that is stored in the 1D buffers is lost. In addition, if a matrix is currently open, it is automatically closed before memory reconfiguration.

You cannot define the size of the matrix workspace, since it is determined by the size of the matrix when you select the **File/Open** command. A very large matrix requires more memory for the matrix workspace.

Sometimes FELIX displays the message "Not enough memory" when you open a matrix or perform certain actions that need temporary memory (e.g., **Linear Predict** or **Optimize Peaks**). If this happens, reconfigure the memory allocation to use a smaller 1D workspace and fewer 1D buffers. To see the current memory allocation, select the **Preference/Memory** command.

## File directories and prefix definitions

FELIX allows you to define pathnames or prefixes for each filetype. This makes it possible for each type of file to reside in a separate directory or even on a different device. For example, you might have 50 macro files, 10 data files, and 2 matrix files. If these files were stored in the same directory, it would be difficult to locate the pair of matrix files among the many macro and data files. It is better to organize the macro files in a single macro directory and the matrix and data files in a different directory.

Prefixes can be viewed or changed by selecting the **Preference/Directories** command. Within the control panel that appears, the

current prefixes are shown. To change these, activate the appropriate entry box and enter the new prefix. Clicking **OK** saves the changes; clicking **Cancel** quits the control panel without making any changes.

> **Caution:** In the init.mac file, most of the prefixes are defined as relative to the "current working directory" (**Start In** folder), designed as "**.**". You can verify the **Start in** folder by clicking the FELIX alias icon with the secondary mouse button ("right-click") and selecting **Properties**, then selecting the **Shortcut** tab. After FELIX is started up, whenever you open or create a database file (*.dba* file), the current working directory is automatically changed to the folder where the .dba file is located. You can change the current working folder by selecting the **Preference/Directory** command.

# The menu interface

FELIX contains an interactive menu system for data processing, displays, and spectrum analysis. By default, the menu interface automatically appears when the program starts. The purpose of the menu interface is to simplify NMR data analysis and speed repetitive processing tasks. The menus provided contain most of the functions needed for processing and analyzing 1-, 2-, 3-, and 4-dimensional data.

In this guide we refer to the various menu items as follows:

- Horizontal menu: **menu bar**.
- Vertical menus: **pulldown menus**.
- Vertical lists of menu items that appear when you select an item that ends with an arrow: **pullright menus**.

## Control panels (dialog boxes)

Control panels are a basic component of the FELIX menu system and are used for communicating text and numbers to FELIX. Any activity that requires information from you opens up a control panel

and waits for you to provide it. Control panels contain several types of information-entry controls:

*Entry box*

The **entry box** is where you can type text or numbers that the program needs; for example, you may enter filenames, plot scale factors, etc. Entry boxes have a *type* of integer, real, or character, which defines how FELIX interprets the characters entered into the box.

*Check box*

A **check box** is used to select one of a set number of options.The current option is shown in the toggle. To change the option, you can click anywhere within the toggle. Clicking several times scrolls through all the options in a circular fashion. Toggles are usually used where the unseen selections are very specific and intuitively obvious.

*Switch (radio button)*

**Switches**, or **radio buttons**, are used to choose one of a small set of options. Radio buttons are used where choosing between alternative selections is not directly obvious. To select a radio button item, click the open circle. The activation and selection of a radio button item is indicated by a contrasting circle within the radio button.

*Combo box*

Clicking on a **combo box** shows all valid choices at once. Drag the cursor to the desired choice and release the mouse button to select that choice.

*List box*

**List boxes** consist of a rectangular area that contains a vertical list of choices with up and down scrolling arrows on the right side. Clicking one choice in the list highlights and selects that choice.

*Button*

**Buttons** are used to exit control panels. When you click the **Cancel** button, the control panel is removed and the program does not update any parameters you may have entered. Clicking **OK** updates the program parameters with the current values in the control panel and removes the control panel. In some instances, choices other than **Cancel** and **OK** are available. The actions of these other buttons should be self-explanatory.

## Pointer types

While in menu mode, you will see several types of pointers (**cursors**). Each has a specific purpose and expects a certain action from you.

*Standard arrow*

The standard arrow pointer indicates that the menu interface is waiting for a click to select a menu item. The menu items that are currently active are highlighted as the arrow moves over them. The arrow pointer is also used within control panels.

*Full-window crosshair*

This pointer covers the entire FELIX graphics window and indicates that a single position (or data-point value) is needed. For example, this pointer appears when you are selecting single cross peaks.

*Spot (small crosshair)*

The spot pointer is a very small crosshair cursor. When the spot pointer appears, it indicates that the program is waiting for a push-drag-release action of the mouse for dragging out a *rubber band box*. Spot pointers are used to define data regions for expanding spectrum displays and for manually adding cross peaks to multidimensional spectra.

> **Caution:** When the spot pointer is present, you must be careful not to simply click and release the left mouse button without dragging out an area, since this might define the beginning and end point as the same value

*Half-crosshair*

The vertical or horizontal half-crosshair pointer is used when only one dimension of information is wanted and a full crosshair might be confusing. One use of the horizontal half-crosshair is for setting the 1D peak-picking threshold. One use of the vertical half-crosshair is for loading a 1D column from a 2D matrix.

*Hourglass*

The hourglass cursor is shown when FELIX is performing a length calculation or other operation.

> **Note:** If the execution of a macro is aborted because of some syntax error, the cursor may remain as an hourglass even if it is no more 'busy'. In such a case you can issue a command:
>
> ```
> cur 0 0 0 1
> ```
>
> in the command window to restore the normal standard arrow cursor.

## Accessing the FELIX command interface

You can issue FELIX commands or execute your own macros from the FELIX interface.

Click inside the FELIX command window above the status bar and type in a command, or type

```
macfile
```

to execute a macro named *macfile*. You may then process and analyze your data using FELIX command statements. You can use up and down arrows to retrieve up to 10 of the most recently issued commands.

Return to menu mode by selecting any menu item.

Please see the *FCL Reference Guide* for further information on running FELIX in command mode.

## Graphics frame layouts

FELIX can display and work with multiple graphics frames (i.e., *child windows*) within the main window. Such a frame can either display a spreadsheet (called table window) or NMR spectrum (called spectral window). This feature lets you work with multiple spectra or display regions simultaneously.

You can create a new spectral window by selecting the **File/New** command or **Window/Add New Frame**. When a new frame is opened, it automatically become current (or active). The current window is identified by a highlighted border. You can also open several frames with pre-defined layouts simultaneously by selecting the **Window/New Layout** command.

The **Window** menu provides several options for you to arrange the child windows. By default, whenever a new window (table or spectral) is open, FELIX automatically re-arranges the layout of the windows, with the tables tiled on the left side, occupying 20% of the main window area; and the spectral windows tiled on the remaining area of the main window. You can modify or turn off this feature by selecting **Preference/Frame Layout** from the main menu and set **Action** to **None**.

> **Note:** FELIX displays one menu bar and possibly one tool bar at a time. When one or more table windows are opened, only the menu bar and the tool bar of the currently activated window are visible. If you want to select a certain menu item or tool bar icon, be sure to click the corresponding window first to activate its menu and tool bar (if any).

Graphics frames can be closed, moved, resized, and maximized using the tools on the frame header.

## Other graphics frame features

*Switching between frames*

If the frames are not maximized, you can switch between FELIX frames by clicking anywhere in a frame. The newly selected frame is brought to the front and the picture is updated.

*Moving frames*

To move a frame, click and hold anywhere in the middle of the frame's header. A grey box appears that lets you move the frame around in the FELIX graphics window. Releasing the mouse button places the frame at the new position.

*Resizing frames*

To resize a frame, click anywhere on an edge of the frame and hold down the primary mouse button. A grey box appears that lets you adjust the position of the edge selected. If you select a corner of the frame, then you can resize the frame along both edges connected to that corner.

*Minimizing frames*

To minimize a frame, click the button with the underscore bar in the upper-right corner. To return the frame to its original size and position, double-click the icon.

*Maximizing frames*

Frames may also be maximized to fill the entire FELIX graphics screen. To maximize a frame, click the maximize widget (the button with the square shape in the upper-right corner). This expands the corners of the current frame to fill the entire region of the FELIX window. To return the frame to normal size, click the maximize widget (now showing two overlapping square shapes) again.

> **Note:** If one frame are maximized, an easy way to switch between frames is to press the Ctrl-Tab keys simultaneously.

### *Closing frames*

To close a frame, click the button with the cross in the upper-right corner. This activates the frame that was last active before the newly closed frame.

## Macros

FELIX macros are the most powerful and innovative feature of this NMR processing software. A collection of FELIX macros is responsible for generating the entire menu interface. A macro is an ASCII text file that contains a list of FELIX commands which are executed in sequence. Macros can employ symbols and expressions, so that you can define specific procedures applicable to your needs. In addition to being able to perform any FELIX command, macros can also contain flow-control statements to define loops, conditional and unconditional branching, and interaction with the user. Further descriptions of macros and their functions are found in the *FCL Reference Guide*.

## The database

FELIX contains a sophisticated database for storing spectrum information. Database files are identified by the .dba extension. By default, when FELIX starts, a database called *file.dba* is built and opened. Throughout your session, spectrum information such as baseline points, integral segments, cross-peak lists, and volume lists are stored in the database.

For transient processing applications like simple 1D peak picking, you may use the same database over and over, rewriting the information as needed. For more involved applications that require repeated use of the database, we recommend closing the default database (file.dba) and building a new one with an appropriate name. This helps prevent you from inadvertently deleting valuable spectrum information.

> **Note:** Activities such as integration, baseline-point picking, and cross-peak picking require a database to be open and cannot be executed without database access.

## The database editor

You can display and edit the entities (tables) in the FELIX database. You can either go through the database contents viewer (see next section) to access all the tables, or use the **Edit** pulldown to access some commonly used tables. Once a table is opened, you can then view, search, and modify the contents of the selected database entity.

## The database contents viewer

The DBA contents viewer lets you peruse the database directory and entity structure graphically. Open this viewer by selecting the **Edit/ Table** command.

On the left of the resulting control panel are the directories within the database.

*View directory contents*

To view the contents of a database directory, select the desired directory with the mouse, then click the **Filter** button (or double-click the desired directory). On the left, you will see any subdirectories in the directory you selected. On the right, you will see the names of the entities that contain spectrum and other information.

*View entity contents*

To view the contents of an entity, select that entity and click **OK** (or you can double-click the entity).

*Quit viewer*

To quit the graphical entity viewer, click **Cancel**.

You can also move quickly between a range of values within a slider by clicking anywhere on the slider.

# 4 Processing, Visualization, and Analysis Interface (1D/2D/ND)

The processing, visualization, and analysis functions in FELIX 2002 are accessible throughout the interface.

## File pulldown

The **File** pulldown (also accessed by simultaneously pressing the <Alt> and <f> keys on the keyboard, abbreviated as <Alt>+f) contains menu items for reading and / or saving datafiles, database files, and entities. It also contains menu items for importing spectrometer data, importing processed data, printing, and exiting the program. Each general menu item (command) in the **File** pulldown is described below.

### File/New

The **File/New** command (<Alt>+fn):

- Brings up a new spectral frame if there is no existing frame.
- Prompts you to choose to either bring up a new frame, or to create a new database file or a new matrix.

FELIX displays a control panel with a list of the files in the database or matrix directory. Select the desired type (**File Type**): either database or matrix. Enter the name of the file in the **Selection** box. To build a matrix, set up the sizes and select the appropriate dimension.

## File/Open

Use the **File/Open** command (<Alt>+fo or <Ctrl>+o) to perform these tasks:

- Open an existing datafile for processing and analysis.
- Open an existing database file, matrix, molecule, or macro for execution.

> **Note:** Before you can perform any operations on a data matrix, you must open the data using this menu item.

FELIX displays a control panel with a list of the files in the data, matrix, molecule, or macro directory. Select the desired file by clicking the filename or by typing in the name. The NMR datafiles can be in any of these formats:

- Old FELIX format
- New FELIX format
- Bruker files from AMX or newer spectrometers (ser or fid)
- Varian FID files
- JEOL Alpha or Delta files
- FELIX FID files.

To read in FELIX old-format data, change the **Format** to **Old Format**. The default is **New Format** for FELIX datafiles.

If you choose to open an existing database file for storing spectral information and a database is currently open, FELIX closes the current database and opens the newly selected one. If the first database has changed in some way, once you select the new database, FELIX prompts you to save or discard the changes to the first database before closing it.

## File/Save

Use the **File/Save** command (<Alt>+fs) to save the current database or the current data in the workspace to disk. If the file already exists, FELIX prompts you to quit or overwrite the file.

## File/Save As

Use the **File/Save As** command (<Alt>+fa) to save the current database, the current data in the workspace, the current matrix, or theoretical matrix to disk with a new name. If the file already exists, FELIX prompts you to quit or overwrite the file. The new database now becomes the current database.

## File/Close

Use the **File/Close** command (<Alt>+fc) to close the current database, the current data in the workspace, or the current matrix. For a write-enabled matrix, FELIX stores the current parameters (displayed limits, display type, current entities) in the header. FELIX uses these parameters the next time you open this matrix to set the limits, display color, peak, and other entities.

## File/Import

Use the **File/Import** (<Alt>+fi) command to generate a pullright menu of controls used to import database entities of tables, including a generic table-import command and some special table-import commands.

### File/Import/Table

Use the **File/Import/Table** (<Alt>+fit) command to read an ASCII file from disk into the binary FELIX database.

> **Caution:** This action is very format-specific. The entity file you read must be in an acceptable format. Use **Selection** to select an ASCII table file. Use **Table Name** to specify an entity that will be created in the database file.

### File/Import/Peaks

Use the **File/Import/Peaks** command (<Alt>+fip) to read an ASCII file that contains peak information into the peak entity. FELIX prompts you to specify an input filename as well as an entity name. Supported formats include:

- FELIX peak entity
- Insight II .pks and .asn formats
- NMRCompass peak file (.dat)
- ASCII

### File/Import/Plot Limits

Use the **File/Import/Plot Limits** command (<Alt>+fil) to read and redefine the plot limits on the current matrix displayed. The file you read in should have been exported earlier using the **File/Export/Plot Limits** command.

## File/Export

Use the **File/Export** (<Alt>+fe) command to open a pullright menu of controls to export database entities of tables. These controls include a generic table-export command and some special table-export commands.

### File/Export/Table

Use the **File/Export/Table** (<Alt>+fet) command to write a FELIX database entity to an ASCII file on disk. In the first control panel, select an existing entity. In the next control panel, enter an ASCII table file as the **Selection**.

### File/Export/Peaks

Use the **File/Export/Peaks** command (<Alt>+fep) to write the peak entity into an ASCII file that contains peak information. FELIX displays the file contents in a text window.

FELIX prompts you to specify a filename as well as the current peak entity name. The supported formats include:

- FELIX peak entity
- Insight II .pks and .asn formats
- NMRCompass peak file (.dat)

- ASCII of all peaks or only of the assigned peaks. These latter two files can contain peaks with multiple competing assignments if the peak assignments were done through the Assign module.

### File/Export/Plot Limits

Use the **File/Export/Plot Limits** command (<Alt>+fel) to write out the current plot limits of the displayed matrix.

## File/Convert

Use the **File/Convert** (<Alt>+ft) to open a pullright menu of converting (filtering) controls. FELIX 2002 converts only processed matrices.

### File/Convert/Matrix

Use the **File/Convert/Matrix** (<Alt>+ftm) command to convert processed spectrometer data from formats that FELIX can filter (translate) into the FELIX matrix format. This makes a new FELIX matrix, which can later be read into the program. These formats are supported:

- Bruker 2rr or 3rrr files
- Varian phase files preprocessed with the sv2d or sv3d Varian macros
- NMRCompass .spc files
- NMRPipe ft2 or ft3 files
- TRIAD .nmr files

## File/Print Setup

Use the **File/Print Setup** command (<Alt>+fr) to set up the printer and other properties of the printing job.

## File/Print Preview

Use the **File/Print Preview** command (<Alt>+fv) to preview the plot.

## File/Print

Use the **File/Print** command (<Alt>+fp) to open a hardcopy plot of the spectra in the current frame.

> **Note:** FELIX 2002 uses the same parameters for screen displaying and hardcopy plotting. To control the size and location of the spectral plot on the paper, select **Preference/ Plot Parameters**; next, click the **Place** button.

## File/Licenses

Use the **File/Licenses** command (<Alt>+fl) to release any features that you don't currently need, allowing others in your workgroup to use them. Also, if one feature is being used by another concurrent user of FELIX 2002 when you started FELIX but becomes available later in your session, select this function to use the feature in your session.

## File/Log File

Use the **File/Log File/Open** command (<Alt>+ffo) to open a log file, which will record the FELIX commands executed from this point.

Use the **File/Log File/Close** command (<Alt>+ffc) to close a log file and terminate the recording of the FELIX commands executed.

Use the **File/Log File/Replay** command (<Alt>+ffr) to open a log file and repeat the commands recorded in the log file.

## File/Exit

Use the **File/Exit** command (<Alt>+fx) to exit FELIX. If changes have been made to the FELIX database, a FELIX dialog box prompts

you to **Save** the database and exit and to **Save** the current environment (which can be restored later).

# View pulldown

The most common way to manipulate displays is to access the **View** pulldown.

## View/Plot

Use the **View/Plot** command (<Alt>+vp or <Ctrl>+p) to redraw the current workspace.

## View/Plottype

To select a type of plot other than the current one, select the **View/Plottype** command (<Alt>+vt). Alternatively, you can select from the combo box in the main tool bar.

FELIX 2002 displays NMR data in the following ways:

- 1D slice
- 2D intensity plot
- 2D contour plot
- Null
- 2D overlay plot

### 1D

FELIX presents the 1D data values in the workspace.

### Intensity plots

In the intensity plot, FELIX presents data values above a set contour level as a filled rectangular region.

The intensity plot is drawn more quickly and coarsely than a contour plot. It is useful for uncritical examinations of the data; for

example, when you are adjusting the contour levels or selecting a region for display as an expanded contour plot.

### Contour plots

The contour plot displays the data as closed lines that follow adjacent data points of a set value. In other words, each closed contour line is the cross-section of a peak taken at a set value. Contour plots display the matrix in the most precise form and are used for data analysis.

### Null

FELIX erases the plot and leave it blank.

### Overlay plots

The contour plot displays the data as closed lines that follow adjacent data points of a set value. In other words, each closed contour line is the cross-section of a peak taken at a set value. Contour plots display the matrix in the most precise form and are used for data analysis.

## View/Tile and Strip Plot

In addition to the standard plot types for *N*D data, FELIX can produce *tile plots*. A typical plot shows a matrix region as several small peaks separated by much space. A tile plot eliminates the intervening space and shows the peaks in much greater detail. This allows FELIX to displays features that are far apart in the actual matrix t on a single plot.

A special tile plot in which one of the axes runs across the full spectrum is a *strip plot*. To turn these types of plot on and off, use the **View/Plottype/Tile/Strip Plot** toggle.

## View/Limits

The items in the **Limits** menu provide numerous options for quickly changing the expansion limits.

### View/Limits/Set Limits

Use the **View/Limits/Set Limits** command (<Alt>+vls or <Ctrl>+s) to specify the spectral limits in real time. When you select this command, FELIX displays a small crosshair cursor. Move this cursor to the region you want to expand and drag out the region to be expanded. FELIX displays the expanded plot when you release the mouse button. If you select too small a region (fewer than four data points in any dimension), FELIX assumes that you accidentally pressed and released the mouse button without selecting a region and displays a warning message without changing the plot limits.

### View/Limits/Manual Limits

Use **View/Limits/Manual Limits** command (<Alt>+vlm) to expand the plot to specific values for the limits in either ppm or points. If you have a 3D or 4D spectrum, specifying 1D limits (or 2D limits for a 4D spectrum) as one point allows you to plot a plane instead of a cube.

### View/Limits/Full Limits

Use the **View/Limits/Full Limits** command (<Alt>+vlf or <Ctrl>+f) to restore the full spectrum display.

### View/Limits/Last Limits

Use the **View/Limits/Last Limits** command (<Alt>+vll) to redisplay the plot with the previous limits. Use this command to rapidly restore an expanded plot.

### View/Limits/Transpose Limits

Use the **View/Limits/Transpose Limits** (<Alt>+vlt>) command to exchange the current horizontal plot limits with the current vertical plot limits. FELIX switches the limits and redraws the plot. This is an easy way to jump to the symmetry-related region in a homonuclear 2D matrix or a homonuclear part of a 3D or 4D matrix.

### View/Limits/Order of Plot

Use the **View/Limits/Order of Plot** (<Alt>+vld) command to control the order in which the matrix dimensions appear on the display.

By default, D1 is plotted along the x axis, and D2 is plotted along the y axis. When you select this command, FELIX displays a control panel where you specify the plot order.

- For a 2D matrix the toggle has two possible states:

  **D1 - D2** specifies that the plot be ordered with D1 along the x axis and D2 along the y-axis

  **D2 - D1** specifies that the plot be ordered with D2 along the x axis and D1 along the y axis.

- For a 3D or 4D matrix you can explicitly specify what the order should be. A FELIX control panel prompts you for the dimension numbers that you want displayed along the x, y, and z-axes.

## 1D vector views

If you have displayed a 2D, 3D, or 4D matrix you can select 1D vectors using the following set of commands.

### View/Limits/1D Horizontal

Use the **View/Limits/1D Horizontal (**<Alt>+vlr) command to select a row from the matrix for viewing as a 1D vector plot. Select the row using a horizontal line cursor by moving the cursor over the desired row and clicking the left mouse button. FELIX loads the selected row into the 1D workspace and plots it in the display. The **plottype** popup on the iconbar changes to **1D**.

### View/Limits/1D Vertical

Use the **View/Limits/1D Vertical** (<Alt>+vli) command to select a column from the matrix for viewing as a 1D vector plot. Select the column using the vertical line cursor. FELIX loads the selected column into the 1D workspace and plots it in the display. The **plottype** popup on the iconbar changes to **1D**.

### View/Limits/1D Orthogonal

Use the **View/Limits/1D Orthogonal** (<Alt>+vlg) command to select a 1D slice that is orthogonal to the current 2D slice of the 3D or 4D matrix for viewing as a 1D vector plot. Select the slice using the crosshair cursor by moving the cursor over the desired spot and

clicking the left mouse button. FELIX loads the selected slice into the 1D workspace and plots it in the display. The **plottype** popup on the iconbar changes to **1D**. For a 4D spectrum the slice runs along the third (**order3**) dimension.

### View/Limits/1D Orthogonal 2

Use the **View/Limits/1D Orthogonal 2** (<Alt>+vln) command to select a 1D slice that is orthogonal to the current 2D slice of the 4D matrix for viewing as a 1D vector plot. Select the slice using the crosshair cursor. FELIX loads the selected slice into the 1D workspace and plots it in the display. The **plottype** popup on the iconbar changes to **1D**. The slice runs along the fourth (**order4**) dimension.

## 2D plane views

If you have displayed a 3D or 4D matrix you can select 2D planes for viewing using the following set of commands.

### View/Limits/Real-time Plane

The **View/Limits/Real-time Plane** (<Alt>+vl-) command brings up the plane-selection slider and lets you view a 2D plane from a 3D or 4D spectrum.

### View/Limits/Select Plane

By using the **View/Limits/Select Plane** (<Alt>+vlp) command you can specify a 2D plane for display. A FELIX control panel prompts you to specify the matrix slice. Select the switch beside the plane that interests you. Enter the value of the point or ppm value for the specific plane in the corresponding box. Set whether ppm or point should be used. Then, select **OK**; FELIX displays the requested 2D slice.

### View/Limits/Horizontal Plane

The **View/Limits/Horizontal Plane** (<Alt>+vlh) command lets you use a cursor to select a slice that is orthogonal and oriented horizontally relative to the current 2D slice of a 3D or 4D spectrum. Selecting this command prompts FELIX to display a horizontal line cursor.

Click the desired line on the currently displayed 2D slice to display the selected orthogonal slice in the frame.

The item in the pulldown menu has a variable directionality connected, that is, if a D1–D2 plane is current, the command appears as **Horizontal Plane (D1-D3)**.

For a 4D matrix, use the **View/Limits/Horizontal 2 Plane** (<Alt>+vlz) command to select along the other orthogonal direction. For example, if your current display is D1–D2, then both a **Horizontal Plane (D1-D3)** and a **Horizontal 2 Plane (D1-D4)** command are available).

## View/Limits/Vertical Plane

Use the **View/Limits/Vertical Plane** (<Alt>+vlv) command to use a cursor to select a slice that is orthogonal and oriented vertically relative to the current 2D slice of a 3D or 4D matrix. FELIX display a vertical line cursor. Click the desired line on the currently-displayed 2D slice; FELIX displays the selected orthogonal slice in the frame.

The item in the pulldown menu has a variable directionality connected, that is, if a D1–D2 plane is current, the command appears as **Vertical Plane (D1-D3)**.

.For a 4D matrix, an additional command (**View/Limits/Vertical 2 Plane** (<Alt>+vlz)) enables you to select along the other orthogonal direction (for example, if your current display is D1–D2, then both a **Vertical Plane (D1-D3)** and a **Vertical 2 Plane (D1-D4)** command are available).

## View/Limits/Orthogonal Plane

Use the **View/Limits/Orthogonal Plane** (<Alt>+vlo) command to use a cursor to select a slice that is orthogonal relative to the current 2D slice of a 4D matrix. FELIX displays a crosshair cursor. Click the desired spot on the currently displayed 2D slice to display the selected orthogonal slice in the frame.

The item in the pulldown menu has a variable directionality connected. That is, if a D1–D2 plane is current, the command appears as **Orthogonal Plane (D3-D4)**.

## View/Export Limits

You can redefine limits in the *current* frame as with the previous **View/Limits** commands, and you can also define *new* limits in another frame using the **View/Export Limits** (<Alt>+ve) commands, as described below.

### View/Export Limits/Export Limits

The **View/Export Limits/Export Limits** (<Alt>+vee) command allows you to export the plot limits from the current frame to another frame.

### View/Export Limits/Export Reference

The **View/Export Limits/Export Reference** (<Alt>+vef) command allows you to export the spectral reference parameters from the current frame to another frame.

### View/Export Limits/Export 1D Horizontal

The **View/Export Limits/Export 1D Horizontal** (<Alt>+ver) command lets you select a row from the matrix for viewing as a 1D vector plot in another frame. Select the row using the horizontal line cursor. FELIX loads the selected row into the 1D workspace and plots it in the selected frame. The frame where the new slice is to be displayed should already exist.

### View/Export Limits/Export 1D Vertical

The **View/Export Limits/Export 1D Vertical** (<Alt>+vei) command allows you to select a column from the matrix for viewing as a 1D vector plot. Select the column using the vertical line cursor. FELIX loads the selected column into the 1D workspace and plots it in the selected frame. The frame where the new slice is to be displayed should already exist.

### View/Export Limits/Export 1D Orthogonal

The **View/Export Limits/Export 1D Orthogonal** (<Alt>+veg) command allows you to select a 1D slice that is orthogonal to the current 2D slice of the 3D or 4D matrix for viewing as a 1D vector plot. Select

the slice using a crosshair cursor by clicking the desired spot. FELIX loads the selected slice into the 1D workspace and plots it in the selected frame. The frame where the new slice is to be displayed should exist. For a 4D spectrum, the slice runs along the third (**order3**) dimension.

### View/Export Limits/Export 1D Orthogonal 2

The **View/Export Limits/Export 1D Orthogonal 2** (<Alt>+ven) command allows you to select a 1D slice that is orthogonal to the current 2D slice of the 4D matrix for viewing as a 1D vector plot. Select the slice using the crosshair cursor by clicking the desired spot. FELIX loads the selected slice into the 1D workspace and plots it in the selected frame. The frame where the new slice is to be displayed should exist. The slice runs along the fourth (**order4**) dimension.

### View/Export Limits/Export 1D Transposed

The **View/Export Limits/Export 1D Transposed** (<Alt>+vet) command allows you to select a 1D slice that is orthogonal to the current 2D slice of the 3D matrix for viewing as a 1D vector plot. Select the slice using the crosshair cursor by clicking the desired spot. FELIX loads the transposed position slice into the 1D workspace and plots it in the selected frame. The frame where the new slice is to be displayed should exist. This feature can be useful for analysis of a 3D HCCH–TOCSY spectrum where you can display an H–H plane along a $^{13}C$ frequency. Clicking, for example, a $H_\alpha$–($C_\alpha$)–$H_\beta$ peak can give a $^{13}C$ slice at the $H_\beta$–$H_\alpha$ position, allowing you to locate the $C_\beta$ frequency.

## 2D plane exporting

If you have displayed a 3D or 4D matrix, you can export 2D planes to a different frame using the following set of commands.

### View/Export Limits/Export Horizontal

The **View/Export Limits/Export Horizontal** (<Alt>+veh) command lets you use a cursor to select a slice that is orthogonal and oriented horizontally relative to the current 2D slice of a 3D or 4D spectrum. Selecting this command prompts FELIX to display a horizontal line

cursor. Click the desired line on the currently displayed 2D slice to display the selected orthogonal slice in the selected frame. The frame where the new slice is to be displayed should exist.

The item in the pulldown menu has a variable directionality connected, i.e., if a D1–D2 plane is current, the command appears as **Export Horizontal (D1-D3)**.

For a 4D matrix use the **View/Export Limits/Export Horizontal 2** (<Alt>+vez)) to select along the other orthogonal direction. For example, if your current display is D1–D2, both an **Export Horizontal (D1-D3)** and an **Export Horizontal 2 (D1-D4)** command are available.

## View/Export Limits/Export Vertical

The **View/Export Limits/Export Vertical** (<Alt>+vev) command lets you use a cursor to select a slice that is orthogonal and oriented vertically relative to the current 2D slice of a 3D or 4D matrix. Selecting this command prompts FELIX to display a vertical line cursor. Clicking the desired line on the currently displayed 2D slice prompts FELIX to display the selected orthogonal slice in the selected frame. The frame where the new slice is to be displayed should exist.

The item in the pulldown menu has a variable directionality connected, i.e., if a D1–D2 plane is current, the command appears as **Export Vertical (D2-D3)**.

For a 4D matrix use the **View/Export Limits/Export Vertical 2** (<Alt>+ve2) command to select along the other orthogonal direction. For example, if your current display is D1–D2, both an **Export Vertical (D2-D3)** and an **Export Vertical 2 (D2-D4)** command are available.

## View/Export Limits/Export Orthogonal

The **View/Export Limits/Export Orthogonal** (<Alt>+veo) command lets you use a cursor to select a slice that is orthogonal relative to the current 2D slice of a 4D matrix. Selecting this command prompts FELIX to display a crosshair cursor. Clicking the desired spot on the currently displayed 2D slice displays the selected orthogonal slice in the selected frame. The frame where the new slice is to be displayed should exist.

The item in the pulldown menu has a variable directionality connected, i.e., if a D1–D2 plane is current, the command appears as **Export Orthogonal (D3-D4)**.

### View/Export Limits/Export Transposed

Use the **View/Export Limits/Export Transposed** (<Alt>+ved) command to use a cursor to select a slice that is transposed relative to the current 2D slice of a 3D matrix. FELIX displays a crosshair cursor. Click the desired spot on the currently displayed 2D slice; FELIX loads an orthogonal 1D slice which is also transposed.

Then FELIX searches through that 1D slice to find the largest peak and uses that peak to define a new plane, which is then displayed in the selected frame. This is useful for the analysis of a 3D HCCH–TOCSY spectrum where you would display an H–H plane along a $^{13}C$ frequency. Clicking, for example, an $H_\alpha$–($C_\alpha$)–$H_\beta$ peak initiates a search for the $^{13}C$ slice at the $H_\beta$–$H_\alpha$ position, allowing you to locate the $C_\beta$ frequency and export the $H_\beta$–($C_\beta$)–$H_\alpha$ plane. The frame where the new slice is to be displayed should be open.

## View/Draw Peaks

Use the **View/Draw Peaks** (<Alt>+vd or <Ctrl>+k) command to display the 1D peaks on a 1D spectrum or the ND cross-peak footprints represented by the current 1D peak or ND cross-peak entity, according to the parameters specified using the **Preference/Peak Display** command. Set the toggle **on** to display the peaks after every spectrum plot.

## View/Draw Frequencies

Use the **View/Draw Frequencies** (<Alt>+vf) command to display the frequencies in the frequency buffer as horizontal and vertical lines through the 2D plane, according to the parameters specified using the **Preference/Frequency Display** command. Set the toggle **on** to display the frequencies after every spectrum plot.

## View/Draw Integrals

Use the **View/Draw Integrals** (<Alt>+vs) command to display the integrals on a 1D spectrum, according to the parameters specified using the **Preference/Integral** command. Set the toggle **on** to display the integrals after every 1D plot.

## View/Draw Basepoints

Use the **View/Draw Basepoints** (<Alt>+vb) command to display the picked baseline points on a 1D spectrum. Set the toggle **on** to display the points after every spectrum plot.

## View/Draw Annotations

Use the **View/Draw Annotations** (<Alt>+va) command to display the annotations in the current annotation file (annfil). Set the toggle **on** to display the annotations after every spectrum plot.

## View/Draw 1D Slices

Use the **View/Draw 1D Slices** (<Alt>+v1) command to specify 1D sections (horizontal, vertical, or both) to export to a different frame (or frames) interactively using the cursor or slider(s) from the current matrix. FELIX prompts you to specify which dimension(s) to take the 1D slices from and which frame(s) is the target to export the slices. FELIX draws a line on the parent plot showing the location from which the slice was extracted.

## View/Draw Multiple 1D Slices

Use the **View/Draw Multiple 1D Slices** (<Alt>+vm) command to specify multiple 1D sections (either horizontal or vertical) to export to a different frame interactively, using the cursors. FELIX draws the selected slices on top of each other in a stack. FELIX prompts you to specify which dimension to take the 1D slices from. FELIX draws lines on the parent plot showing the location from which the slices were extracted.

## View/Draw Thick 1D Slice

Use the **View/Draw Thick 1D Slice** (<Alt>+vk) command to specify a block of 1D slices (either horizontal or vertical) using a rubber band box, and export the sum of them to a different frame. FELIX prompts you to specify which dimension to take the 1D slices from and which frame is to export the results to.

## View/Draw 2D Slices

Use the **View/Draw 2D Slices** (<Alt>+v2) command to specify 2D sections to export from the current 3D or 4D matrix to different frames interactively, using the cursor. FELIX prompts you to specify which dimensions to take the 2D slices from and the target frames to export the selected 2D sections. FELIX draws a line on the parent plot showing the location from which the slice was extracted.

## View/Tile Plot

FELIX display a tile plot as a large rectangular layout of small rectangular tiles, precisely aligned with each other. Each tile is actually the intersection of a narrow region along D1 and a narrow region along D2. A tile plot that is three tiles across and four tiles tall is the result of three regions along D1 intersecting with four regions along D2. The unit of interest is always a line segment, or region, in a single dimension; while the tile plot illustrates the intersections of these segments.

FELIX stores every tile as an entity in the database. The format of the entity is a set of line segments for each dimension, where the segment endpoints are recorded in data-point units. When you build a tile plot, FELIX reads the tile entity to determine which regions to tile.

Use the **View/Tile Plot** (<Alt>+vi) command to open a pullright menu of tile controls, which are described below.

## View/Tile Plot/Tile Plot

Use the **View/Tile Plot/Tile Plot** command (<Alt>+vie) to toggle the tile plot on or off. Use this function after a tile entity is defined with, for example, the **Tile One Peak** or **Tile Regions** command.

## View/Tile Plot/Tile One Peak

Use the **View/Tile Plot/Tile One Peak** (<Alt>+vip) command to create a tile entity and plot from a chosen cross peak. Select a cross peak from which to generate a tile plot with the crosshair cursor.

The peak's footprint defines the first two line segments. FELIX searches in the peak entity for all other peaks that align with the chosen peak. Each of the other peaks contributes at least one line segment to the new tile entity. The resulting tile plot shows all the regions in the matrix where these segments intersect. Use the **View/Tile Plot/Tile Plot Parameters** command to set alignment criteria.

## View/Tile Plot/Tile Regions

Use the **View/Tile Plot/Tile Regions** (<Alt>+vir) command to create a tile entity and plot from a user-generated set of regions. Use the small crosshair cursor to select a rectangular region. Each such region defines up to two line segments for the tile entity. Drag out additional regions to define more segments. Click the <Esc> key to quit selecting regions. The resulting tile plot shows all the regions in the matrix where these segments intersect. FELIX combines segments that are almost identical (those with significant overlap).

## View/Tile Plot/Tile Spin System

This menu item can be used only in conjunction with Assign; i.e. you must have an Assign project set up before you can use this command. Here you can define a tile plot using any combination of frequency clipboards, protopatterns, and patterns.

## View/Tile Plot/Tile ROIs

Use the **View/Tile Plot/Tile ROIs** command to create a tile entity and plot from a user-defined region of interests (ROIs).

### View/Tile Plot/Delete One Column, /Delete One Row

A tile plot may have too many line segments, so that the display shows regions that are not of interest. Use the **View / Tile Plot/ Delete One Column** (<Alt>+vio) and **View/Tile Plot/Delete One Row** (<Alt>+vid) commands to manually remove a single line segment from the entity to generate a new tile plot with fewer tiles. Use the half-crosshair cursor to delete a line segment from the entity. FELIX then displays the new reduced tile plot.

### View/Tile Plot/Tile Plot Parameters

Use the **View/Tile Plot/Tile Plot Parameters** (<Alt>+vit) command to open a control panel that affects many facets of tile plot creation and display. Set the following criteria here:

- Name of the tile entity
- Explicit width for all peak-based line segments
- Behavior for treatment of overlapping line segments
- Criteria for determining cross-peak alignment

Cross-peak alignment can be measured in terms of footprint correlation or line shape correlation, and you can set the correlation threshold to control which peaks are considered aligned.

## View/Strip Plot

### View/Strip Plot/Strip Plot

Use the **View/Strip Plot/Strip Plot** command (<Alt>+vrr) to change a strip plot to a regular plot, or to turn on a previously defined strip plot from a regular plot.

### View/Strip Plot/Strip Plot of Clipboard

The **View/Strip Plot/Strip Plot of Clipboard** menu item (<Alt>-vrc) generates a strip plot representation from the current frequencies in the clipboard.

## View/Strip Plot/Strip

The **View/Strip Plot/Strip Plot of Protopattern** menu item (<Alt>-vrp) generates a strip plot representation from the current frequencies in the selected prototype pattern.

## View/Strip Plot/Strip Plot of Spin System

The **View/Strip Plot/Strip Plot of Spin System** menu item (<Alt>-vrs) generates a strip plot representation from the current frequencies in the selected patterns (spin system).

## View/Strip Plot/Make One Horizontal Strip

Use the **View/Strip Plot/Make One Horizontal Strip** command (<Alt>+vrh) to create a 2D slice in which the y dimension contains all the points and the region covered by the x dimension is defined by the cursor horizontal strip.

## View/Strip Plot/Make One Vertical Strip

Use the **View/Strip Plot/Make One Vertical Strip** command (<Alt>+vrv) to create a 2D slice in which the x dimension contains all the points and the region covered by the y dimension is defined by the cursor vertical strip.

## View/Strip Plot/Make One Orthogonal Strip

Use the **View/Strip Plot/Make One Orthogonal Strip** command (<Alt>+vro) to draw a single strip that is orthogonal to the current 2D plot of a 3D or 4D spectrum.

Define the coordinates and width of the strip plot by dragging the desired area.If you drag horizontally, the short side of the strip will be the horizontal axis; if you drag vertically, the short side of the strip will be the vertical axis. FELIX zooms the plot according to the starting and ending position of the cursor. The long axis of the strip plot will be the current orthogonal dimension.

## View/Strip Plot/Scale Strip Plot

Use the **View/Strip Plot/Scale Strip Plot** command (<Alt>+vrs) to redefine and correct the scaling of too-thin strip plots.

### View/Output

Use the **View/Output** command (<Alt>+vu) to open the output window again after it is closed. The output window is a docking window where the results, instruction, error and warning messages are displayed.

### View/Command Input

Use the **View/Command Input** command (<Alt>+vc) to open the command input window after it is closed. The command input window allows you to issue FELIX commands or execute macros directly.

## Edit pulldown

Use the items in this pulldown to view and edit entities in spreadsheet tables, delete entities, and add annotations to spectra.

### Edit/Table

Use the **Edit/Table** command (<Alt>+et) to open any existing entity from the database in the spreadsheet viewer. Select the entity with the **entity-selection** tool.

### Edit/Peaks

Use the **Edit/Peaks** command (<Alt>+ek) to open the current peak entity in the spreadsheet viewer. The table and the peak entity are interactively linked so that changes to the table (except for sorting) automatically change the peak entity; conversely, changes to the peak entity change the table.

The last column of the table (the intensities) are measured quantities; therefore, changing them in the table has no effect and is not reflected in the entity. The peak table interface has a set of special

command and icons to streamline navigation in the spectrum and interaction with the peaks.

> **Note:** The peak entity contains a different number of columns in different order than the spreadsheet.

> **Note:** The entity differs from its view in the spreadsheet, since patterns are stored in complex, multiply nested entities.

## Edit/Delete Table

Use the **Edit/Table Delete** command (<Alt>+ed) to delete an entity (or table) from the database. Select the entity with the **entity-selection** tool.

> **Remember:** Closing the spreadsheet view of an entity does not remove that entity from the database. Use **Edit/Table Delete to** delete it.

## Edit/NOE_Distance

The **Edit/NOE_Distance** menu item (<Alt>-en) opens the current NOE distance-restraint entity in the spreadsheet viewer. The restraint table interface has a set of special menu items and icons that streamline navigation in the spectrum and interaction with the restraints.

## Edit/3J Restraints

The **Edit/3J Restraints** menu item (<Alt>-e3) opens the current 3J dihedral restraint entity in the spreadsheet viewer. The restraint table interface has a set of special menu items and icons to streamline navigation in the spectrum and interaction with the restraints.

## Edit/Prototype Patterns

The **Edit/Prototype Patterns** menu item (<Alt>-ep) opens the current prototype pattern entity in the spreadsheet viewer, but can be accessed only if you have activated the Assign module and built an

Assign project. The protopattern table interface has a set of special menu items and icons to streamline navigation in the spectrum and interaction with the prototype patterns.

> **Note:** The number and headings of columns in the prototype pattern entity are different from those in the spreadsheet.

## Edit/Spin Systems

The **Edit/Spin Systems** menu item (<Alt>-es) opens the current pattern or spin-systems entity in the spreadsheet viewer, but can be accessed only you have activated the Assign module and built an Assign project. The spin-system table interface has a set of special menu items and icons to streamline navigation in the spectrum and interaction with the patterns.

> **Note:** The entity differs from its view in the spreadsheet, since patterns are stored in complex, multiply-nested entities.

## Edit/Stretches

The **Edit/Stretches** menu item (<Alt>-ec) opens the current stretch of the spin systems entity in the spreadsheet viewer, but can be accessed only if you have activated the Assign module and built an Assign project. The stretch table interface has a set of special menu items and icons to streamline navigation in the spectrum and interaction with the stretches.

## Edit/Residues

The **Edit/Residues** menu item (<Alt>-er) allows you to view the assignment status of each residue in the spreadsheet viewer, that is, which pattern and which frequency are assigned to which residue and which atom. You can access only this menu item if you have activated the Assign module and built an Assign project.

## Edit/Atoms

The **Edit/Atoms** menu item (<Alt>-eo) allows you to view the current atoms entity in the spreadsheet viewer, but can be accessed only if you have activated the Assign module and built an Assign project.

## Edit/Annotation

You may add annotations to any flat graphics plot, whether a simple 1D spectrum, a 2D spectrum, or a 2D slice of an *N*-dimensional matrix.

Annotation positions are specified in normalized device coordinates set on the current plot. The bottom-left and top-right corners of the current plot have coordinate values (0,0) and (1,1), respectively. The menus always place spectral annotations with respect to the current plot. If the spectrum dimensions differ from those used when the original annotations file was created, the annotation items will not be displayed correctly on the new plot with respect to the existing cross peak and spectrum lines. In this case, you must create an entirely new annotation file; or, you must edit the old one to obtain the desired placement of these items.

FELIX saves the annotations as commands in an ASCII file with an extension of .ann. If the current frame has not been associated with an annotation file, use the **Edit/Annotation** (<Alt>+ea) command to select an existing annotation file or create a new one. Otherwise, FELIX prompts you to select from three options:

- To display and edit the current annotation file.
- Not to display it.
- Open another annotation file.

If an annotation file is open, FELIX displays a modeless control panel; use the annotation commands within it to add, move, and delete various annotations. To exit the annotation interface and save your file, select **Done** at the bottom to accept the changes, or select **Cancel** to discard the changes you've made.

Short descriptions of items in the **Annotations** control panel are given in Table 3.

**Table 3 Annotations Control Panel Menu Items**

| Menu Item | Function |
|---|---|
| **Done** | Exit the **Annotation** control panel and accept the changes. |
| **Cancel** | Exit the **Annotation** control panel and discard the changes. |
| **Roman Text** | To place Roman text on the screen, select the **Roman text** command. Set the position of the beginning of the text by clicking the large crosshair cursor, then enter the desired text. The slant, thickness, color, and size of the text can be adjusted by selecting the **Parameters** command and adjusting the appropriate annotation parameters. |
| **Greek text** | To place Greek text on the screen, select the **Greek text** command. Set the position of the beginning of the text by clicking the large crosshair cursor, then enter the desired text. The slant, thickness, color, and size of the text can be adjusted by selecting the **Parameters** command and adjusting the appropriate annotation parameters. |
| **Arrow** | Draw an arrow between two selected positions by dragging: the first mouse-click places the tail of the arrow, and the mouse-release places the arrowhead. The color, line style, and arrow head size can be adjusted with the **Parameters** control panel. |
| **Plumb Arrow** | Draw a plumb (i.e., perfectly vertical or perfectly horizontal) arrow between two selected positions by dragging: the first mouse-click places the tail of the arrow and the plumb point, and the mouse-release defines the arrowhead position. The color, line style, and arrowhead size can be adjusted with the **Parameters** control panel. |
| **Line** | Draw a line between two selected positions by dragging the cursor. The color and line style can be adjusted with the **Parameters** control panel. |
| **Plumb Line** | Draw a plumb (i.e., perfectly vertical or perfectly horizontal) line between two selected positions by dragging: the line color and line style can be adjusted with the **Parameters** control panel. |
| **Vertical Line** | Draw a vertical line from the top to the bottom of a spectrum by selecting a point on your spectrum. The color and line style can be adjusted with the **Parameters** control panel. |
| **Horizontal Line** | Draw a horizontal line through a spectrum by selecting a point on the spectrum. The color and line style can be adjusted with the **Parameters** control panel. |
| **Rectangle** | Draw a rectangle between two selected positions by dragging. The color and line style can be adjusted with the **Parameters** control panel. |

**Table 3 Annotations Control Panel Menu Items (Continued)**

| Menu Item | Function |
|---|---|
| **Header** | Define the upper-left corner for a block of text—the spectrum parameters—by clicking. The color and line style can be adjusted with the **Parameters** control panel. |
| **Delete Region** | Delete the annotations that are completely inside a rectangular region of the plot by dragging. |
| **Move Region** | Move all annotations that are completely inside a rectangular region of the plot as a group. Drag to select the region to be moved, then move the resulting red rectangle to the new location and click again. |
| **Parameters** | List the annotation symbols that can be adjusted to alter the appearance of the subsequent annotations. |
| **Delete All** | Delete all annotations in the current annotation file. |

# Preference pulldown

## Preference/Plot Parameters

This menu item (command) sets various display parameters that affect the plotting of spectrum data.

> **Note:** Various items appear on this menu, depending on whether the data is 1D or ND.

*1D plot parameters*

For 1D data the parameters are divided among five control panels: **Basic**, **Axis**, **Tick**, **Place**, and **Stack**, which can be activated from each other by clicking the appropriate button

**Table 4 1D /Plot Parameters Control Panel Menu Items**

| Control | Function |
|---|---|
| **Stack Depth** | Determine the number of data buffers plotted. A value of zero means to plot the work vector alone; a value of 1 means to plot the work vector and the first data buffer. |
| **Stack Overlap** | How much the plotted vectors overlap: a value of 0 means no overlap between the work vector and data buffer vectors when plotted; a value of 1 means to arrange the plots so they overlap completely. |

**Table 4 1D /Plot Parameters Control Panel Menu Items (Continued)**

| Control | Function |
|---|---|
| **Stack Order** | Choose whether the displayed data buffers are arranged from the top down with buffer one at the top, or from the bottom up with buffer one near the bottom, immediately above the work vector plot. The work vector is always at the bottom of the plot. |
| **Color Scheme** | Select a color from a list. You can select **define** if you want some other color. |
| **Color Number** | Set the initial color for drawing. The default pen colors are shown in Table 5. |
| **Color Cycle** | Define the number of colors used to plot the 1D work vector and data buffer vectors. The first vector is drawn in the color specified by **Color Number**. If **Color Cycle** is set to 0 or 1, all the vectors are drawn in the same color. When **Pen Cycle** is set to 2, the second vector is drawn with a pen color identified by the index **Color Number** +1, the third vector is drawn with the color **Color Number**, and so on. |
| **Plot Mode** | Whether real values, imaginary values, or both are plotted. |
| **Plot Type** | Whether the data values are plotted as points or lines. |
| **Center Plot** | Setting **Center Plot** to **on** sets the center of the y-axis to zero. |
| **Absolute Intensity** | Setting **Absolute Intensity** to **Yes** locks in the plot scaling factor that FELIX calculated for the previous plot. That plot scaling factor ensured that the plot filled the vertical display space. In effect, **Absolute Intensity** turns off automatic scaling, which allows you to compare the magnitudes of two different data sets or the magnitudes of the data in the work vector and the data buffer vectors. |
| **Scale Factor** | Set a multiplier that scales the data plots. The default setting is 1, which causes FELIX to best-fill the vertical space of the plot. A setting of 0.5 produces a plot of half the size. |

**Table 5 Color Settings**

| setting | color | setting | color |
|---|---|---|---|
| **0** | black | **8** | white |
| **1** | white | **9** | red |
| **2** | red | **10** | green |
| **3** | green | **11** | blue |
| **4** | blue | **12** | Blue |
| **5** | Blue | **13** | cyan |
| **6** | cyan | **14** | yellow |
| **7** | yellow | **15** | gray |

**Table 6**

| Control | Function |
|---|---|
| **Axis Units** | Determine the units used to label the x-axis: **PPM**, **Seconds**, **1/cm**, **None**, **Points**, or **Hertz**. |
| **Grid Spacing** | Draw vertical lines on the displayed plot: of 1 means to draw one grid line at each axis labe; 10 means to draw 10 grid lines between each label. Setting a negative **Grid Spacing** value uses tick marks instead of grid lines. |
| **Draw Peaks** | Whether the picked 1D peaks appear on the plot automatically. A valid peak entity must exist. |
| **Peak Entity** | Define the name of the database entity in which the picked peaks are stored. The peaks can be picked using menu items on the **Peakpick** pull-down. |
| **Style** | Set the appearance of the markers for the picked peaks: a line with an arrowhead, an arrowhead alone, or a line alone. |
| **Draw Annotations** | Whether annotations are plotted with the **Draw Workspace** menu item: to include annotations on a hardcopy plot, **Draw Annotations** must be set to **Yes**. |
| **Annotation File** | Name of a file that contains annotation instructions: needed to generate annotations on your plot. |
| **Box Around Plot** | Whether to draw a box around the plot. |
| **Draw 2D Levels** | Draw contour levels on the data in the work vector. You can simplify selection of the contour level setting by loading a matrix vector into the work vector and displaying the contour levels superimposed on the plot. |

**Table 7**

| Control | Function |
|---|---|
| **Manual Plot Size** | Set **Manual Plot Size** to **on** to define the size of the plot by specifying values for the **X Size** and **Y Size** settings. Set **Manual Plot Size** to **No** to automatically draw the plot to fill the space between the **X Offset** and **Y Offset** and the upper-right corner of the display area. |
| **X Offset** | Location of left edge of the plot relative to the FELIX frame, in inches. |
| **Y Offset** | Location of bottom edge of the plot relative to the FELIX frame, in inches. |
| **X Size** | Width of the plot in inches. |
| **Y Size** | Height of the plot in inches. |

## *N*D plot parameters

Similar to 1D data, plot parameters for *N*D data are divided among five control panels: **Basic**, **Axis**, **Tick**, **Place** and **Stack**.

*Basic*

The **Basic** control panel contains tools that control the appearance of the plotted matrix values. You can, for example, choose the base con-

tour level to plot, the number of contour levels to plot, the level multiplier, and the colors of the contour levels.

### Table 8 ND Data Plot Parameters Control Panel Menu Items

| Control | Function |
|---|---|
| **Contour Threshold** | Set the base (lowest) contour level. Base contour level value is the product of **Contour Threshold** and the reserved symbol **mscale** (see the FELIX *Command Language Reference Guide*). |
| **Automatic** | Automatically find the threshold. |
| **Number of Levels** | Specify the number of contour levels to plot. |
| **Level Multiplier** | Determine the difference between values of consecutive contour levels when **Number of Levels** has a value greater than 1. When **Interval Mode** is **Geometric**, a contour level is the product of the previous contour level and the **Level Multiplier** (the **Level Multiplier** must be greater than 1). When **Interval Mode** is **Linear**, a contour level is the sum of the previous contour level and the **Level Multiplier** (the **Level Multiplier** must be greater than 0). |
| **Negative Levels** | Whether contours are drawn for data values less than zero. If you want negative contour levels drawn, set **Negative Levels** to **On**; otherwise, leave it set to **Off**. This parameter is most often set to **On** for contour plots of COSY matrices. |
| **Color Scheme** | Specify a predefined color scheme:<br>**Fire Ramp**: 28-color ramp for positive-only data.<br>**Blue/Green**: 16-color ramp for positive and negative contours.<br>**Green Ramp**: 16-color ramp for positive-only contours.<br>**Blue Ramp**: 16-color ramp for positive-only contours.<br>**Red Ramp**: 16-color ramp for positive-only contours.<br>For any of these predefined color schemes, the **Color Number**, **Color Cycle**, **Number of Levels**, and **Negative Levels** are inactivated. For the next 6 choices (**Red**, **Green**, **Blue Cyan**, **Yellow**) you can set the **Number of Levels** and **Negative Levels**. For the last choice (**Define**) you must manually set all 4 variables (**Color Number**, **Color Cycle**, **Number of Levels**, and **Negative Levels**). |
| **Color Number and Color Cycle** | Pen index for the first contour level. To display more than one contour level, the colors for subsequent levels cycle through the next **Color Cycle** pen indices. When **Color Cycle** is 0, all contours are the same color. For example, with **Number of Levels** set to **4**, **Color Number** set to **1**, and **Color Cycle** set to **2**, the first contour level has color index one, the second has color index two, the third has color index one, and the fourth has color index two. The default pen color indices are shown in Table 7. |

**Table 8 ND Data Plot Parameters Control Panel Menu Items (Continued)**

| Control | Function |
|---|---|
| **Interval Mode** | Whether adjacent contour levels are increased by a multiplicative factor, **Geometric**, or an additive term, **Linear**. |
| **Interpolation Mode** | Method by which FELIX calculates the position of the contour between matrix data points. **None** is the default and draws faster than **Spline**. Set **Interpolation** to **None** unless you are expanding a very small plot region and do not like the boxy look of the contour plot vectors. The **Spline** mode yields smoother looking contours. |

*ND Plot Parameters—Axis*

The **ND Plot Parameters—Axis** control panel contains tools that control the appearance of the other portions of the plot—the axis units for each dimension, projection display, scale factors for each dimension, annotations, box around the plot, and grid lines.

**Table 9 ND Plot Parameters—Axis Control Panel Menu Items**

| Control | Function |
|---|---|
| **Scale D1 and Scale D2 and Scale D3 and Scale D4** | Set scaling factors used to modify the proportions of the plot. When the scale factors are both 1, FELIX proportions the plot to the number of points in each dimension of the displayed region. For example, if one dimension has twice as many points as the other, the plot is rectangular with one side twice the length of the other. To display the plot as a square, set the second scale factor to 2 (or set **Scale D1** to 0.5). When the scale factors are set to -1, then FELIX automatically scales each axis to fill the frame. |
| **Axis D1 and Axis D2 and Axis D3 and Axis D4** | Axis units for each dimension of the matrix: **None**, **Points**, **Hertz**, and **PPM**. Before displaying the matrix with **Hertz** or **PPM** axis units, you must reference the matrix. |
| **Draw Annotations** | Set this parameter to **Off** if you want no annotations on the plot and to **On** if you want annotations drawn each time you make an intensity or contour plot. To produce annotations on a hardcopy plot, this parameter must be **On**. |
| **Annotation File** | Name of the file used to draw annotations. You must create this file using the **Edit/Annotation** command. |
| **Row Projection** | Whether a 1D plot appears along the x axis and the contents of the plot. The possible settings for **Row Projection** are **Off**, **Sum**, **Shadow**, **Buffer 1**, **Buffer 2**, and **Buffer 3**. To turn off the 1D plot, click **Off**. To display a projection calculated as the sum of the values along each column of the matrix, click **Sum**. To display a projection calculated as the maximum data value along each column, click **Shadow**. To display data stored in one of the first three buffers, select **Buffer 1**, **Buffer 2**, or **Buffer 3**. |

**Table 9 ND Plot Parameters—Axis Control Panel Menu Items**

| Control | Function |
|---|---|
| **Column Projection** | Whether a 1D plot appears along the y axis and the contents of the plot. The possible settings for **Column Projection** are **Off**, **Sum**, **Shadow**, **Buffer 1**, **Buffer 2**, and **Buffer 3**. To turn off the 1D plot, click **Off**. To display a projection calculated as the sum of the values along each row of the matrix, click **Sum**. To display a projection calculated as the maximum data value along each row, click **Shadow**. To display data stored in one of the first three buffers, select **Buffer 1**, **Buffer 2**, or **Buffer 3**. |
| **Proj Size (fract)** | Set the size of the 1D projection plots. Specify **Proj Size** as a fraction of the full display size. For example, if you want the matrix to occupy 90% of the display and the 1D projection to occupy 10%, set **Proj Size** to 0.1. |
| **Default Plot Type** | Set the default plot type, which can be **Intensity**, **Contour**, **Stack**, or **Null**. The **Plot Type** command also sets the **Default Plot Type**. |
| **Draw Levels on 1D** | Whether current contour levels are drawn on 1D plots, which assists with setting contour levels. Set **Draw Levels on 1D** to **On** and load a row or column of the matrix into the workspace using the **Display/1D Vector Mode** command. The current contour levels appear as horizontal lines on the 1D plot. |
| **Box Around Plot** | whether the plot is enclosed within a box. By default, **Box Around Plot** is set to **Yes**, and FELIX draws a rectangle around the plot. |
| **Grid Spacing** | An integer that defines the number of grid lines per label to draw on the plot. By default, FELIX draws no grid lines, so **Grid Spacing** is set to 0. When a positive integer is used, the grids are solid lines through the plot. When a negative value is used, tick marks are placed on the outside edges of the plot. |

*ND Plot Parameters—Tick*

The **ND Plot Parameters—Tick** control panel allows you to control the tick mark positions and axis text size for each dimension in a multi-dimensional data set.

**Table 10 ND Plot Parameters—Tick Control Panel Menu Items**

| Control | Function |
|---|---|
| **Tickmark Type** | Whether tickmarks are displayed in the **Default** manner or are **User** adjustable. |
| **Major Tickmarks** | Adjust the separation between major tickmarks for each axis. |
| **Minor Tickmarks** | Adjust the separation between minor tickmarks for each axis. |
| **Axis Text Scale** | Adjust the relative size of **Axis Text**. |
| **Pen Width** | Adjust the pen width of **Axis Text**. |

**Table 10 ND Plot Parameters—Tick Control Panel Menu Items (Continued)**

| Control | Function |
|---|---|
| **Text Slant** | Adjust the slant of **Axis Text**. |
| **Text Thickness** | Adjust the text thickness of **Axis Text**. |

*ND Plot Parameters—Place*

Use the **ND Plot Parameters—Place** control panel to position the plot in the current FELIX frame.

**Table 11 ND Plot Parameters—Place Control Panel Menu Items**

| Control | Function |
|---|---|
| **Manual Plot Size** | Set **Manual Plot Size** to **on** to define the size of the plot by specifying values for the **X Size** and **Y Size** settings. When **Manual Plot Size** is set to **No**, FELIX draws the plot to fill the space between the **X Offset** and **Y Offset** and the upper-right corner of the display area. |
| **X Offset** | The left edge of the plot relative to the FELIX frame, in inches. |
| **Y Offset** | The bottom edge of the plot relative to the FELIX frame, in inches. |
| **X Size** | The width of the plot in inches. |
| **Y Size** | The height of the plot in inches. |

*ND plot parameters—Stack*

Use the **ND plot parameters—Stack** control panel to change the settings for the stack plot.

**Table 12 ND Plot Parameters—Stack Control Panel Menu Items**

| Control | Function |
|---|---|
| **X Axis Skew (fraction)** | Skew applied to subsequent stack plots as a fraction of the x-axis size. To improve plotting speed, the skew value is converted to an integral number of points. A positive value skews the plot to the right and a negative value skews to the left. |
| **Y Axis Skew (fraction)** | The distance, as a fraction of the y-axis size, between the first and last row of a stack plot. To avoid clipping large peaks near the back of the plot, set to a value less than one. |
| **Row Increment** | Stack plot increment value. For example, if **Row Increment** is 2, every second row in the 2D matrix is plotted. Larger values plot faster and show more space between rows but may cause small peaks to be missed. Smaller values plot more slowly and often leave too space between rows. |

**Table 12 ND Plot Parameters—Stack Control Panel Menu Items (Continued)**

| Control | Function |
|---|---|
| **Height Scaling Multiplier** | Vertical scale factor. By default, the factor is 1, so FELIX draws the stack plot to fill the vertical space of the display. The **Height Scaling Multiplier** adjusts the vertical size of the stack plot. |
| **Maximum Peak Height** | Maximum peak height, in inches, for a peak drawn in a stack plot. Peaks above the specified height are clipped so that you can see behind them. |

## Preference/1D Scale

As an alternative to control panel adjustment of 1D plotting parameters, FELIX offers a real-time interface that contains buttons and sliders to manipulate parameters. To activate this feature, select the **Preference/1D Scale** command (<Alt>+pe).

To adjust the spectrum's vertical scale, drag the **Scale** slider or enter a value to the box next to it. You can toggle between absolute and relative scaling. To adjust the vertical offset, click **Offset**; or, if there are multiple 1D slices to display, click **Overlap** to adjust the vertical overlap between them.

To specify any point's (peak's) vertical position, click the **Set Scale** button and click the cursor somewhere in the spectrum display. The point at that position will have the vertical value of the cursor.

To exit the dialog, click the **OK** button to accept the changes, or click the **Cancel** button to discard the changes.

## Preference/2D/ND Levels

As an alternative to control panel adjustment of *N*D plotting parameters, FELIX offers a modeless dialog box to adjust the display. To access this interface, select the **Preference/2D/ND Levels** command (<Alt>+ps). You can adjust the **Contour Threshold**, the **Number of Levels**, and the **Level Multiplier** in real time. To exit click the **OK** button to accept the changes, or click the **Cancel** button to discard the changes.

## Preference/Reference

To reference a spectrum, select the **Preference/Reference** command (<Alt>+pr). FELIX displays different control panels, depending on the dimensionality of the data.

### 1D spectrum referencing

The Reference 1D Data control panel prompts you for the referencing information:

1. Set the **Spectral Frequency** and **Spectral Width** to the values of the spectrometer frequency in MHz and the spectral width in Hz.
2. To enter the **Reference Point** value either type it into the box or click the **Cursor** button in the control panel. Then, move the vertical cursor to the desired reference point on the plot, and click the primary mouse button.
3. FELIX displays the Reference 1D Data control panel again. Enter the reference value in the appropriate entry box, e.g., if you want axis units of Hertz, enter a reference value in the **Reference Hertz** box. For PPM units, enter a reference value in the **Reference PPM** box.
4. Finally, click **OK** to close the panel and redisplay the data with the selected axis units.

### ND spectrum referencing

The Reference Matrix control panel provides tools for referencing your matrix:

1. Set the **Axis type** to **PPM**, **None**, **Points**, or **Hertz**.
2. Next enter the **Spectral Frequency** in MHz, the **Spectral Width** in Hz, the **Reference Point** in data points, and the **Reference Shift** in PPM, then click **OK**.
3. Use one of the three **Cursor** buttons to select the **Reference Point** for a displayed dimension.

## Preference/Pick Parameters

When you select the **Preference/Pick Parameters** command, FELIX offers one of two different control panels, depending on the dimensionality of the data.

### 1D spectrum

Before picking peaks, set the threshold value manually or via the cursor.

Select the **Cursor** option in the popup and click **OK** to use a line cursor for selecting the peak-picking threshold on the plot.

Use **Selection Mode to** specify if you want only positive peaks, only negative peaks, or both.

Use **Points**, **ppm**, **Hz**, or **Assignment** to specify the peak table and the units that should be displayed on the peaks.

Use **Draw Style** to draw one-dimensional peak markers as lines, arrows, or line and arrows.

### ND spectrum

This control panel provides access to the cross-peak entity (table), the cross-peak selection mode (positive peaks, antiphase peaks, negative peaks, and positive and negative peaks), and tools for specifying the antiphase search window size, fixed footprint sizes, and minimum and maximum halfwidth values.

- Click the **Info** button to review help in setting the picking parameters.
- Use **Antiphase Peaks** mode to search for antiphase multiplet peaks, as in a COSY matrix. Select the regular peak picker (which is faster, but sometimes less reliable) or the Stella peak picker (which is slower, depends on example peaks, but, with a good training set, can give superior peak set).
- Set the method for interpolating an extremum by fitting a second-order polynomial (**Quadratic Interpolation** or **Center of Gravity**).

## Stella Peak Picker parameters

The general rule is that more example peaks means longer execution time: for a 2D matrix it can take up to several minutes to pick peaks using 6–10 example peaks. For 3D or 4D sets the time can be even longer.

A local maximum is picked as a peak if:

- Its maximum exceeds the current threshold.
- It has at least a certain number of neighbors whose intensity exceeds the threshold.
- It matches one of the example peaks with a match factor (-1 or +1) that exceeds the **Minimum match factor** threshold (if example peaks are used).

The definable parameters are:

### Table 13 Definable Peak Picking Parameters

| Control | Function |
|---|---|
| **Selection Mode** | Pick positive, negative, or both types of peaks. If you pick negative peaks then you must have negative example peaks, and if you pick positive peaks you must have positive example peaks. |
| **Local maximum method** | Set the local maximum-defining method. You can use the rough maximum, the center of the gravity, or interpolation. |
| **Minimum match factor** | Should be set between -1 and +1, to distinguish between genuine peaks and spurious peaks based on matching the provided example peaks. |
| **Hump tolerance factor** | Used in peak bounds detection. |
| **Extra points for peakbox** | Choose certain points to be added to the found peak limits. |
| **Neighbors above threshold** | Define at least how many neighbors must exceed the current threshold for the local maximum to be considered a peak. |
| **D1 search, D2 search, ...** | Select how many neighbor points should be checked for local maxima in each dimension (**D1 search**, **D2 search**, ...). These search parameters are used to define a box in which the center should be local maximum to be considered as a possible peak. The neighbors are also considered within this box (for a 2D matrix, if **D1 search** = 2 and **D2 search** = 2, then a square containing 25 points is considered at one time). |

**Table 13 Definable Peak Picking Parameters (Continued)**

| Control | Function |
|---|---|
| **Limits** | Whether you want to pick the full spectrum or just a subset. If you set **Limits** to **Define**, then after clicking **OK** a new control panel shows up where you can define the limits of the region you want to pick, in points or ppm. |
| **Output Level** | The amount of information to print at each possible peak (**Quiet**, **Low**, **Medium**, or **High**). |

## Preference/Peak Display

Use the **Preference/Peak Display** (<Alt>+pd) command to control the appearance of the *N*D cross peak footprints. Here you set the display switch, the cross peak entity, and other relevant parameters:

**Table 14 ND Cross Peak Parameters**

| Control | Function |
|---|---|
| **Draw Crosspeaks** | **Off** if you want no cross peaks on the plot and **On** if you want cross peaks drawn each time you make an intensity or contour plot. To draw cross peaks on a hardcopy plot, this parameter must be on. |
| **Crosspeak Entity** | The name of the entity used to draw cross peaks. You must create this entity with commands within the **Peaks** pulldown. |
| **Crosspeak Symbol** | The marker to be drawn at each cross peak: a cross (its size is **Half Width Factor** × 2 × crosspeak_width) with a surrounding box (**Default**), just a cross (**Cross Only**), a half size cross (**Small Cross**), just a small box (**Small Box**), or just the number (**Number**). |
| **Label Peaks** | Whether cross-peak footprints are labelled with nothing, the peak item number, the assignment names, or (for proteins) a shorthand notation. |
| **Label Size** | Size of the cross-peak labels in inches. |
| **Half Width Factor** | Size of the cross-peak footprint, together with the half-widths of a peak. The width of the cross-peak footprint displayed is twice the product of the half-width and the **Half Width Factor**. The footprint size influences the results of many FELIX commands that operate on cross peaks. |

**Table 14 ND Cross Peak Parameters (Continued)**

| Control | Function |
|---|---|
| **Coloring Mode** | Cross-peak footprint coloring mode: one color or (for 3D and 4D spectra) based on the peak position compared to the current plane, or (for assigned peaks) based on assignment or on whether they belong to a specific prototype pattern. The latter two options are available only with Assign. |
| **Color** | Cross-peak footprint color. Specify the color as a pen index, using the **Define** option and entering the number in the **Color Number** box (see **Pen Number**) or referring to it by name. |

## Preference/DQF Parameters

Use the **Preference/DQF Parameters** (<Alt>+pa) command to choose an optimization method for the J-coupling measurement in DQF–COSY spectra (quasi-Newton, Simplex, or simulated annealing).

## Preference/Keypad

Use the **Preference/Keypad** (<Alt>+pk) command to specify the small step size in points for <Alt>+keypad navigation.

## Preference/Frame Layout

Use the **Preference/Frame Layout** (<Alt>+pf) command to specify the options for automatic frame arrangement. By default, whenever a new window (table or spectral) is open, FELIX automatically rearranges the layout of the windows, with the tables tiled on the left side, occupying 20% of the main window area; and the spectral windows tiled on the remaining area of the main window.

You can turn off this feature by setting **Action** to **None**, or your can select another way to arrange the spectral frames, such as **Cascade** or **Tile Horizontally**. However, the table frames are always tiled in

the designated area when automatically arranged. The location and size of the area to tile the table frames can be changed from the **Table Layout** parameters.

> **Note:** You can manually call the automatic arrangement function by selecting the **Window/Auto Arrange** command.

## Preference/Frame Connection

Use the **Preference/Frame Connection** (<Alt>+pc) command to connect and disconnect up to 12 frames containing different views of different spectra.

You can use this connection, for example, to view a $^{15}N$–$^{1}H$ HSQC spectrum in the primary frame and a $^{1}H$–$^{1}H$ view from the corresponding $^{15}N$-separated TOCSY and NOESY spectra in the two secondary frames, having the two secondary frames connected along each dimension and defining the plane selection direction in the HSQC spectrum to be along the $^{15}N$ dimension. Therefore, looking at each $H_N$–N peak in the HSQC spectrum, you can bring up the corresponding $^{15}N$ plane in the TOCSY and NOESY spectra and visually collect each spin system quickly by hand.

You can also connect two frames, each containing a strip along an $H_N$ frequency from an HNCACB and a CBCACONH spectrum at the same $^{15}N$ frequency. This permits a quick scan through the entire two 3D spectra to find the intra- and inter-residual CB–CA–N–HN.

Use the **Custom** option to select a trivial connection between two frames (D1 to D1 or D1–D2 to D1–D2) or specify a more elaborate one.

After a frame connection is set, use the **Disable** option to temporarily disconnect them. Use the **Enable** option to restore the connection.

## Preference/Multiple Cursor

This command is similar to the **Preference/Frame Connection** command, in that you can connect and disconnect the *cursors* using the **Preference/Multiple Cursor** command (<Alt>+pu). You must spec-

ify which axis of one frame should share cursor positions with other frame's axis.

## Preference/Table

Use the **Preference/Table** command (<Alt>+pt) to specify the current table (entity) names for peaks, integrals, volumes, and other items.

## Preference/Directory

Use the **Preference/Directory** command (<Alt>+py) to display a control panel showing the current directory prefixes for each type of file that FELIX uses.

You can edit any box to change a file prefix. The file types that FELIX uses are listed in Table 15. The notation "read only" implies a sharable directory, while "read+write" denotes a directory that should be owned and used by only one person.

**Table 15 File Types Used By FELIX**

| Control | Type | Description |
|---|---|---|
| **Current Directory** | read+write | the current working directory |
| **Data** | read+write | 1D data files |
| **Matrices** | read+write | 2D and *N*D matrix files. |
| **Database** | read+write | FELIX database files. |
| **FELIX Macros** | read only | FELIX macros. |
| **FELIX Dialogboxes** | read only | FELIX control panels. |
| **FELIX IF Macros** | read only | FELIX user interface menu files. |
| **FELIX Icons** | read only | FELIX user interface icon files. |
| **User Macros** | read +write | User-written macros. |
| **NMRRefine DB** | read+write | Insight II files. |
| **Limits** | read+write | Plot limits files. |
| **Annotations** | read+write | Annotation macro files. |
| **Parameters** | read+write | Program context saved parameter files. |
| **Messages** | read only | FELIX error message files. |
| **DB Schema** | read only | Database schema prototype files. |
| **Coordinates** | read+write | Atomic coordinate files. |
| **Text Files** | read+write | All written ASCII text files. |

**Table 15 File Types Used By FELIX (Continued)**

| Control | Type | Description |
|---|---|---|
| **Runtime Files** | read+write | Various temporary runtime files. |
| **Foreign Data** | read +write | Spectrometer data files to be filtered. |
| **Filter Images** | read only | Data filter programs spawned by FELIX. |

## Preference/Memory

Use the **Preference/Memory** command (<Alt>+pm) to display a control panel showing the currently defined buffer size and the number of buffers allocated for FELIX. Here you can change the memory allocation.

## Preference/Macro Debug

Use the **Preference/Macro Debug** command (<Alt>+pb) to display a control panel showing the options for debugging macros.

- If you click **None**, no debugging message is displayed in the text window.
- If your choose to show macro names, FELIX displays the path and filename of each executed macros or menu files in the text window.
- If you choose to show all details, FELIX displays every executed command in the macros in the text window.

The latter two options provide you very useful ways to locate macros and menu files and trace errors.

# Process pulldown

## Process/DC Offset

### Preventing baseline discontinuities

A problem you can encounter when zero filling time-domain data is related to baseline correction. If the right-most data points in the FID are significantly different from zero, and zero filling is performed, a discontinuity (step function) is introduced into the data. The spectrum resulting from the Fourier transformation of data that contain a step function has wiggles or waves in its baseline.

It is crucial to avoid discontinuities when zero filling badly truncated data. This common problem is encountered when processing multi-dimensional data, although it can be prevented by using baseline correction to remove DC offset.

Manipulation of time-domain data prior to Fourier transformation can be used to change the size and appearance of transformed spectra. In addition, some type of spectrum artifacts can be eliminated. Raw FIDs are usually corrected before Fourier transformation to remove any DC offset that may have occurred during data acquisition.

### Setting baseline correction

To correct raw FIDs, select the **Process/DC Offset** (<Alt>+pd) command. FELIX displays a control panel. Set a baseline correction fraction to specify the fraction of the FID, starting from the right side, to be averaged to eliminate the DC offset. The default value of this symbol is 0.2, based on the assumption that most of the signal has decayed to zero in the last 20% of the FID. By averaging the last quarter or so of points in the FID, a good zero level can usually be defined. For complex time-domain data, which contain both real and imaginary parts, the DC offset for each part is calculated independently.

If the baseline-correction function can calculate an accurate zero level, the effect on the transformed spectrum will be to eliminate a spike at the observed frequency. However, if the data are badly trun-

cated (not enough data points were collected), baseline correction may not be able to calculate an adequate zero level. In fact, by applying baseline correction you may *add* a DC offset. If you are worried that your data may be truncated but still want or need to baseline correct the DC offset in your FID, try baseline correcting using a smaller fraction of the FID; that is. set the value of the baseline correction-fraction to 0.05.

## Process/Zero Fill

Use the **Process/ZeroFill** command (<Alt>+pz) to zero-fill spectra. Zero filling is commonly performed on time-domain data. By selecting the **Process/ZeroFill / BC** command, you may increase the number of points in the transformed spectrum and thereby increase the spectrum's apparent digital resolution. Zero filling a spectrum defaults to doubling the size of the data, but you may zero fill to any desired size.

## Process/Solvent Suppression

### Solvent signal suppression

NMR data are frequently composed of signals arising not only from resonances of interest, but also from the solvent used to dissolve the sample. Solvent signals may compromise the analysis of the signals of interest, and in extreme cases may completely obscure important spectrum features.

Although effective methods for minimizing the intensity of solvent signals at acquisition time exist, e.g., through tailored excitation, post-acquisition methods can be extremely useful when undesirable solvent signals persist.

FELIX offers three methods for reducing the intensity of such solvent signals: a linear prediction-based algorithm (LP), a convolution-based method (CNV), and a polynomial-based method.

### LP-Based solvent suppression

The linear prediction-based solvent-reduction routine exploits a technical feature of the LP algorithm to estimate and remove contri-

butions from the most intense components in the spectrum (the intensities of the signals present in the interferogram are effectively ranked). The algorithm relies on the fact that solvent resonance frequently represents the most intense component in the spectrum (as with data acquired in $H_2O$) and explicitly assumes this as a part of its function. If the signal identified as the most intense component is not significantly larger (by default, 5 times the value of the other components), no solvent-peak elimination is done.

To access linear prediction-based solvent reduction, select the **Linear Prediction** option for **Method** in the **Process/Solvent Suppression** command. FELIX displays a control panel prompting you to specify the number of data points to use in the LP calculation and the number of signals to remove. A value of 1 for **Signals To Remove** eliminates only the most intense component of the spectrum, a value of 2 removes the two most intense components, and so on. If the signal identified as the most intense component is not significantly larger than the other components, no solvent-peak elimination occurs. You can view the results of solvent suppression and change the parameters interactively if you specify **Real-Time** for the **Method**.

## CNV-based solvent suppression

The convolution-based solvent-reduction routine conducts a convolution of the data with a sinebell or Gaussian function to first identify the lowest-frequency component, and then subtracts that component from the data (Marion and Bax 1989).

Two parameters are available in this control panel: the convolution function, which can be either a sinebell or Gaussian function (which at best is largely an empirical issue) and the function width. The best value for the function width depends upon the widths of resonances in the spectrum and the resolution. In practice its value is empirically derived.

To access convolution-based solvent reduction, select the **Time-Domain Convolution** option for the **Method** parameter in the **Process/Solvent Suppression** command. FELIX displays a control panel prompting you to specify the convolution function type (sinebell or Gaussian), and the function width (the default value of 10 works well for $^1H$ data acquired in 1–2 K data points). You can view

the results of solvent suppression and change the parameters interactively if you specify **Real-Time** for the **Method**.

### Polynomial-based solvent suppression

The polynomial-based solvent-suppression method uses a polynomial fitting method to remove solvent signals from the time-domain data. The solvent signal is approximated by calculating the mean value of groups of data points and fitting a polynomial to these mean values. The resulting function is then subtracted from the time-domain data. This technique works best when the solvent frequency is close to zero.

To access polynomial-based solvent suppression select the **Polynomial** option for the **Method** parameter in the **Process/Solvent Suppression** command. FELIX displays a control panel prompting you to specify the **Points to Use** and the **Polynomial Order**. The **Points to Use** represents the number of data points in each group of points to average. The **Polynomial Order** represents the order of the polynomial that is used to fit the set of average points. You can view the results of solvent suppression and change the parameters interactively if you specify **Real-Time** for the **Method**.

## Process/Window Function

Time-domain NMR data can be multiplied by window functions that perform digital filtering to reduce noise or increase spectrum resolution. For example, the noise level in 1D NMR data can be attenuated by multiplying the FID by an exponential window function.

Use the **Process/Window Function** command (<Alt>+pw) to select a window function and adjust its parameters by entering parameters directly. You can also set the function interactively while FELIX displays plots of both the window function and the product of the FID (possibly the FT'd spectrum) and the window function.

### Real-time adjustment

The available window functions are:

- **Sinebell**

- **Sinebell^2**
- **Skewed Sinebell**
- **Skewed Sinebell^2**
- **Exponential**
- **Gaussian**
- **Trapezoid**
- **Kaiser**
- **Matched**

Once you select a window function, FELIX displays a new control panel. Enter the parameters and apply the selected window function, or select the real-time option.

If you select the real-time option, a real-time interface panel appears, which consists of three buttons and a group of sliders, depending on the window function. You can apply FT or draw only the FID using the options (**No FFT/FFT/Bruker FT/Digital FT**). You can also **Reset** the parameters to their original values.

When you finish viewing the window function, close the command by clicking **Keep** or **Quit**. These buttons close the real-time interface and either retain the FID as it appeared with the window function or restore the original FID without the window function applied, respectively.

Use the sliders to directly adjust the window function parameters in real time. For example, the real-time interface for the **Sinebell** window function has sliders for the **Window Size** and the **Phase Shift**. You can adjust these parameters by moving the cursor over the slider and dragging. The red slider bar moves and the updated value is displayed within the slider.

Since it is a modeless dialog, you can adjust the plot display through the main commands or the toolbar icons.

## Window function descriptions

### *Matched filter*

Matched filter is an automatic version of exponential multiplication that examines the FID and chooses an appropriate Lorentzian broadening. The matched filter calculates and applies a matched exponential window to the FID. The line broadening is calculated by

performing a least-squares fit to the FID. If the FID has an extremely low signal-to-noise ratio, the fit may fail, and a message to that effect appears on the screen. Note that one large, narrow, softened resonance may dominate the fit. After applying the matched filter, FELIX sets the global line-broadening parameter to the value of the line broadening that was applied.

The **matched filter** command is useful because it allows FELIX to determine the optimal line-broadening parameter for your spectrum, and thus gives the best signal to noise ratios.

To access this function, select it from the interface or enter the following at the command line:

```
> mf
```

For more detailed information, please see the **mf** command in the FELIX *Command Language Reference Guide*.

### *Convolution difference*

Convolution difference is an apodization function that calculates the difference between no line broadening and specified line broadening.

To access this function, enter at the command line:

```
> cd lbroad
```

| | |
|---|---|
| *lbroad* | Adjusts the convolution parameter for the exponential. |

For more detailed information, please refer to the **cd** command in the FELIX *Command Language Reference Guide*.

### *Sinebell*

To access this function, either select it from the interface or enter at the command line:

```
> sb wsize wshift
```

| | |
|---|---|
| *wsize* | Adjusts the number of data points for the window function. |
| *wshift* | Adjusts the phase shift of the window function. |

### *Sinebell squared*

To access this function, either select it from the interface or enter at the command line:

```
> ss wsize wshift
```

| | |
|---|---|
| *wsize* | Adjusts the number of data points for the window function. |
| *wshift* | Adjusts the phase shift of the window function. |

### *Skewed sinebell*

To access this function, either select it from the interface or enter at the command line:

```
> qsb wsize wshift wskew
```

| | |
|---|---|
| *wsize* | Adjusts the number of data points for the window function. |
| *wshift* | Adjusts the phase shift of the window function. |
| *wskew* | Adjusts the skew of the window function. |

### *Skewed sinebell squared*

To access this function, either select it from the interface or enter at the command line:

```
> qss wsize wshift wskew
```

| | |
|---|---|
| *wsize* | Adjusts the number of data points for the window function. |
| *wshift* | Adjusts the phase shift of the window function. |
| *wskew* | Adjusts the skew of the window function. |

### *Exponential line-broadening*

To access this function, either select it from the interface or enter at the command line:

```
> em lbroad
```

| | |
|---|---|
| *lbroad* | Adjusts the line-broadening parameter for the exponential. |

### *Gaussian linebroad-ening*

To access this function, either select it from the interface or enter at the command line:

```
> gm lbroad gbroad
```

| | |
|---|---|
| *lbroad* | Adjusts the line-broadening parameter for the exponential. |
| *gbroad* | Adjusts the Gaussian parameter for the exponential. |

### *Trapezoidal*

To access this function, either select it from the interface or enter at the command line:

```
> tm p1 p2 p3
```

| | |
|---|---|
| *p1* | Adjusts the first point of the trapezoid. |
| *p2* | Adjusts the second point of the trapezoid. |
| *p3* | Adjusts the third point of the trapezoid. |

*Kaiser*

To access this function, either select it from the interface or enter at the command line:

```
> kw wsize alpha
```

| | |
|---|---|
| *wsize* | Adjusts the number of data points for the window function. |
| *alpha* | Adjusts the alpha parameter of the Kaiser window. |

## Process/Linear prediction

Linear prediction estimates the value of a point based on the values of adjacent points. This can be used to replace corrupted values in an FID or to extend an FID.

### First-point prediction

The **Linear Predict First** command uses linear prediction to replace data values at the beginning of the FID. The **Points to use** tool in the control panel defines the number of points used to calculate the LP coefficients. A reasonable value for this parameter would be the number of points in the workspace minus the number of predicted points (the larger the setting of **Points to use**, the longer the action takes to complete). A good value for the **Number of coefficients** setting is one quarter to one third the value of **Points to use**. The **Number of peaks** is included for compatibility with older macros, but is not used in the calculation. The **Number of points to predict** specifies the point at which the **First Points** function begins predicting values. It estimates data values from that point backward to the first point of the FID. For example, to replace the values of the first three points of the FID with predicted values, enter a value of **3** for **First Points**.

### Last-point prediction

Use the **Linear Predict Last** command to predict first points, extend the FID, or replace corrupted points.

Use the **First Point** parameter in the control panel to define the start of points used to calculate the LP coefficients.

Use the **Last Point** parameter to define the end of points to be used to calculate the LP coefficients.

Use the **Start Point** parameter to define the start of points to calculate.

Use the **End Point** parameter to define the end of points to calculate.

A good value for the **Number of coefficients** setting is one quarter to one third the value of points to be used for prediction. The four choices for the **Method** are **Backward**, **Forward**, **Forward-Backward**, and **Mirror**. Use root reflection by turning the **Use Root Reflection** parameter **on**. The **Type of mirror LP** is used only when the **Method** is set to **Mirror**. The **90-180** method is used when the data collection is delayed by one half the dwell time. The **0-0** method is used when there is no delay in the data collection.

## Process/Transform

The commands under **Process/Transform** (<Alt>+st) apply to Fourier transformation, linear prediction, and Hilbert transforms in the workspace.

### Complex FFT

The **Complex FFT** option applies a complex Fourier transform to the data in the work space. For this transform, the data must be true complex data, characterized by simultaneous sample and conversion of the real and imaginary signals.

### Bruker FFT

The **Bruker FFT** option performs a complex Fourier transform on complex data that are unique to some Bruker spectrometers. These spectrometers cannot sample and convert the real and imaginary signals simultaneously; instead, they collect the real and imaginary signals alternately. If your data were collected in this mode, you must use the **Bruker FFT** option in the **Process/Transform** command.

### Real FFT

The **Real FFT** option performs a real Fourier transform on real data in the work space. After the real Fourier transform, the data become complex with the spectrum in the real part of the work space.

### Oversampled FFT

The **Oversampled FFT** option performs a complex Fourier transform on digitally oversampled data collected on Bruker DMX and newer-series spectrometers. If your data were collected using digital oversampling you should use this command to do the transform.

### Inverse FFT

At times you may need to convert frequency-domain data into time-domain data. For this purpose, use the inverse Fourier transform. Select the **Inverse FFT** option in the **Process**/**Transform** command.

### Hilbert transform

To perform a Hilbert integral transform on the data in the work space, select the **Hilbert Transform** option in the **Process**/**Transform** command. The Hilbert transform is valuable for creating a complex spectrum from a real spectrum; that is, it transforms real data in the frequency domain into complex data in the frequency domain. The Hilbert transform is required for rephasing a spectrum after the imaginary part is discarded, which can occur with multidimensional NMR data processing.

## Process/Phase Correction

After Fourier transformation, a spectrum often appears to be out of phase. That is, the resonance lines appear to be a mixture of absorptive and dispersive shapes. This is due to several factors, including finite pulse lengths, acquisition delays, and analog filter response. NMR spectra can be phase-corrected after transformation by multiplying each data point value pair by a phase factor. You may also access this command with the hot keys <Alt>+sp.

## Real-time phase correction

One of the most valuable features of FELIX is real-time phasing capability. To activate this feature, select the **Real-Time** option in the **Process/Phase Correction** command. FELIX displays a modeless dialog box with two sliders and some buttons. Since it is a modeless dialog, you may use the menubar commands or toolbar icons to adjust the display of the spectral plot.

The real-time phase interface includes two active slider bars. To adjust the current values for the zero-order and first-order phase corrections, drag the respective slider bars. The current values of the zero-order phase and the first-order phase are displayed above the sliders. The value ranges of the sliders are displayed beside them. To adjust the value ranges, type in a number directly followed by a carriage return. Or, click the **Coarse** or **Fine** buttons to increase or decrease the value ranges respectively.

To set the pivot for your spectrum, click the **Pivot** button. FELIX displays a vertical hair cursor. Move the cursor to where you want and click the primary mouse button. The current position of the pivot is indicated by a small red triangle at the bottom of your spectrum.

To return to the original phase values, click the **Reset** button. To quit the real-time interface and save and apply the phasing parameters, click **OK**. This updates the reserved symbols **phase0** and **phase1**. To quit without saving the parameters, click **Cancel**.

## Phase correction using parameters

In addition to using the real-time phase interface, you may also phase a spectrum by manually setting values for **phase0** and **phase1**. Select **Parameter** as the **Method** in the **Process/Phase Correction** command.

The values in this control panel are updated when you exit the real-time phasing interface. To phase a spectrum manually, you must first define parameters for the zero-order correction (**phase0**) and the first-order correction (**phase1**). FELIX does not use a separate value for a spectrum pivot; instead, this is incorporated into the values of **phase0** and **phase1**. To apply a phasing correction, update

values for **phase0** and **phase1** in the control panel and click **OK**. To exit the control panel without applying or updating the phase parameters, click **Cancel**.

> **Note:** When you repeatedly process similar spectra and want to apply a known set of phase corrections to a spectrum, it is easier to enter the phase corrections with this method than it is to re-phase a spectrum interactively in the real-time phasing interface.

### Automatic phase correction

In addition to the phasing techniques described above, FELIX provides several functions for automatic phasing of a 1D spectrum. To use one of these methods, select the **Automatic** option in the **Process/Phase Correction** command, then select one of the four **Auto Method** options.

The automatic phasing methods include:

- the PAMPAS and APSL methods for spectra with non-split peaks, such as decoupled $^{13}C$ and DEPT spectra
- a method based on peak integration, for general in-phase 1D spectra
- a basic method intended for common proton spectra

Except for the basic method, you can specify one or more excluded areas, to exclude solvent peaks when calculating the phase parameters. For the PAMPAS and APSL methods, you can also specify a **Filter Width**, which is the minimum peak width (at the peak bottom) required for a sample peak to be used in the calculation of phase parameters.

FELIX selects the default values for **Auto Method** and **Filter Width** automatically, based on the spectrum data in the workspace. However, you can change these settings.

## Process/Baseline correction

The **Process/Baseline Correction** (<Alt>+sb) command contains options that deal with baseline points or do baseline correction.

## Process/Baseline correction/Auto Pick Points

To define baseline points, select the **Auto Pick Points** option in the **Process/Baseline Correction** command. FELIX generates a list of baseline points. Display markers for each baseline point picked in the spectrum are shown at the bottom of the current spectrum.

## Process/Baseline correction/Auto Pick Points w/FLATT

The **Auto Pick Points w/FLATT** option of the **Process/Baseline Correction** command uses the FLATT algorithm (Guntert 1992) for selecting baseline points in a spectrum. The resulting points are stored in the entity whose name is stored in the symbol **basent**.

FELIX first prompts you to specify the **Basepoint Line Width** to use in calculating the **chi** value for the spectrum. Next, FELIX prompts you to specify the **Baseline Width**, **Minimum Chi Square**, **Factor(tau)**, and **Stride** to use in selecting the baseline points. For a more complete description of the required parameters, please see the **abp** and **chi** commands in Appendix A of the FELIX *Command Language Reference* manual.

## Process/Baseline correction/Pick Points via Cursor and Manual Pick Points

To add baseline points singly, use the **Pick Points via Cursor** option or the **Manual Pick Points** option of the **Process/Baseline Correction** command. Select each baseline point via a cursor; or, type in the desired points via a menu interface. To use the cursor, add points by clicking the desired baseline points with the crosshair cursor. To exit this mode, click outside the spectrum.

## Process/Baseline correction/Delete All Points

To delete all the baseline points, select **Delete All Points** in the **Process/Baseline Correction** command. FELIX deletes the current baseline points entity from the database; FELIX prompts you to confirm this action via a dialog box.

## Process/Baseline correction/Delete Points in Region

If you make a mistake while selecting individual baseline points or if you want to modify the current list of baseline points, you may

delete a region of points using the graphical interface. First, select **Delete Points in Region** in the **Process/Baseline Correction** command to create a small crosshair cursor. Then drag out a region of baseline points to delete.

## Process/Baseline correction/Baseline correction

Once the baseline points are defined, you can choose one of several baseline-correction algorithms:

*Process/Baseline correction/Polynomial*

The baseline-correction algorithm generates smoother baseline correction functions from baseline points. The **Polynomial** correction option of the **Process/Baseline Correction** command differs from the cubic spline correction algorithm in that the baseline does not necessarily pass exactly through each baseline point, but a best fit is calculated. In addition, you may set the order of the polynomial (from 2 to 9) in the polynomial control panel. A polynomial of order two yields a smooth parabolic function, and a polynomial of order nine generates a more complex correction function. A polynomial of an order between three and five is usually sufficient to give accurate baseline correction.

*Process/Baseline correction/Real-Time Polynomial*

Use the FELIX real-time baseline-correction feature to adjust the coefficients of a polynomial baseline function while displaying both the resulting baseline function and baseline-corrected spectrum superimposed.

When you select the **Real-Time Polynomial** option of the **Process/Baseline Correction** command, FELIX prompts you to specify a polynomial order for the correction function and an interval width; this is used to average baseline point values as described earlier.

Click **OK**, and FELIX displays the real-time baseline correction interface. When you finish correcting the baseline you can exit the interface and keep the corrected spectrum (**Keep**) or **Cancel** the interface and restore the original spectrum.

To alter the displayed region along the x axis of the spectrum, click **xpand** and using the small crosshair cursor to drag a box around the desired region. To restore the complete spectrum, click **Full**. If you are dissatisfied with any of the baseline points, which are indicated by red ticks below the spectrum, add or remove points by clicking **Add Points** or **Delete Points**. When you are satisfied with the baseline points, click **Fit** to automatically calculate the polynomial coef-

ficients (the calculated baseline appears as a red line superimposed on the spectrum) and click **Apply** to apply the correction to the spectrum.

Use the slider located along the bottom of the interface to adjust the polynomial coefficients individually in real time. First, select the individual coefficient through the popup next to the slider. Again, when the red baseline appears to coincide with the spectrum baseline, click **Apply** to correct the spectrum. You can zero the polynomial coefficients by clicking **Zero** and restore the original spectrum by clicking the **Reset** button. In addition, the displayed spectrum can be shifted and stretched vertically with the keypad.

*Process/Baseline correction/Cubic Spline*

The cubic spline algorithm, applied by selecting the **Cubic Spline** option from the **Process/Baseline Correction** command, generates a baseline that passes exactly through each baseline point. A cubic spline may yield a kinked baseline if the defined baseline data points are close together and noisy. To minimize this problem, adjust the interval width reserved symbol **iwidth** to a number larger than 1. Increasing the interval width minimizes the kinked baseline problem by averaging the data values in an interval of points around each picked point and using that average value as the baseline point.

*Process/Baseline correction/Automatic w/ ABL*

To use a baseline-correction function supported by FELIX that does not require explicit baseline points, select the **Automatic w/ABL** option from the **Process/Baseline Correction** command. FELIX selects noise points and performs a baseline correction for each point. You must input values for the noise level and the peak size in points.

> **Caution:** Depending on the number of points in your spectrum and your line widths, these values may need to be adjusted several times to fit your data. Therefore, you should save a nonbaseline-corrected spectrum before applying the correction. This algorithm was reported by Dietrich et al. (1991) and implemented by W. Massefski.

*Process/Baseline correction/Automatic w/ FLATT*

FELIX also provides the FLATT baseline-correction algorithm, a technique introduced by Guntert and Wuthrich (1992). The FLATT algorithm locates baseline segments in the spectrum and uses linear least-squares to fit a truncated Fourier series to these points.

To use FLATT, select the **Automatic w/FLATT** option in the **Process/Baseline Correction** command. FELIX prompts you to specify the

**Basepoint Line Width**, which is used to calculated the minimum chi-square value. Enter an integer with a value that is small but larger than half the width of the widest peak.

When you click **OK**, FELIX determines a value for the minimum chi square, which should correspond to the contribution of noise to the chi-square value. FELIX displays this value in the next control panel, which opens automatically and prompts you to specify baseline-correction parameters. You must enter a value for the **Baseline Width** as you did for the minimum chi-square estimate. You may adjust the **Minimum Chi Square** value if you want.

The control panel also prompts you to specify the **Points to Correct**, which is the number of Fourier series terms used to fit the baseline, and the **Factor (Tau)**, which specifies how much larger than the minimum chi-square value a segment's chi-square value can be and still be considered baseline. When the chi-square value of a segment exceeds the product of the minimum chi-square value and the **Factor (Tau)**, the segment is considered to contain peak information and is rejected as baseline. To perform this action, click **OK**.

*Process/Baseline correction/Automatic w/ FaceLift*

FELIX also provides the FaceLift baseline-correction algorithm (Chylla & Markley 1993). This signal-recognition based utility identifies baseline points and subtracts the baseline points from the original spectrum data.

To use FaceLift, select the **Automatic w/FaceLift** option in the **Process/Baseline Correction** command. FELIX prompts you to specify the **Filter Width**, which is the half-width of the smoothing data window over which datapoints are sampled. The half-width determines the minimum line width of artifacts that will be removed from the spectrum. The recommended range is 32–64 datapoints (powers of 2 are not necessary). Use the **Number of Standard Deviations** to determine a threshold standard deviation, above which any point is considered to be a signal point. The recommended range is 2.5–3.0. Click **OK** to execute the FaceLift algorithm.

## Process/1D Data Processing

Use the **Process/1D Data Processing** command (<Alt>+s1) to interactively process 1D data or the first FID of an *N*D data set.

First, FELIX displays a control panel where you can specify the **Filter Type** for the kind of data you want to process. In general, if you are working with raw spectrometer data, you should specify **All Files (BRUKER, VARIAN, JEOL:*)** as the **Filter Type**.You can then navigate through the desired directories to get to the data.

At this point you want to select the actual spectrometer datafile. This is usually an FID or SER file. To select the FID double-click the filename; or, click the filename and then click **OK**.

When you click **OK**, FELIX displays a control panel containing the 1D header menu parameters, which are taken from the header information of the spectrometer datafile.

Be sure to check that the proper parameter values are displayed and correct them if necessary. If the **Data Type** parameter is **Complex,** the **Data Size** parameter is in complex points. Spectrometer data generally have a **Data Type** of **Complex**. The **Bruker Data Type** is used only for Bruker QSEQ data where data points are collected alternately as on some older Bruker spectrometers.

When you are satisfied that the header parameters are correct, click **OK**.

FELIX now displays the main 1D data-processing control panel, whose controls are grouped into several sections. The top section lists the individual processing options. These options are similar to

those used for 2D/3D/4D data processing. See Table 16 for more information on the individual processing options.

### Table 16 Main 1D Data-Processing Control Panel Menu Items

| Control | Description |
|---|---|
| **Dimension to Process** | Specify which dimension to process. For the 2D/ND menus you can choose standard processing or reprocessing. You might want to reprocess a dimension if you had first processed it without linear prediction. After fully processing the matrix, you could reprocess a dimension and do linear prediction by using an option such as **D2HT+IFT+LP+FT**. This does a Hilbert transform on the data in D2, inverse transforms it, linear predicts, and then does a normal ft. If you want to reprocess a dimension later on, you should not use a window function during the initial transformation of that dimension.<br>These macros generally assume the following correspondence between the FELIX matrix dimensions D1/D2/D3 and the experimental acquisition dimensions:<br>`For 2D Data` `For 3D Data` `For 4D Data`<br>`D1 --> t2` `D1 --> t3` `D1 --> t4`<br>`D2 --> t1` `D2 --> t2` `D2 --> t3`<br>`D3 --> t1` `D3 --> t2`<br>`D4 --> t1`<br>The FELIX D1 dimension always corresponds to the acquisition dimension. |
| **Output Matrix Filename** | Name of the matrix file that will contain the processed data. This file is built by the D1 transform. |
| **Load Matrix in Memory** | Load the entire matrix in memory instead of keeping it on disk. If you have enough memory this can significantly decrease the amount of time required for processing (not available for the D1 dimension). |
| **Dimension 1 Size** | Number of points in the D1 dimension of the matrix file (which is directly related to the digital resolution in D1). This number must be a power of 2. |
| **Dimension 2/3/4 Size** | Number of points in the indicated dimension of the matrix file, (which is directly related to the digital resolution in that dimension). Processing considerations mandate that these size parameters must be at least as large as the number of data records collected along that dimension. Thus, for a given number of complex points collected along one of the indirect dimensions the **Dimension Size** parameter must be at least twice as large as the number of complex points in order to hold all the data records collected in that dimension. This number must always be a power of 2. |
| **Processing Mode** | Specify how the data are processed. If you select **bundle** then the data are processed on the local workstation using the FELIX "bundle" mode. If you want to process the data using several workstations then select **distributed**. |

**Table 16 Main 1D Data-Processing Control Panel Menu Items (Continued)**

| Control | Description |
|---|---|
| **Correct DC-offset** | Remove a DC value from the time-domain data, provided that the data points represented by the **Fraction** parameter are actually zero ± noise. The **Fraction** parameter indicates the decimal fraction of the final part of the data vector to use for the DC-offset calculation, that is, 0.2 indicates that the final 20% of the FID is used in the calculation that determines how much DC-offset to remove. |
| **Correct 1st-point** | Correct the initial data points using linear prediction or by a simple scaling of the first point (for D2 or D3 processing). If the initial dwell time in the virtual time domain of an *N*D experiment is something other than one-half the normal value, a correction to the value of the first point in the D2(t2) vector and/or the D3(t1) vector may be desirable (Otting et al. 1990). The **Correct 1st-point** parameter allows you to specify a simple scaling factor of one-half or to employ linear prediction to reconstruct the first data points. |
| **Zero Fill** | Specify the final data size for 1D processing (not applicable to 2D/3D/4D processing). |
| **Solvent Suppression** | Three types of solvent suppression are available: a linear-prediction-based algorithm, a convolution-based method, and a polynomial-based method.<br>Parameters for the convolution-based method, the LP-based methods, and the polynomial-based method are discussed in detail in *Process/Solvent Suppression* and in the FELIX *Command Language Reference Guide* under the **cnv**, **lps**, and **pso** commands, respectively. |
| **Window Function** | Specify what type of apodization function to use. The options for **Window Function** are discussed in detail in *Window function descriptions*. |
| **Linear prediction** | Whether to augment the acquired data using a straightforward LP-based extension or the "mirror-image" trick (Zhu and Bax 1990), which is particularly appropriate for data acquired using "constant-time" evolution in t2 and/or t1. The goal is a slight improvement in resolution. |
| **FT Type** | The **D1 transform** command provides options for four types of Fourier transforms: **Real** (rft), **Complex** (ft) for true complex data, **Bruker** (bft) for Bruker data acquired using alternately sampled real and imaginary data points, and **Oversampled** (dft) for Bruker digitally filtered data. If **Oversampled** data is specified, then the appropriate **Decim Factor** and **DSPIVS FACTOR** are taken from the parameter files. |

**Table 16 Main 1D Data-Processing Control Panel Menu Items (Continued)**

| Control | Description |
|---|---|
| **Phasing Mode** | The selection for **Phasing Mode** depends upon whether or not appropriate zero (**Phase0**) and first-order (**Phase1**) phase parameters are known. If such values have previously been determined, then you may set the parameter value to **Use Parameters** and supply appropriate values for the phase parameter input boxes. If the phase parameter values are not known, the **Interactive Mode** opens the real-time phasing interface (see *Real-time phase correction*). If you select the **Interactive** mode, you can enter the number of the FID's you would like to phase in the **FID to Phase** box. |
| **Baseline Correction** | Please see *Process/Baseline correction* for a thorough discussion of the baseline correction utilities. |
| **Absolute Intensity** | Set absolute intensity mode **on**, which means that the spectrum scaling is set by the last displayed spectrum. This allows you to change various processing parameters and still observe the effect on the intensity of peaks. |
| **Reverse vector** | Certain hypercomplex phase-cycling protocols effectively render the complex-conjugate of what is normally expected by the *N*D States processing utility—such data appear to be reversed in the D2 and/or D3 dimension. If prior experience indicates that such a situation prevails for your data, you may specify that data vectors be reversed in D2 and/or D3 as a part of the *N*D processing. You may also delay such correction and use the **Process/Reverse Matrix** command at a later time. |
| **Extract Half Spectrum** | Extract a region of the D1 dimension for further processing. You can specify the left half of the spectrum, the right half of the spectrum, or the region to be extracted (available only for the D1 dimension). |
| **Output Level** | The amount of information written to the text window. The **Quiet** option prints a minimal amount. The **Verbose** option prints information on which vector is being processed so you can follow the progress of the calculation, and slightly lengthens processing time. |

The **Interactive Processing** section of this control panel contains a set of controls for interactively processing the data while varying the parameters for a specific processing action. The bottom section of the control panel specifies the manner in which the processing is done. Click **Apply** to perform the operations specified in the list of

processing operations and redisplay the main processing panel. click **OK** to perform the indicated processing options and close the panel.

> **Note:** To use this control panel, it is a good idea to set up the basic processing first. For example, try setting **FT**, **Set Phase Correct**, and **Real Time** to **on**. Then click **Apply**. This performs the FT operation and starts the real-time phase interface. At this point you can adjust the phase interactively. When you finish phasing (click **Keep** in the phase panel) you are brought back to the main 1D processing control panel, where you can select other specific processing options. Here it is often a good idea to set the **Phase Correct** mode to **Use Current**. The current phasing parameters are then used for subsequent phasing operations.

To interactively vary the parameters for the chosen processing step, click one of the **Interactive Processing** buttons. The **Interactive Processing** options allow you to control the various processing parameters with sliders. As you adjust the sliders you can simultaneously see the effect on the FID. These interactive options also allow you to display the transformed spectrum. This way you can see the effect of varying a processing parameter on both the FID and the transformed spectrum. Click **Keep** in one of these interactive processing panels to return to the main control panel. To exit the main processing panel, click **OK** or **Cancel**.

## Process/2D Data Processing

Use the **Process/2D Data Processing** command (<Alt>+s2) process 2D data. FELIX displays a control panel to specify the **Filter Type** for the kind of data you want to process. In general, if this is the initial processing of raw spectrometer data set you should specify **All Files (Bruker, Varian, Jeol:*)** as the **Filter Type**. You can then navigate through the desired directories to get to the data. You must select the actual spectrometer datafile, which is usually an FID or SER file. To select the FID double-click the filename, or select the filename and then click **OK**.

FELIX next displays a control panel with the 2D header menu parameters. These parameters are taken from the header information of the spectrometer datafile. Be certain that the proper parame-

ter values are displayed and correct them if necessary. Then set the **Data Source** to correspond to the type of data you have. The choices are: **General**, **Bruker**, **Varian**, and **Jeol**. Use the **Data Source** setting to determine how to enter the acquisition information in relationship to how the data were collected.

## General Processing

If you set the **Data Source** to **General**, FELIX displays a general control panel for processing 2D data that are not related to any specific spectrometer type. If you select **General** or **Jeol** processing, you are presented with the same control panel of generalized processing choices. The **Data Type** parameter relates to the form of the FID in the D1 (t2) dimension. The FID in the acquisition dimension is almost always **Complex**.

When FELIX processes 2D data, the method used to collect the data in the indirectly detected dimension affects how the data are processed in the acquisition dimension. This simplifies data processing in the indirect dimension and makes it easier to examine the data along this dimension.

The indirect dimension is almost always processed as **States** or **TPPI**. Thus, for the **Acquisition Mode**, this parameter determines how the D1 (t2) dimension is processed, but is set based on how the indirectly detected dimension was collected. So if data were collected as a Bruker style echo / anti-echo experiment in D2 (t1), the **Acquisition Mode** is set to **Echo/Anti-Echo** and the **Acquisition in D2** parameter is set to **States**. This is because the **Echo/Anti-Echo** experiment results in a complex interferogram along t1.

## Bruker Processing

If you are processing Bruker data, it is generally easier to set the **Data Source** parameter to **Bruker**. FELIX displays a simplified control panel specifically for Bruker data. The **Data Type** parameter is **Complex** unless you have an FID where the data points are sampled sequentially. The **Acquisition in D2** parameter is set based on the Bruker mode of data collection. You can generally determine this from the value of the MC2 parameter in the Bruker proc2s parameter file.

## Varian Processing

If you are processing Varian data it is generally easier to set the **Data Source** parameter to **Varian**. FELIX displays a simplified control panel specifically for Varian data. The **Data Type** parameter is **Complex**. The **Acquisition Mode** parameter is set based on the type of experiment you have. Varian data is most often collected as **States**. If you have a sensitivity-enhanced sequence of the Lewis Kay type, set **Acquisition Mode** to **Sensitivity Enhanced**. If you use the **grad_sort_nd** program to pre-process your data before processing on the Varian, then your data is of the **Sensitivity Enhanced Type**.

After you specify the acquisition parameters, click **OK**. FELIX displays the main 2D control panel. Here you specify the exact sequence of options to use during processing. See Table 20 for more information on the various processing options. When you click **OK** from this control panel FELIX prompts you to supply any needed additional information, and then processing is performed. When you process the D1 dimension of a 2D data set the matrix is first built and then the individual vectors from the input data set are read in, processed, and stored in the matrix.

To process the second dimension of a 2D data set, select the **Process/2D Data Processing** command again and specify a file type of FELIX **Matrix**. In the header menu verify that the **Data Source** parameter is still correct. Then select the D2 dimension for processing in the main 2D control panel. When you click **OK**, the individual vectors from the matrix are read in one at a time, processed, and stored in the matrix.

## Process/3D Data Processing

Use the **Process/3D Data Processing** command (<Alt>+s3) to process 3D data. FELIX displays a control panel where you can specify the **Filter Type** for the kind of data you want to process. In general, if this is the initial processing of raw spectrometer data, you should specify **All Files (Bruker, Varian, Jeol:*)** for the **Filter Type**. You can then navigate through the desired directories to get to the data. You must select the actual spectrometer datafile. This is usually an FID or SER file. To select the FID, double-click the filename or select the filename and then click **OK**.

At this point FELIX displays the 3D header menu parameters. These parameters are taken from the header information of the spectrometer datafile. Verify that the proper parameter values are displayed and correct them if necessary. Set the **Data Source** to correspond to the type of data you have. The choices are **General**, **Bruker**, **Varian**, and **Jeol**. The setting for **Data Source** determines how you enter the acquisition information in relationship to how the data were collected.

## General Processing

If you select **General** as the **Data Source** FELIX displays a general control panel for processing 3D data that are not related to any specific spectrometer type. Selecting **General** or **Jeol** processing opens the same control panel of generalized processing choices. The **Data Type** parameter relates to the form of the FID in the D1 (t3) dimension. The **fid** in the acquisition dimension is almost always **Complex**.

When you process 3D data in FELIX, the method used to collect the data in the indirectly detected dimensions also affects how the data are processed in the acquisition dimension. This simplifies data processing in the indirect dimensions and makes it easier to examine the data along these dimensions.

The indirect dimensions are almost always actually processed as **States** or **TPPI**. Thus for the **Acquisition Mode** this parameter determines how the D1 (t3) dimension is processed but is set based on how the indirectly detected dimensions were collected. So if the data were collected as a Bruker style echo / anti-echo experiment in D3 (t1) and states-TPPI in D2(t2), then **Acquisition Mode** is set to **Echo/Anti-Echo States-TPPI** and the **Acquisition in D2** and **Acquisition in D3** parameters are both set to **States**. This is because the echo / anti-echo experiment results in a complex interferogram along t1.

The **Acquisition Method** determines how the t1–t2 time point values are collected. If the data are collected as a group of four complex FID's corresponding to each t1–t2 time point, then the **Acquisition Method** is **Quartets**. If the data are collected, for example, as pairs of real and imaginary t2 components for all of the real t1 values, followed by the series of pairs for the imaginary t1 values, then this is

referred to as **Planes**. Bruker generally collects data as planes, while Varian generally collects data as quartets.

The **First Incremented** parameter specifies the order in which the FID's were collected. If **First Incremented** is set to **t2**, this means that the t2 parameter was incremented first and then the t1 parameter; that is, for each t1 time increment the *entire* set of t2 time increments is collected before proceeding to the next t1 time value.

- Standard Bruker sequences, which are collected as "3-1-2", should have **First Incremented** set to **t1** because the data are collected as a series of t3–t1 planes.
- Bruker data collected as "3-2-1" should have **First Incremented** set to **t2**.
- Standard Varian data, which are normally collected as "d3, d2", should have **First Incremented** set to **t2** because the data are collected as a series of t3–t2 planes.
- Varian sensitivity-enhanced sequences, which are collected as "d3, d2", should have **First Incremented** set to **t1**.

## Quartet Order Parameter

Use the **Quartet Order** parameter when the **Acquisition Type** has been set to **Quartets**. It determines the order in which the individual elements of the complex quartet are collected and therefore which dimension in the quartet is incremented first.

If the **Quartet Order** parameter is set to **t2**, this implies that the FID's were collected in the following sequence:

| FID# | t1(D3) | t2(D2) |
|---|---|---|
| 1 | real | real |
| 2 | real | imaginary |
| 3 | imaginary | real |
| 4 | imaginary | imaginary |

If the **Quartet Order** parameter is set to **t1**, this implies that the FID's were collected in the following sequence:

| FID# | t1(D3) | t2(D2) |
|---|---|---|
| 1 | real | real |
| 2 | imaginary | real |
| 3 | real | imaginary |
| 4 | imaginary | imaginary |

## Bruker Processing

If you are processing Bruker data it is generally easier to set the **Data Source** parameter to **Bruker**. FELIX displays a control panel specifically for Bruker data. The **Data Type** parameter is **Complex** unless you have an FID in which the data points are sampled sequentially. The **Acquisition in D2** and **Acquisition in D3** parameters are set based on the Bruker mode of data collection. You can generally determine this from the values of the MC2 parameters in the Bruker proc2s and proc3s parameter files.

The **Acquisition Order** parameter is determined by which dimension (t1 or t2) is incremented first. If t1 is incremented first, then the **Acquisition Order** is set to "3-1-2". If t2 is incremented first then it is set to "3-2-1".

## Varian Processing

If you are processing Varian data it is generally easiest to set the **Data Source** parameter to **Varian**. FELIX displays a control panel specifically for Varian data. The **Data Type** parameter should be set to **Complex**. The **Acquisition Mode** parameter is set based on the type of experiment. Varian data are most often collected as **States**. If you have a sensitivity-enhanced sequence of the Lewis Kay type, then set the **Acquisition Mode** parameter to **Sensitivity Enhanced**. If your data were pre-processed using the **grad_sort_nd** program before processing on the Varian, then your data are of the **Sensitivity Enhanced Type**.

Set the **First Incremented** parameter based on which dimension is incremented first. Varian data are most often collected as "d3, d2". Set the **Quartet Order** parameter based on the order in which the elements of the complex quartet of FID's were collected. This parameter is set based on the array parameter in the Varian procpar file. Set

this parameter to **phase**, **phase2**, or **phase2, phase**, depending on the value of the array parameter.

After you specify the acquisition parameters, click **OK**. FELIX displays the main 3D processing control panel. Here you specify the exact sequence of options that will be used during processing. See Table 20 for more information on the various processing options. When you click **OK** in this control panel, FELIX prompts you to supply any needed additional information and then processing is performed. When you process the D1 dimension of a 3D data set the matrix is first built and then the individual vectors from the input data set are read in, processed, and stored in the matrix.

To process the second dimension of a 3D data set, select the **Process/3D Data Processing** command again. Now specify a file type of FELIX **Matrix**. In the header menu, be sure that the **Data Source** parameter is still correct. Then in the main 3D processing control panel, select the D2 dimension for processing. When you click **OK**, the individual vectors from the matrix are read in one at a time, processed, and stored in the matrix.

## Process/3D Plane Processing

Use the **Process/3D Plane Processing** command (<Alt>+sl) to process a 2D plane from a 3D data set. This command is similar to the **Process/3D Data Processing** command described above. The difference is that plane processing creates a 2D matrix instead of a 3D matrix. This function builds the 2D matrix, reads in each vector from the input data set, processes them, and stores them in the 2D matrix. After performing this step, you will have a 2D matrix that is processed in the D1 dimension only. You must then use the **Process/2D Data Processing** command to process the D2 dimension of this new matrix.

In the control panel for processing parameters you can specify a D1-D2 (t3-t2) plane or a D1-D3 (t3-t1) plane for processing. The other options in the various panels are the same as for 3D processing, described above.

## Process/4D Data Processing

Use the **Process/4D Data Processing** command (<Alt>+s4) to process 4D data. FELIX displays a control panel to specify the **Filter Type** for the kind of data you want to process. In general if this is the initial processing of a raw spectrometer data set, you should specify **All Files (Bruker, Varian, Jeol:*)** as the **Filter Type**. You can then navigate through the desired directories to get to the data. Select the actual spectrometer datafile. This is usually an FID or SER file. To select the FID double-click the filename, or select the filename and then click **OK**.

Next FELIX displays the 4D header menu parameters. These parameters are taken from the header information in the spectrometer datafile. Verify that the proper parameter values are displayed and correct them if required. Set the **Data Source** parameter to correspond to the type of data you have. The choices are **Unknown**, **Bruker**, **Varian**, and **Jeol**. 4D data are handled with a general processing scheme that is not specific to any spectrometer type. The acquisition parameters to be entered for 4D data are analogous to those for 3D data with an additional dimension. For more information on entering the acquisition information see “General Processing”. The processing options are the same as those for 2D and 3D data.

## Process/Phase Correct Matrix

Processed data occasionally require re-phasing in one or more dimensions. Use the **Process/Phase Correct Matrix** (<Alt>+ph) command to re-phase a previously processed *N*D dataset that is currently open in the frame. FELIX displays a control panel to specify the dimension to rephase and the method. In addition to automatic phasing, you can give explicit values for the phasing parameters **Phase0** and **Phase1** or adjust the phase parameters interactively. click **OK** to re-phase all vectors in the matrix using these phase parameters (the matrix must be write-enabled).

If you select the automatic phasing function, you can select PAMPAS or APSL as the phase-detection algorithm and can define some excluded areas, to exclude noise while searching the sample peaks for calculation of phase parameters. You can also specify a **Filter**

**Width**, which is the minimum peak width (at the peak bottom) required for sample peaks. FELIX suggests a value for **Filter Width** based on the matrix data, but you can change the value.

## Process/Baseline Correct Matrix

FELIX provides a host of baseline-correction options, and three of the more popular methods, FLT, convolution, and FaceLift, have been used as the basis for a set of post-transform baseline-correction commands. All the available baseline-correction methods are discussed in detail under “Process/Baseline correction”. Use the **Process/Baseline Correct Matrix** command (<Alt>+pb), to specify the method and which dimension the baseline correction should proceed along.

### FLATT method

The **FLATT** command conducts baseline correction on the selected dimension of the transformed *N*D data using the algorithm of Guntert and Wuthrich (1992). The FLATT algorithm discriminates baseline segments and uses a linear least-squares solution to fit a trigonometric series to the baseline points.

Use **Baseline width** to determine a minimum chi-square value. A value that represents the half-width (in points) of the broadest resonance in the spectrum generally yields satisfactory results.

Use **Points to Correct** to specify the number of trigonometric terms to use in the baseline fit. Use **Tau** to specify the factor by which a segment may exceed the minimum chi-square value and still be considered a baseline segment. You may directly specify the source of the chi-square value or allow the utility to derive it (the value then represents the average over all vectors in the matrix). Armed with these values, the function loads and baseline-corrects every vector in the matrix (the matrix must be write-enabled).

### Convolution method

Use the **Convolution** command to conduct baseline correction on the selected dimension of transformed *N*D data using the algorithm of Dietrich et al. (1991) as developed by W. Massefski. This ABL-based utility automatically discriminates baseline segments and

conducts a running-average convolution of the baseline points, while a simple linear correction is applied to the intervening spectrum regions.

Use **Noise size** to specify the convolution width (in points) for the baseline regions. **Peak size** represents the half-width (in points) of the broadest resonance in the spectrum. Using these values, the function loads and baseline-corrects every vector in the matrix (the matrix must be write-enabled).

### FaceLift method

Use the FaceLift function to conduct baseline correction on the selected dimension of transformed *N*D data using the algorithm of Chylla and Markley (1993). This signal recognition-based utility identifies baseline points and filters the high-frequency noise along other dimensions before the baseline matrix is subtracted from the original matrix.

Use **Filter Width** to determine the half-width of the smoothing data window over which datapoints are sampled. The recommended range is 32–64 data points (powers of 2 are not necessary).

Use **Number of Standard Deviations** to determine a threshold standard deviation, above which any point is considered to be a signal point. The recommended range is 2.5–3.0.

Use **D1** (**D2**, **D3**, or **D4**) **Points to Smooth** to calculate the half-width of the smoothing data window that is used to smooth the base-point correction matrix along D1 (D2, D3, or D4). If it is the same dimension as that being baseline corrected, you should use the same value as for the **Filter Width**. Otherwise, a value of 2–4 is recommended (the matrix must be write-enabled).

## Process/Reverse Matrix

Certain hypercomplex phase-cycling protocols effectively render the complex-conjugate of what is normally expected by the **ND States** processing utility. Such data appear to be reversed in the D2 and / or D3 and / or D4 dimension. If prior experience indicates that such a situation prevails for your data, you may specify that data vectors be reversed in D2 and / or D3 and / or D4 as a part of the *N*D processing.

Use the **Process/Reverse Matrix** (<Alt>+pr) command to specify the dimension to reverse in the currently open matrix. Every vector along the specified dimension is reversed (the matrix must be write-enabled).

## Process/Utilities

### Process/Utilities/Squeeze Matrix

Use the **Process/Utilities/Squeeze Matrix** (<Alt>+pss) command to squeeze the current matrix (that is, to discard all the points below the threshold you define). This is useful for retaining only those portions of the matrix where real peaks can be found. Depending on the threshold, you can compress the file quite a bit, which can speed up the redraw and shorten the access time for originally large 3D and 4D spectra.

> **Caution:** This procedure is irreversible, and some actions (for example, volume measurement and peak optimization) may not work well on such a matrix.

### Process/Utilities/Unsqueeze Matrix

Use the **Process/Utilities/Unsqueeze Matrix** (<Alt>+psu) command to create an unsqueezed matrix from a previously squeezed one. This command is not the reverse of **Process/Utilities/Squeeze Matrix**, since it merely inserts zeros in those places in the matrix that were previously (during a squeeze) discarded.

### Process/Utilities/Transpose Matrix

Use the **Process/Utilities/Transpose Matrix** (<Alt>+pst) command to swap two dimensions of a processed matrix.

### Process/Utilities/Projection

Use the **Process/Utilities/Projection** (<Alt>+psp) command you to create a 2D projection of the current 3D or 4D matrix.

### Process/Utilities/Diagonal Plane

Use the **Process/Utilities/Diagonal Plane** (<Alt>+psd) command to extract a 2D diagonal plane from a typically homonuclear 3D matrix.

## Tools pulldown

FELIX 2002 includes several menu items (commands) that affect frequency-domain spectra in the workspace. Most of these functions are directly related to the transformation of multi-dimensional spectra, but several affect the processing of 1D data.

### Tools/Buffers

Buffers are accessed from within the interface by selecting the **Tools/Buffers** command (<Alt>+tb).

#### Tools/Buffers/Store Work to Buffer

To store the current information in the workspace to a buffer, select the **Tools/Buffers/Store Work to Buffer** command (<Alt>+tbs) and enter the buffer number in the control panel. To visualize this information, you must change the stack depth to include that buffer. This action is useful for saving a spectrum when the workspace is needed for some other process.

#### Tools/Buffers/Load Work from Buffer

To load buffer information to the workspace, select the **Tools/Buffers/Load Work from Buffer** command (<Alt>+tbl) and enter the buffer number.

#### Tools/Buffers/Add Work to Buffer

Use the **Tools/Buffers/Add Work to Buffer** command (<Alt>+tba) to add the current contents of the workspace to the specified buffer. This action is especially useful for generating projections of multi-dimensional spectra and for co-adding the absorptive components of hypercomplex data.

### Tools/Buffers/Multiply Work by Buffer

Use the **Tools/Buffers/Multiply Work by Buffer** command (<Alt>-tbm) to multiply the data in the workspace by the contents of the defined buffer. This action is most often used to multiply the data in the workspace by an apodization function that was stored earlier in a buffer.

### Tools/Buffers/Subtract Work from Buffer

Use the **Tools/Buffers/Subtract Work from Buffer** command (<Alt>-tbu) to subtract the contents of the workspace from the defined buffer. This action is especially useful for generating projections of multi-dimensional spectra and for co-adding the absorptive components of hypercomplex data.

### Tools/Buffers/Push Work to Stack Top

Use the **Tools/Buffers/Push Work to Stack Top** command (<Alt>+tbp) to push data into the buffers. FELIX stores the contents of the workspace on the top of the buffer stack. Every time you push onto the stack, you increase the stack depth by one.

### Tools/Buffers/Pop Work from Stack Top

Use the **Tools/Buffers/Pop Work from Stack Top** command (<Alt>+tbt) to load the contents of the top of the buffer stack to the workspace and decrease the stack depth by one. Use this command in conjunction with the **Tools/Buffers/Push Work to Stack Top** command, t.

### Tools/Buffers/Exchange Work/Stack Top

Use the **Tools/Buffers/Exchange Work/Stack Top** command (<Alt>-tbx) to exchange the contents of current workspace with the top of the buffer stack. This is useful for moving data back and forth between workspace and the buffers.

### Tools/Buffers/Zero Stack Depth

Use the **Tools/Buffers/Zero Stack Depth** command (<Alt>+tbz) to reset the stack display to show only the current workspace. This option is useful to bring the program back to a pre-defined state.

## Tools/Lists

One of the most important features of the FELIX database facility is the ability to create fast read-only representations of database entities, or *lists*. Such lists do not actually hold database information per se, but consist of pointers to items in a database entity. Lists exist in the memory that is allocated to the 1D buffers, so extremely long lists may require memory reconfiguration.

The **Tools/Lists** (<Alt>+tl) command displays a pullright sub-menu of items, as described below.

### Tools/Lists/List 1...4

Use the **Tools/Lists/List 1…4** (<Alt>+tl1…4) command to select from among four lists. All subsequent **Lists** commands then operate on the selected list, which is symbolized by the pushed-in toggle button beside it. When you change the list number, cross-peak footprints referenced by that list are *not* automatically drawn.

### Tools/Lists/Draw

Use the **Tools/Lists/Draw** (<Alt>+tld) command to display the contents of the current list, using the color specified for that list.

### Tools/Lists/Color

Use the **Tools/Lists/Color** (<Alt>+tlc) command to specify a footprint color for each of the four lists. The default pen color aliases are listed in Table 7.

### Tools/Lists/Zero

Use the **Tools/Lists/Zero** (<Alt>+tlz) command to re-initialize (or zero) the current list.

## List contents

Lists are composed using any of the six commands in the next group. The actions of all these commands are cumulative, so you can use them to compile complex sets of cross peaks into one list.

### Tools/Lists/Select Displayed

Use the **Tools/Lists/Select Displayed** (<Alt>+tlp) command to add all footprints in the currently displayed region to the current list.

### Tools/Lists/Select Region

Use the **Tools/Lists/Select Region** (<Alt>+tlr) command to add *only* the footprints in the interactively defined region to the current list.

### Tools/Lists/Select Line

Use the **Tools/Lists/Select Line** (<Alt>+tli) command to add footprints which intersect any part of the displayed cursor to the current list.

### Tools/Lists/Find by Name

If the cross peak entity contains assigned peaks, then use the **Tools/Lists/Find by Name** (<Alt>+tln) command to create lists using peak names (using either partial or complete names).

### Tools/Lists/Add One, /Remove One

Interactively add or remove footprints from the current list using the **Tools/Lists/Add One** (<Alt>+tla) and **Tools/Lists/Remove One** (<Alt>+tlo) commands.

## List action

Lists may be organized and reviewed using any of the four commands in the next category.

### Tools/Lists/Merge Lists

Use the **Tools/Lists/Merge Lists** (<Alt>+tlm) command to derive a third list from the union or the intersection of two other lists.

### Tools/Lists/Sort

Use the **Tools/Lists/Sort** (<Alt>+tls) command to sort lists in ascending or descending order, according to the D1-center or D2-center fields in the cross-peak entity.

### Tools/Lists/View

Use the **Tools/Lists/View** (<Alt>+tlv) command to review items that are referenced in the current list. FELIX displays a control panel where you specify whether to display peak centers, widths, and names using data points or ppm units. This displays a spreadsheet of cross peaks, *which should not be edited.*

### Tools/Lists/Write

Use the **Tools/Lists/Write** (<Alt>+tlw) command to create a hard-copy of the items that are referenced in the current list. FELIX displays a control panel to specify whether to display peak centers, widths, and names using data points or ppm units. These written lists are for record-keeping only and cannot be read back into FELIX.

## Tools/Generate Spectrum/FID

Use the **Tools/Generate Spectrum/FID** command (<Alt>+ts) to simulate spectra, FIDs, and noise.

*Generate single-spectrum lines*

Use the **Spectrum from Parameters** option to generate single spectrum lines from an amplitude, frequency, and line widths. You can generate Lorentzian, Gaussian, and Voigt line shapes. The spectrum line can overwrite the workspace or be added to the workspace.

*Generate FID*

Use the **FID from Parameters** option to generate a free induction decay (FID) from an amplitude, frequency, and a time constant. FIDs can overwrite the workspace or be added to the workspace.

*Generate white noise*

Use the **Add Noise** option to generate white noise. The noise can overwrite the workspace or be added to the workspace.

*Generate a 1D theoretical spectrum from output file*

The **Spectrum from File** option generates a 1D theoretical spectrum from a TurboNMR NMR shielding output file.

*Generate a 1D theoretical spectrum from 1D peak entity*

The **Spectrum from Peaks** option generates a 1D theoretical spectrum from a 1D peak entity in the database.

## Tools/Functions

The **Tools/Functions** command (<Alt>+tf) displays a pullright submenu of items for dealing with the workspace and defining specific data values within the workspace.

### Tools/Functions/Reduce to Real

Use the **Tools/Functions/Reduce to Real** command (<Alt>+tfr) to convert a complex spectrum to a real spectrum by discarding the imaginary part of the data in the workspace.

### Tools/Functions/Complex

Use the **Tools/Functions/Complex** command (<Alt>+tfc) to convert a real spectrum into a complex spectrum with a zeroed imaginary part.

### Tools/Functions/Reverse

Use the **Tools/Functions/Reverse** command (<Alt>+tfv) to reverse the data in the workspace by swapping datapoint values. Thus, data point 1 is swapped with data point *N*, and data point 2 is swapped with data point *N*-1, etc.

### Tools/Functions/Complex Conjugate

Use the **Tools/Functions/Complex Conjugate** command (<Alt>+tfo) to negate the imaginary part of the data in the workspace. This action reverses a spectrum if it is performed before executing a Fourier transform.

### Tools/Functions/Magnitude Spectrum

Use the **Tools/Functions/Magnitude Spectrum** command (<Alt>+tfm) to replace the real part of the workspace with the square root of $[(\text{real})^2+(\text{imag})^2]$, or the absolute magnitude of the data, and to replace the imaginary part of the workspace with the arctan (real / imag) or the phase array of the data, in the range -180 to +180°.

## Tools/Functions/Power Spectrum

Use the **Tools/Functions/Power Spectrum** command (<Alt>+tfp) to replace the real part of the data in the workspace with $[(real)^2+(imag)^2]$, or the power spectra, and to set the imaginary part of the data to zero.

## Tools/Functions/Alternate Real/Imaginary

Use the **Tools/Functions/Alternate Real/Imaginary** command (<Alt>-tfa) to change a spectrum that has been separated into real and imaginary parts into alternating real and imaginary parts. Alternating is the standard order where the real and imaginary parts of each data point are adjacent.

## Tools/Functions/Separate Real/Imaginary

Use the **Tools/Functions/Separate Real/Imaginary** command (<Alt>-tfs) to convert a spectrum that is in standard order into separate real and imaginary parts. In separate order, the real parts of all datapoints come first, followed by the imaginary parts of all datapoints.

## Tools/Functions/Exchange Real/Imaginary

Use the **Tools/Functions/Exchange Real/Imaginary** command (<Alt>-tfe) to exchange the real and the imaginary parts of the workspace. This action is most often used when you need to add the real component of a FID (that is, the part of the real serial file) to the real component of a FID that is part of the imaginary serial file.

## Tools/Functions/Shift Data

Use the **Tools/Functions/Shift Data** pullright (Alt-tfd) commands to shift the data in the workspace a specified number of points to the left or right.

- The **Right Shift** option shifts data to the right the specified number of points. The data shifted out is lost and zeroes come in on the other end.
- The **Left Shift** option shifts data to the left the specified number of points. The data shifted out is lost and zeroes come in on the other end.

- The **Circular Right Shift** option shifts data to the right the specified number of points. The data shifted out wraps around and comes back in on the other end.
- The **Circular Left Shift** option shifts data to the left the specified number of points. The data shifted out wraps around and comes back in on the other end.

## Tools/Functions/Set Data Size

The **Tools/Functions/Set Data Size** command (<Alt>+tfi) provides options to alter the size or number of datapoints in the workspace.

- The **Double Size** option doubles the size of the data in the workspace by performing a linear interpolation between existing datapoints.
- The **Halve Size** option halves the size of the workspace by averaging successive pairs of points.
- The **Set Size** option sets the size of the spectrum in the workspace. When the new size is less than the previous size, the right end of the spectrum is thrown away. When the new size is greater than the previous size, the spectrum is padded with zeroed datapoints out to the new size.

## Tools/Functions/Fold Data

The **Tools/Functions/Fold Data** command (<Alt>+tff) provides options for folding and unfolding data in the workspace.

- The **Fold Data** symmetrizes the 1D workspace by co-adding the first and last points, the second and next-to-last points, etc., until a halved symmetrized 1D spectrum is created. Performing a point-generated fold on the workspace decreases the size by a factor of two. This action is convenient for nondiagonal symmetrization of 1D spectra.
- The **Unfold Data** option re-symmetrizes the 1D workspace by regenerating the first and last points, the second and next-to-last points, etc., until a double symmetrized 1D spectrum is created. Performing a point-generated unfolding on the workspace increases the size by a factor of two.
- The **Low Point Fold** option symmetrizes the 1D workspace by keeping the smallest of the first and last points, the smallest of the

second and next-to-last points, etc., until a halved symmetrized 1D spectrum is created. Performing a low-point fold on the workspace decreases the size by a factor of two. This action is convenient for nondiagonal symmetrization of 1D spectra.

## Tools/Mathematics

### Tools/Mathematics/Set Data

The **Tools/Mathematics/Set Data** (<Alt>+tms) command includes three options:

- The **Workspace to Value** option sets all points in the display spectrum to the values specified. Both real and imaginary points can be assigned in the control panel that appears.
- The **Point Range to Value** option sets all points in the specified range to the values entered. Both real and imaginary points can be assigned in the control panel that appears. The range is specified in points.
- The **PPM Range to Value** option sets all points in the specified range to the values entered, with the range specified in PPM.

### Tools/Mathematics/Zero Data

The **Tools/Mathematics/Zero Data** (<Alt>+tmz) command contains several options for zeroing various datapoints in a spectrum:

- The **Zero Workspace** option sets all points in the spectrum to zero. This zeros both the real and imaginary parts of every point.
- The **Zero Real** option sets the real parts of all data points to zero. The imaginary parts are not changed.
- The **Zero Imaginary** option sets the imaginary parts of all data points to zero. The real parts are not changed.
- The **Zero Greater Than** option sets all points with values greater than the specified threshold to zero. This is a rather extreme way to wipe out large peaks in your spectrum and is sometimes used to remove a water peak.

- The **Zero Less Than** option sets all points with values less than the specified threshold to zero. This is also a rather extreme action, used to wipe out baseline noise and small peaks.

## Tools/Mathematics/Multiply Data

Use the **Tools/Mathematics/Multiply Data** command (<Alt>+tmm) to multiply all points in the workspace by a number. If the data in the workspace are complex, then the multiplier may be complex. Use this action to change both the magnitude and phase of the data in the workspace.

## Tools/Mathematics/Add To Data

Use the **Tools/Mathematics/Add To Data** command (<Alt>+tma) to add a number to all points in the workspace. The number may be real or complex.

## Tools/Mathematics/Absolute Value of Data

Use the **Tools/Mathematics/Absolute Value of Data** command (<Alt>-tmv) to replace each point in the workspace with its absolute value.

## Tools/Mathematics/Inverse of Data

Use the **Tools/Mathematics/Inverse of Data** command (<Alt>+tmi) to replace the data in the workspace vector with its inverse. This action takes the reciprocal of each point in the workspace (replace *value* by 1/*value*) and stores the new value in the workspace. Any zero points in the workspace are skipped to avoid a divide-by-zero error. Use this action to create novel and interesting window functions for apodization.

## Tools/Mathematics/Logarithm of Data

Use the **Tools/Mathematics/Logarithm of Data** command (<Alt>+tml) to replace each datapoint in the workspace with its natural (base e) logarithm. Use this action to compute some novel window function for apodization. Any zero points in the workspace are skipped to avoid a divide-by-zero error.

### Tools/Mathematics/Anti-Logarithm of Data

Use the **Tools/Mathematics/Anti-Logarithm of Data** command (<Alt>+tmn)to replace each data value in the workspace with exp(*value*), its natural (base e) anti-logarithm or exponential.

### Tools/Mathematics/Derivative of Data

Use the **Tools/Mathematics/Derivative of Data** (<Alt>+tmd) to push the derivative of the data in the workspace onto the current buffer stack.

### Tools/Mathematics/Integral of Data

Use the **Tools/Mathematics/Logarithm of Data** command (<Alt>+tmt) to push the integral of the data in the workspace onto the current buffer stack.

## Peaks pulldown

Pick 1D or *N*D peaks or resonances with the items in the **Peaks** pull-down (<Alt>+k). The commands in this pulldown work differently, depending on whether the active frame contains a 1D spectrum or an *N*D matrix.

### Peaks/Pick One

Use the **Peaks/Pick One** command (<Alt>+ka) to select one peak at a time. To change the pick parameters, select **Preference/Pick Parameters**.

### Peaks/Pick Region

To select peaks in a sub-region of your display, select the **Peaks** / **Pick Region** command (<Alt>+kg).

If you have a 1D spectrum or an *N*D spectrum for which the **Pick Region Mode** in the **Preference/Pick Parameters** control panel is set

to **Define by Cursor**, FELIX displays a small crosshair cursor that you can use to drag out a region.

Otherwise for *N*D spectra, FELIX picks-peaks in the displayed region (**Pick Region Mode** was set to **Displayed Region**), or you can specify a region through a control panel (if **Pick Region Mode** was set to **Define via Dialog**).

## Peaks/Pick All

Use the **Peaks** / **Pick All** command (<Alt>+kp) pick peaks in the full spectrum. FELIX displays a control panel where you can set the peak-pick parameters (which also can be accessed through the **Preference/Pick Parameters** command).

## Peaks/Remove One

Use the **Peaks/Remove One** command (<Alt>+kn) to delete peaks one by one using the cursor.

## Peaks/Remove Region

Use the **Peaks/Remove Region** command (<Alt>+kr) to delete peaks in a region that you select by dragging out with the small crosshair cursor.

## Peaks/Remove All

Use the **Peaks/Remove All** command (<Alt>+kl) to delete the peak entity.

## Peaks/Edit

Use the **Peaks/Edit** command (<Alt>+ke) to interactively move or adjust the cross-peak shape of a selected cross peak. First click on a cross peak to select it for editing. FELIX signals that the peak is selected by changing the cross peak's footprint color.

Next you have two choices: Adjust the footprint position by clicking near the center of the footprint and dragging, or adjust the cross-peak shape by clicking near the edge of the footprint and then dragging.

This function remains active until you press <Esc> key or click a point in the display at which no cross-peak footprint exists.

## Peaks/Filter

The **Peaks/Filter** (<Alt>+pf) command includes a set of tools for use *after* peak picking. These tools act as filters, eliminating a subset of cross peaks based on the criterion you select. Different numbers of tools are available, depending on the dimensionality of the spectrum.

*1D peak filtering*

Use the **Remove Redundant Peaks** command (<Alt>+kf) to remove redundant peaks for the set of picked 1D peaks.

*ND peak filtering*

Use the **Remove Diagonal Peaks** command to specify a filter tolerance (in datapoints, ppm, or Hz) and then remove cross-peak footprints that lie (within the specified tolerance) on the diagonal (D1=D2). For a 3D data set this diagonal can be any of the plane diagonals (D1=D2, D2=D3, or D1=D3) or the body diagonal (D1=D2=D3).

Use the **Symmetrize Spectrum** command to specify a filter tolerance (in datapoints, ppm, or Hz) and then remove cross-peak footprints that exist on only one side of the diagonal (that is, only those footprints at (D1,D2) which have symmetry-related partners at (D2,D1) are retained). This command also works for 3D spectra.

Use the **Merge Multiplets** command to specify a filter tolerance (in data points, ppm, or Hz) and then combine footprints that have D1/D2 centers within the specified tolerance of one another. The resulting footprints have widths that represent the largest of the widths of the unfiltered footprints in each dimension.

Use the **Filter by Width** command to specify a minimum and maximum footprint halfwidth (in datapoints, ppm, or Hz) and then remove footprints that fall outside the specified halfwidth limits.

Use the **Filter by Volume** command to specify a volume threshold and then removes peaks that have volumes below the specified

threshold. Before you can use this command you need to measure the volumes.

Use the **Filter by Intensity** command to specify a lower and upper intensity threshold and then remove cross peaks that fall either outside or inside the limits.

Use the **Remove Peaks in Region** command to specify a region (in data points, ppm, or Hz) and then remove footprints that fall inside the specified limits.

Use the **Remove Redundant Peaks** command to remove footprints that have D1 and D2 centers at precisely the same position. The resulting footprints have widths that represent the largest of the widths of the unfiltered footprints in each dimension.

Use the **Clean-up Ridges** command to specify a strip width (in data points, ppm, or Hz) and the maximum number of peaks that there should be along each strip. The function then removes the excess low-intensity cross peaks in each strip. Use this action to clean up noise peaks when you know that, at a given frequency (within a certain tolerance), there cannot be more than a certain number of peaks (for example, in an HCCH–TOCSY spectrum along the aliphatic H dimension within 0.1 ppm in C and 0.02 ppm in H, there cannot be more than 10 peaks).

Use the **Filter by Spectrum** command to clean up the current spectrum. You can, for example, clean up a 3D HSQC–NOESY spectrum based on an HSQC spectrum. FELIX prompts you to specify a spectrum and peak table to use (usually a 2D HSQC spectrum). Then specify which dimensions from the 2D to use for accepting peaks in the current spectrum within the given tolerance. FELIX deletes all other peaks (most likely noise peaks).

## Peaks/Optimize

This command (<Alt>+kz) offers a powerful line-fitting interface for deconvolution of complex spectra into individual peaks, which are described by an analytic function of intensity, linewidth, and frequency. These functions allow precise integration of individual peaks.

Depending on the dimensionality of the current spectrum, two different menu items can be accessed:

### *1D peak optimization*

Select the **Peaks/Optimize/Optimize** command to access the 1D line-fitting command and other menu items that display the actual data, the synthetic data, and residual data, either separately or all together in an overlay.

To fit a spectrum, first pick peaks using the **Peak** menu. this determines initial values for each peak's center, height, and width. Next select the **Optimize** command in the **Peaks** / **Optimize** menu. FELIX displays the interface in the current graphics frame with your spectrum in white, overlapped on the synthetic spectrum in red, and the synthetic line of a current peak in green. FELIX displays a modeless dialog containing buttons and sliders that permit you to fit the spectrum.

> **Note:** Since this is a modeless dialog, use the main menu items to add or delete peaks. Thereafter, always press the **Update Peaks** button in the dialog so that the new peaks are accepted for optimization.

The buttons next to the **Current Peak Item#** let you select a current peak among the picked peaks to display its properties and a synthetic line in green. You can press **Previous** or **Next** to access different peaks, or click **Cursor** to select a peak with the mouse.

**Integral** shows the integral of the current peak. If needed, type in an arbitrary value and click the **Normalize** button to normalize the integral of all peaks relative to that value.

Use the slider and readouts in the right half of the control panel to modify the center, height, and width of the selected peak. As you adjust the slider, the synthetic spectrum and synthetic line are updated on the fly, so that you instantly see how your changes fit the real spectrum.

Before starting the automatic fitting, you can click the **Setup** button to activate another control panel where you can set certain initial parameters such as the **Peak shape**, which can be Lorentzian or Gaussian, and optimization locks, which prevent the automatic optimizer from adjusting the specified parameters.

In this control panel you can select the control parameters for simulated annealing function, and the length of the peak tails used to calculate the spectrum. Simulated annealing is always used as the optimization methods.

The peak tails are given as a percentage of the peak height. So, for example, a peak tails setting of 0.01 means that the peaks are calculated from their centers to the point where they fall to 1% of their maximum value. Use this setting to accelerate the optimization by avoiding needless calculation of the contribution of peaks in regions where they have little or no significant amplitude.

Click the **Fit** button to activate the automatic optimizer. During the optimization, FELIX updates the spectra in the display area to show the progress. If you want to abort the fitting before it's completed, press the <Esc> key.

Click **OK** to exit the curve-fitting dialog and accept the optimized peak parameters. The optimized peak data are stored in the existing 1D peak-picking entity. Or click **Cancel** to exit and ignore the changes.

Use **Overlays** to plot the actual, synthesized, and residual data at the same time.

Use **Undo Overlays** to remove the three data types mentioned above and displays only the actual data.

Use **Actual Data** to plot only the actual data.

Use **Synthesize Data** to plot the synthesized data only.

Use **Residual Data** to plot the residual data only.

Use **Save Current Data** to save the current spectrum to a 1D disk file.

## ND peak optimization

Use this menu to optimize and model *N*D cross-peak footprints, shapes and volumes.

> **Note:** All the menu items require that you already have an entity of picked peaks and an entity of measured volumes for at least one mixing time.

### *Background*

At the heart of these peak-optimization and -modeling functions is the notion of a model cross peak. Whereas an actual cross peak is described by a collection of adjacent data points in a matrix having intensities greater than the neighboring region, a *cross-peak footprint* is described by a set of center position and halfwidth values, and a

volume is described as the total intensity of the cross peak's datapoints that lie inside the footprint; a *model cross peak* is defined as the distribution of intensity described by an idealized peak created from an analytic function of line widths, volume, and center positions.

FELIX uses either a Gaussian or Lorentzian line-shape function. The practical implication of these definitions is that FELIX can model the intensity of every cross peak that you have picked and measured a volume for, with a Gaussian or Lorentzian peak shape derived from the footprint and volume information.

The primary utility of these model cross peaks is twofold. First, you can display plots of model data or residual data (real minus model) to help with your early analysis and assignment work. There are many times when visually subtracting out a few well resolved peaks can reveal additional peaks hidden underneath. The second use for model cross peaks is to perform nonlinear least-squares optimizations on cross-peak shapes and intensities to yield the "best fit" values for peak center positions, halfwidths, and volumes; because the optimization algorithms seek to minimize the difference between the real data and the model representation of that data. Optimization can increase the confidence for assignment decisions based on peak alignment and improve volume buildup rate estimates for making distance restraints.

*Implementation*

Use the **Optimize** command to execute a conjugate-gradient minimization algorithm (the quasi-Newton) to improve the values for peak centers, halfwidths, and / or volumes that are stored in the peak and volume entities. The algorithm used is the quasi-Newton, as mentioned in the 1D line-fitting section. FELIX displays a control panel that prompts you for a peak entity, a volume entity, a volume slot number, and the type of line shape to use. You may select which values to optimize and which values to hold constant.

Keep in mind that, for each peak, there are two (or three or four) centers, two (or three or four) widths, and one volume; and that all are fit against a rather small set of data points (the points inside that peak's footprint). Accordingly, the fewer the values that are optimized per peak, the better-determined the algorithm (and the faster it executes). You may find that optimizing just volumes, then just centers, and so on, gives better results than optimizing everything at once. Again, this is primarily a concern when there are relatively few datapoints per footprint.

Once you have selected which values to optimize, click **OK**. FELIX reports how many sets of peaks it will operate on and then announces each set optimized and colors those peaks in green as it completes each set. When all sets are done, FELIX reports the initial and final penalty values in the output window. These penalty units are on an arbitrary scale, yet they represent the RMS deviation in the intensity values at every datapoint in the optimized region of the matrix that has any cross peaks. Lower penalty values correspond to a better fit.

FELIX stores the new values for centers, halfwidths, and volumes in their respective entities.

> **Caution:** The results of this action cannot be undone, so you may want to save backup files of your peaks and volumes first.

## Data modeling

So far we have only discussed the notion of actual matrix data. When you do a plot, the actual datapoints in the matrix are what get fed to the plot routine. In addition to actual data, FELIX can also display and analyze model and residual data.

Use the **Actual Data** command to use the actual matrix data. This reverses the actions of the **Model Data** and **Residual Data** commands.

Use the **Model Data** command to synthesize data from the current peaks and volumes, instead of reading datapoints from the matrix. All subsequent plots and analyses are performed on the synthesized model data. To return to using the actual matrix data, select the **Peaks/Optimize/Actual Data** command.

Use the **Residual Data** command to use a blend of real datapoints from the matrix and synthesized data from the current peaks and volumes. Specifically, the blend is “one part real minus one part model”; most commonly called the residual. All subsequent plots and analysis are performed on this residual data. Use this feature to measure the residual volume inside peak footprints that cannot be attributed to the peak, which is a sigma or uncertainty measurement for the real peak volume. To return to using the actual matrix data, select the **Peaks/Optimize Peaks/Actual Data** command.

Use the **Model Parameters** command to select whether Gaussian or Lorentzian line shapes are used in the modeling process.

> **Caution:** Be sure you remember to explicitly return to using actual data when you are finished investigating model and residual data.

## Peaks/Brother Peak

Use the **Peaks/Brother Peak** command (<Alt>+kb) to explicitly extend assignments to other cross peaks along the D1 or D2 dimensions. You first select, with the crosshair cursor, a cross peak that already has assignments in one or more dimensions. You then explicitly select cross peaks that share the same D1 or D2 assignments.

## Peaks/List

Use the **Peaks/List** (<Alt>+kl or <Ctrl>+l) command to check the status of individual peaks (chemical shifts, widths, peak assignments, and frequency assignments), which are listed in the output window or zoomed on in the peak table if it is open. You can click on a cross peak and the table scrolls to the location so that the row of that peak is visible and highlighted. Press <Esc> or click off a peak to exit.

## Peaks/Find

Use the **Peaks/Find (<**Alt>+kd) command to find a specific cross peak. Search for a cross peak by its assignment name or its number. Or use the **List/Name** command to create a list of cross peaks that match a search criterion.

If you choose to search the cross-peak entity for a specified assignment name, or partial name, FELIX creates a list that contains the discovered peak(s). You may use a wildcard character (*****) as part of the assignment name. If FELIX locates a peak, its display expands to show the found peak or the peak changes color, depending on your choice.

### Peaks/Name One Peak

Use the **Peaks/Name One Peak (**<Alt>+ko) command to select a peak interactively and then assign names to the peak.

## Measure pulldown

### Measure/Cursor Position

The **Measure** pulldown includes functions for obtaining point numbers, ppm values, and corresponding data values. When you select the **Measure/Cursor Position** command (<Alt>+up) while a 1D spectrum is displayed, FELIX displays a vertical half-cursor. FELIX reports the current axis position and data value as you move the cursor. The axis position is in axis-based units. If your axis is in points, it tracks in points; if your axis is set to ppm, it tracks in ppm. To quit the cursor-tracking mode, press <Esc>. The data value shown is the actual data value stored at that location in the workspace.

For *N*D spectra, accurate peak positions and heights can be read interactively from the display using this menu item. When you select the **Measure/Cursor Position** command, FELIX displays a crosshair cursor and reports the cursor position in x- and y-axis units (possibly the third and fourth dimension, as well); FELIX also reports the matrix value at that point.

FELIX continuously updates the information as you move the cursor, until you quit the cursor-tracking mode by pressing <Esc>.

### Measure/Correlated Cursors

Use the **Measure/Correlated Cursors** (<Alt>+uc) command to initiate a multiple-cursor mode. Use this mode to correlate peaks in more than one simultaneously displayed plot. For example, you may plot a TOCSY, COSY, and NOESY spectrum in three different frames. When you select the **Measure/Correlated Cursors** command and move the cursor into one of the frames, the cursor becomes a crosshair cursor and crosshair cursors appear and track

the identical positions in the other two frames. If the matrices are referenced and displayed with axis units of ppm, use the cursors to correlate peaks in the three spectra. Press the mouse button to report the current position. To quit multiple-cursor mode, press <Esc>.

## Measure/Distance/Separation

For a 1D spectrum, in addition to spectrum positions and intensities, you can calculate the separation between any two spectrum features in a display using the **Measure/Distance/Separation** command (<Alt>+ud). Use a crosshair cursor to select two locations on the display. FELIX reports the separation in points, ppm, and Hertz. To exit, press <Esc>.

For an *N*D spectrum, use the **Measure/Distance** command to find the distance between the two atoms defining a particular peak (either through peak assignment or through frequency assignment) by clicking the peak. FELIX measures the distance in the currently active molecule.

## Measure/Integral/Volume

### 1D Integral

You may use FELIX to integrate the entire spectrum as a single integral or as shorter segments. To integrate the entire spectrum, select the **View/Draw Integrals** command (<Alt>+vs). If you want additional options dealing with integrals, use the **Measure/Integral/Volume** command (see below).

*Add Segment*

Use the **Add** button to define integral segments. Add integral segments by dragging out a segment region with the mouse. To exit this mode, press the <Esc> key. FELIX displays integral curves after exiting the add segment mode.

*Remove Segment*

If you make a mistake while selecting individual segments or if you want to modify the current list of segments, you may delete a small subregion of segments graphically. Click the **Remove** button to create a small crosshair cursor. Then drag out a region of segments to delete.

*Remove All Segments*

To delete all the segments, click the **Remove All** button in the **Measure/Integral/Volume** command. This deletes the current integral segments entity from the database and requires confirmation via a dialog box.

*Adjust Integral Display*

Click the **Draw** button to redraw the integrals curves. Click the **Dismiss** button to remove the integral curves from the plot.

Click **Adjust** to adjust the display of the integrals. FELIX displays another dialog. Here you can select different display types, adjust the slope, bias, or overlap of the integral curves.

> **Caution:** By adjusting the slope and bias, you can set an integral value to any value. *Use these adjustments cautiously.*

Click the **Normalize** button if you want to normalize the display values of the integrals.

## *N*D volume measurement

FELIX calculates the volume of a cross peak as the integral of all data-point intensities inside the cross-peak footprint. The reserved symbol *hafwid* controls the relative size of all footprints, affecting the resultant volume measurements. (Please see the FELIX *Command Language Reference Guide* for more detailed information about this symbol).

Because the peak picker determines the cross-peak footprint widths from the peak's half-width at half-height, tall peaks, which have better signal-to-noise ratios, have relatively smaller footprints than small peaks with worse signal-to-noise (and thus relatively larger footprints). In this way, the volume of a strong peak is the sum of relatively few data points of high intensity, while the volume of a weak peak is the sum of relatively many data points of low intensity.

*Show One Volume*

Use the **Show One Volume** option to calculate and display the volume of a single cross peak. The arrow cursor becomes a large crosshair; use it to click the peak of interest. FELIX calculates the volume for the selected cross peak and displays it in the output window. When there is a current volume entity, FELIX displays both the raw calculated volume and the stored volume from the volume entity. As long as a peak is selected, FELIX repeats the action. To end the action, click in a region of no peaks or press <Esc>.

*Measure All Volumes*

Use the **Measure All Volumes** option to measure the volumes of all cross peaks. FELIX displays a control panel and prompts you to specify the cross peak entity name, the volume entity name, a slot number, and the mixing time for the current matrix in seconds. If the named volume entity does not yet exist, FELIX prompts you to specify the total number of slots to be built into the new volume entity. When the volume entity *does* exist, FELIX first checks to make sure that the number and IDs of all cross peaks match the number and IDs of all volumes. This assures that the two entities are compatible. Specifying a slot number that you have already used overwrites those volumes.

The measured volumes depend on the center and width of each cross peak and the reserved symbol *hafwid*, which can be adjusted using the **Preference/Peak Display** command. Measuring volumes is very fast for small cross-peak sets, but can take tens of seconds for very large numbers of cross peaks.

*Measure Buildups*

Use the **Measure Buildups** option to measure the volumes not only in one spectrum but in a series of spectra within the same buildup. It stores the result in the consecutive slots of the same volume entity.

*Remeasure Buildup for One Peak*

Use the **Remeasure Buildup for One Peak** option to remeasure a buildup for a particular peak.

*Delete Volumes*

Use the **Delete Volumes** option to delete the current volume entity, after prompting you for confirmation. There is no way to delete a portion of the volumes.

## Measure/Buildup

### Measure/Buildup/Show Buildup

Use the **Measure/Buildup/Show Buildup (**<Alt>+ubs) command to display a graphical representation of all the stored volume slots for a selected cross peak. Select a cross peak with the large crosshair cursor. FELIX produces a 2D graph of volume versus time, showing the volume buildup for that cross peak. The initial datapoint of zero volume at time zero is explicitly included.

### Measure/Buildup/Fit Buildup

Use the **Measure/Buildup/Fit Buildup** (<Alt>+ubf) command to fit the buildup data to any of seven different functions.

## Measure/J Coupling

FELIX provides a set of tools for extracting J-coupling constants from a variety of spectra. These coupling constants then can be used to calculate torsion angle restraints for structure determination studies.

Because of the complexities of overlapping COSY peaks and multiplet "splitting" effects within any one peak, you can calculate the J-coupling constants for only one peak at a time. Based on the quality of the results, you are then free to add each J-coupling measurement to the J-coupling entity one at a time.

### Measure/J Coupling /DQF

The **Measure/J Coupling/DQF** (<Alt>+ujd) command uses a sophisticated line-fitting algorithm to calculate the true centers of each lobe of a DQF–COSY multiplet and then measures the separation in Hertz. This command works only on non-overlapping peaks with four primary lobes (down, up, up, down). It cannot robustly handle peaks with more lobes due to "splitting". Use a crosshair cursor to select a peak. FELIX then calculates and reports the J coupling and sigma in the output window.

The algorithm works as follows:

1. For each dimension, FELIX divides the cross-peak footprint in half and then projects each half (sum the points) down to a 1D line segment. This yields two 1D line segments; each represents the line shape of two lobes of the cross peak (one up, one down). The number of datapoints in each line segment depends on the size of the cross-peak footprint in that dimension.
2. Next, each of these line segments is peak-picked (to yield two peaks; one positive, one negative) and then passed through the 1D curve fitter to optimize the two peak centers, widths, and heights to best-fit Gaussian line shape models. This optimization step is responsible for finding the "true" centers from the "appar-

ent" centers given by the peak-pick routine, for each line segment. This yields two independent measurements for the separation in each dimension.

3. FELIX reports the average separation as the J-coupling value and reports the deviation from that average as the sigma, or uncertainty, of that J-coupling value, for each dimension.

## Measure/J Coupling/Manual ECOSY

Use the **Measure/J Coupling/Manual ECOSY** command (<Alt>+ujy) to measure a coupling between two subpeaks of a typically ECOSY spectrum. Use the cursor to define the boundaries of two subpeaks; then, FELIX uses the optimizer (similar to DQF) to measure the distances.

## Measure/J Coupling/Heteronuclear ECOSY

Use the **Measure/J Coupling/Heteronuuclear ECOSY** command (<Alt>+uje) to measure J couplings in heteronuclear ECOSY-type 2D experiments (Griesenger et al. 1986).

## Measure/J Coupling/Heteronuclear FIDS

Use **Measure/J Coupling/Manual FIDS** command (<Alt>+ujf) to measure a $^{3}J$ coupling on two HSQC spectra using the FIDs (fitting of doublets and singlets), where you measure the fully decoupled and a partially coupled HSQC spectrum. After peak picking, the coupling constants can be extracted using time-domain fitting (Schwalbe et al. 1993).

## Measure/J Coupling/Heteronuclear FIDS/ECOSY

Use the **Measure/J Coupling/Heteronuuclear FIDS/ECOSY** command (<Alt>+ujs) to measure the $^{2}J_{CN}$ and $^{3}J_{CN}$ coupling constants on three 2D HSQC experiments, using the combined $C^{1}$–FIDS or FIDS–ECOSY method (Rexroth et al. 1995a).

## Measure/J Coupling/Heteronuclear DQ/ZQ

Use the **Measure/J Coupling/Heteronuuclear DQ/ZQ** command (<Alt>+ujz) to measure the J couplings on a pair of double-quantum

and zero-quantum 2D experiments like HN(CO)CA or HNCA (Rexroth et al. 1995b).

## Measure/J Coupling/Heteronuclear 3D ECOSY

Use the **Measure/J Coupling/Heteronuuclear 3D ECOSY** command (<Alt>+uj3) to measure the J couplings in a 3D ECOSY-type experiment.

## Measure/J Coupling/HSQC-J

Use the **Measure/J Coupling/HSQC-J** command (<Alt>+kjh) to measure the J coupling via a series of HSQC experiments, finding the coupling constant by interpolating the zero crossing of the volume series.

> **Note:** You must have picked peaks in a series of HSQC spectra and have measured volumes measured and stored them in the volume entity.

## Measure/J Coupling/Manual Separation

The **Measure/J Coupling /Manual Separation** (<Alt>+ujm) command is a primitive tool, compared to the line-fitting algorithm described above. You select a peak with a large crosshair cursor. FELIX plots the chosen peak in an expanded blown-up frame to represent as much peak shape as possible. Then, use a small crosshair to drag out a rectangular shape. Try to align the four corners of the rectangle on the true centers of the four lobes of the peak. **Measure/J Coupling /Manual Separation** then calculates the separation in Hertz from your cursor corners and reports the J-coupling values in the output window. There are no sigma terms with this method, since there is only one measurement for each dimension.

## Measure/J Coupling/Volume Ratio

Use the **Measure/J Coupling/Volume Ratio** command (<Alt>+ujv) to display the volumes of two peaks (e.g., an off-diagonal and a diagonal) selected via the cursor from the currently displayed *N*D spectrum (usually triple resonance). Use this volume ratio for calculating J coupling with an external program.

## Measure/Relaxation

The **Measure/Relaxation** menu offers a suite of tools that allow you to analyze a series of heteronuclear 2D relaxation spectra. FELIX assumes that peaks have been picked in one of the spectra or in a similar one with the same spectral widths.

### Measure/Relaxation/Measure Heights/Volumes

Use the **Measure/Relaxation/Measure Heights/Volumes** command (<Alt>+uv) to evaluate peak heights or volumes in the series of spectra. This feature includes optimization options to accommodate slight peak displacements relative to the initial peak table and moderate peak overlaps.

### Measure/Relaxation/S/N Ratio

Use the **Measure/Relaxation/S/N Ratio** command (<Alt>+us) to determine the signal-to-noise ratio by analyzing one or more duplicate spectra and extrapolating to the remaining time points.

### Measure/Relaxation/View Timecourse via Cursor

Use the **Measure/Relaxation/View Timecourse via Cursor** command (<Alt>+uc) to point at a peak in the displayed 2D spectrum and display the series of peak heights or volumes of that peak with error bars as determined by the previous two menu options. FELIX also displays the best-fit exponential decay curve if one has been fitted to the data, and displays the relaxation rate in the status bar and the output window.

### Measure/Relaxation/View Timecourse via Item

Use the **Measure/Relaxation/View Timecourse via Item** command (<Alt>+ut) to choose a peak number whose time course you want displayed.

### Measure/Relaxation/Fit R1/R2/NOE

Use the **Measure/Relaxation/Fit R1/R2/NOW** command (<Alt>+uf) to analyze the peak height or peak volume data. FELIX fits the R1 and R2 timecourses to appropriate exponential decay curves, taking

into account the experimental uncertainties. FELIX displays the resulting relaxation rates in the output window and stores them in a database table. The database table also holds the other parameters of the fitted curve in order to reconstruct it if necessary. FELIX evaluates the heteronuclear NOE by taking the ratio of the peak heights in spectra acquired with $^{1}H$ saturation and without $^{1}H$ saturation. Signal-to-noise evaluation is built into this procedure by analyzing a second pair of spectra.

### Measure/Relaxation/Modelfree input

Use the **Measure/Relaxation/Modelfree Input** command (<Alt>+um) to extract the relevant parameters from the FELIX database and prepare a rudimentary input file for the ModelFree program of A.Palmer (available at http://cpmcnet.columbia.edu/dept/gsas/biochem/labs/palmer/).

## Measure/Scalar/Normalize

*1D integral normalization*

FELIX can normalize the integral of any segment of the spectrum to an arbitrary value. Four normalization menu items are available under the **Measure/Scalar/Normalize** command (<Alt>+un). After normalization, FELIX updates the volume element in the integral segment entity to the normalized value.

> **Note:** This function can also be accessed directly from the **Measure/Integral/Volume** command for 1D spectrum.

### By Item Number of Segment

Use the **By Item Number of Segment** command to create a list box where you graphically select the segment to normalize based on its beginning and ending point. You must also give a normalization value for this segment.

### By Data Point Limits

Use the **By Data Point Limits** command to display a control panel in which you enter a low and high point to define a normalization range, as well as the normalization value.

## Select Segment via Cursor

Use the **Select Segment via Cursor** command to generate a small crosshair cursor to select the segment to normalize by dragging to enclose the segment.

## Raw Absolute Integrals

Use the **Raw Absolute ntegrals** command to store and display each segment's integral as its raw intensity, with no normalization at all.

*ND scalar*

The rate of volume buildup for a cross peak in a set of NOESY matrices is related to the distance between the two corresponding atoms in the molecule of study. Before deriving distance restraints from a set of volume buildup rates, you must define a scaling constant taken from cross peaks that correspond to fixed interatomic distances in the molecule. Define a small set of reference cross peaks that all correspond to a single fixed distance. FELIX then averages the buildup rates of these peaks to determine the scaling constant used to convert volume buildup rates into distance restraints.

> **Caution:** Due to the method of averaging these rates, it is very important that all cross peaks selected as scalar peaks correspond to one single interatomic distance in angstroms. The only exception to this rule is when the **Empirical Fit** method of restraint calculation is used. Here, the scalar entity consists of a series of scalar peaks that correspond to a range of interatomic distances.

> **Note:** We cannot overemphasize the importance of these reference peaks: they are crucial in defining NOE distance restraints.

## Add One

Use the **Measure/Scalar/Normalize/Add One** command to add one more cross peak to the scalar peak entity. FELIX displays a control panel and prompts you for the assignment names in D1 and D2 and a distance in angstroms. FELIX then searches the current cross-peak

entity for a peak with those names, verifies that peak is not already a scalar, and then adds one more entry to the scalar-peaks entity.

> **Caution:** Remember that all scalar peaks should represent *one single distance*. Basing the scaling constant on an average of different distances is not valid. The only exception to this rule is when the **Empirical Fit** method of restraint calculation is used (see above).

## Add One via Cursor

Use the **Measure/Scalar/Normalize/Add One via Cursor** command to add the peak by selecting it with the cursor.

## Delete One

Use the **Measure/Scalar/Delete One** command to remove one peak from the scalar entity. FELIX displays a control panel and shows you a list of all scalar peaks, by assignment name. Select one scalar to remove from the entity. Repeat this action to delete other scalar peaks.

## Clear All

Use the **Measure/Scalar/Clear All** command to delete the entire scalar-peaks entity. FELIX prompts you for confirmation.

## Change

Use the **Measure/Scalar/Normalize/Change** command to displays a control panel that shows the name of the current scalar entity. Only one scalar entity can be current at a time. To change to another scalar entity, enter a different name and click **OK**.

## Normalize/View

Use the **Measure/Scalar/Normalize/View** command to create a spreadsheet of the scalar entity. Here you view the individual scalar peaks and interatomic distances. In the spreadsheet, you can edit the distance.

## Measure/DISCOVER Restraints

One of the principal goals of analysis of 2D NOESY experiments is accumulating interatomic distance restraints for use in various molecule structure-determination studies. FELIX provides a set of menu items for turning volume buildup rates into distance restraints. Once you have an entity of scalar peaks, you are ready to generate distance bounds from assigned cross peaks. You can generate restraints in DGII/Discover format or in XPlor format, using the **Measure/DISCOVER Restraints** and **Measure/X-PLOR Restraints** commands, respectively.

*NOE distance*

Several restraint classes are supported in FELIX. The basic 2D NOESY peaks can be used in structure generation and refinement as NOE-distance restraints. Based on an assigned peak entity and measured volumes (optionally buildups), FELIX can create new restraints, interactively show one restraint, and use a list of violations to recalculate restraints.

## NOE Distance Define

Use the **NOE Distance Define** option to create a new entity **(msi:noe_dist)** of distance restraints from volume buildup rates. Defining restraints overwrites any existing restraints if **Action** in the control panel is set to **New**. If **Action** is set to **Append**, the restraints produced are appended to the restraint entity. The entity name containing the **scalar** information used in calculating the restraints is then input. You then input values for the **Lower Force Constant**, **Upper Force Constant**, and **Maximum Force**. Then you select the method used to derive the buildup rates. You can choose between using a **Single mixing time** or a "best fit" rate based on the **First N mixing times**, as calculated by a first-order polynomial (**Straight Line**), a **Second Order Polynomial**, or an **Empirical Fit**. In an **Empirical Fit** calculation, FELIX uses a scalar entity containing a series of cross-peak intensities that correspond to a range of interatomic distances to determine an empirical relationship between NOE intensity and distance.

Although NOE volumes, in reality, increase exponentially and are damped by the exponential T2 relaxation, there are simply not enough mixing times to yield a robust fit to such an exponential-type function. Since only the initial buildup rate is needed (not the

entire function), FELIX provides fit functions that can robustly calculate the initial rate from a small set of volume observations.

You can define, for the case of a symmetric spectrum, which peaks are to be considered in the restraint generation: all peaks or peaks on a specific side of the diagonal (for up to three user-defined regions) or only the lower-intensity peaks (symmetry selection).

All the above methods (except **Empirical Fit**) use the inverse $6^{th}$-power relation between buildup rate and interatomic distance to calculate a single distance in angstroms from the volume buildup rate for each cross peak and the scaling constant derived from the scalar peaks. That one resultant distance is then used to create a restraint having lower and upper distance bounds, based on the chosen method.

Use the **Exact Distance** method to create an entity where each restraint is an exact distance. FELIX displays a control panel that prompts you to specify the minimum and maximum distance allowed for any restraint. For all restraints, the lower- and upper-bound distances are both set to the one calculated distance.

Use the **Strong-Medium-Weak Bins** method to create an entity where each restraint is grouped into one of three distance-bound bins. FELIX displays a control panel that prompts you to specify the minimum and maximum distances allowed, the distance boundary between the strong and medium bins, and the distance boundary between the medium and weak bins. For all restraints, the lower and upper bounds are one of these three explicit distance ranges, depending on which bin the calculated distance falls in.

Use the **Van Der Waals-Exact** method to create an entity where each restraint uses a generic van der Waals hard-sphere radius for the lower bound and sets the upper bound to be the calculated distance. FELIX displays a control panel that prompts you to specify the van der Waals lower bound and the maximum upper bound.

Use the **Percentage of Distance** method to create an entity where each restraint uses a percentage of the calculated distance as the lower and upper bounds. FELIX displays a control panel that prompts you to specify the minimum and maximum distances, a lower-bound percentage, and an upper-bound percentage. For each restraint, the lower bound is determined by subtracting the lower-bound percentage of the calculated distance from the calculated distance, while the upper bound is obtained by adding the upper-

bound percentage of the calculated distance to the calculated distance. In this manner, short distances translate into narrower bounds while longer distances have wider bounds.

Also, you may specify that only the non-overlapped peaks be used in creating restraints. You can choose what percentage of the peak box area overlap is to be handled differently, by setting the **Area Threshold**. You can then discard those peaks (set **Partial Overlap** to **Discard**) or use a different method to turn them into restraints (e.g., **Use as Qual**).

## NOE Distance Calculate One

The **NOE Distance Calculate One** command is similar to the **NOE Distance Define** command above, except that only a single restraint is calculated and the restraint value is not saved to the database. Use the cursor to specify the cross peak used in the restraints calculation.

## NOE Distance Redefine

The **Measure/DISCOVER Restraints/NOE Distance Redefine** command serves as a refinement tool. After a set of peaks is assigned and a set of restraints is extracted based on that assignment, you can try to generate structures by using distance geometry (**Insight II/NMR_Refine/DGII**) or simulated annealing (**Insight II/NMR_Refine/MD_Schedule**).

After a successful run, several hot-spots are normally discovered; for example, certain assignments or restraints may not be right. If so, use the **NOE Distance Redefine** command to loosen, tighten, or delete some restraints showing the highest violations (or the highest number of violations within a family). To do this, you must first load the restraints on all the refined molecules into to the Insight II program, using the **NMR_Refine/Restraints/Read *molname**** command.

Then you may execute the **NMR_Refine** / **Distance/List** command. The provided *numvioltofelix* script redirects the output into another file.

At this point, you can use the **Measure/DISCOVER Restraints/NOE Distance Redefine** command on this file. You should specify the **Restraint entity** you want to work with (usually the *accelrys:noe_dist*) and the **Buildup Rate Calculation Method**.

FELIX next shows the violation table, from which you can zoom in on each peak for which the defined restraint was violated. From the table you can see the calculated distance, the restrained values, and the violation statistics. You then can select an **Action**: **Leave as is**, **Redefine bounds**, **Delete Restraints**.

## NOE volume Define

From an assigned peak entity and the corresponding volume entity, FELIX can calculate NOE volume restraints, which contain volume lower bounds and upper bounds as restraining entities. These can then be used in Discover.

Use the **NOE Volume Define** command to calculate and stores NOE-volume restraints for all the assigned peaks.

## NOE Distance Overlap Define

In certain instances some peaks can have multiple possible assignments. Those assignments (made in **Assign**) can be used in Discover to help in the refinement. These restraints are called NOE distance overlapped restraints.

You must have already defined singly assigned peaks as scalar peaks and also have defined the volumes to be measured before you can generate overlapped restraints. You can use the **NOE Distance Overlap Define** option to define such restraints from 2D NOE spectra or from heteronuclear edited 3D or 4D NOE spectra. Each NOE distance overlap restraint contains a set of possible atom name pairs (multiple possible assignments), as well as an effective distance upper and lower bound. You can export these restraints to the Insight II program and use them as ambiguous restraints in a Discover simulated annealing or rMD, rEM run.

## NOE Volume Overlap Define

The peaks with multiple possible assignments can be used in Discover directly—that is, without turning them into effective distances—to help in the refinement. These restraints are called NOE volume overlapped restraints.

You must have already measured the volumes (buildups) before you can generate overlapped restraints. You can use the **NOE Volume Overlap Define** option to define such restraints from 2D NOE

spectra or from heteronuclear edited 3D or 4D NOE spectra. Each NOE volume overlap restraint contains a set of possible atom name pairs (multiple possible assignments), as well as restraining volume(s). You can export these restraints to the Insight II program and use them as ambiguous restraints in a Discover simulated annealing or rMD, rEM run.

### 3J Dihedral

After measuring $^{3}$J couplings using the above described menu items and then assigning peaks, use the **3J Dihedral** option to create $^{3}$J-dihedral restraints.

## Measure/X-PLOR Restraints

### NOE

Several restraint classes are supported in FELIX. The basic 2D NOESY peaks can be used in structure generation and refinement as NOE-distance restraints. Based on an assigned peak entity and measured volumes (optionally buildups), FELIX can create new restraints, interactively show one restraint, and use a list of violations to recalculate restraints.

*Define*

Use the **Define** option to create a new entity (**msi:noe_dist**) of distance restraints from volume buildup rates.

> Defining restraints overwrites any existing restraints if the **Action** parameter is set to **New**.

If **Action** in the control panel is set to **Append**, FELIX appends the restraints produced to the restraint entity. Next, input the entity name containing the **scalar** information used in calculating the restraint. Then you select the method used to derive the buildup rates. You can choose between using a **Single mixing time** or a “best fit” rate based on the **First N mixing times**, as calculated by a first-order polynomial (**Straight Line**), a **Second Order Polynomial**, or an **Empirical Fit**. In an **Empirical Fit** calculation, FELIX uses a scalar entity containing a series of cross peak intensities that correspond to a range of inter-atomic distances to determine an empirical relationship between NOE intensity and distance.

Although the NOE volumes, in reality, increase exponentially and are damped by the exponential T2 relaxation, there are simply not enough mixing times to yield a robust fit to such an exponential-type function. Since only the initial buildup rate is needed (not the entire function), FELIX provides fit functions that can robustly calculate the initial rate from a small set of volume observations.

You can define, for the case of a symmetric spectrum, which peaks are to be considered in the restraint generation: all peaks or peaks on a specific side of the diagonal (for up to three user-defined regions) or only the lower-intensity peaks (symmetry selection).

All the above methods (except **Empirical Fit**) use the inverse $6^{th}$-power relation between buildup rate and interatomic distance to calculate a single distance in angstroms from the volume buildup rate for each cross peak and the scaling constant derived from the scalar peaks. That one resultant distance is then used to create a restraint having lower and upper distance bounds, based on the chosen method.

The **Exact Distance** method creates an entity where each restraint is an exact distance. A control panel asks for the minimum and maximum distances allowed for any restraint. For all restraints, the lower- and upper-bound distances are set to one calculated distance value.

The **Strong-Medium-Weak Bins** method creates an entity where each restraint is grouped into one of three distance bound bins. A control panel prompts you for the minimum and maximum distances allowed, the distance boundary between the strong and medium bins, and the distance boundary between the medium and weak bins. For all restraints, the lower and upper bounds are one of these three explicit distance ranges, depending on which bin the calculated distance falls in.

The **Van Der Waals-Exact** method creates an entity where each restraint uses a generic van der Waals hard-sphere radius for the lower bound and sets the upper bound to be the calculated distance. A control asks prompts for the van der Waals lower bound and the maximum upper bound.

The **Percentage of Distance** method creates an entity where each restraint uses a percentage of the calculated distance as the lower and upper bounds. A control panel prompts you for the minimum and maximum distances, a lower-bound percentage, and an upper-

bound percentage. For each restraint, the lower bound is obtained by subtracting the lower-bound percentage of the calculated distance from the calculated distance, and the upper bound is obtained by adding the upper-bound percentage of the calculated distance to the calculated distance. In this manner, short distances are translated into narrower bounds, and longer distances have wider bounds.

You may specify that only non-overlapped peaks be used in creating restraints. You can choose what percentage of the peak box area overlap is to be handled differently by setting the **Area Threshold** parameter. You can then discard those peaks (set **Partial Overlap** to **Discard**) or use a different method to turn them into restraints (for example, **Use as Qual**).

## Ambiguous NOE

In certain instances some peaks can have multiple possible assignments. Those assignments (made in **Assign**) can be used in XPlor to help in the refinement. These restraints are called ambiguous NOE restraints.

You must have already defined singly assigned peaks as scalar peaks and also have defined the volumes to be measured before you can generate overlapped restraints. You can use the **Ambiguous NOE** and **Define** options to define such restraints from 2D NOE spectra or from heteronuclear edited 3D or 4D NOE spectra. Each ambiguous NOE restraint contains a set of possible atom name pairs (multiple possible assignments), as well as effective distance upper and lower bounds. You can export these restraints to the Insight II program and use them as ambiguous restraints in a XPlor simulated annealing or rMD, rEM run.

## Dihedral

After measuring $^{3}J$ couplings using the above described menu items and assigning peaks, you can create dihedral restraints using the **Dihedral** option.

## NOE-Intensity

Using an assigned peak entity and the corresponding volume entity, FELIX can calculate NOE intensity restraints, which contain volume

lower and upper bounds as restraining entities in the **NOE-Intensity** and **Define** options. These can then be used in XPlor.

### Ambiguous NOE-Intensity

Peaks with multiple possible assignments can be used in XPlor directly (that is, without turning them into effective distances) to help in the refinement. These restraints are called ambiguous NOE-intensity restraints.

You must have already measured the volumes (buildups) before you can generate overlapped restraints. You can use the **Ambiguous NOE-Intensity** and **Define** options to define such restraints from 2D NOE spectra or from heteronuclear edited 3D or 4D NOE spectra. Each ambiguous NOE intensity restraint contains a set of possible atom name pairs (multiple possible assignments), as well as restraining volume(s) bound. You can export these restraints to the Insight II program and use them as ambiguous restraints in an XPlor simulated annealing or rMD, rEM run.

### NOE–NOE

If you have an assigned 3D NOE–NOE spectrum where you measured volumes, you can turn them into 3D NOE–NOE restraints using the **NOE-NOE** and **Define** options.

## Window pulldown

The **Window** pulldown (<Alt>+W) contains commands that rearrange or reset the frames for spectral display.

### Window/Cascade

The **Window/Cascade** command (<Alt>+wc) allows you to rearrange the windows by cascading them.

## Window/Tile Horizontally

The **Window/Tile Horizontally** command (<Alt>+wh) allows you to rearrange the windows as horizontal tiles.

## Window/Tile Vertically

The **Window/Tile Vertically** command (<Alt>+wv) allows you to rearrange the windows as vertical tiles.

## Window/Add New Window

The **Window/Add New Window** command (<Alt>+ww) allows you to add a new spectral frame.

> **Note:** If there is an existing spectral frame, the newly opened frame inherits all the reserved and user-defined symbols from it.

## Window/New Layout

The **Window/New Layout** command (<Alt>+wn) allows you to remove all the current frames, and create a new set of frames.

> **Note:** The New Layout options delete the contents of existing spectral frames.

**Table 17 Window/New Layout Control Panel Menu Items**

| Control | Description |
|---|---|
| **1 Frame** | Open a single graphics frame. |
| **2 Frames Left/Right** | Open two graphics frames, positioned side by side. |
| **1 square + 2 Left/Right** | Open three frames. The first frame takes up the left half of the FELIX window and two vertical frames are to the right of this frame. |
| **2 Frames Up/Down** | Open two frames, arranged vertically. |
| **3 Frames Up/Down** | Open three frames, arranged vertically. |
| **4 Frames Up/Down** | Open four frames, arranged vertically |

**Table 17 Window/New Layout Control Panel Menu Items (Continued)**

| Control | Description |
|---|---|
| **1 square + 2 Up/Down** | Open three frames. The first frame takes up the left half of the FELIX window and two windows, vertically arranged, are to the right of this frame. |
| **1 square + 3 Up/Down** | Open four frames. The first frame takes up the left half of the FELIX window and three windows, vertically arranged, are to the right of this frame. |
| **4 Small Square Frames** | Open four square graphics frames and arranges them to cover the viewable canvas. The frames are square if the canvas is square. |
| **4 Large Square Frames** | Open four large square graphics frames (size of the viewable canvas) and arrange them in two rows and two columns. The frames are square if the canvas is square. |

# Help pulldown

The **Help** pulldown (<Alt>+h) contains commands that provide information on using FELIX 2002.

## Help/About

The **Help/About** command (<Alt>+ha) gives you information about the current version of FELIX.

## Help/Topic

The **Help/Topic** command (<Alt>+ht) opens a Netscape browser window and allows you to access the online documentation.

## Help/Keypad

The **Help/Keypad** command (<Alt>+hk) gives you information about the currently available navigation commands and their keypad shortcuts.

# 5 Assign User Interface

## Project menu items

The first step in NMR spectrum assignment is to set up the database. The Assign module organizes an NMR research project in the FELIX database management system into an entity. To start an assignment project, therefore, you must first define this entity.

> **Note:** None of the **Assign** menu items work until a project is opened.

## Assign/Project

Using the set of **Project** menu items you can build a new project, open an existing one, show it, delete it, add new experiments, read the library, or redefine the molecule whenever necessary (for example, when assignment and refinement is done in an iterative fashion, new assignments are made after a refinement step, so the redefinition of molecule coordinates is important).

The project entity is built using the **Assign/Project** (<Alt>-ap) menu item and contains information about the molecule or complex under investigation and the NMR spectra (preferences, file names, etc.). The project entity contains several other entities, such as patterns and frequencies, which can be defined in the building stage.

For Assign to function properly, you should specify the name of the molecule or complexes (MSI .car or XPlor .pdb format). This file is read in and used throughout the whole module.

If the project already exists (that is, if it was built in a previous session) this menu item will open it. If FELIX fails to find the name of

the project on its own, you need to enter the name in the control panel.

> **Note:** Even though nothing prevents you from defining multiple projects in a database, it is a strongly recommended that you open a separate database for each project.

Opening the project opens the Experiments table, which enables you to quickly switch between spectra (see the "The Experiments table–Assign/Experiment" section below).

If you have already opened the project, using this menu item again shows the entity in a spreadsheet table.

## The Experiments table–Assign/Experiment

Once the project is built, you can specify the spectra that were measured on the molecular system under investigation. You can also redefine the plotting or other attributes of the experiment, and you can delete an experiment from the database. The first time you use this menu item, the **Experiments** table is created and displayed. From that point on, you must use the table to add, delete, change, or switch between experiments.

### Experiment/Select

Accessing the **Experiment/Select** menu item (<Alt>-xs) from the **Experiments** table quickly retrieves spectra from a project entity, so that you can display contour or density plots from related spectra with the same limits. When you switch from one spectrum to another, all parameters are automatically reloaded. This makes it unnecessary to manually switch entities (that is, the peak entity is reloaded after changing spectra).

### Experiment/Add

To add an experiment to the project, select the **Experiments/Add** (<Alt>-xa) menu item from the **Experiments** table. During the experiment setup, you must specify your preferred plotting parameters, which are saved in the project entity.

To insure proper functioning, the spectra included must be referenced in ppm. You can also specify various parameters of the spectrum, such as temperature, pH, solvent, the measured nuclei, the order of measurement, the folding, the type of experiment (J or NOE), the number of J steps for J-type experiments, mixing time for NOE, and several other items, such as the peak and volume entities.

You can define up to 12 spectra in one project.

### Experiment/Change Attributes

With the **Experiment/Change Attributes** (<Alt>-xc) menu item from the **Experiments** table, you can redefine plotting or other parameters of an experiment at any time.

> **Note:** If you redefine the plotting parameters outside this menu item, the effect of that redefinition remains as long as you do not chose a different experiment or exit the program.

### Experiment/Delete

You can delete an experiment from the project by using the **Experiment/Delete** (<Alt>-xd) menu item from the **Experiments** table. This menu item is useful if you work with more than the currently available 12 spectra in your project. (For example, after you are done with spin-system assignment using a couple of spectra, you can delete them and add new ones for which you want to make peak assignments–typically these are NOE-type spectra.)

## Assign/Define Library

FELIX also provides you with a library, pd.rdb, which was compiled using data from the literature. You can (and really *should*) read this library using the **Assign/Define Library** (<Alt>-al) menu item. This library is an ASCII file and contains information concerning the frequent residue types that occur in biopolymers. Currently it contains information about standard amino acid residues and deoxynucleotides. If you want to work with ribonucleic acids you should read in the rna.pdb file. You can edit these library files, add new residue types, or create a whole new library, but be sure to maintain the library syntax. Also, it is important that you maintain the bio-

sym.alias file if you add new residue types.Please see Chapter 2, *Task: Adding modified residues to the Assign database* for more information.

The following is an example item from the library:

```
! ASN
!
! H - N    H2   Od
!     |     |    |
! H - Ca - Cb - Cg - Nd - H2
!     |
! O = C
!
RESIDU ASN N
RESATM H HN 1 8.16 0.62
RESATM H HA 2 4.74 0.30
RESATM H HB1 3 2.74 0.32
RESATM H HB2 3 2.88 0.27
RESATM H HD21 7 7.75 0.32
RESATM H HD22 7 6.99 0.55
RESATM N N 8 123.0 5.0
RESATM C CA 10 53.0 2.0
RESATM C C 9 150.0 50.0
RESATM O O 15 0.0 999.0
RESATM C CB 11 38.0 3.0
RESATM C CG 12 100.0 999.0
RESATM N ND2 16 125.0 50.0
RESATM O OD1 17 100.0 999.0
!      NH AH B1 B2 D1 D2 N A C O Cb Cg Nd Od
CONECT 0
CONECT 3 0
CONECT 4 3 0
CONECT 4 3 2 0
CONECT -6 -5 -4 -4 0
CONECT -6 -5 -4 -4 2 0
CONECT 1 2 3 3 5 5 0
CONECT 2 1 2 2 4 4 1 0
CONECT 3 2 3 3 5 5 2 1 0
CONECT 4 3 4 4 6 6 3 2 1 0
CONECT 3 2 1 1 3 3 2 1 2 3 0
CONECT 4 3 2 2 2 2 3 2 3 4 1 0
CONECT 5 4 3 3 1 1 4 3 4 5 2 1 0
CONECT 5 4 3 3 1 1 4 3 4 -5 2 1 2 0
ENDRES
```

## Assign/Read Coordinates

Although it is necessary to read in a molecule at the stage of building the project, you still may want to reread the molecule file. The most obvious reason would be that the starting model was just a linear chain, and at one point you may have a refined model, so you must reread the molecule into FELIX. To do this, you can use the **Assign/Read Coordinates** (<Alt>-ao) menu item, since the number of atoms was not changed.

# Manual spin-system picking–the Assign/Frequency Clipboard menu items

The **Assign/Frequency Clipboard** pullright (<Alt>-af) of menu items deals with the frequency clipboard. A clipboard is a data structure that allows you to store and manipulate an array of chemical shifts and associated nuclei. This clipboard is mainly used for picking spin systems manually (which then can be promoted to patterns) or for manipulating a collection of frequencies copied from patterns or prototype patterns. You can also display it as lines through 2D plots, and tiling, or strip plots can be spawned from it.

## Assign/Frequency Clipboard/Zero Clipboard

Before you start picking a spin system manually, you must first make sure that there are no frequencies in the clipboard, using the **Assign/Frequency Clipboard/Zero Clipboard** menu item (<Alt>-afz). This allows you to delete all frequencies from the list.

> **Note:** This action occurs immediately, without asking for confirmation.

## Assign/Frequency Clipboard/Add One

The **Assign/Frequency Clipboard/Add One** menu item (<Alt>-afa) allows you to add a frequency to the frequency clipboard by picking a chemical shift from a plot of a spectrum.

## Assign/Frequency Clipboard/Delete One

The **Assign/Frequency Clipboard/Delete One** menu item (<Alt>-afd) allows you to delete frequencies from the frequency clipboard using the control panel.

## Assign/Frequency Clipboard/Swap Two

The **Assign/Frequency Clipboard/Swap Two** menu item (<Alt-aft) allows you to exchange the order of two frequencies in the frequency clipboard.

## Assign/Frequency Clipboard/Remove Duplicates

The **Assign/Frequency Clipboard/Remove Duplicates** menu item (<Alt>-afm) allows you to purge frequencies from the frequency clipboard that are considered duplicate entries, based on a chemical shift tolerance that you define.

## Assign/Frequency Clipboard/Compare Frequencies

The **Assign/Frequency Clipboard/Compare Frequencies** menu item (<Alt>-afc) allows you to compare the current collection of frequencies (frequency clipboard) with each of the patterns or prototype patterns. This allows you to see how many fuzzy similarities there are with each of them. Therefore you must specify the target to compare against (patterns or prototype patterns). If patterns are used, a comparison should be based on general shifts or on spectrum-specific shifts. If the fuzzy similarity exceeds the **Min Similarity** parameter, you are notified. For comparison, the **Tolerance** should also be used. This menu item is useful when try-

ing to find out if the frequencies in the frequency clipboard constitute a novel pattern.

## Assign/Frequency Clipboard/Copy Clipboard To Pattern

Once a frequency clipboard is compiled and sorted, you can copy it to a pattern using the **Assign/Frequency Clipboard/Copy Clipboard To Pattern** menu item (<Alt>-afl).

You can choose to append to frequencies at the end of the pattern, overwrite (when each frequency in the pattern is overwritten by a frequency from the list), or substitute (when each frequency in the pattern is replaced by the one from the list that lies closest to it). Also, you can select **new**, which creates new pattern at the same time.

## Assign/Frequency Clipboard/Copy Clipboard To Proto

With this menu item (<Alt>-afo) you can copy a clipboard to a new prototype pattern.

## Assign/Frequency Clipboard/Copy Pattern To Clipboard

The **Assign/Frequency Clipboard/Copy Pattern To Clipboard** menu item (<Alt>-afn) performs the opposite action of the **Copy Clipboard To Pattern:** you can copy frequencies from a pattern to the frequency clipboard and manipulate them through the frequency clipboard's menu items.

## Assign/Frequency Clipboard/Copy Proto To Clipboard

The **Assign/Frequency Clipboard/Copy Proto To Clipboard** menu item (<Alt>-afp) allows you to choose a prototype pattern as a source of frequencies for a frequency clipboard.

### Assign/Frequency Clipboard/View Clipboard

With this menu item (<Alt>-afv) you can list the actual contents of the frequency clipboard in the output window.

### Assign/Frequency Clipboard/Sort Clipboard

The **Assign/Frequency Clipboard/Sort Clipboard** menu item (<Alt>-afs) allows you to sort the frequencies in the clipboard in ascending or descending order.

### Assign/Frequency Clipboard/Tile Clipboard

The **Assign/Frequency Clipboard/Tile Clipboard** menu item (<Alt>-afi) allows you to spawn a tile plot from the frequencies in the clipboard.

### Assign/Frequency Clipboard/Strip Plot Clipboard

The **Assign/Frequency Clipboard/Strip Plot Clipboard** menu item (<Alt>-afr) allows you to spawn a strip plot from the frequencies in the clipboard.

### Assign/Frequency Clipboard/Draw Clipboard

The **Assign/Frequency Clipboard/Draw Clipboard** menu item (<Alt>-afw) allows you to draw straight lines on the plot along the frequencies in the clipboard.

## Prototype Pattern menu items

The menu items in the third subsection of the **Assign** pulldown contains menu items relating to the prototype pattern entity. The prototype patterns are the generally rough spin systems that are the results of automated spin-system detection algorithms.

## Assign/Collect Prototype Patterns

The **Assign/Collect Prototype Patterns** (<Alt>-ac) menu item contains options used in automated spin-system detection. In Assign, several choices are available: those based on systematic search of 2D spectra (i.e., COSY-type only, TOCSY-type only, TOCSY and NOESY type spectra, or the COSY, TOCSY, and NOESY spectra-based method); those using optimization to find spin systems (in TOCSY or TOCSY and COSY spectra); and those utilizing systematic searching in 3D homonuclear spectra, 3D double or triple resonance spectra, or 4D triple resonance spectra.

Note that the TOCSY type can refer to any homonuclear 2D experiment in which magnetization is transferred through J coupling and the correlation extends over more than three bonds (e.g., RELAY or DOUBLE_RELAY experiments).

The double-resonance menu items use the information of a $^{15}$N-separated TOCSY spectrum ($^{15}$N HSQC-TOCSY, $^{15}$N HMQC-TOCSY, or $^{15}$N TOCSY-HSQC), with or without information from a 2D $^{15}$N-HSQC spectrum.

The methods available for spin-system detection in 3D or 4D heteronuclear triple-resonance spectra are contained in **Triple Resonance** options. Currently you can use the following types of spectra:

- HNCA, HN(CO)CA.
- HNCO, HNCA, HN(CO)CA.
- CBCANH (or HNCACB), CBCA(CO)NH.
- 2D HSQC, CBCANH, CBCA(CO)NH.
- HNCA, CBCANH, CBCA(CO)NH.
- HNCO, CBCANH, CBCACO(N)H.
- HNHA, CBCANH (or HNCACB), CBCA(CO)NH.
- HNCO, HC(CH)-TOCSY-NH, HC(CH)-TOCSY-(CO)NH.
- 4D HACANH, 4D HACA(CO)NH.

The algorithms used by the options in the **Assign/Collect Prototype Patterns** menu item are described in Chapter 1, *Assign building*

*blocks*. What follows is a short description of the controls within each of these menu items.

## Assign/Collect Prototype Patterns options

*2D Homonuclear*

You can choose between two different automatic seed / expansion areas, depending on the molecule–one that is suitable for peptides / proteins and one that is suitable for nucleic acids. Here we describe these two different options together:

- **Systematic Search** –Within the **Systematic Search** menu item, you must first choose the method you want to use (COSY only, TOCSY only, TOCSY+NOE, or COSY+TOCSY+NOE), then you must choose the experiments to use, although Assign does attempt to find which experiments correspond to a particular method.

  For the **Seed/Expansion Area**, you can select **Use Defaults** to use the default values (which you can review later by clicking the **More** button), or you can enter them in the next control panel by selecting the **Review** option. Also, there are two interactive options (which require a seed spectrum to be currently plotted)– you can start spin-system detection from a peak (**Select via Peak**) or by dragging an area with the cursor (**Drag with Cursor**). If you choose to enter the seed area you should enter it in ppm. This area is used to collect the seed peaks. Assign attempts to disentangle the spin systems from these peaks. Assign uses the **Frequency Collapse Tolerance** value (in ppm) to decide if a candidate frequency already belongs to the prototype pattern. This tolerance should reflect the chemical shift variations expected among different spectra.

  If you choose at this point to start the control panel's action (by clicking **OK**), the spin-system detection uses default values for the remaining settings. If you want to review and possibly change the defaults, click the **More** button. The new control panel contains these controls:

  a. The **Remove Intraproto Frequencies** control indicates whether peaks that belong to already existing prototype patterns from an earlier run should be removed from consideration. This can help to avoid duplicate entries if several prototype pattern-detection runs are applied and the protos entity is not purged between runs.

b. The **Number of Frequencies in Proto Min** and **Max** controls set the minimum and maximum numbers of frequencies a prototype pattern can contain. For proteins, a good starting point are the values 3 and 8, respectively. This option can also be used to selectively detect spin systems (e.g., glycines or AMX spin systems). For this, **Min** should be set to the number of expected frequencies *minus* one (to account for coinciding frequencies); and **Max** should be set to the number of expected frequencies *plus* one.

c. The **Number of Iterations** control governs the number of extension loops that can be run for each seed peak.

d. You can use the **Frequencies per Iteration** control to specify how many candidate frequencies to add to the prototype pattern in each extension loop (usually one or two).

The next set of controls to set are the tuning parameters: the number of contacts required for a candidate frequency to be eligible or membership in a developing prototype pattern. For a prototype pattern containing 2, 3, 4, or more frequencies, you should enter at least how many COSY, COSY, and TOCSY, and COSY and TOCSY and NOESY contacts should have a new frequency to be considered as a candidate. All the values should be $\geq 0$ and any number entered for $N$ frequencies should be $\geq$ the required number for $N$-1 frequencies.

The last controls to set are the ppm filter definitions. You may impose filtering conditions to select out any unwanted prototype pattern, based on the number of frequencies in certain chemical-shift ranges. For proteins you might specify, for example, that between 6 and 12 ppm there should be only one frequency, and between 3 and 5.5 ppm there should be at least one and at most three frequencies.

- **Simulated Annealing** –When using the **Simulated Annealing** menu item, you should first select which method you want to use for spin-system detection: TOCSY only; TOCSY and COSY; or TOCSY, COSY, and $^{13}C$-$^{1}H$ HSQC. Depending on the method, you should set the experimental values accordingly. You can detect spin systems selectively (by setting the Boolean variable to **Specific** and selecting a residue type from the list) or detect *all* spin systems (by setting the Boolean variable to **All Residue**). When detecting spin systems selectively, you can also define the expected number of spin systems—the **Occurrence**.

You can remove peaks that are already part of earlier detected spin systems from consideration by setting the **Remove first** control to **True**. You can specify the number of standard deviations to be used in chemical-shift comparison by setting the **Max Std Dev** control. **Output Level** can be set to **Quiet**, **Low**, **Medium**, or **High**. Finally, you must define the inter-spectrum chemical-shift tolerance between TOCSY and COSY spectra and between TOCSY and HSQC spectra.

*3D Homonuclear*

With the **Assign/Collect Prototype Patterns/3D Homonuclear** menu item, you should first set the experiment you want to use, then, similar to the method used in the **2D Systematic Search** menu item, you must specify the seed area and the expansion area in ppm (e.g., **Seed area W3 Low** and **High**). Then you should define which two frequencies from each seed peak to use as the two first frequencies of a new prototype pattern (**Use seed peak W3 True/False**, **W2 True/False**, and **W1 True/False**). You must also define what dimension of a candidate peak is to be used as a new frequency in the spin system (**Use exp. peak W3 True/False**, **W2 True/False**, and **W1 True/False**). The remaining controls are set like those discussed under the **2D Systematic Search** menu item.

*Double Resonance*

Under this option you can choose from three different methods:

- **3D HS(M)QC-TOCSY**–The **3D HS(M)QC-TOCSY** option is a spin-system search method that was developed on a $^{15}$N separated TOCSY spectrum. The function collects the spin systems, starting on an N-HN-HN peak and then collecting all the frequencies along a line defined by the $^{15}$N and HN frequency. Therefore, after defining the proper experiment you must specify which dimension is the **Root** (i.e., what dimension contains the HN frequencies), then the seed area (usually for $^{1}$H dimensions it is the amide region, and for $^{15}$N the appropriate region). The expansion area is usually the amide-(alpha, side chain)-$^{15}$N region. Then you must define which dimension's frequencies are to be used from particular peaks (seed or expansion) as frequencies in the prototype pattern. The remaining controls are similar to those discussed above under the **Protos/Automated Detect/ 2D Systematic Search** menu item.

- **2D HS(M)QC + 3D HS(M)QC-TOCSY**–With the **2D HS(M)QC + 3D HS(M)QC-TOCSY** menu item, FELIX searches for spin systems, starting with two resonances for a $^{15}$N HSQC spectrum (seed spectrum). You must define which dimension (usually W2)

the **Root** (i.e., the $H_N$) is in, as well as the seed areas. FELIX then searches the HSQC-TOCSY spectrum for connected peaks and stores the resulting frequencies as spin systems or prototype patterns.

- **3D HC(C)H-TOCSY**–FELIX searches the $^{13}C$-separated TOCSY spectrum for spin systems starting from the **Root** dimension ($^{13}C$-$^{1}H$ dimension).

*Triple Resonance*

The options for **Triple Resonance** are:

- **HNCA + HN(CO)CA**–The **HNCA + HN(CO)CA** menu item is based on two 3D triple-resonance spectra: the HNCA and the HN(CO)CA. The resulting prototype patterns contain the $H_{N,i}$-$N_i$-$C_{\alpha i}$-$C_{\alpha i-1}$ frequency quartets. You should set the **HNCA** and **HN(CO)CA experiments** from the spectra defined in the project if FELIX fails to recognize them correctly–in which case you would just acknowledge the **Use Default Experiments**, then for each experiment you should set the dimensions (**Carbon**, **Nitrogen**, and **Proton**). Finally, you can set the interspectrum **Tolerances** and the **Number of iterations** (how many times the program tries to expand, starting from all HN(CO)CA peaks, to collect the $C_\alpha i$ and $C_\alpha i_{-1}$ frequencies). At each iteration the interspectrum tolerances are increased by the **Tolerance factor** you supply. You may also set the **Output level**.

- **HNCO + HNCA + HN(CO)CA** –The **HNCO + HNCA + HN(CO)CA** menu item is based on three 3D triple-resonance spectra: the HNCO, the HNCA, and the HN(CO)CA. The resulting prototype patterns contain the $H_{N\alpha i}$-$N_i$-$C_{\alpha,i}$-$C_{\alpha,i-1}$ frequency quintets. You should set the **HNCO**, **HNCA**, and **HN(CO)CA experiments** from the spectra defined in the project, then for each experiment you should set the dimensions (**Carbon**, **Nitrogen**, and **Proton**). Finally, you can set the interspectrum **Tolerances** and the **Number of iterations**. At each iteration the interspectrum tolerances are increased by the **Tolerance factor** you supply. You may also set the **Output level**.

- **CBCANH + CBCA(CO)NH** –The **CBCANH + CBCA(CO)NH** menu item is based on two 3D triple-resonance spectra: an HNNCACB or CBCANHN spectrum and a CBCA(CO)NHN spectrum. The resulting prototype patterns contain the $H_{N,i}$-$N_i$-$C_{\alpha,i}$-$C_{\beta,i}$-$C_{\alpha,i-1}$-$C_{\beta,i-1}$ frequency sextets. You should set the **HNCACB** and **CBCA(CO)NHN experiments** using the spec-

tra defined in the project, then you should set the dimensions for each experiment (**Carbon**, **Nitrogen**, and **Proton**). For both experiments you should set the intensity pattern expected (usually positive in CBCA(CO)NHN for both $C_\alpha$ and $C_\beta$; positive and negative in HNCACB for $C_\alpha$ and $C_\beta$, respectively). Finally, you can set the interspectrum **Tolerances** and the **Number of iterations**. At each iteration the interspectrum tolerances are increased by the **Tolerance factor** you supply. You may also set the **Output level**.

- **2D HSQC + CBCANH + CBCA(CO)NH** –This option is similar to the previous one, in that you can detect spin systems in a pair of triple-resonance experiments (CBCANH and CBCA(CO)NH) with the help of a 2D HSQC spectrum, which is used as a starting point for each spin system. The results of the run are similar protopatterns ($H_{N,i}$-$N_i$-$C_{\alpha,i}$-$C_{\beta,i}$-$C_{\alpha,i-1}$-$C_{\beta,i-1}$) and the variables to be set are similar.
- **HNCA + CBCANH + CBCA(CO)NH** –This option is similar to the previous one. The difference is that you can use as a starting point the HNCA spectrum instead of the 2D HSQC spectrum. The results of the run are similar protopatterns ($H_{N,i}$-$N_i$-$C_{\alpha,i}$-$C_{\beta,i}$-$C_{\alpha,i-1}$-$C_{\beta,i-1}$) and the variables to be set are similar.
- **HNCO + CBCANH + CBCACO(N)H** –If you measure a CBCACO(N)H spectrum instead of the CBCA(CONH spectrum which is sometimes advantageous because it helps to resolve resonance overlap problems), then the **HNCO + CBCANH + CBCACO(N)H** menu item can be used to detect spin systems in the HNCO, CBCANH, and CBCACO(N)H experiments triplet.The resulting spin systems contain seven frequencies: $H_{N,i}$-$N_i$-$C_{i-1}$-$C_{\alpha,i}$-$C_{\beta,i}$-$C_{\alpha,\ i-1}$-$C_{\beta,\ i-1}$.
- **HNHA + CBCANH + CBCA(CO)NH** –You can selectively find glycine spin systems using this menu item, where the HNHA(Gly) is utilized in excess to CBCANH and CBCA(CO)NH, as described by Wittekind et al. (1993).
- **HNCO+ H(CC)-TOCSY-NH + H(CC)-TOCSY-CO(N)H** –You can find spin systems using this menu item, where the detection starts from each HNCO peak and the extension is done along the TOCSY lines. The resulting spin system contains a variable number of frequencies, depending on the efficiency of the TOCSY

transfer and the spin-system type: $H_{N,\,i}$-$N_i$-$C_{i-1}$-$H_{\alpha,\,i}$-$(H_{\xi,}i)_n$-$H_{\alpha,\,i-1}$-$(H_{\xi,}\,i_{-1})_m$.

- **4D HACANH + HACA(CO)NH** –The **4D HACANH + HACA(CO)NH** menu item is based on two 4D triple-resonance spectra: an HNNCAHA or HACANHN spectrum and an HACA(CO)NHN spectrum. The resulting prototype patterns contain the $H_{N,i}$-$N_i$-$C_{\alpha,i}$-$H_{\alpha,i}$-$C_{\alpha,i-1}$-$H_{\alpha,i-1}$ frequency sextets. You should set the **Intraresidue experiment** and **Interresidue experiment** controls using the spectra defined in the project, then for each experiment you should set the dimensions (**Proton, Carbon**, **Nitrogen**, and **Amide**). Finally, you can set the **Interspectral tolerances** and the **Number of iterations**. Also, at each iteration the interspectrum tolerances are increased by the **Tolerance factor** you supply. As in the above menu items, you may specify the **Output level**.

### *User Settable*

If you would like to use a different combination of spectra, you can use the **Assign/Collect Prototype Patterns/User Settable** menu item. You have to define a primary spectrum (e.g., an HSQC or an HNCO) and secondary spectra (e.g., an HSQC-TOCSY or a pair of HNCACB and CBCA(CO)NH). You also have to specify in which direction the search should proceed and what the expected atom types found along each dimension are.

### *Semiautomated Spin System Detection*

Besides the built-in methods for spin-system collection, you can use virtually any combination of spectra. The method is a semiautomatic detection–that is, you need to click a position (or peak) in a primary spectrum (such as HSQC or HNCO) and then, according to the setup procedure, a spin system gets collected and stored, if there are correlated peaks in the spectra you defined. To achieve this, you must first set up your experiments and define how the connection between the peaks is expected to be, using the **Assign/Collect Prototype Patterns/Semiautomated Setup** menu item. Then you can use the **Assign/Collect Prototype Patterns/Semiautomated Collect** menu item (<Ctrl-=) to collect a new spin system. You can step through your spectra and do all the spin-system semi-automated detection until you use the **Assign/Collect Prototype Patterns/Semiautomated Unset** menu item.

- **Semiautomated Setup**–With this menu item you can set up semiautomated protopattern detection. You need to define a primary spectrum, for example, an HSQC or an HNCO spectrum, and one or more secondary spectra. This menu item connects the

frames with the spectra you defined. So if you zoom in on an active frame, the other connected frames change their limits according to the definition.

- **Semiautomated Collect** –With this menu item (<Ctrl-=) you can detect protopatterns by clicking a peak in the primary (seed) spectrum after a setup (**Assign/Collect Prototype Patterns/ Semiautomated Setup**) is finished. Activate this menu item by activating the frame where the primary spectrum is. With the resulting crosshair cursor, click the peak where the semiautomated collection is to be started.
- **Semiautomated Unset**–With this menu item you can finish semiautomated detection.

### *Extend Prototype Patterns*

The **Assign/Collect Prototype Patterns/Extend Prototype Patterns** menu items helps to collect new frequencies (that is, extend spin systems) into already existing prototype patterns. For example, if you detect the sextets ($H_{N,i}$-$N_i$-$C_{\alpha,i}$-$C_{\alpha,i}$-$C_{\alpha,i-1}$-$C_{\alpha,i-1}$) on a pair of experiments, then you can use an HAHB(CO)NH spectrum to extend the spin systems to include the $H_{\alpha,i-1}$ and $H_{\beta,i-1}$ frequencies. This is mainly useful when the backbone spins of a protein are collected via a triple-resonance method, and the sidechain resonances with a triple-resonance or double-resonance method.

- **Extend Prototype Pattern Along One Axis** –By using this menu item with the current 3D experiment, you can add new frequencies to existing ones in current spin systems.
- **Extend Prototype Pattern Using Two Spectra**–This menu item can add new frequencies to the existing ones in current spin systems using two spectra. For example, after first collecting spin systems in HNCO and HSQC-TOCSY with the **Assign/Collect Prototype Patterns/User Settable** menu item, you can use **Extend Prototype Pattern Using Two Spectra** on a combination of HNCACB and CBCA(CO)NH to add $C_{\alpha,i}$, $C_{\beta,i}$, $C_{\beta,i-1}$, and $C_{\beta,i-1}$ frequencies.
- **Extend Prototype Patterns Using HCCH-TOCSY**–By using this menu item with the current 3D HCCH-TOCSY experiment, you can collect sidechain spin systems on the current spin systems, containing either $H_\alpha$, $C_\beta$, or $C_\alpha$, $C_\beta$ frequencies.

## Assign/Edit Prototype Pattern

This menu item provides you with several options:

- **Graphical Edit Frequency** allows you to graphically edit the frequencies of a prototype pattern.
- **Edit Shifts** allows you to edit the prototype pattern, delete frequencies, assign spin types, and perform other manipulations.
- **Delete One Prototype Pattern** deletes *one* prototype pattern.
- **Delete All Prototype Patterns** deletes *all* prototype patterns.

## Assign/Promote Prototype Patterns

After prototype patterns are collected, you can use the **Assign/Promote Prototype Patterns** menu items to copy all of them to patterns or you can copy patterns back to prototype patterns.

### Copy Prototype Patterns to Spin Systems (Patterns)

With this menu item you can promote all spin systems (prototype patterns) detected with any of the automated collection methods to patterns. For homonuclear and double-resonance spin systems, FELIX will just copy over the frequencies and create new patterns. For triple-resonance experiments, FELIX will search through the list for potentially sequential spin systems using the tolerance (which you can set) and store that information in patterns as neighbors. Also, if you request it, this function can search for potential profiles and store them. Finally, FELIX can compare each potential pattern against all existing ones to discover if the new candidate pattern is unique. If it finds that the new pattern is *not* unique, it asks you to allow for deletion. This deletion option slows the speed of the function's execution, but allows you to avoid duplication of the patterns. In certain cases, proline spin systems can be detected as part of a protopattern showing up only as neighbors (i-1). This can occur if protopattern detection was done using CBCA(CO)NH and CBCANH spectra.

## Copy Spin Systems (Patterns) to Prototype Patterns

By using this menu item you can copy final spin systems (patterns) back to prototype patterns.

## Assign/Zoom Prototype Pattern

The **Assign/Zoom Prototype Pattern** (<Alt>-an) menu item is useful for 3D and 4D spectra when you want to visually inspect the spin systems resulting from automated detection. Using this menu item, you can define which prototype pattern's frequencies should define the region. Selecting a prototype pattern number from the **Source** list and then clicking the **Show** button fills in the **Frequencies** list with the frequencies of that prototype pattern. If you then select the desired **Orientation**, specifying which frequency should be at which dimension, then also specify the required ranges, you can bring up the region by clicking **OK**.

### Protopatterns table

The majority of the visualization menu items useful for analyzing prototype patterns are accessible through the prototype pattern table. The table is usually opened by using the automated collection menu items, but you can open it with the **Edit/Prototype Patterns** (<Alt>-ep) menu item. Before you can execute any menu items in the table, you need to select (click) a row or select multiple rows, depending on the action you want to perform. The menu items accessible through the table interface are briefly described below.

*Action/Zoom*

The **Action/Zoom** (<Alt>-az) menu item from the Protopatterns table is useful for 3D and 4D spectra when you want to visually inspect the spin systems resulting from automated detection. You can select a prototype pattern by clicking its number (highlighting the row) for which prototype pattern's frequencies should define the region. Then you need to either use the **Action/Zoom** menu item or click the **Zoom** icon in the Protopatterns table. To set the preferences for this menu item, you need to use the **Preferences/Zoom** menu item in the Protopatterns table.

*Action/Draw*

You can visualize a prototype pattern by drawing straight lines on the actual spectrum along the frequencies of a particular prototype pattern. First you need to select two protos from the table and then

either click the **Action/Draw** (<Alt>-ad) menu item or click the **Draw** icon in the Protopatterns table. Using the **View/Draw Frequencies** or the **Clear Frequencies** menu item in the right-mouse menu, you can then turn these lines off. To set the colors you can use the **Preference/Draw** menu item in the Protopatterns table.

*Action/Tile Plot*

The **Action/Tile Plot** (<Alt>-at) menu item is used to define a tile from any pair of prototype patterns. You first need to select two prototype patterns by clicking two rows in the table. Then you need to select the **Action/Tile Plot** menu item or use the **Tile Plot** icon in the Protopatterns table. The tile plot preferences can be set with the **Preference/Tile Plot** menu item in the Protopatterns table. To turn off the tile plot you can use the **View/Plottype/Tile/Strip Plot** menu item.

*Action/Strip Plot*

The **Action/Strip Plot** (<Alt>-as) menu item from the Protopatterns table is used to define a strip plot from a specific prototype pattern. First you need to select a prototype pattern by clicking the needed row in the table. Then select the **Action/Strip Plot** menu item or use the **Strip Plot** icon in the Protopatterns table. The strip plot preferences can be set in the **Preference/Strip Plot** menu item in the Protopatterns table. To turn off the strip plot you can use the **View/Plottype/Tile/Strip Plot** menu item.

*Action/ND Strip Plot*

This menu item (<Alt>-an) allows you to define a strip plot from a 3D / 4D spectrum, where the individual strips originate from potentially different planes of different spin systems. You first need to select the needed prototype patterns by clicking the rows in the table (after the first click use <Shift>-click). Then you need to click the **Action/ND Strip Plot** menu item or use the **ND Strip Plot** icon in the Protopatterns table. The strip plot preferences can be set in the **Preference/ND Strip Plot** menu item in the Protopatterns table. To turn off the tile plot you can use the **View/Plottype/Tile/Strip Plot** menu item.

*ProtoPattern/Add Frequency via Cursor*

This menu item (<Alt>-nf) in the Protopatterns table allows you to graphically add a frequency to the selected prototype pattern.

*ProtoPattern/Promote One Proto to Spin System*

The **ProtoPattern/Promote One Proto to Spin System** menu item (<Alt>-np) in the Protopatterns table copies over the frequencies from the selected prototype pattern to a new pattern (spin system).

> **Caution:** If the selected prototype pattern is a triple-resonance spin system with frequencies from the neighboring spin system, this information is lost during the promotion–that is, this information is not transferred to neighbor probability, which is different from the way the **Assign/Promote Prototype Patterns** menu item works.

*ProtoPattern/List Membership*

The **ProtoPattern/List Membership** (<Alt>-nl) menu item in the Protopatterns table is used to find the particular peak to which a prototype pattern belongs, if any. The result is listed in the output window.

```
  Peak # D1 D2
------ -- --
53 9.3042 121.3826
Proto D1 Distance Proto D2 Distance
----- -- -------- ----- -- --------
5 9.31 0.580025e-02 5 121.425 0.0424
```

# Pattern (Spin Systems) menu items

The next section in the **Assign** pulldown contains menu items dealing with patterns or full spin systems–that is, those which have neighbor probabilities, residue type probabilities, and assignments.

## Assign/Spin System

The **Assign/Spin System** menu (<Alt>-as) contains menu items for creating, deleting, and editing patterns.

### Creating and Deleting Spin Systems

*Assign/Spin System/Add One*

The **Assign/Spin System/Add One** menu item (<Alt>-aso) allows you to create a new empty pattern and add it to the database. You may define the name, color, comment, and root frequency, and the unique identification number is defined by the program.

*Assign/Spin System/Delete One*

The **Assign/Spin System/Delete One** menu item (<Alt>-asl) allows you to delete patterns from the database by clicking the unique number of the pattern to be deleted.

*Assign/Spin System/ Clear All*

The **Assign/Spin System/Clear All** menu item (<Alt>-asc) allows you to delete all patterns from the database.

> **Important:** Accidental use of this menu item can destroy several weeks worth of work.

## Editing Spin Systems

After a pattern is established, you may edit it. New frequencies can be added to or deleted from a pattern. Attributes or shifts can also be edited.

*Assign/Spin System/ Add Frequency via Cursor*

This menu item (<Alt>-asa) is similar to the one in the **Frequency Clipboard**, but it allows you to directly add frequencies to the patterns.

*Assign/Spin System/ Add Frequency Manual*

This menu item (<Alt>-asm) allows you to type in a new frequency which is then appended to a pattern.

*Assign/Spin System/ Delete Frequency*

The **Delete Frequency** menu item (<Alt>-asd) allows you to delete frequencies from the patterns.

*Assign/Spin System/ Edit Attributes*

The **Assign/Spin System/Edit Attributes** menu item (<Alt>-asi) allows you to change the name, color, comment, and root frequency of the pattern.

*Assign/Spin System/ Auto Root*

The **Assign/Spin System/Auto Root** menu item (<Alt>-asr) allows you to copy the generic shift of the first frequency of the pattern to its root frequency attribute. This frequency is then used in the homonuclear neighbor detection routines of **Assign/Neighbor** menu items.

*Assign/Spin System/ Copy Generic Shift To Specific*

The **Copy Generic Shift To Specific** menu item (<Alt>-asg) allows you to copy the generic shifts to spectrum-specific shifts for frequencies in all patterns or in a specific pattern. You must specify the experiment for which specific shifts are to be redefined. The proper spectrum-specific shifts of assigned frequencies in patterns are crucial to the success of automated peak assignment.

*Assign/Spin System/ Copy Specific Shift To Generic*

The **Copy Specific Shift To Generic** menu item (<Alt>-ass) allows you to copy the spectrum-specific shifts of frequencies of one or all patterns to generic shifts. You must specify which experiment's spectrum-specific shifts to copy.

*Assign/Spin System/ Auto Update Specific Shifts*

This menu item (<Alt>-asu) automatically goes through the specified spectrum and tries to update the spectrum-specific shifts for patterns using possibly intrapattern peaks within specified tolerances.

*Assign/Spin System/ Edit Frequency Chemical Shift*

The **Edit Frequency Chemical Shift** menu item (<Alt>-ase) allows you to change the chemical shifts of frequencies of pattern. You can edit the generic shift or the spectrum-specific shift. If you select the latter, you automatically select the experiment which will undergo a shift change.

*Assign/Spin System/ Graphical Edit Frequency*

The **Assign/Spin System/Graphical Edit Frequency** menu item (<Alt>-asf) allows you to change the chemical shifts of frequencies of a pattern in a graphical way. You are asked which pattern's chemical shifts to edit, then lines are drawn across the spectrum at these frequencies. You must select the frequency to be edited with the crosshair cursor, then position the crosshair at the new frequency location and click. You can continue to edit the frequencies in that pattern until you want to exit the menu item (by pressing <Esc>).

*Assign/Spin System/ Tile+Show+Edit Frequencies*

The **Tile+Show+Edit Frequencies** menu item (<Alt>-ast) is similar to the **Graphical Edit Frequency** menu item, but after asking which pattern is to be edited, it tiles the pattern, shows the frequencies, and then allows you to edit. Press <Esc> to exit this menu item.

*Assign/Spin System/ Rename Spin Systems*

The **Rename Spin Systems** menu item (<Alt>-asn) allows you to change the default names of all or particular patterns to the residue they belong to. If you assign all frequencies in a pattern (for example, pa2) to a specific residue (for example, ALA_32), then this action updates the name of the pattern and, in subsequent actions (for example, tiling and listing), the name of that pattern shows as that residue.

## Spin System Extension–Assign/Extend Spin Systems

Three automated methods for adding frequencies to spin systems (patterns) are accessible through the **Assign/Extend Spin Systems** (<Alt>-ad) menu item:

### Extend via HCCH-TOCSY

Using this method on the current 3D HCCH-TOCSY experiment allows you to automatically add new frequencies to the patterns starting from $C_\alpha$, $H_\alpha$ frequency pairs.

### Extend via H(CCH)(CO)NH or C(CH)(CO)NH

Using this method on the current 3D H(CCH)-TOCSY-(CO)NH or 3D C(CH)-TOCSY-(CO)NH experiment allows you to automatically add new frequencies to the patterns.

> **Note:** This option works only if the neighbor relationships between the spin systems are unambiguous (the probabilities are set to 1), since it adds aliphatic $H_\alpha$ or $C_\alpha$ frequencies not to the $i^{th}$ pattern but to the *i-1* neighbor.

### Extend via HBHA(CO)NH

Using this method on the current 3D HBHA(CO)NH experiment allows you to automatically add new frequencies to the patterns.

> **Note:** This option works only if the neighbor relationships between the spin systems are unambiguous (the probabilities are set to 1), since it adds aliphatic $H_\alpha$ and $H_\beta$ frequencies not to the $i^{th}$ pattern but to the *i-1* neighbor.

## Residue Type Probability–Assign/Residue Type

The items in the **Assign/Residue Type** menu (<Alt>-ay) deal with the residue type probabilities of patterns.

### Assign/Residue Type/Score Residue Type

The **Assign/Residue Type/Score Residue Type** menu item (<Alt>-ays) allows you to automatically match patterns and residue types.

Two algorithms are available for scoring. One is adapted for proton or all-atom scoring. Here, for each atom in the library, Assign tries to find a matching frequency in the pattern. The best-matching frequency may lie no more than a user-defined standard deviation

away from the expected value. If a required number of atoms can be matched, a score is computed and stored in the database. The scores are then stored in the database as probabilities that one particular pattern belongs to a certain residue type.

The other method uses the $C_\alpha$ and / or the $C_\beta$ chemical shift distribution to score patterns. This option can be triggered by setting the **Scoring method** to **CACB only**.

These probabilities can then be used by the **Assign/Sequential** menu items to make sequence-specific assignments.

### Assign/Residue Type/Match Residue Type

The **Assign/Residue Type/Match Residue Type** menu item (<Alt>-aym) allows you to compare a pattern with a residue type defined in the library.

For each atom in the residue and for each frequency in the pattern, a result is printed: [ $\delta$ = (actual shift – expectation value) / standard deviation]; therefore the lower the value of $\delta$, the higher the probability that the particular frequency belongs to the atom.

### Assign/Residue Type/Set Residue Type

The **Assign/Residue Type/Set Residue Type** menu item (<Alt>-ayt) allows you to manually set the probability (or “score”) that a particular pattern belongs to a certain residue type. You can then list it using the **Assign/Residue Type/Show Residue Type** menu item.

### Assign/Residue Type/Show Residue Type

The **Assign/Residue Type/Show Residue Type** menu item (<Alt>-ayw) allows you to list the score you set with the **Assign/Residue Type/Set Residue Type** menu item or calculated with **Assign/Residue Type/Score Residue Type** menu item.

## Neighbor Probability–Assign/Neighbor

The **Assign/Neighbor** pullright menu (<Alt>-an) contains a set of menu items that deal with the neighbors of patterns (the patterns belonging to consecutive residues in the sequence).

The first part of the pullright contains menu items for automatically detecting potential neighbor patterns in a 2D NOESY spectrum (**Find Neighbor Via 2D NOE**), in a homonuclear 3D [J,NOE], [NOE,J] or [NOE,NOE] spectrum (**Find Neighbor Via 3D NOE**), in a $^{15}$N-separated NOE spectrum (**Find Neighbor Via 3D N-15 NOE**), or in triple-resonance experiments (**Find Neighbor Via 3D/4D**). Please see Chapter 1, *Assign* for a description of the algorithms for the **Find Neighbor** menu items.

## Assign/Neighbor/Find Neighbor Via 2D NOE

In the **Assign/Neighbor/Find Neighbor Via 2D NOE** (<Alt>-an2) menu item the following controls can be set:

### Table 18 Assign/Neighbor/Find Neighbor Via 2D NOE controls

| Control | Function |
|---|---|
| **Experiment** | Specify which peak entity to run neighbor detection on. |
| **Root Frequency Tolerance** | Define the tolerance for root frequencies in ppm units, for use in the algorithm. |
| **Frequency Collapse Tolerance** | Define the frequency collapse tolerance in ppm units. This tolerance is used to judge whether a candidate frequency is new or old. You can specify a pattern for which the neighbors are searched or you can set this parameter to detect all patterns at once. |
| **Normalize Scores** | Set the scores for all patterns to yield a sum of one. |
| **Store Scores** | Specify if the newly calculated scores should replace or be added to the old ones. |
| **Number of Neighbors to Store** | Specify the maximum number of candidate neighbors to store. |
| **Output Level** | Specify how much detail to show in the text window during the neighbor search. |
| **Range of frequencies to Use** | Specify the range for the pattern in ppm. Only the NOESY peaks that lie in this range are used. Normally for proteins this would be the $H_\alpha$, $H_\beta$ region, but you may also include the $H_N$ region. |
| **Maximum # of frequencies to Use** | Specify the maximum number of frequencies of the particular patterns to use in the detection algorithm. |
| **Range of candidates to Use** | Specify the ppm range of candidates to use. For proteins this would be the amide region. |

**Table 18 Assign/Neighbor/Find Neighbor Via 2D NOE controls (Continued)**

| Control | Function |
|---|---|
| **Minimum # of NOE contacts** | Specify how many frequencies of the pattern the candidate should have NOE contact with. |
| **Minimum # of NOE peaks** | Specify how many NOE peaks should be between the frequencies of the pattern and this candidate (this number should be ≥ the number of NOE contacts). |

## Assign/Neighbor/Find Neighbor Via 3D NOE

In the **Assign/Neighbor/Find Neighbor Via 3D NOE** (<Alt>-an3) menu item, you should specify the **Range of freq** for the pattern in ppm. Only the NOESY peaks that lie in this range are used. Normally for proteins this would be the $H_\alpha$, $H_\beta$ region, but you may also include the $H_N$ region.

**Table 19 Assign/Neighbor/Find Neighbor Via 3D NOE controls**

| Control | Function |
|---|---|
| **Max # of freq's to use** | Specify the maximum number of frequencies of the particular patterns to use in the detection algorithm. |
| **Range of cands to use** | Specify the ppm range of candidates to use. For proteins this would be the amide region. |
| **Range of seqs to use** | Filter out peaks having their w2 out of the range specified. |
| **Use reverse cons** | Reverse connectivity peaks (peaks arising in a homonuclear TOCSY-NOE spectrum) from, for example, $H_{\alpha,i+1}$-$H_{N,i+1}$-$H_{N,i}$. The **Range of reverse cons** must also be specified for this operation. |
| **Range of reverse cons** | Specify the ppm range of the expected reverse connectivity peaks. |
| **Min # of NOE contacts** | Specify how many frequencies of the pattern the candidate should have NOE contact with. |
| **Min # of NOE peaks** | Specify how many NOE peaks should be between the frequencies of the pattern and this candidate (this number should be ≥ the number of NOE contacts). |
| **Root freq tol** | Define the tolerance for root frequencies in ppm units, for use in the algorithm. |

**Table 19 Assign/Neighbor/Find Neighbor Via 3D NOE controls (Continued)**

| Control | Function |
|---|---|
| **Freq coltol** | Define the frequency collapse tolerance in ppm units. This tolerance is used to judge whether a candidate frequency is new or old.You can specify a pattern for which the neighbors are searched or can set this parameter to detect all patterns at once. |
| **Normalize scores** | Set the scores for all patterns to yield a sum of one. |
| **Store neigh.** | Specify the maximum number of candidate neighbors to store. |
| **peak entity** | Specify which peak to analyze with neighbor detection. |
| **level of output** | Specify the degree of detail to show in the output window during the neighbor search. |

## Assign/Neighbor/Find Neighbor Via 3D N-15 NOE

In the **Assign/Neighbor/Find Neighbor Via 3D N-15 NOE** (<Alt>-ann) menu item, you can use the default parameters or specify the maximum number of frequencies of the particular patterns to use in the detection algorithm (**Number of Frequencies to use in Pattern**). You should then specify the atom type and ranges in ppm to use for the **First Type** (usually the $H_N$ frequencies) and for the **Second Type** (usually $^{15}N$).

Then you should specify the dimension to use for candidate search (**Candidate (NOE) Dimension**) and the region in ppm. Only the NOESY peaks that lie in this range are used. Normally for proteins this would be the $H_\alpha$, $H_\beta$ region, but you may also include the $H_N$ region.

Several additional parameters can be modified:

**Table 20 Assign/Neighbor/Find Neighbor Via 3D N-15 NOE controls**

| Control | Function |
|---|---|
| **Minimum # of NOE Peaks** | Specify how many NOE peaks should be between the frequencies of the pattern and this candidate. |
| **Frequency Collapse Tolerance** | Define the frequency-collapse tolerance (ppm units), which is used to decide whether a candidate frequency is new or old.You can specify a pattern for which the neighbors are to be searched or specify all patterns. |
| **Normalize Scores** | Set the scores for all patterns to yield a sum of one. |
| **Store Scores** | Whether newly calculated scores should replace old ones or be added to old ones. |
| **Number of Neighbors to Store** | Specify maximum number of candidate neighbors to store.You should also enter the peak entity on which the neighbor detection should run and the level of output. |

## Assign/Neighbor/Find Neighbor Via 3D/4D

This menu item (<Alt>-an4) allows you to find neighbors for existing patterns in a triple-resonance 3D or 4D experiment, such as HN(CO)CA.

## Assign/Neighbor/List Neighbors

The **Assign/Neighbor/List Neighbors** menu item (<Alt>-anl) lists the neighbor relation between patterns set in manual or automated routines.

## Assign/Neighbor/Show Neighbors Via Strip Plot

This menu item (<Alt>-anr) allows you to view possible neighbors found in automated routines by spawning strip plots on a 3D spectrum (e.g., HSQC-NOESY or CBCANH).

## Assign/Neighbor/Set Neighbors Manual

The **Assign/Neighbor/Set Neighbors Manual** menu item (<Alt>-ans) allows you to set one pattern as another's neighbor ($i$ - $i$ + 1 connectivity or $i$ - $i$ - 1 connectivity) with a certain probability.

### Assign/Neighbor/Delete Neighbors

The **Assign/Neighbor/Delete Neighbors** menu item (<Alt>-and) deletes a previously set neighbor relation.

## Search for Sequential Assignment –Assign/Sequential

The **Assign/Sequential** (<Alt>-aq) pullright contains two menu items to make automated sequence-specific assignments and several menu items to review them.

### Assign/Sequential/Systematic Search

The **Assign/Sequential/Systematic Search** (<Alt>-aqs) menu item allows you to find potential matchings of the collection of patterns onto the sequence using a systematic search. The patterns should have neighbors and residue type assignments. The molecule and the library should also be defined.

> **Note:** This option works only on unbranched biomacromolecules.

**Table 21 Assign/Sequential/Systematic Search controls**

| Control | Function |
|---|---|
| **Min Individual Assignment Prob** | Tell the program when a pattern is considered to be assigned to a residue type. |
| **Min Neighbor Prob Score** | Indicate which candidate neighbors to consider from a possibly longer list. |
| **First Residue (#) to Consider** | Specify the first residue (by number) from which to generate assignments. |
| **Last Residue (#) to Consider** | Specify the last residue from which to generate assignments. |
| **Min Length of Assigned Stretches** | Specify the minimum length of the sequence for which to make assignments. |
| **Max # of Assignments to Generate** | Prevent the program from running in an infinite loop (i.e., quit after generating this many assignments). |
| **Sort Generated Assignments** | Print assignments as they are generated or have them stored and sorted (by setting to **True**) at the end of the procedure, so that only the best ones are listed. Besides sorting, Assign also removes any assigned stretches that are contained in a longer stretch. |

**Table 21 Assign/Sequential/Systematic Search controls (Continued)**

| Control | Function |
|---|---|
| **Max # of Assignment to Print** | Specify the maximum number of sequential assignments to print out and store in the database (if **Sort generated assignments** is **true**). |
| **Output Level** | Specify the degree of detail in the search output. |

## Assign/Sequential/Simulated Annealing

The **Assign/Sequential/Simulated Annealing** (<Alt>-aqa) menu item finds optional matchings of the collection of patterns to the sequence, using the simulated annealing method. The patterns should have neighbors and residue-type assignments. The molecule and the library should also be defined.

- The **Min individual assignment prob** option tells the program when a pattern is considered to be assigned to a residue type.
- The **First residue (#) to consider** and **Last residue (#) to consider** for which the assignments are to be generated should be entered by number.
- Setting **Discard previous assignment** to **True** causes FELIX to disregard previous assignments of patterns. The **Store generated assignments** parameter specifies that the result of the assignment should be stored, besides reporting it. The storage happens by assigning each individual frequency in the pattern to a form, e.g., 1:VAL_37:*.

The next three settings are used to fine-tune the simulated annealing algorithm: you can change the **Temperature factor** to a lower number if most of the sequence is well defined or set the number of iterations in **Iteration factor**. The **Sequential/individual factor** setting can be used to change the weight of the neighbor information vs. spin-system probability scores.

Finally, select the appropriate **Output level**.

## Assign/Sequential/Show Suggested Via Strip

This menu item (<Alt>-aqr) allows you to show the stored results of an Assign **Sequential/Systematic Search** run as strip plots on a 3D spectrum.

### Assign/Sequential/Compose Stretch with Strip Plot

This menu item composes a “stretch” of spin systems, starting with a specified pattern, using only neighbor probabilities. The stretch is then stored in the stretch entity. The action also creates a strip plot for the current 3D triple resonance matrix.

### Assign/Sequential/Suggest Assignment For Strip

This menu item suggests a sequence-specific assignment for a specified stretch, using the residue-type probabilities.

## Spin System Assignment–Assign/Assign Spin System

The last step in assigning spin systems is to name the resonances within them. The menu items to do this are in the **Assign/Assign Spin System** (<Alt>-aa) pullright.

### Assign/Assign Spin System/Residue Type

The **Assign/Assign Spin System/Residue Type** (<Alt>-aar) menu item allows you to assign all frequencies in a particular pattern to a residue type (but not number) in the sequence (i.e., a non-sequence-specific assignment). A typical assignment is that each frequency has a name like 1:VAL_*:*, where wildcards (*****) mean that the exact residue number and the atom name is not defined yet.

### Assign/Assign Spin System/Sequence Specific

The **Assign/Assign Spin System/Sequence Specific** (<Alt>-aas) menu item allows you to assign all frequencies in a particular pattern to a residue type and number in the sequence (i.e., a sequence-specific assignment) but no atom name is specified. A typical assignment is that each frequency has a name like 1:VAL_27:*, where the wildcard (*****) mean that the exact atom name is not defined yet. Meanwhile, you can request that the program make **suggestions** based on matching the chemical shifts of the frequencies against the libraries and naming the frequencies with atom names (**Assign Frequencies** should be set to **Yes**). In this case the frequencies in the pattern will have names like 1:VAL_27:HN.

### Assign/Assign Spin System/Frequency

The **Assign/Assign Spin System/Frequency** (<Alt>-aaf) menu item allows you to manually assign the frequencies of the patterns.

First, you select the frequency of the assignment to be built. You can do this by clicking the **Frequency** control panel and then clicking **SELECT**. Next you select a residue type from the **Residues** list and click **FILTER**. The program then fills in the **#s** list with valid residue numbers and the **Nuclei** list with all protons (or carbons or nitrogens, depending on the type of frequency) of that particular residue type. If you now click one number from the **#s** list and one nucleus from the **Protons** list, clicking the **BUILD** button fills in the atom specification for that particular frequency.

If the atom specification is acceptable, click **ADD** or **Return** to transfer it to the **Possible Assignment** list and store it in the database. Alternatively, you may directly type in the name of the frequency in the **Atom Spec** box. Up to six possible assignments can be stored with each frequency in the database. Unique or most-probable assignments can be specified in the database by selecting them from the list and clicking **SPECIFY**, in which case this assignment shows up in the top line under the Assignment header. You also can despecify–that is, make all assignments equally possible–by clicking **SPECIFY** without highlighting any assignment. You can delete any assignment by clicking **DELETE**.

### Assign/Assign Spin System/Unassign Spin System

The **Assign/Assign Spin System/Unassign Spin System** (<Alt>-aau) menu item allows you to unassign all frequencies in a pattern.

## Assign/Chemical Shift Index (CSI)

If the resonance assignments are done for all the patterns, then you can derive some secondary structural information–this method is called the “Chemical Shift Index.”

### Assign/Chemical Shift Index/HA Chemical Shift Index

Calculate the chemical shift indices based on assigned $H_\alpha$ chemical shifts in current patterns (<Alt>-aih).

### Assign/Chemical Shift Index/CA Chemical Shift Index

Calculate and tabulate the CSI, based on assigned $C_\alpha$ shifts in current patterns (<Alt>-aia).

### Assign/Chemical Shift Index/CB Chemical Shift Index

Calculate and tabulate the CSI, based on assigned $C_\beta$ shifts in current patterns (<Alt>-aib).

### Assign/Chemical Shift Index/C Chemical Shift Index

Calculate and tabulate the CSI based on assigned C¢ shifts in current patterns (<Alt>-aic).

### Assign/Chemical Shift Index/Consensus CSI

Calculate a consensus CSI table, if the $H_\alpha$, $C_\alpha$, $C_\beta$ and C¢ CSI's are already calculated and stored (<Alt>-aio).

### Assign/Chemical Shift Index/Dihedral Restraint

Based on a consensus CSI table, this menu item calculates dihedral angle restraints for Ψ and Φ for consensus secondary structural regions (<Alt>-aid).

## Reporting Spin Systems–Assign/Report Spin System

After the assignments are done, you can create reports of particular spin systems with the **Assign/Report Spin System** (<Alt>-ar) menu item, which either lists information to the output window or stores the frequencies, frequency names, neighbors, and generic and spectrum-specific shifts in a file.

## Assign/Zoom Spin System

The **Assign/Zoom Spin System** menu item (<Alt>-az) is useful for 3D and 4D spectra when you want to visually inspect the final spin systems or patterns. You can define which pattern's frequencies should define the region. If you select a pattern name from the **Source** list and then click the **Show** button, the **Frequencies** box is

filled in with the frequencies of that pattern. If you then select the desired **Orientation** and specify which frequency should be at which dimension, also specifying the required ranges, the region can be brought up by clicking **OK**.

## Spin System Table–Edit/Spin Systems

Most of the visualization menu items useful for analyzing patterns (spin systems) are accessible through the spin-system table. The table is opened by using the **Edit/Spin Systems** (<Alt>-es) menu item. Before you can execute any menu items in the table, you need to select (click) a row or multiple rows, depending what you want to do.

### Action/Zoom

The **Action/Zoom** (<Alt>-az) menu item from the Spinsystems table is useful for 3D and 4D spectra when you want to visually inspect the frequencies in spin systems. You can select a pattern by clicking its number (highlighting the row) for which pattern's frequencies should define the region. Then you need to use the **Action/Zoom** menu item or click the **Zoom** icon in the Spinsystems table. To set the preferences for this menu item, use the **Preferences/Zoom** menu item in the Spinsystems table.

### Action/Draw

You can visualize a pattern by drawing straight lines on the actual spectrum along the frequencies of particular spin systems. First you need to select one or two patterns from the table and then click the **Action/Draw** (<Alt>-ad) menu item or click the **Draw** icon in the Spinsystems table. By using the **View/Draw Frequencies or the Clear Frequencies** item in the right-mouse menu, you can turn these lines off. To set the colors, you can use the **Preference/Draw** menu item from the Spinsystems table.

### Action/Tile Plot

The **Action/Tile Plot** (<Alt>-at) menu item is used to define a tile from any pair of patterns. You first need to select two patterns by clicking two rows in the table. Then you need to select the **Action/Tile Plot** menu item or use the **Tile Plot** icon in the Spinsystems

table. Tile plot preferences can be with in the **Preference/Tile Plot** menu item in the Spinsystems table. To turn off the tile plot you can click the **View/Plottype/Tile/Strip Plot** toggle.

## Action/Strip Plot

The **Action/Strip Plot** (<Alt>-as) menu item in the Spinsystems table is used to define a strip plot from a specific pattern. First you need to select a pattern by clicking the needed row in the table. Then you select the **Action/Strip Plot** menu item or use the **Strip Plot** icon on the Spinsystems table. Strip plot preferences can be set with the **Preference/Strip Plot** menu item in the Spinsystems table. To turn off the strip plot you can click the **View/Plottype/Tile/Strip Plot** toggle.

## Action/ND Strip Plot

This menu item (<Alt>-an) allows you to define a strip plot from a 3D/4D spectrum, where the individual strips originate from potentially different planes of different spin systems. You first need to select the needed patterns by clicking the rows in the table (after the first click, use <Shift>-click). Then you need to select the **Action/ND Strip Plot** menu item or use the **ND Strip Plot** icon on the Spinsystems table. Strip plot preferences can be set with the **Preference/ND Strip Plot** menu item in the Spinsystems table. To turn off the tile plot you can click the **View/Plottype/Tile/Strip Plot** toggle.

## Spinsystem/Add Frequency via Cursor

This menu item (<Alt>-sf) in the Spinsystem table allows you to graphically add a frequency to the selected pattern.

## Spinsystem/List Residue Type

The **Spinsystem/List Residue Type** menu item (<Alt>-st) in the Spinsystem table allows you to list the “score” that you set with the **Assign/Residue Type/Set Residue Type** menu item or that you calculated via the **Assign/Residue Type/Score Residue Type** menu item for the previously highlighted pattern.

### Spinsystem/List i+1 Neighbors

The **Spinsystem/List i+1 Neighbors** menu item (<Alt>-sl) in the Spinsystem table lists the *i*+1 neighbor patterns from the database for the current (highlighted) spin system.

### Spinsystem/Show i+1 Neighbors Via Strip Plot

This menu item (<Alt>-sn) allows you to view possible *i*+1 neighbors for the current spin system in the database by spawning strip plots on a 3D spectrum (e.g., HSQC-NOESY or CBCANH).

### Spinsystem/List i-1 Neighbors

The **Spinsystem/List i-1 Neighbors** menu item (<Alt>-s-) in the **Spinsystem** table lists the *i*-1 neighbor patterns from the database for the current (highlighted) spin system.

### Spinsystem/Show i-1 Neighbors Via Strip Plot

This menu item (<Alt>-as) allows you to view possible *i*-1 neighbors for the current spin system in the database by spawning strip plots on a 3D spectrum (e.g., HSQC-NOESY or CBCANH).

### Spinsystem/Delete Neighbor

The **Spinsystem/Delete Neighbor** menu item (<Alt>-sd) in the Spinsystem table allows you to interactively delete neighbors that were shown as strip plots for a particular spin system. You need to click a particular strip to delete the spin system defining that strip as a possible neighbor.

### Spinsystem/Perpendicular Strips

The **Spinsystem/Perpendicular Strips** menu item (<Alt>-sp) in the Spinsystem table allows you to interactively define a set of strip plots that are perpendicular to the currently viewed plot of a 3D double- or triple-resonance spectrum. For example, first display an HN-N plane of an HNCA on CBCANH or $^{15}$N NOESY spectrum at a spin system's intraresidue or interresidue $^{13}$C or HA frequency. Then select the **Spinsystem/Perpendicular Strips** menu item in the Spinsystems table, click the same peak (the target spin system), then click the peaks that are in the same plane (the possible neighbor spin

systems) and from which strip plots will be spawned. (If FELIX cannot find a peak in the spin system table, you should go back and manually add that spin system to the spin system table using the frequency clipboard.) When there are no more candidate peaks left in the plane, press <Esc>. FELIX then draws the strip plots, target spin system first.

The menu item works properly if the **ND Strip** preference is set (e.g., which is the NH axis and which is the $^{15}$N axis, etc.).

To set then the neighbor relations you can use the **Spinsystem/Set i+1 Neighbor** or **Spinsystem/Set i-1 Neighbor** menu item in the Spinsystems table. You need to set the appropriate neighbor relationship (i.e., *i*+1 or *i*-1) depending on whether the spectrum is double or triple resonance and whether the target spin system's peak was selected as an intraresidue or interresidue peak.

## Spinsystem/Set i+1 Neighbor, Spinsystem/Set i-1 Neighbor

The **Spinsystem/Set i+1 Neighbor** (<Alt>-se) and **Spinsystem/Set i-1 Neighbor** (<Alt>-si) menu items in the Spinsystem table allow you to visually set neighbors for a spin system if an *N*D strip plot is displayed.

Select the **Spinsystem/Set i+1 Neighbor** or **Spinsystem/Set i-1 Neighbor** menu item from the Spinsystems table then use the crosshair cursor to select first the target spin system's strip and then the neighbor spin system's strip. If **Set i+1 Neighbor** is selected, the neighbor spin system is marked as the *i*+1 neighbor of the target spin system. If **Set i-1 Neighbor** is selected, the target spin system is marked as the *i*-1 neighbor of the neighbor spin system.

## Spinsystem/Assign

The **Spinsystem/Assign** menu item (<Alt>-sa) from the Spinsystem table is similar to the The **Assign/Assign Spin System/Sequence Specific** menu item.

## Spinsystem/Report

After the assignments are done you can list the frequencies, frequency names, neighbors, and generic and spectrum-specific shifts for the current spin system in the output window, using this menu item (<Alt>-sr) from the Spinsystem table.

## Stretches of Spin Systems–Edit/Stretches

A set of menu items dealing with stretches of spin systems is accessible within the Stretches table. These stretches of spin systems can be composed with either the **Assign/Sequential/Systematic Search** menu item or the **Assign/Sequential/Compose Stretch** menu item with the **Strip Plot** controls. The table is automatically opened by these menu items when it is created, or alternatively, you can open the table with the **Edit/Stretches** (<Alt>-ec) menu item. Before you can execute any menu item in the table, you must first select a stretch by highlighting (clicking) its row.

### Action/ND Strip Plot

The **Action/ND Strip Plot** (<Alt>-an) menu item in the Stretches table allows you to define a strip plot from a 3D or 4D spectrum, where the individual strips originate from potentially different planes of the spin systems of the stretch. You first must select the stretch by clicking its row in the table. You then select the **Action/ ND Strip Plot** menu item or use the **ND Strip Plot** icon on the Stretches table. You can set strip plot preferences using the **Preferences/ND Strip Plot** menu item in the Stretches table. To turn off the strip plot, click the **View/Plottype/Tile/Strip Plot** toggle.

### Stretch/New

The **Stretch/New** (<Alt>-sn) menu item in the Stretches table allows you to define a new stretch. You must select a pattern from the control panel to add a new stretch to the table. You can then manually add other spin systems.

### Stretch/Compose Stretch with Strip Plot

The **Stretch/Compose Stretch with Strip Plot** (<Alt>-sc) menu item in the Stretches table is the same as the **Assign/Sequential/ Compose Stretch with Strip Plot** menu item. This menu item composes the stretch of spin systems, starting from a specified pattern, and using only neighbor probabilities. The stretch is then stored in the stretch entity and displayed in the table. The menu item also creates a strip plot for the current 3D matrix.

### Stretch/Add Pattern

The **Stretch/Add Pattern** (<Alt>-sa) menu item in the Stretches table allows you to manually add a spin system to the current (active) stretch, using the control panel.

### Stretch/Swap Pattern

The **Stretch/Swap Pattern** (<Alt>-sp) menu item in the Stretches table allows you to swap out a pattern from the current stretch. You must first click the pattern in the strip plot with the large crosshair cursor and then select a new pattern from the control panel. The menu item then swaps the two patterns in the entity and in the strip plot.

### Stretch/Delete Pattern

The **Stretch/Delete Pattern** (<Alt>-sd) menu item in the Stretches table allows you to interactively delete a pattern by selecting the strip plot of the current stretch with the large crosshair. The pattern is also deleted from the table.

### Stretch/Suggest Assignment for Stretch

The **Stretch/Suggest Assignment for Stretch** (<Alt>-ss) menu item in the Stretches table suggests a sequence-specific assignment for the current stretch of spin systems using the previously defined residue-type probabilities.

### Stretch/Assign One Stretch

The **Stretch/Suggest Assignment for Stretch** (<Alt>-ss) menu item in the Stretches table performs a sequence-specific resonance assignment for the patterns in the current stretch, using either suggested assignments (from the **Assign/Sequential/Systematic Search** menu item or from the **Stretch/Suggest Assignment** menu item in the Stretch table) or assignments you specify.

## Assign Residue Table–Edit/Residues

After you have made sequence-specific assignments for some spin systems, you can look at the assignments not only in the Spinsystem

table, but also in the Residues table. This table is activated through the **Edit/Residues** (<Alt>-er) menu item.

### File/Save As

The **File /Save As** (<Alt>-fa) menu item saves a report about the current resonance assignments for the given residue to a .txt file.

### Action/Zoom

The **Action/Zoom** (<Alt>-az) menu item in the Residues table is useful for 3D and 4D spectra when you want to visually inspect the frequencies in spin systems assigned to specific residues. You can select a residue that is assigned to a pattern by clicking its number (which highlights the row), which should the define the region for the pattern's frequencies. You then use the **Action/Zoom** menu item or the **Zoom** icon in the Residues table. To set the preferences for this action, use the **Preferences/Zoom** menu item in the Residues table.

### Action/Draw

You can visualize an assigned residue by drawing straight lines on the actual spectrum along the frequencies of the particular spin system to which it is assigned. First you select one or two residues from the table, then select the **Action/Draw** (<Alt>-ad) menu item or click the **Draw** icon in the Residues table.

### Action/Tile Plot

The **Action/Tile Plot** (<Alt>-at) menu item is used to define a tile from any pair of assigned residues. You first select two patterns by clicking two rows of residues in the table, then select the **Action/Tile Plot** menu item or click the **Tile Plot** icon in the Residues table. You can set tile plot preferences with the **Preferences/Tile Plot** menu item in the Residues table. To turn off the tile plot, click the **View/ Plottype/Tile/Strip Plot** toggle.

### Action/Strip Plot

The **Action/Strip Plot** (<Alt>-as) menu item in the Residues table is used to define a strip plot from a specific assigned pattern. First you select a pattern by clicking the needed row of residues in the table. You then select the **Action/Strip Plot** menu item or click the **Strip**

**Plot** icon in the Residues table. Strip plot preferences can be set with the **Preferences/Strip Plot** menu item in the Residues table. To turn off the strip plot, click the **View/Plottype/Tile/Strip Plot** toggle.

## Action/ND Strip Plot

The **Action/ND Strip Plot** (<Alt>-an) menu item allows you to define a strip plot from a 3D/4D spectrum, where the individual strips originate from potentially different planes of different spin systems. You first select the needed patterns by clicking the rows of the residues in the table (<Shift>-click to select more than one row). You then select the **Action/ND Strip Plot** menu item or click the **ND Strip Plot** icon in the Residues table. You can set strip plot preferences with the **Preferences/ND Strip Plot** menu item in the Residues table. To turn off the tile plot, click the **View/Plot type/Tile/Strip Plot** toggle.

## Action/Show Residue

The **Action/Show Residue** (<Alt>-ar) menu item in the Residues table can display one or more residues in the molecule by blanking the other residues. You can only use this menu item if you have activated the 3D molecule display through the **Model/Draw Molecule** menu item.

## Residue/Assign

The **Residue/Assign** (<Alt>-ra) menu item in the Residues table can be used to assign the frequencies for a selected pattern to the atoms of the current residue.

## Residue/Report

The **Residue/Report** (<Alt>-rr) menu item prints a report in the output window about the current resonance assignments for the residue.

# Peak Assignment menu items

## Assign/List Peak

With the **Assign/List Peak** (<Alt>-at) menu item you can query the database for a particular peak's positions, assignment status, and membership in spin systems.

## Assign/Peak Assign

The **Assign/Peak Assign** (<Alt>-ak) pullright. contains several menu items for changing the assignments of a peak. A peak is considered to be assigned if the frequencies defining it are assigned *and* if this assignment is transferred to the peak itself. Therefore, in an Assign strategy, the frequencies in the patterns are the primary targets of the assignment. Once the frequencies are assigned, you can transfer this information to peaks automatically (**Assign/Peak Assign Peak/Autoassign Peaks**) or manually (**Assign/Peak Assign/Manual Assign Singly** or **Assign/Peak Assign/Manual Assign Multiply**).

### Assign/Peak Assign/Manual Assign Singly

The **Assign/Peak Assign/Manual Assign Singly** (<Alt>-aks) menu item is used to manually assign peaks. If you assign peaks manually, FELIX first checks to see whether the frequencies defining that peak are already assigned in patterns. If the frequencies are assigned, they show up as possible peak assignments. You can then select from the listed names for a peak or create a different assignment.

First you select the **Dimension** for which the assignment is to be built by choosing from the **Dimension** popup. Then you select a residue type from the **Residues** list and click the **FILTER** button. The program then fills in the **#s** list with valid residue numbers, and the **Nuclei** list with, for example, all protons of that particular residue type. Clicking one number from the **#s** list and one nucleus from the **Nuclei** list, then clicking the **BUILD ASN** button fills in the atom specification for that particular frequency. To accept these specifications, press <Enter>, which transfers this atom specification to the

peak and also stores it in the database. You can also select a frequency assignment from the **Use Frequency Assignments** list and make it the actual peak assignment by pressing <Enter>. You may delete any peak assignments by clicking the **DELETE ASN** button or just clear an erroneous one from the box with the **CLEAR ASN** button. You can use several utilities in the **Show** or the **Restraints** pulldown. To use the **Restraints/NOE_DIST** or **NOE_VOL** options you must have scalar peaks defined and also must have measured the peak volumes. With these options you can define or redefine restraints peak by peak. With the **Show** utilities you can: measure distances between the constituent atoms of possible frequency assignments in the current model, see the NOE buildup (if you measured one previously), see unassigned atoms that are within 7 Å of the atoms to which this peak can possibly be assigned, and draw the frequencies on the peak box to help to distinguish visually between the possibilities. Finally, if you are working with a 2D NOE spectrum you can jump to the symmetric peak and assign it, verify the assignments, or identify overlap problems.

## Assign/Peak Assign/Manual Assign Singly 3D

The **Assign/Peak Assign/Manual Assign Singly 3D** (<Alt>-akd) menu item is used to manually assign peaks in a 3D HSQC-NOESY type experiment. You must first select and display a 3D HSQC-NOESY type experiment before selecting this menu item. This menu item changes the cursor to one that you can use to select the peak you want to assign. FELIX then checks the cross-peak table for the HSQC-NOESY experiment to see if assignments already exist. If they do, they are displayed along with the chemical shift and tolerance for each dimension.

The control panel that appears contains two list boxes. The list box on the left displays matching assignment pairs for the HSQC dimension. To be listed, both assignments must be from the same residue and the frequencies for each assignment must both be within the given tolerance for that dimension, as shown in the control panel. Along with the assignments, the difference in chemical shift between the selected cross-peak and the assigned frequency is show for both HSQC dimensions. Often, the correct assignment is the only one listed. If multiple possible assignments are listed, the correct one can usually be selected based on the chemical shift deltas. If no possible assignments are listed or if you want to display more possibil-

ities, then you can increase the tolerance for the corresponding dimensions and click the **Refresh** button. All assignment pairs matching the new tolerances are then displayed.

The list box on the right shows the possible assignments in the NOESY dimension. To generate this list, FELIX goes through all assigned frequencies and displays those that are within the specified tolerance for the NOESY dimension. The distance to this proton is displayed, along with the difference in the chemical shift between the selected cross-peak and the assigned frequency. This list is sorted by distance. After you select a new HSQC assignment pair in the left list box or change the NOESY dimension tolerance, update the list by clicking the **Refresh** button.

After you are satisfied that you have selected the correct assignments, you can transfer the assignments to the HSQC-NOESY cross-peak table by clicking the **Assign** button in the **Store Assignment** rectangle. You can then click **NEW PEAK** to select a new peak for assignment and start the process over, or click **Cancel** to exit the control panel.

## Assign/Peak Assign/Manual Assign Multiply

In FELIX it is possible to have peaks with multiple competing assignments stored. You can use automated assignment to store that information, or you can use the **Assign/Peak Assign/Manual Assign Multiply** (<Alt>-akm) menu item to store, alter, or delete such information. If a peak has multiple possible assignments stored, then in the control panel this will show up. Then you can delete from those multiple assignments, add new ones, or promote one particular multiple assignment to a single assignment. You can use utilities from the **Show** or the **Restraints** pulldown. To use the **Restraints/NOE_DIST_OVLP** or **NOE_VOL_OVLP** options, you must define scalar peaks and measure the peak volumes. With these menu items you can define or redefine restraints peak by peak. From the **Show** pulldown you can: measure distances between the constituent atoms of possible frequency assignments in the current model, see the NOE buildup (if you measured one previously), see unassigned atoms that are within 7 Å of the atoms to which this peak can possibly be assigned, and draw the frequencies on the peak box to help to distinguish visually between the possibilities. Finally, if you are working with a 2D NOE spectrum, you can jump to the

symmetric peak and assign it, verify the assignments, or identify overlap problems.

## Assign/Peak Assign/Autoassign Peaks

The **Assign/Peak Assign/Autoassign Peaks** (<Alt>-aka) menu item is used to transfer frequency assignments to peak assignments. Therefore it is crucial that you be finished with frequency assignments before you use this menu item. You should also set and adjust the spectrum-specific shifts for the spectrum to which peaks are to be assigned, if necessary. Since the algorithm uses the molecule's structural information to decide whether the two atoms that the NOE transfer are assigned to are within the specified cutoff, the results depend on the actual structure: fewer peak assignments are made for a linear-chain protein than for a partially or fully folded one. The algorithm is described in Chapter 1, *Assign building blocks* You can specify the cutoff distance for NOEs (**Rejection Cutoff**) and the **Peak Entity** to assign. You can also specify whether to strictly enforce this distance cutoff (**Strictly Enforce Dist**). With this setting, when there is only one assignment possible for a given peak, then the assignment is retained, although the distance between the two atoms is more than the rejection cutoff. This option can be useful if all the resonances are assigned in a molecule and the structure is unfolded. You can choose to let the program multiply assign peaks; if so, it is possible to define a second cutoff distance (**Unambiguous Cutoff**), which is used to distinguish between multiple possible assignments. You can choose not to assign peaks that are already fully assigned (**Skip Fully Assigned**) or multiply assigned (**Skip Multiply Assigned**). Finally, you can select the **Output Level**.

## Assign/Peak Assign/Reassign Peaks

The **Assign/Peak Assign/Reassign Peaks** (<Alt>-akr) menu item reassigns certain peaks based on a list. You would typically use this menu item after hotspots were found in generated structures. Hotspots can result from misassignment or overlapped peaks. For misassignment, you use a list to reassign (or unassign) those peaks. You can use a simple ASCII file that just has the two (or three or four, depending on dimensionality) names in a row separated by a blank, or you can create a list from within Insight II using the following procedure: first load the restraints on all the refined molecules into Insight II using the **NMR_Refine/Restraints/Read** *molname** com-

mand. Then execute the **NMR_Refine/Distance/List** command. The provided **numvioltofelix** script redirects the resulting output to another file. This is the file you can use in the **Filename** parameter. Then you proceed similarly to the previously described **Manual Assign Singly** menu item, but instead of selecting the peaks by cursor, the peaks are automatically centered in your current frame after selecting them from the table and the corresponding control panel is opened.

### Assign/Peak Assign/Check Consistency

The **Assign/Peak Assign/Check Consistency** (<Alt>-akc) menu item checks the consistency between frequency assignments and peak names. In FELIX, each peak has a stored pointer to a frequency and a name for each dimension. If after assigning a peak to a frequency, you reassign that frequency, there can be an inconsistency–that is, the peak name will be different from the frequency name (that it points to). This can happen if, for example. you assign a frequency first to an HB1 and assign the peak accordingly, and then reassign that frequency to HB2. The peak name is still HB1, but the frequency to which it points is named HB2. To correct this kind of discrepancy you need to use the **Check Consistency** menu item. In the control panel, the **Peak Entity** defaults to the current experiment's peak entity. You can change this setting.

### Assign/Peak Assign/Unassign Peaks

The **Assign/Peak Assign/Unassign Peaks** (<Alt>-aku) menu item serves to unassign all peaks in the currently active experiment.

> **Important:** There is no way to cancel this menu item.

## Pattern/List Membership

You can use the **Pattern/List Membership** (<Alt>-nl) menu item to find out which pattern a particular peak belongs to (i.e., which is the closest pattern within the spectrum-specific tolerance).

# Pattern/List Frequency

This menu item allows you to determine which pattern(s) a particular cursor position can belong to (<Alt>-ny).

# 6 Autoscreen User Interface

## Project menu items

The first step in using Autoscreen to study protein-ligand interactions is to set up the project. The Autoscreen module organizes a research project in the FELIX database management system into a group of *entities*. Using the **Project** menu items, you can build a new project, open an existing project, add, delete, or display experiments, and define molecule files for the protein and ligands.

> **Note:** None of the Autoscreen menu items work until a project is open.

### Autoscreen/Project

The project entities are built (or loaded if they exist) using the **Autoscreen/Project** (<Alt>-np) menu item. The entities contain information about the NMR experiments and associated ligand structures, parameters for processing, scoring and displaying spectra, scoring results, and other data. The project entities are organized in a database directory with the same name as that of the project.

When you select this menu item, it prompts you to specify a project name if an Autoscreen project does not exist. This name is used to create a directory (in the database) in which several project entities are also created.

If a project already exists (that is, if it was built in a previous session) this menu item opens it.

You also have to specify whether 1D or 2D spectra you are dealing with. The default setting is 2D. Most of the command descriptions below assume you are using 2D data except where noted.

> **Note:** You can create only one Autoscreen project per database. When a project is finished, you can select **File/New** to open a new database, and then repeat this step to create a new project.

When creating a new project, FELIX tries to use the parameters for spectrum display and scoring that were used in the most recent project in the same session. If these are not available, it assigns default values for them. (But note that parameters for spectrum processing are not inherited from project to project.) The spectrum-display parameters related to the display limits, references, and threshold are reset based on the control spectrum so that they are, in effect, not strictly inherited either.

When creating a new project, the VERIFY DIRECTORIES control panel allows you to define the paths used to access or save the following files:

- Spectrometer Data (For 2D project only, represented by the FELIX symbol **forpfx**).
- Matrix Files (for 2D projects only, represented by the FELIX symbol **matpfx**).
- ASCII text files, including file list and results (represented by the FELIX symbol **txtpfx**).
- Processed 1D Files (For 1D project only, represented by the FELIX symbol **datpfx**).

You can click **Browse** next to any of the paths to select a directory interactively, or you can type in the directory directly. When you click **OK** to close the control panel, an empty Autoscreen Experiments Table opens.

**Table 22 Description of items in the Autoscreen Experiments Table**

| Column | Description |
|---|---|
| **id** | ID of the experiment. |
| **score** | Total score of the experiment. |
| **thresh** | Threshold used for peak picking if scored. |
| **status** | Status of the experiment, with 0 standing for nonprocessed, 1 for processed, 2 for scored, and 9 for control spectrum. |
| **fid** | File name of the FID file, if any. |

**Table 22 Description of items in the Autoscreen Experiments Table (Continued)**

| Column | Description |
|---|---|
| **type** | File type of the FID file, with ser for Bruker serial file, bruker_fid for Bruker fid file, fid for Varian FID file, mat for FELIX matrix, 2rr for Bruker processed file, var for Varian processed file, spc for NMR Compass file, ft2 for NMRPipe file, and nmr for TRIAD file. |
| **struc** | Filename of a molecule, with extension .pdb standing for PDB file, .car for CAR file, and .mol for MDL file. It can also be filename of a list of molecules if another extension is used. |
| **comment** | Comments, or ligand concentration for titration data. |

## Autoscreen/Experiment

Once the project is built, you can import the NMR experimental data and molecular data. You can use the **Autoscreen/Experiment** pull-right menu *or* the menu items on the Autoscreen Experiments Table to add, delete, and display experiments, or to define project paths.

Table 23 shows the types of raw or processed data that can be used by Autoscreen. If a processed data matrix is used, it is converted (or simply copied, if it is a FELIX matrix) to a FELIX matrix with a filename the same as the experiment ID. The symbol is used in the Autoscreen Experiments Table or a file list for indicating the datatype.

**Table 23 Types of NMR data used by Autoscreen[1]**

| Data file type | Symbol |
|---|---|
| Bruker raw 1D data | bruker_fid |
| Bruker raw 2D data | ser |
| Varian raw data | fid |
| Bruker processed data | 2rr |
| Varian processed data | var |
| FELIX processed data | mat |
| NMRPipe processed data | ft2 |
| TRIAD processed data | nmr |
| NMRCompass processed data | spc |

[1] Although various types of data can be imported into an Autoscreen

project, you are advised to stick to one of them for all experiments in a particular project instead of mixing them.

## Autoscreen/Experiment/Add From File List

This menu item allows you to add experiments to the project with a file list. A file list is an ASCII file that specifies the experiments and associated information to be added to the project. Each line in the file specifies the experiment ID, datafile, file type, structural filename (optional), and comments (optional), as explained below:

1. An experiment ID is one word used as the identification of the experiment. It is also used as the filename of the processed (or converted) FELIX matrix.
2. A datafile is the filename of the time-domain data (or processed data) prefixed by an absolute path or a path relative to the project path forpfx.
3. The type of the raw or processed data is specified as in Table 23.
4. A structural filename can be a .pdb, .car, or .mol file. Any other filename is taken as a list of structural files. A filename can have an absolute path or have a path relative to the project path xyzpfx. If you do not have a molecule but want to input comments, enter **null** as the filename.
5. The optional comment can be one or more words; FELIX concatenates them with underscores (_).

The following sample file list specifies information for five experiments:

```
Control    1/ser    ser  demo       control
Test1      2/ser    ser  test.car    test-1
Test2      3/ser    ser  mols.txt
Test3      4/ser    ser
Test4      5/ser    ser   null        The last experiment
```

## Autoscreen/Experiment/Add All Files

This menu item allows you to add all Bruker experiments in a certain directory to the project.

**Remember:** This function expects numbered Bruker experiments. If the experiments are not consecutively numbered, you can instruct it to skip one or more between every two experiments, or it will automatically ignore nonexistent experiments.

### Autoscreen/Experiment/Add One

This menu item allows you to add experiments to the project interactively. You can also specify other information for each experiment, such as the filename of the associated structure.

### Autoscreen/Experiment/Verify Directories

This menu item allows you to verify and change the important paths used by the project. See "Autoscreen / Project" on page 315 for more detail.

### Autoscreen/Experiment/Show Experiments Table

This menu item displays the Autoscreen Experiments Table if it is not open yet.

### Autoscreen Experiments Table menu items

*Edit/Add Experiment/ Add From File List* — This menu item is equivalent to **Autoscreen/Experiment/Add From File List**. See "Autoscreen / Experiment" on page 317 for more detail.

*Edit/Add Experiment/ Add All Files* — This menu item is equivalent to the **Autoscreen/Experiment/Add All Files** menu item. See "Autoscreen / Experiment" on page 317 for more detail.

*Edit/Add Experiment/ Add One* — This menu item is equivalent to **Autoscreen/Experiments/Add One**. See "Autoscreen / Experiment" on page 317 for more detail.

*Edit/Delete Experiment* — This menu item deletes the highlighted experiment(s) in the Autoscreen Experiments Table from the project.

*Edit/Verify Directories* — This menu item is equivalent to **Autoscreen/Experiments/Verify Directories**. See "Autoscreen / Experiment" on page 317 for more detail.

## Processing and scoring menu items

After adding experiments to the project, you can process the control spectrum, pick or import peaks, import assignments, set up scoring parameters, and score the test spectra. The relevant menu items are found on the **Autoscreen** menu and in the Autoscreen Experiments Table. Two menu items in the Peak Displacements Table also allow you to manually edit scoring results.

### Autoscreen/Calibrate Control Peaks

This menu item is used to interactively correct systematic errors of peak locations. To do this, zoom in on a spectrum region so that a peak and a peak label are clearly visible. Then select the menu item. Click the peak label, drag the cursor to the correct location, and release the mouse button. The displacement defined by the mouse movement is added to all control peaks. You can repeat this step until you are satisfied and then press <Esc> to exit the calibration.

### Autoscreen/Import Assignments

This menu item is used to import resonance assignments. The files can be in FELIX spin system, Insight resonance file, or BMRB assignment table format.

### Autoscreen/Setup Scoring

The **Autoscreen/Setup Scoring** menu item is used to set up the scoring parameters and create a score matrix, if necessary, where the results will be saved. A score matrix usually has a filename *score_DB.mat*, where *DB* is the name of the current database. After you perform this action, a new score matrix is generated if it does not exist, if you've added or deleted control peaks, or if you're switching to a new control-peak table.

When you select this menu item, the 2D SCORING PARAMETERS control panel appears, which allows you to define the basic parameters for scoring. If you click the **Advanced** button, you can define more advanced parameters in the ADVANCED PARAMETERS FOR 2D SCORING control panel. The parameters are explained below. For more details. please refer to Chapter 1, *Theory*.

### Basic parameters

Basic parameters, which are controlled via the 2D SCORING PARAMETERS control panel, include:

- **Control Peak Table**: Select **Use One** to use the same set of control peaks for all test spectra, or select **Prompt** so that FELIX prompts for different set for each test spectrum to score. The default, **Use One**, is for high-throughput screens; you have to specify the control peak table entity name. The **Prompt** option is for interactive analysis of titration data, where you can take advantage of the step-wise peak displacements among a titration series for more accurate peak tracing.
- **Test Peaks**: Select **Auto Pick** to allow FELIX to automatically pick the test peaks when scoring a spectrum; or select **Prompt** so that it prompts for a prepared test peak entity and uses it for scoring. When **Auto Pick**, the default option, is selected, you can specify a peak entity name to store the automatically picked test peaks. Note that this peak entity saves the peaks for only one spectrum. If more than one spectra are scored in a batch, only the peaks for the last scored experiment are retained. The name of the peak table to be used as control peaks.
- **Peak Displacement Limits: D1** (or **D2**) **Minimum** is the minimum chemical-shift difference in D1 (or D2) required for a peak to contribute to the score. If the displacements in both dimensions are smaller than these limits, they are ignored. **D1** (or **D2**) **Maximum** is the maximum possible peak displacement expected in D1 (or D2). A test peak is not matched to a control peak if their chemical-shift difference in either dimension is larger than the limit. All values are in ppm.
- **Scale Factors:** The coefficients used to calibrate the chemical-shift displacements in D1 (or D2). Default values are 1.0 and 0.2 for D1 and D2, respectively. If you want to use $^{15}$N chemical shifts only, set D1 to 0 and D2 to 1.

### Advanced parameters

Advanced parameters, which are controlled via the ADVANCED PARAMETERS FOR 2D SCORING control panel, include:

- **Peak Picking Parameters for Test Spectra**: In this section, the **Threshold Method** determines how to set the threshold for peak picking in the test spectra. If the default, **Automatic**, is selected, FELIX automatically adjusts the threshold so that peaks are picked with a reasonable quantity and quality. If **Control** is selected, the same threshold as the one used for the control spec-

trum is used for all test spectra. If **Define** is selected, the value you enter in the associated entry box is used.

- All other parameters in this section are the same as for standard 2D peak picking. The same values as for the control spectrum are recommended for them.
- **Peak Matching:** Parameters in this section determine how test peaks are matched to control peaks during automatic scoring.

  The options **Use Peak Widths** and **Use Peak Heights**, determine whether allow the peak widths and peak heights, respectively, are to be used for calculation of shape similarity between a control peak and a test peak. You can choose not to use them, use them only for peak matching, use them only for scoring, or use them for both peak matching and scoring.

  If peak shape is used, **Minimum Shape Similarity** is a threshold for two peaks to match. This is an extra category for two peaks to match in addition to the **Peak Displacement Limits** in the basic parameters group.

  **Search Methods** provides two alternative algorithms for searching for a best match between the test and control peaks. If **Tree Search**, the default method, is selected, a heuristic depth-first search method is applied. Since this can be time-consuming, you can limit the CPU time spent on each test spectrum by defining a value for **CPU Time Limit** (default value is **10 s**). If **Simulated Annealing** is selected, the stochastic method is applied. The latter is usually fast (so **CPU Time Limit** is not used), yet does not guarantee a best match. The latter method is recommended when the spectra are so complicated that tree searching does not give satisfactory results in a reasonable amount of CPU time.
- **Unmatched Peaks:** Parameters in this section determine how to deal with peaks that do not have a match in either the control or test spectrum.

  For each unmatched control peak, if **Fit to Test** is selected, it is fitted to the test spectrum using the peak-optimization function if the percentage of unmatched control peaks has not exceeded the **Maximum (%)**. (For details please see “Peaks / Optimize”in Chapter 4, *Processing, Visualization, and Analysis Interface (1D/2D/ ND)* If the fitting is successful, the optimized peak is taken as

their matched test peak. Otherwise it remains unmatched, and a **Penalty** (default 0.60) contributes to the score of the experiment

For unmatched test peaks, **Selection** allows you to select the method to define them. If **None** is selected, such peaks are ignored. If **Close to Control Peaks** is selected, only those that are close to at least one control peak, namely with displacements no larger than the **Maximum D1** (or **D2**) **Peak Displacement Limits**, are included. Otherwise, if **All** is selected, all peaks are included. When determining if it is a legitimate test peak, the peak widths and height of a test peak are compared with the statistics for all the matching test peaks. The parameter **Num.of RMSD** (default 2.0) allows you to define a tolerable deviation from the average peak widths and height. The **Penalty** is the contribution of each unmatched test peak to the score of the experiment. The default (0.2) is smaller than that for an unmatched control peak (0.6), because automatic peak picking of test peaks is usually less reliable than picking of control peaks, which is normally done manually where refinement is possible.

## Autoscreen Experiments Table menu items

*Action/Process Control Spectrum*

This menu item allows you to process the highlighted experiment as a control spectrum. The processing parameters are saved and are later used for automatic processing of all other test experiments.

> **Important:** If you've already processed or scored any spectra, you will have to process and score those spectra again.

*Action/Select Control Spectrum*

This menu item allows you to select a processed experiment as the control spectrum. This is useful when you are using processed data to set up an Autoscreen project (hence you do not need to process a control spectrum) or when you want to switch to another control spectrum after processing a control spectrum and some test spectrum. In the latter case, you are warned about losing the current scoring results, if any.

*Action/Process Selected Spectra*

This menu item processes the highlighted experiment(s) as test spectra. The same processing settings as for the control spectrum are used. When a spectrum is successfully processed, its status is updated to 1 in the Autoscreen Experiments Table.

*Action/Score Selected Spectra*

This menu item scores the highlighted experiment(s). If a highlighted experiment is not processed, it processes the experiment before scoring. You must set up scoring parameters before using this menu item. When a spectrum is successfully scored, its status is updated to 2 in the Autoscreen Experiments Table, and the test spectrum and control spectrum are displayed as an overlay of contours. Arrows are displayed over the spectra showing the displacements of peaks. The Peak Displacements Table is updated with scoring information such as the contributions of individual peaks.

*Action/Go*

Using this menu item processes all nonprocessed test experiments and scores all nonscored experiments. If an experiment is already processed or scored, it is not processed or scored again. This menu item is useful when you trial-processed and -scored a few experiments and now want all the remaining experiments to be scored in batch mode.

Upon completing this action, a histogram of scores vs. experiment is displayed.

*Action/Rescore All Spectra*

This menu item scores all processed experiments, whether or not they were previously scored. This menu item is useful when you have changed ROI peaks and want the changes to be reflected in the scores.

Upon completing this action, a histogram of scores vs. experiment is displayed.

*Action/Reprocess/ Rescore All*

This menu item processes and scores all experiments in batch mode, no matter what their status.

Upon completing this action, a histogram of scores vs. experiment is displayed.

For the 1D case the data is processed / reprocessed using the same parameters as the control spectrum. Scoring and histogram generation is not yet supported.

*Updating the displayed tables*

After the **Autoscreen/Setup Scoring** action is successfully completed, a Peak Displacements Table is displayed. This table shows the control peaks and their contributions to the score of a current scored experiment. The ID of the current experiment, if any, is displayed next to the title of the table. This table is automatically updated after you select **Action/Score Selected Spectra**. You can also update it to reflect the scoring of a certain experiment by double-clicking the experiment in the Autoscreen Experiments Table or

by displaying its peak contribution histogram. If the table is closed, you can select **Autoscreen/Show Displacement Table** to open it.

The Peak Displacements Table contains the columns shown in Table 24.

**Table 24 Columns in the Peak Displacement Table**

| Column | Description |
|---|---|
| **id** | Numbering of the control peak in Peaks-xpk:peaks table. It is always used for identifying a control peak. |
| **asg1** | The assignment of the control peak in D1. If not assigned, value is “null”. |
| **asg2** | The assignment of the control peak in D2. If not assigned, value is “null”. |
| **cntrib** | The contribution of the peak to the total score of the experiment. It is usually calculated based on the **shift1**, **shift2** and **weight**. |
| **shift1** | The absolute chemical shift displacement between the matched peak pair in ppm along D1. If no matching test peak, it remains zero. |
| **shift2** | The absolute chemical shift displacement between the matched peak pair in ppm along D2. If no matching test peak, it remains zero. |
| **shape** | The similarity of the shapes of the matched peak pair; 0 = least similar, 1 = identical. If the peak shape is not used for matching, value = 1. |
| **weight** | Weight of contribution of the peak to the total score of the experiment. An ROI (region of interest) peak has weight greater 0. By default all peaks in Peaks-xpk:peaks table are taken as ROI peaks when setting up scoring. You can change the weight of a peak manually from the table. |
| **tstcen1** | The chemical shift of the matching test peak in ppm along D1. If no matching test peak, it remains zero. |
| **tstcen2** | The chemical shift of the matching test peak in ppm along D2. If no matching test peak, it remains zero. |

## Peak Displacements Table menu items

*Edit/Remove Displacement*

This menu item allows you to click a certain control peak to remove its displacement. This peak is then scored as an unmatched control peak. This process can be repeated until you press <Esc> to exit.

This action is equivalent to that of the **Remove Displacement** icon on the tool bar of the same table.

*Edit/Change Displacement*

This menu item allows you to change an existing displacement or add a new one. Click a control peak and drag to a destination test peak. When you release the mouse button, a violet arrow shows the new displacement. You can repeat this process as often as you want to, then press <Esc> key to exit.

A user-defined displacement is scored in the same way as an automatically determined one, except that there are no minimum and maximum limits for the displacement. The destination is not checked either, so it can be anywhere in the spectrum. In addition, peak shapes are not considered when calculating their contributions to the score. All the changes you've made are reported in the text window and updated in the Peak Displacements Table.

> **Note:** A control peak can be matched to only one test peak, which means that this menu item replaces the original displacement, if any.

This action is equivalent to that of the **Remove Displacement** icon on the tool bar of the same table.

## ROI menu items

By default, the displacements of all control peaks are used for scoring. In practice, you may be interested in only a subset of peaks that are particularly informative for the study. This subset, designated region of interest (ROI), can be defined and displayed using several methods offered by the **Autoscreen/Define ROI** pullright menu of the **Edit** menu of the Peak Displacements Table.

As displayed in the Peak Displacements Table, an ROI peak has a weight greater than zero. When a peak is included as ROI, its weight is set to 1.0. When a peak is removed from ROI, its weight is set to zero. In addition to using the following menu items, you can always directly modify the weight of a peak in the Peak Displacement Table.

> **Tip:** ROI peaks can be defined only after setting up scoring. (See "Autoscreen / Setup Scoring" on page 320.) The first time you set up scoring, all peaks are taken as ROI, with weight equal to 1.0. Every time you change the ROIs, you can select **Action/Rescore All** (from the Autoscreen Experiments Table) to rescore all spectra based on the new ROIs.

### Autoscreen/Define ROI/Add One Peak

This menu item displays a cross-hair cursor, allowing you to add ROI peaks one-by-one by clicking each peak. To exit this mode, press <Esc> or click a blank area.

### Autoscreen/Define ROI/Add Displayed Peaks

This menu item includes all displayed peaks as ROI.

### Autoscreen/Define ROI/Add Region

This menu item allows you to drag out a rectangle around a set of peaks to include all peaks in the rectangle as ROI.

### Autoscreen/Define ROI/Add by Residue Numbers

This menu item allows you to include ROI on the basis of their assignments. In the FIND 2D ROI PEAKS BY RESIDUE NUMBERS control panel, enter the residue numbers, as individual numbers, ranges, or a combination of both, separated by commas (for example, **1,3,6-12**).

### Autoscreen/Define ROI/Add by Residue Name

This menu item also allows you to include ROI on the basis of their assignments. In the Find control panel, all assigned peaks are displayed in a list box. You can select one of them and click **OK** to include it as ROI.

### Autoscreen/Define ROI/Remove One Peak

This menu item displays a cross-hair cursor which allows you to remove ROI peaks individually, by clicking a peak at a time. To exit this mode, press <Esc> or click a blank area.

### Autoscreen/Define ROI/Remove Region

This menu item allows you to drag out a rectangle so that all peaks in the rectangle are excluded from ROI.

### Autoscreen/Define ROI/Remove by Residue Number

This menu item allows you to exclude peaks from ROI on the basis of their assignments. In the FIND 2D ROI PEAKS BY RESIDUE NUMBERS control panel, enter the residue numbers as individual numbers, ranges, or a combination of both, separated by commas (for example, **1,3,6-12**).

### Autoscreen/Define ROI/Remove All

This menu item designates all peaks as non-ROI peaks.

### Autoscreen/Define ROI/Draw ROI

This menu item highlights the ROI peaks in yellow by default. You can change the color by using the **Autoscreen/Define Display** menu item.

The ROI peaks are automatically highlighted when you draw a spectrum with an Autoscreen menu item. However, sometimes redrawing the spectrum (for example after you resize the spectral window) turns off the ROI peak highlighting. You can select this menu item again to highlight them.

### Autoscreen/Define ROI/Tile ROIs

This menu item enhances the display of ROI peaks by reserving a separate window for each one. This focuses the display directly on the interesting areas.

#### Peak Displacements Table menu items:

- **Edit/Add To ROI** – This menu item adds the currently highlighted peaks (rows) in the Table to be included to the ROI.
- **Edit/Remove From ROI** – This menu item removes the currently highlighted peaks (rows) in the Table from the ROI.

## Display and print menu items

Autoscreen allows you to display and print spectra in single or overlay mode. You can set many display options and save them along with the project. The relevant menu items are found on the **Auto-**

**screen** menu and in the Autoscreen Experiments and Peak Displacements tables.

## Autoscreen/Save Display and Reference

This menu item saves the following display settings in the project. These are usually changed using the general FELIX menu items, as detailed below:

- **References** is normally changed with the **Preference/Reference** menu item. You must use **Autoscreen/Save Display and Reference** to save the reference.
- **Threshold** is normally changed with the **Preference/Plot Parameters** menu item. (You can also change this setting using the **Autoscreen/Set Display** menu item. If so, you do not need to select the **Autoscreen/Save Limits and Reference** menu item to save the setting.)

These settings are used for display and scoring of all experiments in the project.

> **Caution**: If you make changes to these settings and want to save them, be sure to use the **Autoscreen/Save Limits and Reference** menu item to insure that they are not lost.

## Autoscreen/Setup Display

This menu item allows you to select some frequently used display options in a control panel. You can save the changes in the project by clicking **OK**.

The options include the following for displaying the contours of control and test spectra:

- **Contour Threshold**. For the control spectrum, you can select **Automatic** so that the threshold is automatically calculated before each display. Alternatively, set **Contour Threshold** to define **Define** and manually enter a threshold value in the associated entry box. The value you enter is always used as the threshold for the control spectrum.
- For test spectra, you can select **Automatic** or **Define** as described above, or select **Control** so that the threshold is the same as that used for the control spectrum. If you select **Automatic** and the

spectrum has been scored, the value in the **thresh** column in the Autoscreen Experiments Table is used.

- **Number of Levels**. The maximum number of contour levels to display.
- **Level Multiplier**. A factor used to calculate contours.
- **Negative Levels**. Whether to display negative peaks.
- **Color**. Used for displaying contours.
- **Color Cycle**. Number of colors to cycle while drawing contours. When displaying the overlay contours, the same number of colors is used to display the experiment IDs of both the test and control spectra.

The options also include the following for displaying control peaks:

- **Draw Cross Peaks on Control spectrum**. Whether to draw cross peaks.
- **Crosspeak Symbol**. Set to **Default** (box), **Cross Only**, **Small Cross**, or **Small Box**.
- **Non-ROI Color**. Color for displaying peaks that are not in the region of interest.
- **ROI Color**. Color for displaying peaks that are in the region of interest.
- **Peak Label**. Set to **None**, **Number**, **Assignment**, **Short Assignment**, or **Residue**.
- **Label Size**. Size of peak labels.

**Note:**

- Only the peaks of the control spectrum can be displayed. By default, peaks of a test spectrum are picked on the fly during scoring but not saved in the database.
- As described in “Autoscreen / Project” on page 315, the above parameters are automatically inherited from the previous project if there was one in the same session.

## Autoscreen Experiments Table menu items

*Draw icon*

Clicking this icon displays the highlighted experiment in the Autoscreen Experiments table. If you select multiple experiments, they are displayed in turn. If the control spectrum is included, the picked peaks, if any, are displayed. Otherwise no peaks are displayed. If you select a nonprocessed test spectrum (that is, one with a status less than 1), it is automatically processed before it is displayed.

Many display attributes can be changed. See "Autoscreen / Setup Display" on page 329 for more information.

> **Note:** The control spectrum must be already processed before you use the **Draw** icon.

*Draw Next icon*

The **Draw Next** icon displays the experiment *next to* the currently highlighted one in the Autoscreen Experiments Table. If you select multiple experiments, the one next to the first selected one is displayed.

*Draw Previous icon*

This icon displays the experiment *previous to* the one currently highlighted in the Autoscreen Experiments Table. If you select multiple experiments, the one previous to the first selected one is displayed.

*Overlay icon*

This icon overlays contours (2D case) of the highlighted test experiment on top of those for the control experiment. If you select multiple experiments, the first test experiment is used. If you select only the control experiment, FELIX ignores the menu item. If control peaks and displacement arrows exist, these are also displayed.

For the 1D case this icon displays the highlighted experiment along with the control spectrum for comparison. If you have scored the selected test, this menu item also updates the Peak Displacements Table.

Many display attributes can be changed. See "Autoscreen / Setup Display" on page 329 for more information.

> **Note:** You must process both the control spectrum and selected test spectrum before you click this icon.

*Overlay Next icon*

Clicking this icon overlays the contours (2D case) of the test experiment *next to* the currently highlighted experiment over those of the

control experiment. If you select multiple experiments, the experiment next to the first selected experiment is used.

For the 1D case this icon displays the test experiment after the currently highlighted experiment along with the control spectrum for comparison.

*Overlay Previous icon*

Clicking this icon overlays the contours (2D case) of the test experiment *previous to* the currently-highlighted experiment over those of the control experiment. If you select multiple experiments, the experiment previous to the first selected experiment is used.

For the 1D case this icon displays the test experiment previous to the currently highlighted experiment along with the control spectrum for comparison.

*Overlay Multiple icon*

Clicking this icon overlays the contours (2D case) of the one or more highlighted test experiments over those of the control experiment. If the control experiment is highlighted, it is ignored.

> **Note:** In contrast to **Overlay**, this icon does not display displacement arrows even if the test spectra were scored. It also does not update the Peak Displacements Table.

For the 1D case this icon allows you to overlay different combinations of the selected 1D spectra and to compare the spectra using a real-time adjustment tool. Before clicking on the icon you highlight a group of 1D spectra that you wish to include in the analysis group. You can highlight multiple spectra by clicking the mouse on the item number in the table and dragging the mouse over the group you want to include. Or you can use the Shift key (to select groups of rows) or the Control key (to add individual rows) in conjunction with the mouse to add multiple rows.

When you have highlighted the desired group of experiments and you click on the **Overlay** icon this brings up the 1D Overlay Setup menu. The **Number of Spectra to Display** parameter is the number of spectra or combinations that you wish to have on the screen at one time. You can adjust the **Spectrum Overlap** parameter to determine the degree of overlap (0.0 = no overlap, 1.0 = total overlap). For each of the items in the overlap display you can specify whether they are individual spectra (**Method** = **A**) or a subtraction of two spectra (**Method** = **A-B**). After you set up each of the rows in the overlap display you can click on **Apply** to see the updated display.

For rows that are a subtraction of two spectra you can choose to do a real-time adjustment of the combination by clicking **Adjust**. This brings up the Real-Time Subtraction menu. You can adjust the intensity of either the A or B spectrum along with the point shift between the two. You can use this tool to examine differences between the various spectra.

*Peak Contribution Histogram icon*

For each highlighted test spectrum, this icon displays the histogram of contributions vs. peaks if it is scored. This menu item also updates the Peak Displacement Table.

In the PEAK CONTRIBUTION HISTOGRAM OPTIONS control panel, the **X Coordinates** can be set to **Peak IDs** or **Residue Numbers**. If **Peak IDs** is selected, the item numbers of the peaks in the Peaks Table are used as the *X* coordinates. Otherwise the numbering of residues assigned to the peaks is used and unassigned peaks are ignored.

The five choices for **Y Coordinates** are:

- **Contributions:** Use the contributions of all control peaks (displayed in the **contrb** column in the Peak Displacements Table).
- **D1** (or **D2**) **Displacements:** Use the absolute chemical-shift displacements in the D1 (or D2) dimension (displayed in the **shift1** (or **shift2**) column in the Peak Displacements Table).
- **D1 + D2 Displacements:** Use the sum of the absolute chemical shifts in D1 and D2.
- **Shape Similarity:** Use the similarities of peak shapes (displayed in the **shape** column in the Peak Displacements Table).

## Peak Displacements Table menu items

*Edit/Sort Contributions*

This menu item sorts the peaks in *descending* order of their contribution to the score. This action is equivalent to that of the **Sort Contributions** icon on the toolbar in the same table.

*Edit/Undo Sort Contributions*

This menu item sorts the peaks in *ascending* order of their item number. This action is equivalent to that of the **Undo Sort Contributions** icon on the toolbar in the same table.

*View/Zoom on Peaks*

This menu item zooms the overlay spectrum display into highlighted peak(s) in the Peak Displacements Table, providing a closer view of the displacement of peak(s) that you are interested in.

This action is equivalent to that of the **Zoom on Peaks** icon on the toolbar of the same table. If you are interested in one particular peak, you can simply double-click that row to zoom in on it.

*View/Zoom Next*

This menu item zooms the overlay spectrum display into the peak *next to* the currently highlighted one in the Peak Displacements Table. If you highlight multiple peaks, only the first one is considered.

This action is equivalent to that of the **Zoom Next** icon on the toolbar of the same table.

*View/Zoom Previous*

This menu item zooms the overlay spectral display into the peak *previous to* the currently highlighted one in the Peak Displacements Table. If you highlight multiple peaks, only the first one is considered.

This action is equivalent to that of the **Zoom Previous** icon on the toolbar of the same table.

*View/Locate in Peak Table*

This menu item highlights, in the Peaks table, the corresponding peaks of the highlighted peaks in the Peak Displacements table, providing a straightforward way to locate control peaks in the Peaks table.

This action is equivalent to that of the **Locate in Peak Table** icon on the toolbar of the same table.

## Menu items for presenting results

The scoring results can be presented in various ways for visualization and export, using menu items on the **Autoscreen** menu and on the Peak Displacement Table, as described below.

### Autoscreen/Show Displacements Table

This menu item shows the Peak Displacements Table, which displays the scoring information of the current test spectrum. This table is automatically displayed when you set up scoring parameters and is updated when you display the overlay or score histogram of a test spectrum.

## Autoscreen/Display Scores vs. Experiments

This menu item displays a histogram of scores vs. the experiments. It is useful for visually identifying highly displaced experiments, which usually correspond to high-affinity ligands.

## Autoscreen/View Clusters

This menu item groups experiments that share common displaced peaks, providing a way to identify individual binding subsites in a protein. The VIEW CLUSTERS control panel allows you to define a threshold, so that peaks with displacements smaller than that value are ignored. Some clustering details are displayed in the text window.

> **Tip:** The experiment numbers and peak numbers are reshuffled, so you must use the crosshair cursor (automatically displayed after selecting **Autoscreen/View Cluster)** to identify the peaks and experiments in the clusters. Press <Esc> when you are done. If you want to return to the crosshair cursor after pressing <Esc>, select **Autoscreen/View Clusters** again.

## Autoscreen/Export Score

This menu item opens the EXPORT ACTIVITY control panel, allowing you to export the scoring results in several formats:

- Select **All Scores** for **Contents** and either **Tab** or **Space** for **Delimiter**. Enter a filename under **Selection** (the .dat suffix is automatically added to the filename if you do not provide it). Click **OK**, and all the experiments and their scores are listed in the file in the same order as that of the experiments in the Autoscreen Experiments Table.
- Similar to the previous bullet item, except select **All Scores Sorted** for **Contents**. All experiments and scores are listed in descending order of scores.
- Similar to the first bullet item, except select **Scores and Titration** for **Contents**. The scores of the top *X* experiments, in descending order of score, and the contributions of the top *Y* peaks that have the greatest sum of contributions to the *X* experiments, are listed in the file. The values of *X* and *Y* can be set as **Number of Experiments** and **Number of Peaks**, respectively. This function

gives a summary of “interesting peaks in interesting experiments.”

- Similar to the first bullet item, except select **C2 QSAR Table** for **Contents**. All experiments and scores are listed in a format suitable for QSAR study with the Cerius$^2$ program.
- For calculation of $K_d$, first highlight the interesting experiments in the Autoscreen Experiments Table, then highlight the interesting peaks in the Peak Displacements Table. Set **Contents** to **Titration Selected**. Toggle **Use Comments as Concentration** to **on** if you’ve entered the ligand concentrations in the comment column in the Autoscreen Experiments Table. Select **Tab** or **Space** for **Delimiter**. Click **OK**, and the contributions of the selected peaks to the selected experiments are exported along with the concentration data, if any.
- For calculation of $K_d$, for all peaks set **Contents** to **Titration All Peaks**. Make sure the concentrations are entered in the comment column of the Autoscreen Experiment Table. Select **Tab** or **Space** for **Delimiter**. Click **OK**, and the contributions of all peaks to the experiments are exported with a valid concentration along with the concentration data.

**Tip:** You can <Shift>-click to select multiple consecutive rows in a table or <Ctrl>-click to select noncontiguous rows. Note that the order in which you select the rows is used when exporting multiple rows to the file.

## Autoscreen Experiments Table menu items

*Action/Color Score*

This menu item exports the contributions of the assigned peaks to a text file that can be used by Insight II or Cerius$^2$ to color the residues in the protein, providing a way to visualize the subsites that have close contacts with the ligand.

**Note:** Please refer to Chapter 5, Using Autoscreen in the FELIX *Tutorials* for an example of coloring scores using Insight II.

## Peaks Displacements Table menu items

*View/Titration*

This menu item plots a histogram of contributions vs. experiment for the highlighted peak in the Peak Displacements Table, providing a way to see the affinity of different ligands to a particular residue.

This action is equivalent to that of the **Titration** icon on the toolbar of the same table.

*Locate in Peak Table icon*

This icon highlights the corresponding control peak in the Peaks Table for the one currently highlighted in the Peak Displacements Table. This facilitates viewing the details of a selected control peak.

# A References


Akke, M.; Skelton, N. J.; Kordel, J.; Palmer III, A. G.; Chazin, W. J. "Effects of ion binding on the backbone dynamics of calbindin $D_{9k}$ determined by $^{15}N$ NMR relaxation" *Biochemistry* **32**, 9832–9844 (1993).

Anglister, J.; Grzesiek, S.; Wang, A. C.; Ren, H.; Klee, C. B.; Bax, A. "$^{1}H$, $^{13}C$, $^{15}N$ nuclear magnetic resonance backbone assignments and secondary structure of human calcineurin B" *Biochemistry* **33**, 3540–3547 (1994).

Banks, K. M.; Hare, D. R.; Reid, B. R. "Three-dimensional solution structure of a dna duplex containing the *Bcl*I restriction sequence: Two-dimensional NMR studies, distance geometry calculations, and refinement by back-calculation of the NOESY spectrum" *Biochemistry* **28**, 6996 (1989).

Barkhuijsen, H.; de Beer, R.; Bouee, W. M. M. J.; van Ormondt, D. "Retrieval of frequencies, amplitudes, damping factors, and phases from time-domain signals using a linear least-squares procedure" *J. Magn. Reson.* **61,** 465 (1985).

Billeter, M.; Braun, W.; Wuthrich, K. "Sequential resonance assignments in protein $^{1}H$ nuclear magnetic resonance spectra: Computation of sterically allowed proton–proton distances and statistical analysis of proton–proton distances in single crystal protein conformations" *J. Mol. Biol.* **155**, 321–326 (1982).

Blake, P. R.; Park, J. B.; Bryant, F. O.; Aono, S.; Magnuson, J. K.; Eccleston, E.; Howard, J. B.; Summers, M. F.; Adams, M. W. "Determinants of protein hyperthemostability: Purification and amino acid sequence of rubredoxin from the hyperthermophilic archaebacterium *Pyrococcus furiosis* and secondary structure of the zinc adduct by NMR" *Biochemistry* **30**, 10885 (1991).

Boucher, W.; Laue, E. D.; Campbell–Burk, S. L.; Domaille, P. J. "Improved 4D NMR experiments for the assignment of back-

bone nuclei in $^{13}C/^{15}N$ labelled proteins" *J. Biomol. NMR* **2**, 631–637 (1992).

Campbell–Burk, S. L.; Domaille, P. J.; Starovasnik, M. A.; Boucher, W.; Laue, E. D. "Sequential assignment of the backbone nuclei ($^{1}H$, $^{15}N$ and $^{13}C$) of c-H-*ras* p21 (1-166) GDP using a novel 4D NMR strategy" *J. Biomol. NMR* **2,** 639–646 (1992).

Chylla, R. A.; Markley, J. L. *J. Mag. Reson. Series B* **102**, 148–154 (1993).

Clubb, R. T.; Ferguson, S. B.; Walsh, C. T.; Wagner, G. "Three-dimensional solution structure of *Escherichia coli* periplasmic cyclophilin" *Biochemistry* **33**, 2761–2772 (1994).

Constatine, K. L.; Goldfarb, V.; Wittekind, M.; Friedrichs, M. S.; Anthony, J.; Ng, S.-C.; Mueller, L. "Aliphatic $^{1}H$ and $^{13}C$ resonance assignments for the 26-10 antibody $V_L$ domain derived from heteronuclear multidimensional NMR Spectroscopy" *J. Biomol. NMR* **3**, 41–54 (1993).

Dietrich, W.; Rudel, C. H.; Neumann, M. "Fast and precise automatic baseline correction of one- and two-dimensional NMR Spectra" *J. Magn. Reson.* **91**, 1–11 (1991).

Dzakula, Z. "Phase angle measurement from peak areas (PAMPAS)" *J. Magn. Reson.*146, 20-32 (2000).

Eccles, C.; Guntert, P.; Billeter, M.; Wuthrich, K. "Efficient analysis of protein 2D NMR spectra using the software package EASY" *J. Biomol. NMR* **1**, 111 (1991).

Ernst, R. R. "Numerical Hilbert transform and automatic phase correction in magnetic resonance spectroscopy." *J. Magn. Reson.* **1**, 7–26 (1969).

Farmer II, B. T.; Müller, L.; Nikonowicz, E. P.; Pardi, A. "Unambiguous resonance assignments in $^{13}C$,$^{15}N$-labeled nucleic acids by 3D triple-resonance NMR" *J. Amer. Chem. Soc.* **115**, 1040–1041 (1993).

Freund, C.; Ross, A.; Plückthun, A.; Holak, T. "Structural and dynamic properties of the $F_v$ fragment and the single-chain $F_v$ fragment of an antibody in solution investigated by heteronuclear three-dimensional NMR spectroscopy" *Biochemistry* **33**, 3296–3303 (1994).

Gladden, L. F.; Carpenter, T. A.; Klinowski, J.; Elliot, S. R. “Quantitative interpretation of exponentially broadened solid-state NMR signals” *J. Magn. Reson.* **66**, 93–104 (1986).

Griesinger, C.; Sørenson, O. W.; Ernst, R. R. *J. Chem. Phys.* **85**, 6837 (1986).

Gross, K. H.; Kalbitzer, H. R. “Distribution of chemical shifts in $^{1}H$ NMR spectra of proteins” *J. Magn. Reson.* **76**, 87–99 (1988).

Grzesiek, S.; Bax, A. “An efficient method for sequential backbone assignment of medium-sized isotopically enriched proteins” *J. Magn. Reson.* **99**, 201–207 (1992).

Grzesiek, S.; Bax, A. “Amino acid type determination in the sequential assignment procedure of uniformly $^{13}C/^{15}N$-enriched proteins” *J. Biomol. NMR* **3**, 185–204 (1993).

Grzesiek, S.; Döbeli, H.; Gentz, R.; Garotta, G.; Labhardt, A. M.; Bax, A. “$^{1}H$, $^{13}C$, and $^{15}N$ NMR backbone assignments and secondary structure of human interferon-γ” *Biochemistry* **31**, 8180–8190 (1992).

Guntert, P.; Wuthrich, K. “FLATT: A new procedure for high-quality baseline correction of multidimensional NMR spectra” *J. Magn. Reson.* **96**, 403–407 (1992).

Hajduk, P.; Meadows, R.; Fesik S. “Discovering high-affinity ligands for proteins” *Science*, **278**, 497–499 (1997).

Hamming, R. W. *Digital Filters*, Prentice Hall: New Jersey, 3rd Ed., (1989).

Hansen, A. P.; Petros, A. M.; Meadows, R. P.; Nettesheim, D. G.; Mazar, A. P.; Olejniczak, E. T.; Xu, R. X.; Pederson, T. M.; Henkin, J.; Fesik, S. W. “Solution structure of the amino-terminal fragment of urokinase-type plasminogen activator” *Biochemistry* **33**, 4847–4864 (1994).

Heuer, A. “A new algorithm for automatic phase correction by symmetrizing lines” *J. Magn. Reson.* **91**, 241–253 (1991).

Kleywegt, G. J. “Computer-assisted assignment of 2D and 3D NMR spectra of proteins”, *Ph.D. Thesis*, University of Utrecht, Utrecht, The Netherlands (1991).

Kleywegt, G. J.; Boelens, R.; Cox, M.; Linas, M.; Kaptein, R. “Computer-assisted assignment of 2D $^{1}H$ NMR spectra of proteins:

Basic algorithms and applications to phoratoxin B." *J. Biomol. NMR* **1**, 23–47, (1991).

Kleywegt, G. J.; Lamerichs, R. M. J. N.; Boelens, R.; Kaptein, R. "Toward automatic assignment of protein $^{1}$H NMR spectra" *J. Magn. Reson.* **85**, 186–197 (1989).

Kleywegt, G. J., Vuister, G. W.; Padilla, A.; Knegtel, R. M. A.; Boelens, R.; Kaptein, R. "Computer-assisted assignment of homonuclear 3D NMR spectra of proteins. Application to pike parvalbumin III" *J. Magn. Reson.* **B102**, 166–176 (1993).

Kumaresan, R.; Tufts, D. W. "Estimating the parameters of exponentially damped sinusoids and pole-zero modeling in noise" *IEEE Trans. ASSP-30* **6**, 833 (1982).

Legault, P.; Farmer II, B. T.; Müller, L.; Pardi, A. "Through-bond correlation of adenine protons in a $^{13}$C-labeled ribozyme" *J. Amer. Chem. Soc.* **116**, 2203–2204 (1994).

Makhoul, J. "Linear prediction: A tutorial review" in *Modern Spectrum Analysis*, Childers, D. G., Ed., Wiley: New York (1978).

Mandel, A. M.; Akke, M.; Palmer III, A. G. "Backbone dynamics of *Escherichia coli* ribonuclease H1: Correlations with structure and function in an active enzyme" *J. Mol. Biol.* **246**, 144–163 (1995).

Marion, D.; Ikura, M.; Bax, A. "Improved solvent suppression in one- and two- dimensional NMR spectra by convolution of time-domain data" *J. Magn. Reson.* **84**, 425–430 (1989).

Marino, J. P.; Prestegard, J. H.; Crothers, D. M. "Correlation of adenine H2 / H8 resonances in uniformly $^{13}$C labeled RNAs by 2D HCCH–TOCSY: A new tool for $^{1}$H assignment" *J. Amer. Chem. Soc.* **116**, 2205–2206 (1994).

Medvedeva, S.; Simorre, J.-P.; Brutscher, B.; Guerlesquin, F.; Marion, D. "Extensive $^{1}$H NMR resonance assignment of proteins using natural abundance gradient-enhanced $^{13}$C–$^{1}$H correlation spectroscopy" *FEBS Lett.* **333**, 251–256 (1993).

Morelle, N.; Simorre, J.-P.; Marion, D. "Computer-assisted assignment of *Chlorobium limicola* ferrocytochrome C555 $^{1}$H NMR spectra using simulated annealing", poster communication, 35th Experimental NMR Conference, Pacific Grove, CA, p. 135 (1994).

Morelle, N.; Brutscher, B.; Simorre, J.-P.; Marion, D. “Computer assignment of the backbone resonances of labelled proteins using two-dimensional correlation experiments” *J. Biomol. NMR* **5**, 154 (1994).

Otting, G.; Widmer, H.; Wagner, G.; Wuthrich, K. “Origin of $t_1$ and $t_2$ ridges in 2D NMR spectra and procedures for suppression” *J. Magn. Reson.* **66**, 187–193 (1990).

Redfield, C.; Smith, L. J.; Boyd, J.; Lawrence, G. M. P.; Edwards, R. G.; Gershater, C. J.; Smith, R. A. G.; Dobson, C. M. “Analysis of the solution structure of human interleukin-4 determined by heteronuclear three-dimensional nuclear magnetic resonance techniques” *J. Mol. Biol.* **238**, 23–41 (1994).

Rexroth, A.; Szalma, S.; Weisemann, R.; Bermel, W.; Schwalbe, H.; Griesenger, C. “Determination of $^3J(H_{Ni},C'_i)$-coupling constants in proteins with the C′-FIDS method” *J. Biomol. NMR* **6**, 237 (1995a).

Rexroth, A.; Schmidt, P.; Szalma, S.; Geppert, T.; Schwalbe, H.; Griesenger, C. “New principle for the determination of coupling constants that largely suppresses differential relaxation effects” *J. Amer. Chem. Soc.* **177**, 10389 (1995).

Schwalbe, H.; Samstag, W.; Engels, J. W.; Griesinger, C.; *J. Biomol. NMR* **3**, 479–486 (1993).

Shuker, S.; Hajduk, P.; Meadows, R.; Fesik S. “Discovering high-affinity ligands for proteins: SAR by NMR” *Science*, **274**, 1531–1534 (1996).

Skelton, N. J.; Palmer III, A. G.; Akke, M.; Kordel, J.; Rance, M.; Chazin, W. J. “Practical aspects of two-dimensional proton-detected $^{15}N$ spin relaxation measurements” *J. Magn. Reson.* **B102**, 253–264 (1993).

Sklenár, V.; Peterson, R. D.; Rejante, M. R.; Wang, E.; Feigon, J. “Two-dimensional triple-resonance HCNCH experiment for direct correlation of ribose H1′ and base H8, H6 protons in $^{13}C,^{15}N$-labeled RNA oligonucleotides” *J. Amer. Chem. Soc.* **115**, 12181–12182 (1993).

Wand, A. J.; Nelson, S. J. “Refinement of the main chain directed assignment strategy for the analysis of $^1H$ NMR spectra of proteins” *Biophys J.* **59**, 1101–1112 (1991).

Wittekind, M.; Gosbach, M.; Friedrichs, M.; Dreyfuss, G.; Mueller, L. "$^{1}H$, $^{13}C$, and $^{15}N$ NMR assignments and global folding patterns of the RNA-binding domain of the human hnRNP C proteins." *Biochemistry* **31**, 6254–6265 (1992).

Wittekind, M.; Metzler, W. F.; Mueller, L. "Selective correlations of amide groups for glycine alpha protons in proteins." *J. Magn. Reson.* **B101**, 214–217 (1993).

Wuthrich, K. *NMR of Proteins and Nucleic Acids*, Wiley Interscience: New York (1986).

Zhu, G.; Bax, A. "Improved linear prediction for truncated signals of known phase." *J. Magn. Reson.* **90**, 405–410 (1990).

Zhu, G.; Bax, A. "Improved linear prediction of damped NMR signals using modified 'forward-backward' linear prediction." *J. Magn. Reson.* **100**, 202–207 (1992).

Zolnai,. Z.; Macura, S., Markley, J. L. "Phasing two- and three-dimensional NMR spectra by use of the Hilbert transform can save computer time and space." *J. Magn. Reson.* **89**, 94–101 (1990).

# B Keyboard Shortcuts and Accelerator Keys

Most menu items have *keyboard shortcuts*, which you can use when or if you prefer not to access them with the mouse. First access the dropdown menu by pressing the <Alt> key, and then pressing a key as the underscored letter in the menu name you want to access. Next select a menu item from the dropdown menu by pressing the hot key again as the underscored letter in the item name. For example, press <Alt+f> for the **File** menu. Then press o to access the **Open** item. (In all FELIX documents, this is abbreviated as <Alt>+fo.)

You can figure out the other shortcut keys from the underscores in the menu names and menu items, so there is no need to provide an exhaustive list in this chapter.

Some menu items have *accelerators*, which usually consist of the <Ctrl> key and a letter (as shown in the tables below).

Accelerators are not based on the menu hierarchy, so they are not as obvious, but they are more convenient, since they are accessed by a single keystroke, e.g., <Ctrl>+o to open a file.

## Accelerator Keys

The following accelerator keys are accessed by pressing the <Ctrl> key and another letter simultaneously. Please note that they are effective only when a spectral frame (rather than a table frame) is active.

<Ctrl+o> File / Open

<Ctrl+p> View / Plot

<Ctrl+1> View / Plottype / 1D

<Ctrl+i> View / Plottype / Intensity

<Ctrl+c>View / Plottype / Contour

<Ctrl+s> View / Limits / Set Limits

<Ctrl+f> View / Limits / Full Limits

<Ctrl+k>View / Draw Peaks

<Ctrl+l> Peak / List

## Pulldown-independent functions

The following functions are accessed by pressing only the key indicated (it is not necessary to press the <Alt> or <Ctrl> key). Note the numbers are keypad numbers (not from the keyboard), and they are effective only when a spectral frame (instead of a table frame) is active.

*If the active display is an ND spectrum*

| Command | Keystroke |
|---|---|
| Pan left and down | 1 |
| Pan down | 2 |
| Pan right and down | 3 |
| Pan left | 4 |
| Grab and drag | 5 |
| Pan right | 6 |
| Pan left and up | 7 |
| Pan up | 8 |
| Pan right and up | 9 |
| Step one plane up *(3rd dimension in 3D or 4D)* | <up arrow> |
| Step one plane down *(3rd dimension in 3D or 4D)* | <down arrow> |
| Step one plane up *(4th dimension in 4D)* | <right arrow> |
| Step one plane down *(4th dimension in 4D)* | <left arrow> |
| Select new plane *(if frame connection is on and the primary frame is the active frame)* | . |
| Zoom in *(half limits)* | - |
| Zoom out *(double limits)* | – |

*If the active display is a 1D slice*

| Command | Keystroke |
|---|---|
| Pan left | 4 |
| Pan right | 6 |
| increase intensity | 8 |
| decrease intensity | 2 |
| half limits | - |
| double limits | – |

.

# C FELIX Startup

Once installation is complete, FELIX 2002 is ready to use. You can start FELIX by either double clicking the FELIX 2002 alias icon on your desktop, or by selecting **Start/Programs/Accelrys Felix 2002/ Felix 2002**.

> **Note:** If FELIX complains about invalid licensing, you have to install the licensing pack by selecting **Start/Programs/ Accelrys License Pack/License Utilities**, or you have to install a demo license by selecting **Start/Programs/Accelrys Felix 2002/Demo License Installer**. Please ask your system administrator for details.

## The felixrc.ini file

On startup, FELIX 2002 looks for the *felixrc.ini* file in the installation directory. This file defines paths for FELIX to search to find macros, menus, data, and other files. You can always edit the felixrc.ini file to modify the individual directory paths.

The line numbers shown below do *not* appear in the felixrc.ini file, but are used here for reference. A line starting with # is ignored by FELIX.

### The felixrc.ini file format

```
1  #
2  # Default user paths
3  #
4  env 'ACCELRYS_FELIX' biosym
5  def annpfx ./
6  def biopfx ./
7  def datpfx ./
8  def dbapfx ./
```

```
 9  def etcpfx ./
10  def ezppfx ./
11  def limpfx ./
12  def matpfx ./
13  def objpfx ./
14  def parpfx ./
15  def txtpfx ./
16  def xyzpfx ./
17  def conpfx ./
18  def forpfx ./
19  def umacpfx ./
20  def menus menus
21  def mnumod 1
22  #
23  # User local macro and menu directories go here
24  #
25  def macpfx &biosym/macros/felix/&menus/mac/
26  def macpf1 &biosym/macros/felix/&menus/model/mac/
27  def macpf2 &biosym/macros/felix/&menus/assign/mac/
28  def macpf3 &biosym/macros/felix/&menus/ht/mac/
29  def macpf4 &umacpfx
30  def nmacpfx 4
31  def mnupfx &biosym/macros/felix/&menus/mnu/
32  def mnupf1 &biosym/macros/felix/&menus/model/mnu/
33  def mnupf2 &biosym/macros/felix/&menus/assign/mnu/
34  def mnupf3 &biosym/macros/felix/&menus/ht/mnu/
35  def motpfx &biosym/macros/felix/&menus/mot/
36  def icopfx &biosym/macros/felix/&menus/ico/
37  def msgpfx &biosym/data/felix/msgs/
38  def schpfx &biosym/database/felix/schema/
39  def filpfx &biosym/exe/filters/
40  def asglib &biosym/data/felix/asglib/
41  #
42  # System paths go here
43  # CHANGE ONLY AT YOUR OWN RISK
44  #
45  SYSTEM MACROS &biosym/macros/felix/&menus/mac/
46  SYSTEM MACROS &biosym/macros/felix/&menus/assign/mac/
47  SYSTEM MACROS &biosym/macros/felix/&menus/model/mac/
48  SYSTEM MACROS &biosym/macros/felix/&menus/ht/mac/
49  SYSTEM MENUS &biosym/macros/felix/&menus/mnu/
50  SYSTEM MENUS &biosym/macros/felix/&menus/assign/mnu/
51  SYSTEM MENUS &biosym/macros/felix/&menus/model/mnu/
52  SYSTEM MENUS &biosym/macros/felix/&menus/ht/mnu/
53  SYSTEM MOTIF &biosym/macros/felix/&menus/mot/
54  SYSTEM ICONS &biosym/macros/felix/&menus/ico/
55  SYSTEM MSGS &biosym/data/felix/msgs/
56  SYSTEM SCHEMA &biosym/database/felix/schema/
```

```
57  SYSTEM FILTERS &biosym/exe/filters/
58  SYSTEM ASGLIB &biosym/data/felix/asglib/
59  #
```

**Line 4:**

The FELIX command **env** maps the environment variable ACCELRYS_FELIX to the user symbol **biosym**. This symbol is used to set the paths for the system files which are macros, menus, msgs, and schema.

**Lines 5–19:**

The default paths for searching for the files you create (e.g., text files and data files) are set here. You can change this to whatever paths you would like FELIX to search for particular files. The example shown above let FELIX search all the files you create in the current working directory, which is usually the directory where you create the database file when you start up FELIX.

**Lines 20-21:**

These lines are obsolete..

**Lines 25–40:**

The paths for your local macros, menus, and other files are set here. By default, these paths are set to the system (or installation paths). If you want to modify a macro for your use, you need to place this macro in a directory (e.g., mymacros) and set the macpfx symbol to point to this directory. By default, the following strategy is used by FELIX to search for files (e.g., macros and menus):

1. Look in the user's current working directory.

2. Look in the directories pointed to by the **macpf***, **mnupf***, and similar prefixes.

When the specified file is found, the search ends and that file is used. This gives you more flexibility in editing and changing macros for your own use, while not affecting other users of FELIX.

You may define up to 10 directories to be searched for each file type. Keep in mind, though, that execution speed may be slowed by long search paths.

**Lines 44–58:**

These lines are obsolete.

## The initialization macro

The init.mac file is the FELIX macro file that is executed when FELIX starts. It determines every detail of how FELIX starts. FELIX uses a two step search to find the init.mac file. FELIX first looks in the **Start In** folder for the init.mac file. If it is not found there, FELIX next searches the paths defined by the **macpf1**, **macpf2**, ... symbols in your felixrc.ini file (see previous section).

**Warning:** FELIX cannot run without the init.mac file.

The init.mac file has the following commands that you may want to customize:

```
>   def verify 0
```

This suppresses the macro debug mode. You can change the value to 1 or 2 to show macro file names and macro contents for debugging.

```
>   def blkwht 0
```

This sets the background color of the spectral frames to be black. If you set it to 1, the background color will be white.

```
>   cfg 32768 6
```

This sets six buffers for 1D data swapping, and each buffer as 32768 complex points.

```
>   def maxdim 4
```

Initializes the maximum number of dimensions that FELIX will be able to handle.

# D Data Files

## File types

FELIX 2002 reads and writes datafiles of several different types. Each file type is identified by a three-letter suffix (or *extension*), preceded by a dot. Some types of user-generated files do not require prefixes (e.g., 3D object files). File names may contain a maximum of 32 characters. The standard default FELIX file types and their extensions are:

| Extension | File type |
|---|---|
| .dat | data files |
| .mat | matrix files |
| .mac | macro files |
| .mnu | menu files |
| .msg | message files |
| .dba | database files |
| .sch | schema files |
| .ann | annotation files |

Additionally, the .pks and .ppm file formats, used by FELIX and by other Accelrys products, are described in the *Common File Formats* book available here:

http://www.accelrys.com/doc/life/insight2K/formats980/Files980TOC.doc.html

Although the default file extensions are convenient, they are not mandatory. Commands that access only specific types of files assume the default extension unless a nonstandard extension is specified. For example, the read command (**re** or **rn**) is usually used to read unprocessed data files that have .dat default extension. However, FELIX will read datafiles with nonstandard extensions, so long as the file extensions are explicitly stated:

```
> re sample.ser

> re sample.001
```

The default extensions merely help you to organize your files conveniently, to avoid confusion.

## File format for ASCII data files

In addition to the old and new 1D-data file formats, there is also an ASCII datafile format. Historically, there was no way to move data files between computers with different binary data formats. The original FELIX data files (old format) could be read only by the same type of computer that had written them. This is because these data files contained binary representations of the data. The use of ASCII-formatted datafiles was the only way to move data to a different type of computer.

The new format for FELIX datafiles eliminates this problem by using a machine-independent data representation. Now, any computer that uses ANSI standards for its binary data can read and write FELIX 1D datafiles, and the need for ASCII formatted data files has been greatly reduced.

However, for completeness, and for those rare situations where no other method is appropriate, the ASCII data format is still useful. The format is described below.

This data format can contain only a single 1D data spectrum. To hold a complete multidimensional experiment, you need multiple ASCII files, one for each FID.

### ASCII data file example

An example, with explicit line numbers, is shown. The line numbers are for reference only; they are *not* part of the file.

```
1             params        16
2             2048,    0.20000000E+04
3             1,       0.50000000E+03
4             0,       0.00000000E+00
5             1,       0.00000000E+00
```

```
6              0,      0.00000000E+00
7              0,      0.10020406E+02
8              0,     -0.23724947E+02
9              0,      0.00000000E+00
10             0,      0.00000000E+00
11             0,      0.00000000E+00
12             0,      0.00000000E+00
13             0,      0.00000000E+00
14             0,      0.00000000E+00
15             0,      0.00000000E+00
16             0,      0.00000000E+00
17             0,      0.00000000E+00
18            data          2048
19     0.29346375E+05 0.81563688E+05 0.29839797E+05
0.82501023E+05
20     0.29120594E+05 0.84976672E+05 0.24982078E+05
0.86691141E+05
21     0.19801203E+05 0.84221766E+05 0.17831391E+05
0.79039844E+05
22     0.18796516E+05 0.75148625E+05 0.19913953E+05
0.72823125E+05
23     0.21470313E+05 0.70696391E+05 0.24038313E+05
0.70447773E+05
24     0.24443797E+05 0.72803344E+05 0.21105609E+05
0.73622492E+05
. . .
```

## ASCII data file format description

### Line 1:

This is the parameter header line, which contains the keyword params and the number of parameter data lines. The format is:

```
params, i8
```

### Lines 2–17:

These are the data parameters. These lines all contain one integer and one real number. Many of these parameters are unused. The essential parameters are named explicitly. For parameters that are equivalent to a FELIX reserved symbol, the symbol name is given in parentheses. The format is:

```
1x,i15,2x,e15.8
```

**Line 2:**

This line contains the number of data points (**datsiz**) and the spectrum width (**swidth**).

**Line 3:**

This line contains the data type (**datype**) and the spectrometer frequency (**sfreq**).

**Line 4:**

The first datafield in this line is unused, the second is the reference shift (**refsh**).

**Line 5:**

This line contains the axis type (**axtype**) and the reference point (**refpt**).

**Line 6:**

This line is unused.

**Line 7:**

The first datafield in this line is unused, while the second is the zero-order phase (**phase0**).

**Line 8:**

The first datafield in this line is unused, while the second contains the first-order phase (**phase1**).

**Lines 9–17:**

These lines are unused.

**Note:** The reference shift and point symbols are only read in when the axis type is non-zero. The phases are only read in when they are non-zero.

**Line 18:**

This is the data header line, which contains the keyword “data” and the number of datapoints. The format is:

```
data',2x,i8)
```

**Lines 19–24:**

The remaining lines contain datapoint values. The format of all these lines is identical. The identity of each value depends on the data type. When the data are complex, the real-number portion precedes the imaginary-number portion for each datapoint. The format is:

```
1x,4e15.8
```

Sample data segment for real-number data:

```
Real_1 Real_2 Real_3 Real_4
Real_5 Real_6 Real_7 Real_8
...
```

Sample data segment for complex-number data:

```
Real_1 Imag_1 Real_2 Imag_2
Real_3 Imag_3 Real_4 Imag_4
Real_5 Imag_5 Real_6 Imag_6
Real_7 Imag_7 Real_8 Imag_8
...
```

For more detailed information on reading files, see the *Felix Command Language Reference*, which is available here:

http://www.accelrys.com/doc/life/insight2K/felix/cref/CrefTOC.doc.html

.

# Index

## A



## B



## C

## D



## E



## F



## G

## H



## I



## K



## L



## M

## N



## O



## P



## R



## S

## T



## U



## V



## W



## X



## Z

.