

NMRView: A computer program for the visualization and analysis of NMR data*

Bruce A. Johnson^{a,**} and Richard A. Blevins^b

*Departments of ^aBiophysical Chemistry and ^bMolecular Systems, Merck Research Laboratories, RY80Y-103,
P.O. Box 2000, Rahway, NJ 07065, U.S.A.*

Received 2 February 1994

Accepted 29 March 1994

Keywords: Proteins; Resonance assignments; Data analysis; Software; Structure determination

SUMMARY

NMRView is a computer program designed for the visualization and analysis of NMR data. It allows the user to interact with a practically unlimited number of 2D, 3D and 4D NMR data files. Any number of spectral windows can be displayed on the screen in any size and location. Automatic peak picking and facilitated peak analysis features are included to aid in the assignment of complex NMR spectra. NMRView provides structure analysis features and data transfer to and from structure generation programs, allowing for a tight coupling between spectral analysis and structure generation. Visual correlation between structures and spectra can be done with the Molecular Data Viewer, a molecular graphics program with bidirectional communication to NMRView. The user interface can be customized and a command language is provided to allow for the automation of various tasks.

INTRODUCTION

One of the greatest challenges to the modern NMR spectroscopist is to visualize, analyze and manage the great variety of information that is obtainable through NMR techniques. The available information has increased dramatically in the past few years, as a result of the rapid development of new pulse sequences. The increased use of isotopic labeling and the development of 3D and 4D NMR experiments has especially contributed to this advance. The greatest value is derived from these experiments, not by examining them in isolation, but rather by the correlation of information between the different experiments. Along with the increase in the number of data sets that are correlated, the total amount of information that is abstracted from the data has

*Inquiries concerning the availability of NMRView and the Molecular Data Viewer should be sent via email to johnsonb@merck.com or to Bruce A. Johnson, Merck Research Laboratories, RY80Y-103, P.O. Box 2000, Rahway, NJ 07065, U.S.A.

**To whom correspondence should be addressed.

increased as well. Spectra can contain thousands of cross peaks that must be tabulated and analyzed. Furthermore, the analysis of the spectra of larger molecules is best done in an iterative fashion, where the increasingly refined structure feeds back information to aid in the assignment of the spectra.

Most NMR software has constrained the user to visualize the data from only one experiment at a time and without direct links to the structural data. The most common method for correlating data from multiple experiments used to be superimposition of hard-copy plots. More recently, programs have appeared that provide at least some limited ability to visualize multiple data sets (Oh et al., 1988; Kraulis, 1989; Zolnai et al., 1990; Eccles et al., 1991; Kjær et al., 1991). Generally, these programs are limited to 2D data and often provide significant constraints on the number of active files and the number and layout of spectral display windows.

In this paper we describe a new program, NMRView, for the visualization and analysis of NMR spectra and related molecular structure data. We have tried to create a program that provides a minimum number of limits on the user, yet stresses ease of use and automatic intuitive behavior. To facilitate the use of the information, NMRView contains a variety of database and analysis routines that are integrated with the spectral display features. Moreover, the program has intrinsic structure analysis capabilities as well as bidirectional interfaces to molecular structure programs, so that spectral analysis and structure generation can be a tightly coupled process.

METHODS

NMRView was written in the C programming language. The initial version was written on a Sun Microsystems workstation, using the XView toolkit for the graphical interface. Currently, the program will compile and execute on either Sun or IBM RS/6000 workstations (using a port of the XView toolkit and libraries). A port of the program using the Motif graphical user interface is in progress and should run on Sun, IBM, SGI and other UNIX workstations.

Achieving the goal of reducing limitations on open files and display windows was aided by the use of dynamic memory allocation of the data set and window parameter structures and a bidirectional linked-list method to keep track of these structures. A simple scheme, based on bitwise shift operators and memory pointers, was designed to allow rapid access to any point of any open file. This method is illustrated below for a 2D file consisting of a series of data blocks. Given B_1 and B_2 , the number of data points along each of the two dimensions of a block, and N_1 , the number of blocks along the first dimension of the file, then the file block that a particular point i,j occupies can be calculated from:

$$s_1 = \log_2(B_1)$$

$$s_2 = \log_2(B_2)$$

$$o_2 = \log_2(N_1)$$

$$\text{block} = (i \gg s_1) + ((j \gg s_2) \ll o_2)$$

If the given block is not currently in memory, it is read from disk. Then, the point within the block can be calculated from:

$$p_2 = \log_2(B_1)$$

$$\text{point} = (i \& (B_1 - 1)) + ((j \& (B_2 - 1)) \ll p_2)$$

This algorithm has somewhat greater overhead to recover a particular data point than some alternative methods. However, it is simple and efficient to implement, elegant, and because of the bitwise shift and logical operators used it has very good performance. If the user specifies the above file parameters and the file header size, it should read most block-structured NMR file formats. At present, NMRView can directly extract the necessary parameters from files produced by FELIX (Biosym, Inc.) and a conversion utility can be used to transform VNMR (Varian, Inc.) data sets into a compatible format.

An efficient contouring algorithm was considered crucial to the facile use of NMRView. Contouring algorithms used by NMR software generally are designed to accommodate drawing to pen plotters. The software is thereby constrained to minimize up/down motion of the plotter pen. This creates a significant performance penalty when drawing multiple contour levels. NMRView, which is optimized for graphical displays and Postscript printers, avoids this by drawing all specified contour levels in each grid region (area bounded by four data set points). Drawing at multiple contour levels is thereby only marginally slower than drawing at a single contour level.

RESULTS AND DISCUSSION

Multiple data-set access

A primary design goal of NMRView was to allow the user to have facile access to multiple data sets. This goal was obtained, with the result that there is effectively no limit to the number of spectral data sets that can be open simultaneously. One limitation necessary to achieving this goal was to restrict NMRView to accessing data sets that have a block-oriented structure. Files with this structure have the advantage that only the portion of the file that is currently of interest need be read from the disk file. This allows portions of multiple files to exist in memory simultaneously. The default memory use of NMRView allows 1024 blocks, typically 16 Kb in size, to be in memory simultaneously. These memory blocks could be from one large file or portions of several files. When the available memory blocks are full, and a request is made for data not in memory, NMRView replaces the block that has been in memory the longest with the new data. All these operations appear transparent to the user, who simply perceives that there are no practical limits to the number of files that can be open.

Multiple display windows

To make effective use of multiple files requires the ability to visualize more than one file at a time. Indeed, effective use of even a single file benefits from the ability to visualize expansions of separate regions of the file at the same time. These capabilities are attained by providing the user interface with multiple spectral display windows (Fig. 1). As with the number of files, there are no practical limits on the number of windows that can be displayed. Furthermore, windows can be of any size, up to the limits of the display screen, and can be displayed at any location on the screen. Windows can overlap and be closed to icons. Existing windows can be resized and the spectrum will be automatically redrawn. The initial size of windows (and many other parameters)

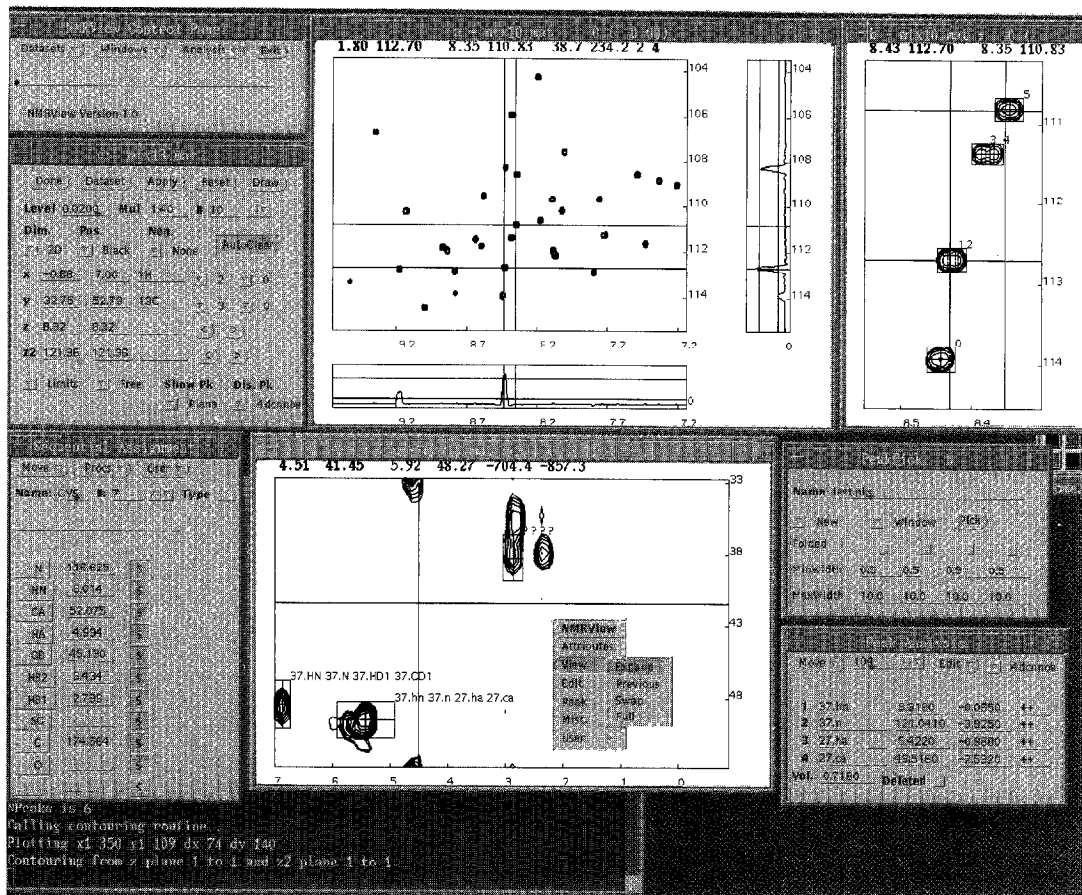


Fig. 1. An image of the computer screen during an NMRView session. The session illustrated has three screen windows open that are displaying regions of a ^1H - ^{15}N -HSQC experiment (windows with headers 'a-mtx10.mat' and 'b-mtx10.mat') and a ^1H - ^{13}C plane of a 4D ^{15}N - ^{13}C -NOESY experiment (window with header 'c-mtx13.mat'). Spectral window b resulted from copying and pasting a region of spectral window a. The window with header 'a-mtx10.mat' has two daughter windows, set to show 1D slices through the spectrum. Note the correlation of cursors among the various subwindows of windows a and b. A TCL procedure is set to redraw the two 1D windows whenever the mouse button is double-clicked in the 2D window. The spectral attributes, sequential assignment, peak-picking and peak-analysis panels are also shown. All illustrated spectra were collected on a Varian Unity 500 NMR spectrometer and processed with FELIX (Biosym, Inc.).

can be specified in a user preferences file. A window manager facilitates control over window display.

At times, it is useful to group related windows. This can be attained in NMRView by subdividing a single display frame into multiple spectral display windows (Fig. 1), for example to display a series of strips of a given spectrum. There are no practical limits to the number of windows, or their size and position, that can be displayed in a single display frame.

Any of the spectral windows can display any region of any file. Thus, each window could be displaying a different file, or different regions of the same file, or any combination thereof. Data from 2D, 3D or 4D data sets can be displayed as 1D vectors or as 2D contour plots (Fig. 1). The 1D vectors and 2D slices can be taken in any orthogonal orientation through the data sets. The

2D displays are always generated as true contour plots, rather than raster displays. Clearing the display window prior to displaying the spectra can be disabled, so that different spectra can be overlaid in a single window. Since plot limits are calculated in ppm, rather than data points, this is possible even if the spectra have different sizes or sweep widths.

Record and playback of contours

Even with the rapid contouring algorithm and memory management features described above, displaying contours of very large files can be demanding and a method of storing contour data is useful (Kraulis, 1989). A means of recording and playing back contours was implemented using a VCR paradigm. The user can select a record mode in the spectral-attributes window and then any display in that window is recorded in a contour file. The user can then select play mode, and any display operation obtains its contour vectors from the stored contour file. All normal expansion, plane selection, display orientation, etc. operations work the same. The user only notices that the display is significantly faster, and is restricted to contour levels at or above those that were recorded. The user can readily toggle back and forth between live mode (with contouring of the actual data file) and play mode.

Cursor correlation

The usefulness of multiple window displays is restricted, unless there is some mechanism to provide a correlation of information between the windows. The primary method for this correlation is through the use of the cursors. Each spectral window has two cross-hair cursors, whose motion can be correlated with the cursors in all other spectral windows. Requiring the user to explicitly instruct NMRView to correlate individual cursors, even among a small number of windows, would have placed a significant burden on the user and significantly diminished the ease of use of the program. Instead, a method was devised that correlates cursors in an automatic and intuitive manner. Each axis of a data set has an associated label that is specified by the user. When the user moves a cursor in a particular window, the cursors in all other windows that have the same label are moved (Fig. 1). This is independent of whether the common labels are on the same screen axis (horizontal or vertical) or whether the common labels are on the same file dimension. The cursor in each window only tracks the motion of the moved cursor if the plot limits of the window overlap the position of the moved cursor. Cursor tracking can be disabled in a window by temporarily changing the window's axis label(s).

Copy/Paste protocol

The cursors can be used as in conventional NMR software, to select regions for expansion. Plot limits for previous expansions are saved in each window, so the user can back up through them. A disadvantage, however, of using expansions is that one loses the visual context of the full spectrum or the larger region encompassing it. As an alternative to this conventional method of viewing expanded regions, NMRView implements a copy/paste protocol. The region within the cursor box can be copied to a buffer, and then pasted to a new window (Fig. 1). The user can then inspect the expanded region, while still being able to see the original plot region. What is copied is actually a data structure describing the plot region, not just a pixel map of the display area, so the new window is fully active with all NMRView functions. Alternatively, just the plot limits can be applied to the new window, so that the corresponding region of a different spectrum could be selected.

Peak picking

Visualization of NMR spectra, using features such as those described above, is at present essential to the proper analysis of the spectra. Most of the useful information present in NMR spectra can, however, be extracted in the form of peak lists containing information about peak positions, shapes, widths and intensities. Accordingly, NMRView provides significant capabilities for the extraction and analysis of peak data. The first tool required is a peak picker. NMRView incorporates a simple but fairly robust peak picker that is similar to that used in some other NMR programs (for example, Kleywegt et al., 1990). Peaks are identified as local maxima in the spectrum. The actual peak position is determined by interpolating to the maximum of a polynomial fit to the three data points nearest to the maximum. The peak bounds are determined by interpolating to the position where the peak crosses below the lowest displayed contour. The determination of position, width and bounds is repeated along all dimensions of the data set. All local maxima are picked, but each peak is flagged according to criteria of minimum and maximum width and overlap with the spectral edge. The minimum and maximum limits are user specifiable.

Some spectra, especially 3D and 4D spectra, may be collected with a sweep width narrower than the frequency range of the spectral peaks (for example, Bax et al., 1990,1991). In these spectra, peaks are folded to a position that differs from their true position by an integer multiple of the sweep width. Peaks on the edge of such spectra 'wrap' around to the opposite edge and conventional peak pickers will not find the correct peak position and width. The NMRView peak picker can be set to wrap at the edge, so that the peak width and position are properly determined. It is not always simple to determine the number of times a given peak has been folded, so NMRView does not automatically correct the peaks to their true position. It would be possible, however, to incorporate preexisting knowledge, such as the spectral sweep width, the chemical shift dispersion, whether delayed acquisition was used to invert peaks folded an odd number of times (Bax et al., 1991), and the expected ^1H and ^{13}C chemical shifts, into a TCL script (see below) that could correct for folding after the peaks have been picked.

Facilitated peak analysis

For most peaks, it is generally hard to improve on the peak position that results from the automatic peak picker. It is, however, useful to visually confirm the results of the peak picking, perhaps making minor adjustments in peak positions. Furthermore, while the peak picker automatically recognizes some artifactual peaks by using criteria of minimum and maximum width, many other artifacts are not readily recognizable. Accordingly, it can be useful as well to interactively delete spurious peaks. This interactive analysis can be tedious, even for peak lists with few peaks. Inspecting thousands of peaks in a 4D spectrum could be nearly impossible. To ease this process, NMRView provides capabilities for facilitated peak analysis. Any spectral window can be set to have its display limits controlled by a peak-analysis panel. Selecting a peak in the peak-analysis window results in the display of the peak region in any of the windows that have the show-peak attribute set. In this manner, the user can rapidly step through even large peak lists, observing every peak. Multiple windows can be controlled so that, for example in 4D spectra, two orthogonal views of the peak could be shown.

The display of peak lists on spectra presents a similar problem to that of the correlation of cursors across windows. For example, how can a peak list obtained from one spectrum be

displayed on a different spectrum that may have different dimensions? The solution used by NMRView is similar to that used for correlating cursors. The axis labels are inherited by the peak list. When displaying a peak list, an appropriate correlation of the dimensions of the peak list and that of the spectral display is obtained by comparison of the peak dimension labels with those of the spectral display. Thus, for example, a peak list from a 2D ^{15}N -HSQC spectrum will display properly on a 4D ^{13}C , ^{15}N HCANNH experiment, if the ^1HN and ^{15}N axes have common names.

We have not emphasized the use of complex pattern-recognition algorithms for analyzing the coupling patterns present, for example, in 2D COSY spectra (Pfändler et al., 1985). These techniques seem to be less useful in our applications to larger proteins, or where isotopic labeling is available. One method for reducing the complexity of spectra due to spin-spin coupling is, however, included (see also Eccles et al., 1991). Peak pairs that are doublets due to coupling can be collapsed to single peaks. This operation recognizes any pair of peaks whose centers are within a specified tolerance. Typically, a tolerance slightly larger than the expected coupling is specified in one dimension, and a tolerance slightly larger than the expected uncertainty in peak position is specified in the other dimension. The algorithm can be iterated to successively reduce additional couplings.

A second algorithm has been found to be very useful in correlating peaks between different peak lists, or across the diagonal in a single spectrum. This algorithm is based on reciprocal matching of the peaks in the two lists. Consider two peaks A_i and B_j from two peak lists A and B. The distance between peaks A_i and B_j must be within user-specified tolerances for each dimension. Furthermore, these peaks match each other, if and only if, B_j is the peak in list B that is closest to peak A_i , and A_i is the peak in list A that is closest to peak B_j . The comparison is performed only for specified dimensions of the peak list. For example, the ^1H and ^{15}N dimensions of an HCANNH and an HCA(CO)NNH experiment could be correlated, without requiring the HA and CA dimensions to correlate. This routine forms one aspect of automated analysis capabilities that are currently in development.

Assignment tables

One of the primary uses of the peak data is to obtain sequence-specific assignments of the peak positions. This is facilitated by a residue panel that displays the atom names, the current chemical shift assignment, and whether or not the atom is assigned stereospecifically (Fig. 1). The atom names, types and connectivities can be obtained by importing a PEGASUS format sequence file (Johnson and Sugg, 1992), a PDB (Protein Data Bank, Bernstein et al., 1977) file or an XPLOR PSF file (Brünger, 1992).

Command language

The above visualization and analysis features of NMRView provide a broad array of capabilities to the NMRView user. We cannot, however, envision every possible use for such a program. Furthermore, NMR is a rapidly changing field and ideas for new features are suggested often. Therefore, we considered it important to provide a means to allow users to add new capabilities to NMRView. We wanted to allow extendibility without relying on changing the source code. Source code changes are slow to implement because the code needs to be recompiled, they cannot be implemented during an active session with the program, are difficult for the non-programmer, and can cause inscrutable errors in the execution of the program. The alternative is to embed an

interpreted language into the program. Instead of writing our own language, we chose to incorporate an existing command language. TCL (pronounced tickle), tool command language, is a powerful and flexible command language that can readily be embedded in any application (Ousterhout, 1990). It provides scalar, array and list variables. TCL scripts can include callable procedures with arguments and local variables. Multiple procedures can be resident in memory simultaneously and a library of procedures can be loaded automatically at the program startup. Flow control is possible with if and case statements, and looping with for, do and while statements. TCL includes many built-in commands for manipulating and analyzing strings and lists, performing arithmetical operations and logical comparisons, and reading, writing and examining the status of files. A powerful command language like TCL provides many features that would otherwise be unattainable without relying on changes at the source-code level.

We extended the many built-in commands of TCL with commands that give access to nearly all the functions of NMRView. These commands can be entered individually in a command field, or can be included in TCL procedures. All aspects of the spectral windows, for example the window size and placement, display dimensions, plot limits and peak display modes can be controlled with these commands. A large number of NMRView commands are available for analyzing and manipulating peak lists. For example, commands exist to find peaks within a region of a spectrum, find the peak closest to a specified point, find peaks whose labels match specified criteria, and delete peaks in specified regions. Using the peak-related commands, scripts can be generated to filter peak lists to eliminate peaks based on various criteria, such as nearness to the diagonal, minimum or maximum widths, intensities or width/height ratios.

NOE identification and analysis

Peak lists form the basis of NOE constraint generation. Accordingly, there are specific NMRView commands to aid in assignment of NOESY peaks. The NOE identification command can use 2D, 3D or 4D criteria for matching to the chemical shift assignment data (similar procedures using 2D data have been previously described by Kline et al. (1990) and by Johnson and Sugg (1992)). For example, the user can request all possible assignments consistent with an NOE between a proton at 3.2 ppm that is bound to a carbon at 37.3 ppm, and a proton at 7.2 ppm that is bound to a nitrogen at 117.3 ppm. NMRView would return a list of proton pairs that are within a certain tolerance of the specified shifts. If one or more molecular structures have been read in to memory, the program will calculate the shortest, average and longest distances between the various proton pairs. This provides additional criteria for selecting a specific assignment for the peak. The integration of this command into the TCL command language, along with the ability to import and analyze molecular structures, forms the basis of manual or automated NOE assignment that can be used in an iterative structure refinement process.

The rapid use of an iterative refinement process is in part dependent on facile generation of the constraint lists. NMRView can generate constraint lists from assigned NOE cross peaks. NOEs can be calibrated using a binning method or an analytical calculation (Güntert et al., 1991). Pseudoatom corrections are automatically applied where needed. The constraints involving diastereotopic substituents that are not stereospecifically assigned are optimized by taking account of the available experimental data (Güntert et al., 1991). The constraints can then be exported in either PEGASUS (Johnson and Sugg, 1992) or XPLOR format (Brünger, 1992) or a custom TCL script can be written to export in any other format. In order to obtain stereospecific assignments

as well as torsion angle constraints, a grid search procedure has been implemented within NMRView. The procedure is similar to some previously described (Güntert et al., 1989; Nilges et al., 1990), but allows for the inclusion of additional couplings that can be measured from hetero-nuclear experiments.

Molecular structure analysis

This iterative structure refinement process is also facilitated in NMRView by a variety of commands for the manipulation and analysis of molecular structures. NMRView can read molecular structures in the format of PDB files. Structures can be superimposed and average structures can be generated. NMRView commands are available that can extract a variety of structural information, such as interatomic distances and dihedral angles. These can also form the basis of TCL scripts to generate tables and figures of structure data for publications.

While NMRView by itself is incapable of displaying molecular graphics, the package can be coupled to molecular graphics programs through UNIX message queues (this will be converted to UNIX sockets in a future release). At present we use MDV (Molecular Data Viewer, Blevins, R., unpublished results). Any structure(s) stored in NMRView can be sent to the MDV for display. TCL scripts can be written to link any NMRView data to the structure (Fig. 2). For example, clicking on a spectral peak could highlight the corresponding atoms in the display.

XY plotting and analysis

In the analysis of structures and NMR data, it is useful to display data in the format of XY plots. NMRView incorporates a simple XY plotting facility (Fig. 2). Any data that can be extracted using NMRView commands can be stored in an XY array and plotted. In addition, a nonlinear regression routine is included for fitting equations to the data. Several built-in equations are provided for fitting NMR relaxation data. The XY plotting and nonlinear regression routines can be driven both through the GUI (Graphical User Interface) and with NMRView commands. With this capability one can, for example, analyze the multiple data sets of a 2D heteronuclear relaxation study in a few minutes. As one steps through the peak list, the corresponding cross peak from each spectrum can be displayed, the XY plot of the intensity versus data can be displayed, and the relaxation rate data can be calculated and stored to a file. Besides analysis of relaxation data, the XY facility has many uses, including observing NOE buildup curves, displaying Ramachandran plots and displaying rmsd versus residue plots.

Programmable user interface

Using TCL scripts provides a great amount of flexibility and extendibility to the program, but deviates somewhat from the design goal of having the program be controlled through the graphical user interface. To counter this, we provide three methods to integrate the graphical interface with the TCL command scripts. First, mouse events in spectral windows can trigger the execution of user-specified TCL procedures. Up/down mouse button motion, double-clicking the mouse buttons and dragging the cross-hair cursor can each trigger specific TCL procedures. Second, the user can add custom entries to the pop-up menu in each spectral window and many of the interface panels. These entries are the names of user-specified TCL procedures that are executed

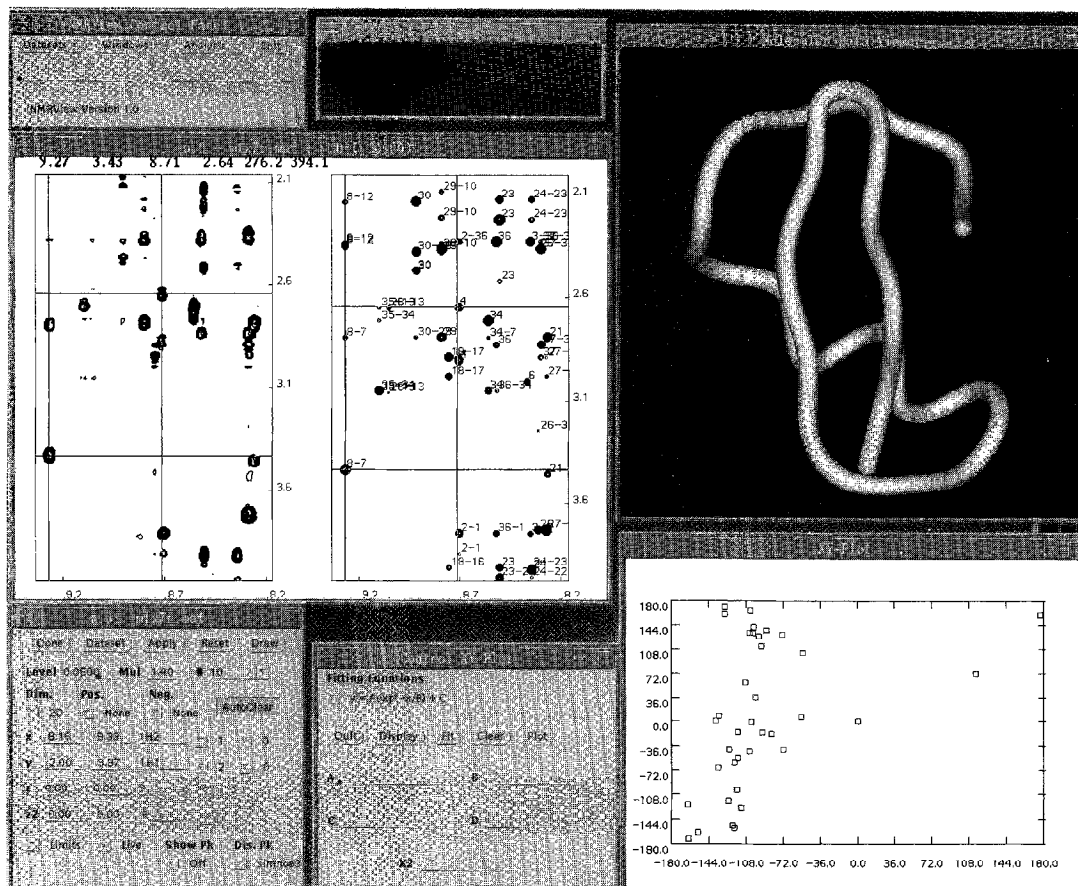


Fig. 2. A second view of an NMR session. During this session two spectral windows are in use within a single frame. A NOESY spectrum is displayed in the left window and a simulated NOESY based on the current molecular structure is displayed in the right window. The Molecular Data Viewer is in use to display a structure currently stored in NMRView memory. The XY plotting facility is being used to display a Ramachandran plot of the structure displayed in the Molecular Data Viewer.

upon selecting the new menu entry. Third, pressing any alphanumeric key sends that key's character and the window name to a user-specified TCL procedure. Together, these methods provide a mechanism to customize the user interface to the particular task at hand.

An application example

Finally, we briefly describe how NMRView has been used in an actual application, the determination of the structure of a 39-residue protein. Four-dimensional HCANNH and HCA-(CO)NNH experiments (Boucher et al., 1992) were automatically peak-picked. The folding option of the peak picker was used on the appropriate dimensions of the spectra. Two spectral windows were used to display two sets of orthogonal views of each peak. The spectral region displayed was controlled by the peak-analysis window. Each picked peak was quickly inspected, adjusted if necessary or deleted if artifactual. The two peak lists were automatically correlated

using the double matching algorithm described above. This was first done using all four peak dimensions, to eliminate interresidual cross peaks from the HCANNH spectra. Next, the match was based on the HA and CA dimensions to correlate overlapping HA,CA pairs and then on the HN and N dimensions to correlate overlapping HN,N pairs. These correlated peaks were then placed in order by the common peaks. All the expected glycine- or proline-bounded stretches of residues were found (Campbell-Burk et al., 1992). These were then manually assigned to the sequence, based on length and glycine and proline starts and stops.

Next, a 4D HACACH TOCSY (Johnson, B.A., unpublished results) experiment was performed. A TCL procedure was assigned to an entry on the user menu of the residue panel. The procedure gets the HA and CA chemical shifts and selects the corresponding plane of the spectra for display of the side-chain C,H dimensions. Next, it draws boxes on the spectra at the expected peak position for the side-chain atoms of that residue type. The actual peaks were then correlated to the expected peak positions to complete the assignment for that residue.

Having completed the sequential assignment process, the NOESY spectra were assigned. Four-dimensional ^{13}C , ^{15}N and ^{13}C , ^{13}C NOE experiments were peak-picked. Regions containing artifactual peaks, such as the stripe corresponding to residual water, were interactively deleted. TCL procedures were then used to automatically delete peaks that violated minimal or maximal widths. Next, the remaining peaks were interactively examined, as described above. The PLAY mode was used, so that stepping through the peak list was very rapid. Every peak of the two 4D NOESY spectra was individually examined.

Next, a TCL procedure was used that displays the spectral region corresponding to the cross peaks expected between any given proton and the three or four other protons that are closest to it. This procedure was assigned to the user menu of the residue panel to allow rapid checking and correction of the chemical shift list so that it corresponds to the current experiment (Eccles et al., 1991). Adjusting the chemical shift list in this way allows one to reduce the peak tolerances used in the NOE identification process and increases the number of peaks that can be unambiguously identified, based on chemical shift alone. Sufficient unambiguous NOE peaks (approximately 213) were assigned, using the NOE identification facility of NMRView to generate a constraint list. Initial structures were generated using the program PEGASUS (Johnson and Sugg, 1992). These structures were then imported into NMRView. The NOE identification procedure was repeated, using the additional criteria of the interproton distances from the trial structures for the assignment procedure. This process was iterated until a well-defined structure was obtained. The structures were then used in NMRView to generate synthetic spectra, assuming a simple r^{-6} distance dependence of peak intensity (Wüthrich, 1986). These were compared to the experimental 2D NOESY spectra to check for the presence of unassigned cross peaks. Additional cross peaks were assigned in this way. A total of 273 NOE constraints were ultimately used in the structure generation. The grid-search procedure was used with coupling constant and intrare-sidue and sequential NOE data to obtain stereospecific assignments and torsion angle constraints. Final structures were generated with PEGASUS.

TCL procedures were then written to statistically analyze the resulting structures. Files were generated containing rmsd superpositions of the overall structures, as well as residue-based atomic coordinate and dihedral angle deviations. The chemical shift table was automatically dumped to a file in a format suitable for import into a word processor.

CONCLUSIONS

NMRView forms the basis of an integrated approach for analyzing NMR data. A variety of unique features combine to provide a facile method of interacting with multiple data files and provide a tight coupling between the spectral analysis and structure-generation process. The graphical user interface provides advanced visualization capabilities that can be customized to the task at hand. The built-in commands provide many capabilities for peak analysis. The command language provides a mechanism for customization and automation. The example of the use of NMRView described here used a facilitated approach that combines manual and automated procedures. We are currently working to extend the program to more fully automate processes of spectral assignment of heteronuclear NMR spectra.

ACKNOWLEDGEMENTS

We thank Drs. Paul Gooley and John O'Connell for suffering through the early versions of NMRView and providing many suggestions for improvements.

REFERENCES

- Bax, A., Clore, G.M. and Gronenborn, A.M. (1990) *J. Magn. Reson.*, **88**, 425–431.
- Bax, A., Ikura, M., Kay, L.E. and Zhu, G. (1991) *J. Magn. Reson.*, **91**, 174–178.
- Bernstein, F.C., Koetzle, T.F., Williams, G.J.B., Meyer Jr., E.F., Brice, M.D., Rodgers, J.R., Kennard, O., Shimanouchi, T. and Tasumi, M. (1977) *J. Mol. Biol.*, **112**, 535–542.
- Boucher, W., Laue, E.D., Campbell-Burk, S. and Domaille, P.J. (1992) *J. Am. Chem. Soc.*, **114**, 2262–2264.
- Brünger, A.T. (1992) *X-PLOR*, V. 3.1, A System for X-ray Crystallography and NMR, Yale University Press, New Haven, CT.
- Campbell-Burk, S., Domaille, P.J., Starovasnik, M.A., Boucher, W. and Laue, E.D. (1992) *J. Biomol. NMR*, **2**, 639–646.
- Eccles, C., Güntert, P., Billeter, M. and Wüthrich, K. (1991) *J. Biomol. NMR*, **1**, 111–130.
- Güntert, P., Braun, W., Billeter, M. and Wüthrich, K. (1989) *J. Am. Chem. Soc.*, **111**, 3997–4004.
- Güntert, P., Braun, W. and Wüthrich, K. (1991) *J. Mol. Biol.*, **217**, 517–530.
- Johnson, B.A. and Sugg, E.E. (1992) *Biochemistry*, **31**, 8151–8159.
- Kjær, M., Andersen, K.V., Shen, H., Ludvigsen, S., Windekilde, D., Sørensen, B. and Poulsen, F.M. (1991) In *Computational Aspects of the Study of Biological Macromolecules by Nuclear Magnetic Resonance Spectroscopy* (Eds, Hoch, J.C., Poulsen, F.M. and Redfield, C.) NATO ASI series A, Vol. 225, Plenum Press, New York, NY, pp. 291–302.
- Kleywegt, G.J., Boelens, R. and Kaptein, R. (1990) *J. Magn. Reson.*, **88**, 601–608.
- Kline, T.P., Brown, F.K., Brown, S.C., Jeffs, P.W., Kopple, K.D. and Mueller, L. (1990) *Biochemistry*, **29**, 7805–7813.
- Kraulis, P.J. (1989) *J. Magn. Reson.*, **84**, 627–633.
- Nilges, M., Clore, G.M. and Gronenborn, A.M. (1990) *Biopolymers*, **29**, 813–822.
- Oh, B.H., Westler, W.M., Darba, P. and Markley, J.L. (1988) *Science*, **240**, 908–911.
- Ousterhout, J.K. (1994) *TCL and the Tk Toolkit*, Addison-Wesley, Reading, MA.
- Pfändler, P., Bodenhausen, G., Meier, B.U. and Ernst, R.R. (1985) *Anal. Chem.*, **57**, 2510–2516.
- Wüthrich, K. (1986) *NMR of Proteins and Nucleic Acids*, Wiley, New York, NY.
- Zolnai, Z., Westler, W.M., Ulrich, E.L. and Markley, J.L. (1990) *J. Magn. Reson.*, **88**, 511–522.