Protein Structure Modeling using Chemical Shifts and Residual Dipolar Coupling Homology, in the Context of an Integrated Software System for Spectral Processing and Analysis

Frank Delaglio
delaglio@ nih.gov

Section on Biophysical Nuclear Magnetic Resonance
Laboratory of Chemical Physics
National Institutes of Health, NIDDK
5 Center Dr MSC 0505
Bethesda MD 20892-0505 USA

Lab: 301 496-1207 Fax: 301 496-0825 Cell: 301 806-0867

Advisor:
Prof. Yuji Kobayashi
Biophysical Chemistry
The Graduate School of Pharmaceutical Sciences
Osaka University

Submitted:
Thursday, May 31, 2001

Last Revised:
Friday, September 7, 2001

Table of Contents

Abstract

A novel approach to NMR structure calculation is presented, based on searching the Protein Databank (PDB) for structural fragments whose simulated residual dipolar couplings and empirically estimated chemical shifts give a best match to measured shifts and couplings of the protein under study. This approach can be exploited in many possible ways, including for prediction of phi and psi backbone angle constraints, estimation of the alignment tensor, and in searches for larger scale structural homology. This new approach also offers the potential to greatly decrease the time required for NMR structure determination, as it avoids the need for large numbers of NOE assignments. One possible implementation of the approach is illustrated using the protein ubiquitin, for which complete sets of dipolar couplings have been measured in samples with two different alignment tensors. In one scheme applied to ubiquitin, phi and psi angles from an overlapping collection of the 70 best seven-residue fragments found by the PDB search were used in combination to reconstruct an overall backbone fold. This overall fold was then refined by perturbation of phi and psi to optimize agreement with the complete set of couplings and shifts. In preliminary results with ubiquitin, this scheme produced a backbone fold with an rmsd of better than 1.5 Å relative to the X-ray structure, without the use of NOEs or any other distance constraints. Alternatively, schemes can be devised which also employ such easily derived information as sequential NOE distances and H-bond data, and these are discussed.

The approach builds on concepts introduced in the earlier TALOS system, which predicts protein backbone angles by secondary shift homology using a database of 20 high-resolution protein structures and their corresponding chemical shifts. Extensions to this method, which reverse the TALOS procedure in order to predict chemical shifts from protein backbone angles, are described. This prediction strategy can be used in homology search schemes, and to allow structure refinement directly against chemical shifts. To implement the dipolar coupling structure determination approach, a new program called DYNAMO has been developed. This new software uses the flexible TCL/TK scripting language as a command interface, and provides facilities for homology searching, structure manipulation and evaluation, simulation of dipolar couplings and other NMR parameters, and facilities for conventional NMR structure calculation by restrained dynamics. This new program is integrated into the NMRPipe software system for spectral processing and analysis. Therefore, the details of NMRPipe are also presented, with attention to related aspects such as special methods for quantifying coupling constants.

This work is part of a broader goal to develop a complete, convenient, and effective software system for all stages of NMR structure determination and analysis. To date, the existing software has been applied in over 500 studies of macromolecular structure.

Introduction

Early NMR protein structure applications relied primarily on the measurement of the maximum possible number of short-range (*ca* 5 Å) ^{1}H-^{1}H distances from 2D NOE spectra, in schemes pioneered primarily by Wüthrich and co-workers (Wüthrich, 1986). The principal challenge to these early applications is that the NOE spectra were not only the main source of structural information, but also the main tool for sequential assignment. Development of protein stable isotope enrichment techniques and corresponding ^{15}N and ^{13}C separated 3D NMR experiments helped by reducing the degree of spectral overlap, but did not circumvent the basic problem of the interdependence of NOE analysis and sequential assignment (Fesik and Zuiderweg, 1988; Marion et al. 1989; Oh et al. 1988; Ikura et al., 1990). This problem was answered by the advent of novel sequential triple resonance methods, which revolutionized the applicability of NMR to macromolecular structure determination by making extraction of chemical shift assignments mostly independent from molecular conformation and NOEs (Bax et al., 1991; Clore and Gronenborn, 1991; Grzesiek and Bax, 1992A,B,C; Palmer et al., 1992). In the same spirit, recent advances in the measurement and use of dipolar couplings offer an opportunity to extract substantial amounts of structural information independently from extensive NOE analysis. This can be augmented by local structure information that is also derived independently from NOE analysis, especially from J couplings (Karplus, 1959; Bystrov. 1976; Bax et al., 1994; Biamonti et al., 1994, Hu and Bax, 1997; Vuister et al., 1999) and the chemical shifts themselves (Spera and Bax, 1991; Wishart et al., 1991; Kuszewski et al., 1995; Cornilescu et al., 1999).

It is important to note that these methodological advances cannot be exploited without corresponding computational tools and methods. In this presentation, we describe many of the software and analysis issues that have accompanied the evolution of multidimensional NMR of biomolecules, and give examples of how several of these issues can be addressed.

From the point of view of software and analysis, we can identify some key areas and tasks involved in the application of NMR to biomolecules. Each one of these areas is a rich topic on its own, and can overlap with the others:

- **Spectral processing.** This includes the general task of multidimensional Fourier transform, as well as specialized reconstruction methods. It also includes the less interesting but still critical task of spectrometer format conversion: rigorous establishment of chemical shift calibration parameters, definition of data sizes, accommodation for digital oversampling, etc.
- **Peak detection and quantification.** This includes both automated peak detection methods, and interactive interfaces for peak editing. Furthermore, all NMR structure parameters are ultimately derived from some combination of the basic signal properties of amplitude, frequency, linewidth, and phase, so methods to estimate these parameters and their uncertainties are critical.
- **Resonance assignment.** In the case of proteins, this task is often divided into assignment of the backbone resonances, and assignment of sidechain resonances.
- **Derivation of structure parameters.** This includes calculation of torsion angles from J couplings, measurement of distance bounds from NOEs, etc.
- **NOE assignment.** This is commonly viewed as the most difficult and time-consuming step in NMR structure determination.

- **Structure calculation.** The task of building physically realistic structures which are consistent with the NMR parameters.
- **Visualization and evaluation of structure.** This includes measurement and presentation of the agreement between structure and NMR parameters, as well as tests to show whether the structure is physically realistic.
- **Exploitation of structure.** For example, in drug screening and binding studies.

In addition to providing facilities to perform these tasks, software and analysis approaches can also strive to further these goals:

- Increase molecular size and complexity which can be studied by NMR
- Confirm validity of structure
- Increase precision of structure
- Reduce time and labor required, minimize chance for error, increase throughput

In **Section 1**, we describe the NMRPipe software environment, which is the basis for all of the computational methods described in this work. The NMRPipe system is a UNIX software environment of processing, graphics, and analysis tools designed to meet current routine and research-oriented multidimensional processing requirements, and to anticipate and accommodate future demands and development. The system is based on UNIX pipes, which allow programs running simultaneously to exchange streams of data under user control. In an NMRPipe processing scheme, a stream of spectral data flows through a pipeline of processing programs, each of which performs one component of the overall scheme, such as Fourier transformation or linear prediction. In our approach, complete multidimensional processing schemes are constructed as simple UNIX shell scripts. The processing modules themselves maintain and exploit accurate records of data sizes, detection modes, and calibration information in all dimensions, so that schemes can be constructed without the need to explicitly define or anticipate data sizes or storage details of real and imaginary channels during processing. The asynchronous pipeline scheme provides other substantial advantages, including high flexibility, favorable processing speeds, choice of both all-in-memory and disk-bound processing, easy adaptation to different data formats, simpler software development and maintenance, and the ability to distribute processing tasks on multi-CPU computers and computer networks. Through use of an extensible scripting language called TCL/TK, we also show how the tools of the NMRPipe system can be extensively customized and combined to build targeted interactive and automated systems for a wide variety of NMR processing and analysis tasks.

In **Section 2**, we describe an example of spectral quantification which is a synthesis of methods for spectral processing, presentation and analysis, and therefore an ideal example of the utility of the NMRPipe approach. An interactive procedure is described which determines ^{1}H-^{1}H couplings from fitting the cross peak multiplets in regular phase-sensitive COSY spectra. Robustness and simplicity of the method relies on the fact that a given cross peak intensity is not an independent variable in the fitting procedure, making it possible to measure couplings accurately even from individual cross peaks with unresolved multiplet structure.

In **Section 3**, we describe an approach for exploiting chemical shifts for quantitative prediction of protein backbone conformation. Chemical shifts of backbone atoms in proteins are known to be exquisitely sensitive to local conformation, and homologous proteins show quite similar patterns of secondary chemical shifts. The inverse of this relation is used to search a database for triplets of adjacent residues with secondary chemical shifts and sequence similarity which provides the best match to the query triplet of interest. The database contains $^{13}C^{\alpha}$, $^{13}C^{\beta}$, $^{13}C'$, $^{1}H^{\alpha}$ and ^{15}N chemical shifts for 20 proteins for which a high resolution X-ray structure is available. The computer program TALOS was developed to search this database for strings of residues with chemical shift and residue type homology. The relative importance of the weighting factors attached to the secondary chemical shifts of the five types of resonances relative to that of sequence similarity were optimized empirically. TALOS yields the 10 triplets which have the closest similarity in secondary chemical shift and amino acid sequence to those of the query sequence. If the central residues in these 10 triplets exhibit similar phi and psi backbone angles, their averages can reliably be used as angular restraints for the protein whose structure is being studied. Tests carried out for proteins of known structure indicate that the root-mean-square difference (rmsd) between the output of TALOS and the X-ray derived backbone angles is about 15s. We find that TALOS can commonly make predictions for roughly 65% of the residues in a given protein, although approximately 3% of the predictions made by TALOS are found to be in error.

As an extension to the TALOS database search approach, we also show how the TALOS database can be used to build surfaces which give the typical secondary chemical shift associated with a given ϕ and ψ backbone conformation. These surfaces can be used in turn to simulate chemical shifts empirically based on secondary structure.

In **Section 4**, a novel procedure for determining the three-dimensional structure of large segments of a protein backbone is described and demonstrated for the small globular protein ubiquitin. The method is based on the experimental measurement of a large number of dipolar couplings for backbone atom pairs in a $^{13}C/^{15}N$ enriched protein weakly aligned in a dilute liquid crystalline phase, which yield orientational information for the interatomic vectors. The first stage of the method searches the Protein Data Bank repository for protein segments whose simulated dipolar couplings and chemical shifts agree with those measured for a given query segment in the query protein under study. This yields fragments which have good structural homology with the query segment, with root-mean-square deviations (rmsd) of 10-12° for the backbone torsion angles ϕ and ψ. A simple gradient based minimization procedure, applied to a starting structure built on the basis of these initial estimates, subsequently refines the structure such that all dipolar couplings predicted by the model are in agreement with the experimental dipolar couplings. The procedure is demonstrated for the protein ubiquitin, for which two complementary sets of dipolar couplings, measured in two different liquid crystalline phases, are available. The backbone coordinates for the structure agree to within an rmsd of better than 1.5 Å with the X-ray crystal structure, in the absence of any NOE, hydrogen bond or other experimental distance restraints.

In the **Concluding Remarks Section**, we review the results of this work, as part of a broader goal to develop a complete, convenient, and effective software system for all stages of NMR structure determination and analysis.

1. Multidimensional NMR spectral processing and analysis based on UNIX pipes and scripting languages

Introduction

As use of multidimensional NMR has become widespread, demands on multidimensional spectral processing software have increased. Software must keep pace with both NMR applications research, and with the routine use of NMR for biomolecular structure determination. Routine use requires software to accommodate increasing numbers of experiments, larger data sizes, more complicated processing schemes, and common use of 4D NMR (Pelczer and Szalma, 1991; Bax and Grzesiek, 1993). Various vendor-specific modes of quadrature detection and data storage must also be addressed. At the same time, NMR technique development research requires software to serve as a platform for testing and evaluation of new experiments and acquisition methods, as well as new spectral analysis and enhancement approaches.

The user community for multidimensional processing software is also changing, and many practitioners of biological NMR are not necessarily familiar with NMR computer applications or signal processing. In addition, there are generally increasing expectations for software that is graphically oriented, error free, and which works harmoniously with other applications on a variety of networked computers. Correspondingly, current software development approaches often favor creation of several small, well-targeted applications coordinated by standard graphics and command tools.

We present here the NMRPipe system, a comprehensive new multidimensional NMR data processing system which addresses the growing needs for ease of use, efficiency, and flexibility of multidimensional spectral processing in the laboratory network. The NMRPipe system is a UNIX pipeline-based software environment for multidimensional processing, coordinated with spectral graphics and analysis tools. The system was implemented in the C programming language (Kernighan and Ritchie, 1988) using the program development tools of UNIX (Kernighan and Pike, 1984).

Several other multidimensional NMR data processing packages have been developed over the past decade, including the popular FELIX (BIOSYM Technologies Inc., San Diego CA), as well as AZARA (W. Boucher, unpublished results), Dreamwalker (Meadows et al., 1994), GIFA (Delsuc, 1988), NMR Toolkit (Hoch, 1985), NMRZ (New Methods Research Inc., Syracuse NY), Pronto (Kjaer et al., 1994), PROSA (Güntert et al., 1992), and TRIAD (Tripos Inc., St. Louis MO). The NMRPipe system incorporates a novel approach to spectral processing which is complementary to other methods, and provides many advantages. Spectral processing is performed using modules connected by UNIX pipes, which allow programs running simultaneously to exchange streams of data under user control. In this approach, a stream of spectral data flows through a pipeline of processing programs, each of which performs one component of the overall scheme, such as Fourier transformation or mirror-image linear prediction.

The processing programs of the NMRPipe system work in the same way as ordinary UNIX commands; this means that complete multidimensional processing schemes can be constructed as standard UNIX command scripts, which are easy to learn and manipulate. The pipeline approach

provides favorable processing speeds, while at the same time allowing the choice of both all-in-memory and disk-bound processing, easy adaptation of new algorithms and differing data formats, and simpler software development and maintenance. Since processing is achieved via a series of programs running simultaneously, the NMRPipe pipeline approach also provides a way to exploit the capabilities of multi-processor computers or to distribute processing tasks across a network.

In addition to the general advantages of the pipeline approach, there are other advantages arising from specific details of NMRPipe's implementation. For example, the components of NMRPipe are engineered to maintain and exploit accurate records of data size, detection mode, calibration information, and processing parameters in all dimensions. This means that schemes can be created and reused easily, since parameters can be specified in terms of spectral units, and there is no need to explicitly define or anticipate data sizes during processing. The parameter record also allows NMRPipe modules to assemble the correct combination of real and imaginary data for a given dimension automatically; this permits dimensions to be processed and reprocessed in any order with schemes that are generally the same regardless of acquisition mode and vendor-specific storage details.

Methods

The NMRPipe approach relies on the UNIX operating system concepts of data streams, filters, and pipes, so they are discussed in some detail here. By necessity, these concepts are becoming increasingly familiar to the biomolecular NMR community, since modern spectrometers are commonly controlled by UNIX computers, and molecular structures are usually generated and visualized on UNIX workstations.

UNIX Commands and Filters. UNIX has no strong distinction between commands built into the operating system and programs which are part of "external" applications such as spectral processing. This means that application programs can potentially be used like ordinary UNIX commands, and the standard UNIX facilities for combining and manipulating them can be exploited. For example, one or more UNIX commands can be placed into an ordinary text file, called a *shell script.* Such a shell script can then be executed by its name, just as if it were also a UNIX command.

A UNIX *filter* is a general term for a command or program which reads input, processes it in some way, and produces an output. One example of a filter is the UNIX command **sort,** which reads lines of text and writes them out again sorted in alphabetical order. Another example is the UNIX command **tr**, which translates characters (e.g. from upper-case to lower-case) in its input before writing them. Depending on the nature of the task involved, UNIX filters may read and process their input data in small parts, such as **tr** (which can process one character at a time), or in its entirety, such as **sort** (which must read the entire input first in order to sort it).

In UNIX terminology, a filter's source of input data is called *standard input* and its destination for output data is called *standard output.* By default, standard input is data entered from the keyboard, and standard output is data displayed on the computer screen. UNIX allows filters to take their input from an existing file instead of the keyboard; this is called *input redirection*, and it is performed using the < character. Correspondingly, filters can send their output to a file

instead of to the screen; this is called *output redirection*, and it is performed using the > character. The following two UNIX commands show examples of redirection. The first command will sort the lines in file "old.text", and write the sorted results to file "new1.text"; the second command will convert the text in file "new1.txt" from lower-case to upper-case, and store the result in file "new2.text":

```
sort < old.text > new1.text
tr 'a-z' 'A-Z' < new1.text > new2.text
```

Commands like these illustrate the concept of a *data stream*, where data "flows" from an input source, travels through a filter, and collects at an output destination.

UNIX Command Line Arguments. The use and behavior of a UNIX command can be adjusted by command-line arguments, which are additional parameters specified after the command. The parameters are usually identified by words or letters prefixed by the - character. For instance, while the UNIX command **sort** will sort text in alphabetical order, adding the argument **-r** will cause text to be sorted in reverse alphabetical order:

```
sort -r < old.text > new1.text
```

Each UNIX command has its own list of possible command-line arguments, which are described in the command's *manual page*, a brief document (but often more than one page) that is available on-line. UNIX manual pages have a standard format, and new manual pages can be added easily, so that application programs can make use of the same on-line help system used by other UNIX commands.

UNIX Pipes. UNIX pipes allow commands to be connected together in a series, where the output of one command is used directly as the input to the next command. A series of programs connected in this way is often called a *pipeline*. A pipe is specified in a UNIX command line by the | character inserted between commands. For example, we can combine the sorting and character translation commands into a single pipeline:

```
sort < old.text | tr 'a-z' 'A-Z' > new2.text
```

In this pipeline, data travels from the input file through the **sort** filter, and the sorted result travels via pipe through the **tr** filter and then to the output file. As shown, pipes allow simple commands to be combined to perform complex tasks, while avoiding the need for intermediate results to be saved in files. Pipeline communication is also relatively fast, since UNIX pipes are generally implemented via physical memory buffers in the operating system (Stevens, 1992).

Pipelines, like UNIX command lines in general, can be split over several lines of text. This is especially useful when the pipeline contains many components. In the UNIX idiom, the \ (backslash) character is used at the end of a line to continue a command onto the next line. For example, a functionally equivalent version of the **sort** pipeline above could be entered as follows:

```
sort < old.text \
| tr 'a-z' 'A-Z' > new2.text
```

Spectral Processing Function as a UNIX Filter. The concept of a UNIX filter command can be extended directly to spectral processing. By analogy, a spectral processing function can be implemented as a UNIX filter which reads an input stream of unprocessed spectral data vectors, applies a spectral processing function to each vector, and writes the result as stream of processed vectors. We have implemented this concept as a program called **nmrPipe**, the central module of the NMRPipe system.

The **nmrPipe** program applies a given processing function to a stream of spectral data. The processing function is selected via a "function name" argument **-fn**, and corresponding processing modes and parameters are specified by other optional command-line arguments. For example, the following three commands are filters which apply a forward Fourier transform (FT), an inverse Fourier transform, and a 90-degree zero order phase correction (PS), respectively:

A Forward Transform Filter: `nmrPipe -fn FT`
An Inverse Transform Filter: `nmrPipe -fn FT -inv`
A Phase Correction Filter: `nmrPipe -fn PS -p0 90`

The required input stream for **nmrPipe** consists of a header describing the data, followed by the binary data vectors themselves, usually in a sequential order. The output stream consists of the header, which is updated to reflect processing, followed by the processed data vectors. The stream format is meant to resemble the contents of an ordinary 2D file plane, so that such a file can be used directly with **nmrPipe**.

As with other UNIX filters, **nmrPipe** reads and writes streams via standard input and standard output, but for convenience explicit input and output file names can be specified by the command-line arguments **-in** and **-out**. For example, the following two commands perform the same task; they both apply a Fourier transform to all the data vectors in file "spec.fid", and save the result in file "spec.ft":

```
nmrPipe -fn FT < spec.fid > spec.ft
nmrPipe -fn FT -in spec.fid -out spec.ft
```

The **nmrPipe** program includes implementations of many common 1D processing functions, as well as several other useful elements; these are listed in Table 1.1, and several are discussed in more detail below.

Spectral Processing Scheme as a UNIX Pipeline. The concept of a spectral processing function performed as a UNIX filter leads directly to the idea of a spectral processing scheme implemented as a UNIX pipeline; this is the central concept of the NMRPipe system. In this method, spectral data flows through a pipeline of processing filters, each performing one aspect of the processing scheme. In practice, this is achieved by using multiple instances of the

```
bruk2pipe -in ser                                    \          Spectrometer-Format Input
~ -xN       1024   -yN        104   -zN         64 \          Total Points in File
~ -xT        512   -yT         52   -zT         32 \          Complex Points Acquired
~ -xMODE Complex   -yMODE Complex   -zMODE Complex \          Acquisition Mode
~ -xSW   7575.76   -ySW   8445.95   -zSW   1515.15 \          Spectral Width, Hz
~ -xOBS  500.130   -yOBS   125.76   -zOBS  50.6800 \          Observe Frequency, MHz
~ -xCAR    4.683   -yCAR     46.0   -zCAR   117.00 \          Carrier Position, PPM
~ -xLAB       HN   -yLAB     CACB   -zLAB        N \          Axis Labels
~ -ndim        3   -aq2D   States                   \          Dimension Count, 2D Mode
~ -out fid/cbcaconh%03d.fid -verb -ov                          Output File Series
```

Figure 1.1. Annotated format conversion script used for a 3D CBCA(CO)NH FID acquired on a Bruker AMX spectrometer. The general form of the conversion script is the same for other spectrometers. Parameters for each dimension are specified via arguments prefixed by **-x**, **-y**, **-z** and **-a** for the X-axis, Y-axis, Z-axis, and A-axis of the data. In order to accommodate padding which may have been performed by the spectrometer, there are separate parameters for the number of points stored in the input file, and the number of points actually acquired. The acquisition modes are specified by keywords such as "Sequential" (Redfield and Kunz, 1975), "Complex" or "States" (States et al., 1982), "TPPI" (Marion and Wüthrich, 1983), "States-TPPI" (Marion et al., 1989), etc., which define the Fourier transform mode and sign manipulation required; chemical shift calibration parameters are also recorded. The NMRPipe format output series is specified by the argument **-out**. Complete argument details are given in the Appendix.

---

nmrPipe program, each with different command-line arguments to select a processing function and optional parameters. For example, the following scheme applies a sinusoid-to-a-power window function (SP), zero fill (ZF), Fourier transform (FT), and deletes the imaginary part of the result (**-di**). In the absence of additional arguments, the processing functions in this scheme use default parameters, so that the SP function applies a sine-bell, the ZF function doubles the data size, and the FT function applies a complex forward transform:

```
nmrPipe   -fn SP     -in spec.fid \
| nmrPipe -fn ZF                  \
| nmrPipe -fn FT -di -out spec.ft
```

Considered in more detail, the scheme above consists of three instances of **nmrPipe**, connected by pipes, and running "simultaneously". This means that the UNIX operating system will alternate CPU time and other resources between the instances of **nmrPipe** while the scheme is executing. During execution, the first instance of nmrPipe reads a data vector from the input file "spec.fid", applies the window function SP, and writes the result vector to the pipeline. The second instance of **nmrPipe** reads the apodized vector from the pipeline when it becomes available, applies zero filling, and writes the result to the next stage of the pipeline. The third instance of **nmrPipe** reads the apodized, zero-filled vector from the pipeline when it becomes available, applies a Fourier transform, and writes the result to file "spec.ft"; meanwhile, the earlier instances of **nmrPipe** may have already begun to read and process the next vector. This procedure continues until all vectors have passed through the pipeline.

Spectrometer Format Conversion. Many of the advantages of the NMRPipe system stem from the fact that relevant acquisition parameters for all dimensions are established during conversion of data from the spectrometer format to the NMRPipe format. A typical 3D conversion script is

given in Figure 1.1. As shown, the conversion establishes the acquisition modes, data sizes and chemical shift calibration information for each dimension. The parameters are usually entered manually, but most of these could be extracted automatically from spectrometer parameter files (D. Benjamin, private communication).

The conversion programs themselves have been engineered to compensate for vendor-specific differences in the way that real and imaginary data are interleaved for each dimension, so that the converted result always provides the real and imaginary data for all dimensions in a predictable order. This allows subsequent processing schemes to be independent of spectrometer vendor. Currently, the NMRPipe system includes conversion facilities for GE Omega export format, JEOL GX and Alpha formats, Chemagnetics formats, Varian formats, and Bruker analog and digital sampling formats.

Like **nmrPipe**, the conversion programs are also implemented as UNIX filters. This means that the output stream of a conversion command can be sent directly into a processing pipeline, without the need to save an intermediate converted result on disk. It also means that a conversion program can read data produced by another pipeline command as an alternative to reading data directly from a file. One useful example of this is the ability to convert data directly from a tape drive by using a tape reading command (such as the UNIX command **dd**) as the data source. Another example is the ability to convert versions of spectrometer data which were compressed to save space by using a decompression command (such as the UNIX command **zcat**) as the data source.

Multidimensional Processing via Pipelines. The NMRPipe system includes two approaches to extend the pipeline method to multiple dimensions. One approach is to insert an appropriate matrix transpose command into the interior of a processing pipeline. Another approach is to use commands at the beginning or end of the pipeline which are capable of reading or writing vectors from an arbitrary dimension of a multidimensional spectrum. The two approaches can be used alone or in combination.

In a pipeline, a transpose function acts like a reservoir, which accumulates an intermediate result in memory before sending the transposed version down the remainder of the pipeline. Therefore, functions before a transpose receive and process a stream of vectors from a given dimension, and then functions after the transpose receive and process a stream of vectors from the exchanged dimension. Depending on which dimensions are being exchanged, a transpose function may require only enough memory for a 2D plane from the data, or it may require enough memory for an entire 3D or 4D matrix, so it is not generally applicable.

As noted above, the pipeline approach can be extended to multidimensional processing simply by adding two kinds of modules, as an alternative to in-memory transpose. The first module is a program at the head of the pipeline, which creates a data stream by reading vectors from a given dimension of a multidimensional input. The second module is a program at the tail of the pipeline, which gathers processed vectors and writes them to a given dimension of a multidimensional output. We have implemented two such programs, **xyz2pipe** and **pipe2xyz**, which are suitable for reading and writing multidimensional data in the multi-file 2D plane format suggested by Kay et al. (Kay et al., 1989). The programs take their names from the

```
xyz2pipe -in fid/hnco%03d.fid -x  -verb           \        Read Vectors from X-Axis
 | nmrPipe -fn SOL                                \        Solvent Filter
 | nmrPipe -fn SP -off 0.4 -end 0.98 -pow 2 -c 0.5 \       Window, 1st Point Scale
 | nmrPipe -fn ZF                                 \        Zero Fill
 | nmrPipe -fn FT                                 \        Fourier Transform
 | nmrPipe -fn PS -p0 43 -p1 0.0 -di              \        Phase, Delete Imaginaries
 | nmrPipe -fn EXT -x1 11ppm -xn 5.5ppm -sw       \        Extract Amide Region
 | nmrPipe -fn TP                                 \        2D Transpose X/Y
 | nmrPipe -fn SP -off 0.4 -end 0.95 -pow 1       \        Window
 | nmrPipe -fn ZF                                 \        Zero Fill
 | nmrPipe -fn FT                                 \        Fourier Transform
 | nmrPipe -fn PS -p0 -90 -p1 180 -di             \        Phase, Delete Imaginaries
 | pipe2xyz -out ft/hnco%03d.ft2 -y                        Write Vectors to Y-Axis

xyz2pipe -in ft/hnco%03d.ft2 -z -verb             \        Read Vectors from Z-Axis
 | nmrPipe -fn SP -off 0.4 -end 0.95 -pow 1 -c 0.5 \       Window, 1st Point Scale
 | nmrPipe -fn ZF                                 \        Zero Fill
 | nmrPipe -fn FT                                 \        Fourier Transform
 | nmrPipe -fn PS -p0 0.0 -p1 0.0 -di             \        Phase, Delete Imaginaries
 | pipe2xyz -out ft/hnco%03d.ft3 -z                        Write Vectors to Z-Axis
```

Figure 1.2. Annotated processing script for 3D amide-proton detected data, illustrating use of 2D transpose. In this scheme, the X-axis and Y-axis are read, processed, and written in the first pass, and the Z-axis is read, processed and written in the second pass. Each pass consists of a pipeline beginning with the **xyz2pipe** program and ending with the **pipe2xyz** program; these programs use the arguments **-x**, **-y**, **-z**, and **-a** to specify which dimension is being read or written. The input and output file series are specified by the template arguments **-in** and **-out**. Complete argument details are given in the Appendix.

nomenclature X-axis, Y-axis, Z-axis, A-axis, etc. which we use to describe the dimensions of the spectral data. Correspondingly, the dimension to be read or written is specified simply as a command-line argument **-x**, **-y**, **-z**, or **-a**. When reading or writing from a given dimension, the programs alter the sequential order of the other dimensions in the data stream in a regular, predictable way, by a multidimensional rotation. This means that schemes can be created to conserve the original data order, or change it to accommodate a particular processing or analysis strategy. The programs require at most enough physical memory to contain only four or so 2D planes from the data. In addition, the programs have been engineered to allow in-place processing (i.e. same input and output files), and to provide the correct combinations of real and imaginary data so that dimensions can be processed in any order. In the simplest multidimensional scheme, each dimension of the data is processed in a separate pass, which requires reading the entire input from disk, and writing the entire result. Such a scheme can be simplified and made more efficient by adding one or more in-memory transpose steps, which eliminates the need to save an intermediate result on disk. A typical 3D processing script employing a 2D transpose approach is shown in Figure 1.2. In this script, the X-axis and Y-axis are processed together in the first pass, and then the Z-axis is processed in a second pass. Such a script represents an effective compromise between disk access and physical memory use, since in practice only a small number of 2D planes are being manipulated in memory at any given time by the various programs in the pipeline. If large amounts of physical memory are available, schemes with 3D or 4D in-memory transpose steps can also be constructed, again reducing the need to save intermediate results. The overall approach provides basic multidimensional schemes

```
bruk2pipe -in ser $ARGS                              \        Convert Bruker Format
| nmrPipe -fn SP -off 0.35 -end 0.95 -pow 2 -c 0.5 \        Window, Scale 1st Point
| nmrPipe -fn ZF -size 512                           \        Zero Fill
| nmrPipe -fn FT -di                                 \        Fourier Transform
| nmrPipe -fn TP                                     \        2D Transpose X/Y
| nmrPipe -fn SP -off 0.35 -end 1.0 -pow 1  -c 0.5 \        Window, Scale 1st Point
| nmrPipe -fn ZF -size 128                           \        Zero Fill
| nmrPipe -fn FT -di                                 \        Fourier Transform
| pipe2xyz -out ft/noe%02d%03d.DAT -y                        Write Vectors to Y-Axis

xyz2pipe -in ft/noe%02d%03d.DAT -z -verb             \        Read Vectors from Z-Axis
| nmrPipe -fn SP -off 0.35 -end 0.95 -pow 1 -c 1.0 \        Window
| nmrPipe -fn ZF -size 64                            \        Zero Fill
| nmrPipe -fn FT -di                                 \        Fourier Transform
| pipe2xyz -out ft/noe%02d%03d.DAT -z -inPlace               Write Vectors to Z-Axis

xyz2pipe -in ft/noe%02d%03d.DAT -a -verb             \        Read Vectors from A-Axis
| nmrPipe -fn SP -off 0.35 -end 0.95 -pow 1 -c 1.0 \        Window
| nmrPipe -fn ZF -size 64                            \        Zero Fill
| nmrPipe -fn FT -di                                 \        Fourier Transform
| pipe2xyz -out ft/noe%02d%03d.DAT -a -inPlace               Write Vectors to A-Axis
```

Figure 1.3. Annotated 4D format conversion and processing script for a (256*)(64*)(16*)(16*) point 4D 13C-13C correlated 1H-1H NOE FID, illustrating use of 2D transpose (the asterisk denotes complex data). Acquisition parameters have been abbreviated by $ARGS and phase correction steps have been omitted to save space. In this scheme, the results of the format conversion program **bruk2pipe** are sent directly to the processing pipeline without the need to save an intermediate converted FID on disk. The size of the final result is (512)(128)(64)(64) points. Processing time: 8 hr. 20 min. on a Sun Sparc 10 workstation.

---

which require only modest amounts of memory for 3D or 4D processing, but which can be altered easily to take advantage of large memory systems. Complementary examples in the case of 4D processing are given in Figures 1.3 and 1.4.

The script shown in Figure 1.3 converts and processes a 4D spectrum in three passes, using only 2D in-memory transpose. In this case, the spectrometer format conversion, X-axis processing, and Y-axis processing are all performed in the first pass, the Z-axis is processed in the second pass, and the A-axis is processed in the third pass. The corresponding script in Figure 1.4 performs the same processing, but it has been rearranged so that the spectrum is processed in only two passes by the addition of a 3D in-memory transpose function. The first pass performs the spectrometer format conversion and the processing for the X-axis, Y-axis and Z-axis. The A-axis is processed in the second pass. As these examples show, in-memory processing is achieved at the discretion of the user, simply by use of appropriate transpose functions. Only minor alteration of a given processing scheme is needed, and no reconfiguration or recompilation of the software is required. Instead, the transpose functions, like all other functions of the NMRPipe system, allocate suitable amounts of memory automatically.

Processing Functions and Options. The NMRPipe system makes use of a relatively small number of processing functions, but these are augmented by a variety of modes and options; the

```
bruk2pipe -in ser $ARGS                              \        Convert Bruker Format
| nmrPipe -fn SP -off 0.35 -end 0.95 -pow 2 -c 0.5 \        Window, Scale 1st Point
| nmrPipe -fn ZF -size 512                           \        Zero Fill
| nmrPipe -fn FT -di                                 \        Fourier Transform
| nmrPipe -fn YTP                                    \        2D Transpose X/Y
| nmrPipe -fn SP -off 0.35 -end 1.0 -pow 1  -c 0.5 \        Window, Scale 1st Point
| nmrPipe -fn ZF -size 128                           \        Zero Fill
| nmrPipe -fn FT  -di                                \        Fourier Transform
| nmrPipe -fn ZTP                                    \        3D Transpose X/Z
| nmrPipe -fn SP -off 0.35 -end 0.95 -pow 1 -c 1.0 \        Window
| nmrPipe -fn ZF -size 64                            \        Zero Fill
| nmrPipe -fn FT -di                                 \        Fourier Transform
| pipe2xyz -out ft/noe%02d%03d.DAT -z                        Write Vectors to Z-Axis

xyz2pipe -in ft/noe%02d%03d.DAT -a -verb             \        Read Vectors from A-Axis
| nmrPipe -fn SP -off 0.35 -end 0.95 -pow 1 -c 1.0 \        Window
| nmrPipe -fn ZF -size 64                            \        Zero Fill
| nmrPipe -fn FT -di                                 \        Fourier Transform
| pipe2xyz -out ft/noe%02d%03d.DAT -a -inPlace               Write Vectors to A-Axis
```

Figure 1.4. Annotated 4D format conversion and processing script for a (256*)(64*)(16*)(16*) point 4D 13C-13C correlated 1H-1H NOE FID, illustrating use of both 2D and 3D transpose. Acquisition parameters have been abbreviated by $ARGS and phase correction steps have been omitted to save space. This scheme performs the same processing as the script shown in Figure 3, but in this version, a 3D in-memory transpose is used to avoid saving one of the intermediate results. The size of the final result is (512)(128)(64)(64) points. Processing time: 7 hr. 55 min. on a Sun Sparc 10 workstation.

---

processing functions listed in Table 1.1 and include over 300 options and parameters. For example, the functions POLY (polynomial fitting) and LP (linear prediction) each have a rich collection of parameters which allows them to perform many tasks. The POLY function can be used as a solvent filter in the time-domain, as well as for manual or automated correction according to a reliable in-house algorithm, and the corrections can be limited to selected spectral regions if desired. The linear prediction function LP can be used to predict points in either the start, end, or interior of existing data, in backward, forward or mixed forward-backward mode, with or without mirror-image methods and root-reflection. In addition to this flexibility, the LP function has also been implemented using a matrix inversion procedure in place of the iterative (and often unstable) root-searching approach, making it especially robust (G. Zhu and A. Bax, unpublished results).

The NMRPipe processing functions make extensive use of default parameter settings. This helps to make argument lists more concise, since individual parameters can be adjusted while leaving default settings intact. For example, when used with no other arguments, LP will apply linear prediction and root reflection with 8 complex coefficients to extend the original data to twice its size. The number of coefficients (the LP order) can be changed via the **-ord** option, and the number of predicted points can be changed independently via the **-pred** parameter. Mirror image LP can be selected simply by adding either flag **-ps0-0** or **-ps90-180** to any LP command line, depending on whether data have no acquisition delay, or a half-dwell delay.

Table 1.1. Processing functions of the nmrPipe program[a]

Name	Function	Comments
MAC	Macro Interpreter	User-Written functions in a subset of C
FT	Fourier Transform	Complex, Real, Inverse, Sign Adjust, Auto mode, etc.
HT	Hilbert Transform	Ordinary, Mirror Image, Auto Mode
LP	Linear Prediction[b]	Forward-Backward[c], Mirror-Image[d], etc.
MEM	Maximum Entropy Method[e]	1D to 4D, Two Channel[f], Deconvolution[g]
EM	Exponential Window	First point scaling, Inverse mode
GM	Lorentz/Gauss Window	First point scaling, Inverse mode
TM	Trapezoid Window	First point scaling, Inverse mode
SP	Sine to a Power Window	First point scaling, Inverse mode
ZF	Zero Fill	Inverse mode
EXT	Extract a Region	By Points, Hz, PPM, %, or Left, Right, etc.
PS	Phase Correction	Frequency shift, Inverse Mode
MC	Modulus Calculation	Modulus or Power Spectrum
SOL	Solvent Filter	Time-Domain Convolution[h]
POLY	Polynomial Solvent Filter	Time-Domain Polynomial Subtraction[i]
POLY	Polynomial Baseline Correction	Manual or Automatic[j], All or selected region
MED	Model-Free Baseline Correction	Automatic Median Method[k]
BASE	Linear Baseline Correction	Manually selected series of regions
CBF	Constant FID Correction	DC Correction of FID
QART	Quad Artifact Reduction[l]	Manual or Automatic
SMO	Smoothing Convolution Filter	Adjustable filter length and coefficients
TP	2D X/Y Transpose	In-memory; identical to YTP
ZTP	3D X/Z Transpose	In-memory, All combinations of real and complex data
ATP	4D X/A Transpose	In-memory, All combinations of real and complex data
REV	Reverse Data	Updates calibration
LS RS CS	Left Shift, Right Shift, Circular Shift	Updates calibration, Can invert signs of shifted data
FSH	Shift via Fourier Transform	Provides non-integer shifts
SHUF	Various Shuffling Functions	Complex Interleave, Byte Swap, etc.
SIGN	Various Sign Manipulations	Negate all, Negate half, Sign alternate, etc.
DX	Derivative	Derivitive by Central Difference
INTEG	Integral	Integral by Simple Sum
COADD	Simple Real Linear Combination of Rows	Combination of points, vectors, or planes, N Inputs, One Output
QMIX	General Complex Linear Combination	Combination of points, vectors, or planes, N Inputs, M Outputs
SET	Set Data to Constant (Also ADD, MULT)	All data or specified region

[a] Several functions are described in more detail in the Appendix. [b] Kumaresan and Tufts, 1982; Barkhuijsen et al., 1985; Stephenson, 1988; Hoch, 1989; Olejniczak and Eaton, 1990; Zhu and Bax, 1992A,B. [c] Delsuc et al., 1987; Zhu and Bax, 1992. [d] Zhu and Bax, 1990. [e] Maximum Entropy Reconstruction (Sibisi, 1983; Skilling and Bryan, 1984; Hore, 1985; Laue et al., 1985; Stephenson, 1988; Kauppinen and Saario, 1993; Schmieder et al., 1994) is implemented according to the method of Gull and Daniell (Gull and Daniell, 1978; Wu, 1984). [f] Laue et al., 1985; Hoch et al., 1990. [g] Ni and Scheraga, 1986; Ni et al., 1986; Mazzeo et al., 1988. [h] Marion et al., 1989. [i] Callaghan et al., 1984. [j] Details of automated baseline detection are given in the Appendix entry for function POLY. [k] Friedrichs, 1995. [l] Parks and Johannesen, 1976; the automated mode uses a grid search to minimize the integral of an interactively selected artifact.

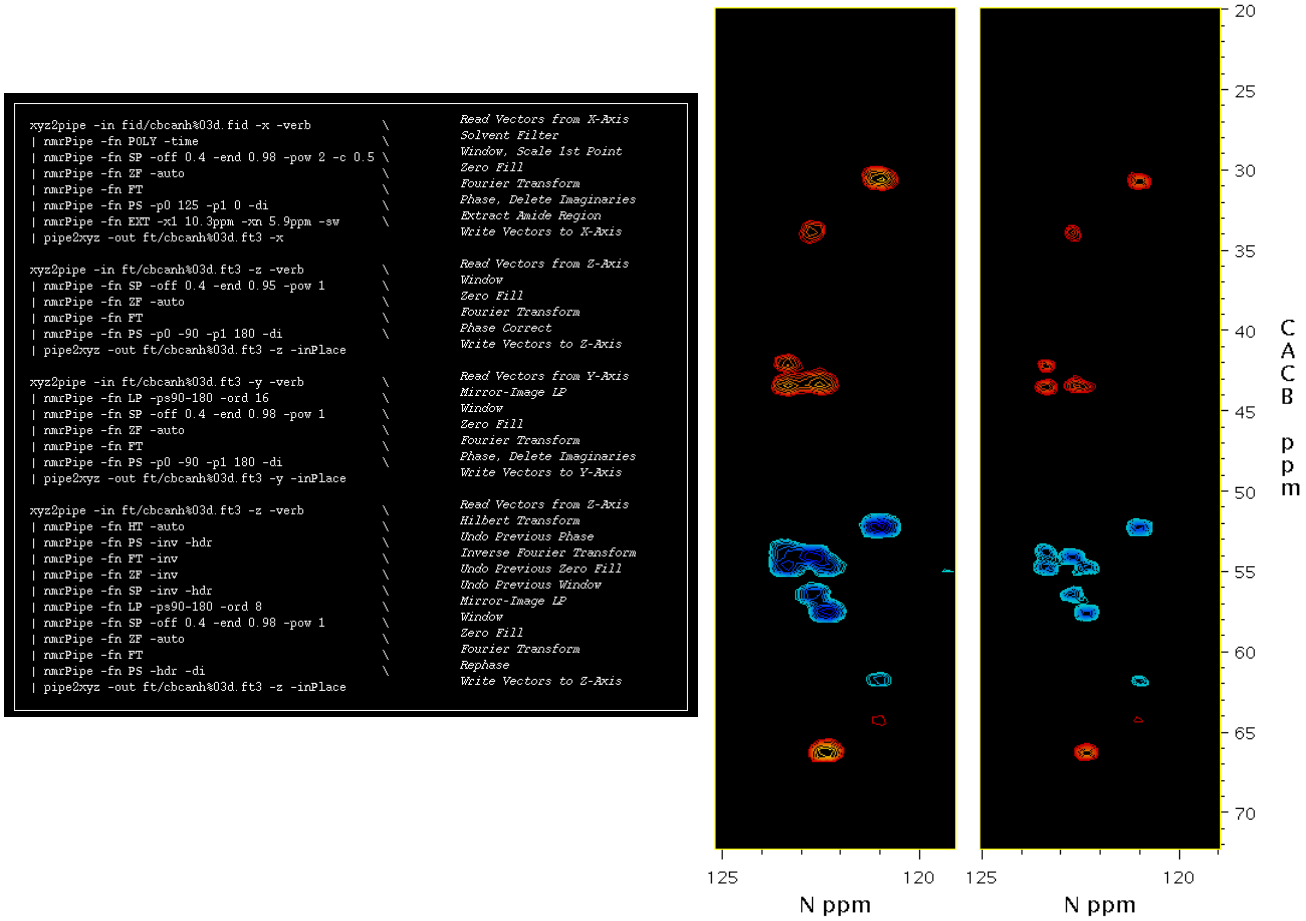
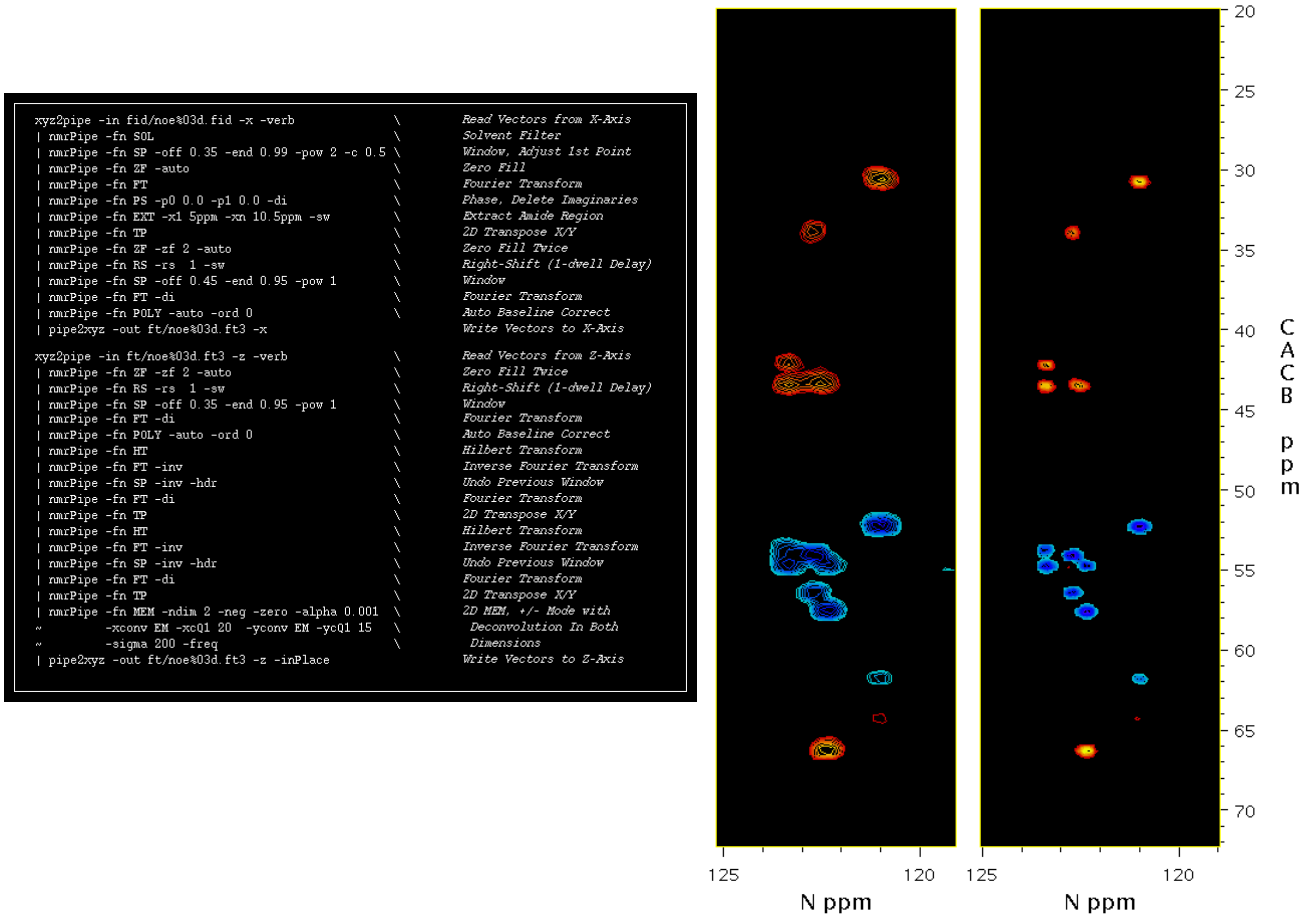


Figure 1.5. Annotated 3D processing script and example result for amide-detected data, illustrating the use of inverse processing features in a linear prediction scheme. The scheme took 4 hr. 55 min. to perform on a Sun Sparc 10 workstation with a 3D CBCA(CO)NH FID of (512*)(52*)(32*) points. The result is based on an intermediate amide-proton dimension size of 1024 points, yielding a 3D spectrum of (299)(256)(128) points after extraction of the amide-proton region and deletion of imaginary data. In the scheme, LP is used on the indirectly detected Y-axis and Z-axis of the data. This scheme is arranged so that when LP is applied to double the size of a given dimension, the other dimensions have been completely processed with a window function, zero filling, and phasing. This localizes the signals as much as possible in the other dimensions and thus simplifies the signal content of the dimension to be predicted (Kay et al., 1991). In the scheme, the X-axis is processed in the first pass, the Z-axis is processed in the second pass, the Y-axis is extended via LP and processed in the third pass, and the Z-axis is inverse-processed, extended via LP, and reprocessed in the fourth pass. The contour plots show a comparison of a region in a CBCANH spectrum reconstructed by an ordinary Fourier transform (left), and by the Linear Prediction scheme (right). Since this scheme increases spectral intensities, the two contour plots are drawn with different level settings.

Many of the functions exploit or update the spectral header parameters during processing. For example, apodization, zero-fill, and phase correction details are recorded, and chemical shift calibrations can be updated automatically by any function which extracts or shifts the data. The functions also keep track of the valid time-domain size of the data, as influenced by time-domain shifts or frequency-domain extractions. Where appropriate, parameters can be specified in PPM or Hz as well as in points.



```
xyz2pipe -in fid/noe%03d.fid -x -verb              \     Read Vectors from X-Axis
| nmrPipe -fn SOL                                  \     Solvent Filter
| nmrPipe -fn SP -off 0.35 -end 0.99 -pow 2 -c 0.5 \     Window, Adjust 1st Point
| nmrPipe -fn ZF -auto                             \     Zero Fill
| nmrPipe -fn FT                                   \     Fourier Transform
| nmrPipe -fn PS -p0 0.0 -p1 0.0 -di               \     Phase, Delete Imaginaries
| nmrPipe -fn EXT -x1 5ppm -xn 10.5ppm -sw         \     Extract Amide Region
| nmrPipe -fn TP                                   \     2D Transpose X/Y
| nmrPipe -fn ZF -zf 2 -auto                       \     Zero Fill Twice
| nmrPipe -fn RS -rs  1 -sw                        \     Right-Shift (1-dwell Delay)
| nmrPipe -fn SP -off 0.45 -end 0.95 -pow 1        \     Window
| nmrPipe -fn FT -di                               \     Fourier Transform
| nmrPipe -fn POLY -auto -ord 0                    \     Auto Baseline Correct
| pipe2xyz -out ft/noe%03d.ft3 -x                  \     Write Vectors to X-Axis

xyz2pipe -in ft/noe%03d.ft3 -z -verb               \     Read Vectors from Z-Axis
| nmrPipe -fn ZF -zf 2 -auto                       \     Zero Fill Twice
| nmrPipe -fn RS -rs  1 -sw                        \     Right-Shift (1-dwell Delay)
| nmrPipe -fn SP -off 0.35 -end 0.95 -pow 1        \     Window
| nmrPipe -fn FT -di                               \     Fourier Transform
| nmrPipe -fn POLY -auto -ord 0                    \     Auto Baseline Correct
| nmrPipe -fn HT                                   \     Hilbert Transform
| nmrPipe -fn FT -inv                              \     Inverse Fourier Transform
| nmrPipe -fn SP -inv -hdr                         \     Undo Previous Window
| nmrPipe -fn FT -di                               \     Fourier Transform
| nmrPipe -fn TP                                   \     2D Transpose X/Y
| nmrPipe -fn HT                                   \     Hilbert Transform
| nmrPipe -fn FT -inv                              \     Inverse Fourier Transform
| nmrPipe -fn SP -inv -hdr                         \     Undo Previous Window
| nmrPipe -fn FT -di                               \     Fourier Transform
| nmrPipe -fn TP                                   \     2D Transpose X/Y
| nmrPipe -fn MEM -ndim 2 -neg -zero -alpha 0.001  \     2D MEM, +/- Mode with
~         -xconv EM -xc01 20  -yconv EM -yc01 15   \      Deconvolution In Both
~         -sigma 200 -freq                         \      Dimensions
| pipe2xyz -out ft/noe%03d.ft3 -z -inPlace         \     Write Vectors to Z-Axis
```


Figure 1.6. Annotated 3D processing script and example result for amide-detected data, illustrating the use of inverse processing features in a 2D Maximum Entropy Reconstruction scheme. The scheme took 16 hr. 45 min. to perform on a Sun Sparc 10 workstation for a 3D 15N-NOE FID of (512*)(128*)(64*) points. The result is based on an intermediate amide-proton dimension size of 1024 points, yielding a 3D spectrum of (420)(512)(128) points after extraction of the amide-proton region and deletion of imaginary data. In the scheme, 2D MEM is applied to planes in the indirectly detected Y-axis (^{1}H) and Z-axis (^{15}N) of the data, which were each acquired with a one-dwell delay. The scheme is arranged to temporarily reorder the data so that the MEM function is provided with a stream of data planes from the indirect dimensions (the original Y-axis and Z-axis). The indirect dimensions are first processed by right-shifting, Fourier processing, and automated zero-order baseline correction to compensate for the one dwell time acquisition delay; the Fourier processing includes use of window functions to increase the effectiveness of the automated baseline correction. The planes are then reprocessed so that they are presented for Maximum Entropy reconstruction already phased, baseline corrected, and extensively zero-filled, but transformed without any window functions. Additional argument details are given in the Appendix. The contour plots show a comparison of a region in a CBCANH spectrum reconstructed by an ordinary Fourier transform (left), and by the MEM scheme (right). Since this scheme increases spectral intensities, the two contour plots are drawn with different level settings.

Inverse Processing. Multidimensional enhancement schemes commonly call for inverse processing, so several functions have been implemented with an inverse mode for convenience. For instance, window functions support an inverse mode which divides by the window function, and zero filling supports an inverse mode which strips away previous zero padding. These conveniences make it possible to construct complicated inverse processing protocols concisely, and if parameters are selected appropriately the original data can commonly be recovered to a

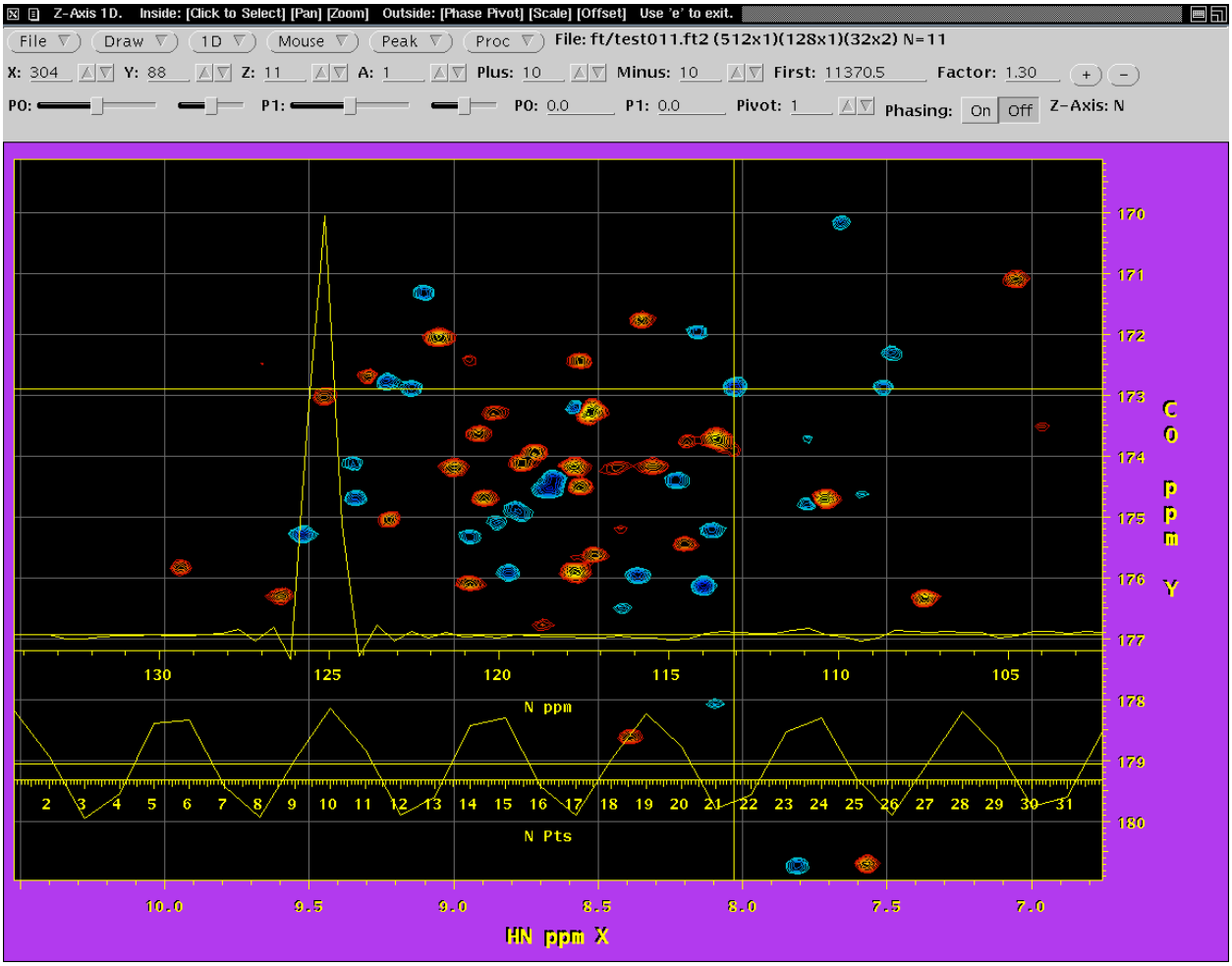


Figure 1.7. The NMRDraw graphical processing and analysis interface, illustrating interactive processing of a 1D vector extracted from the Z-axis of a 3D interferogram. The topmost border of the program window describes the current functions of the mouse buttons. The command panel along the top contains graphical tools for executing commands, selecting the region of data to view, setting contour parameters, and adjusting phase values. The 2D contour display shows the eleventh transformed HN/13CO plane from a partially transformed HNCO spectrum (Z-axis (15N) data is still in the time-domain), with positive data drawn in a continuous range of blue colors, and negative data in a range of red colors. The small window over the contour display at the top left is a pop-up command area for entering **nmrPipe** processing commands. The cross-hair superimposed over the contour display shows the user-selected location for extraction of the Z-axis 1D vector. The time-domain vector itself is drawn along the bottom of the display. The Fourier processed version of the vector, also prepared interactively, is drawn above the 1D time-domain data.

precision of better than one part in 105. Examples are given in Figure 1.5 and 1.6 of forward/inverse processing scripts for applying linear prediction and Maximum Entropy reconstruction in the two indirectly detected dimensions of a 3D spectrum. In the case of the LP scheme in Figure 1.5, forward and inverse processing is used to minimize the number of signals which must be predicted in any given vector in order to increase the prediction's stability and incidentally decrease the time required (Kay et al., 1991). In the case of the MEM scheme in Figure 1.6, forward and inverse processing are used to allow a more stable automated baseline correction by using data processed with window functions, before data is reprocessed without window functions for Maximum Entropy reconstruction.

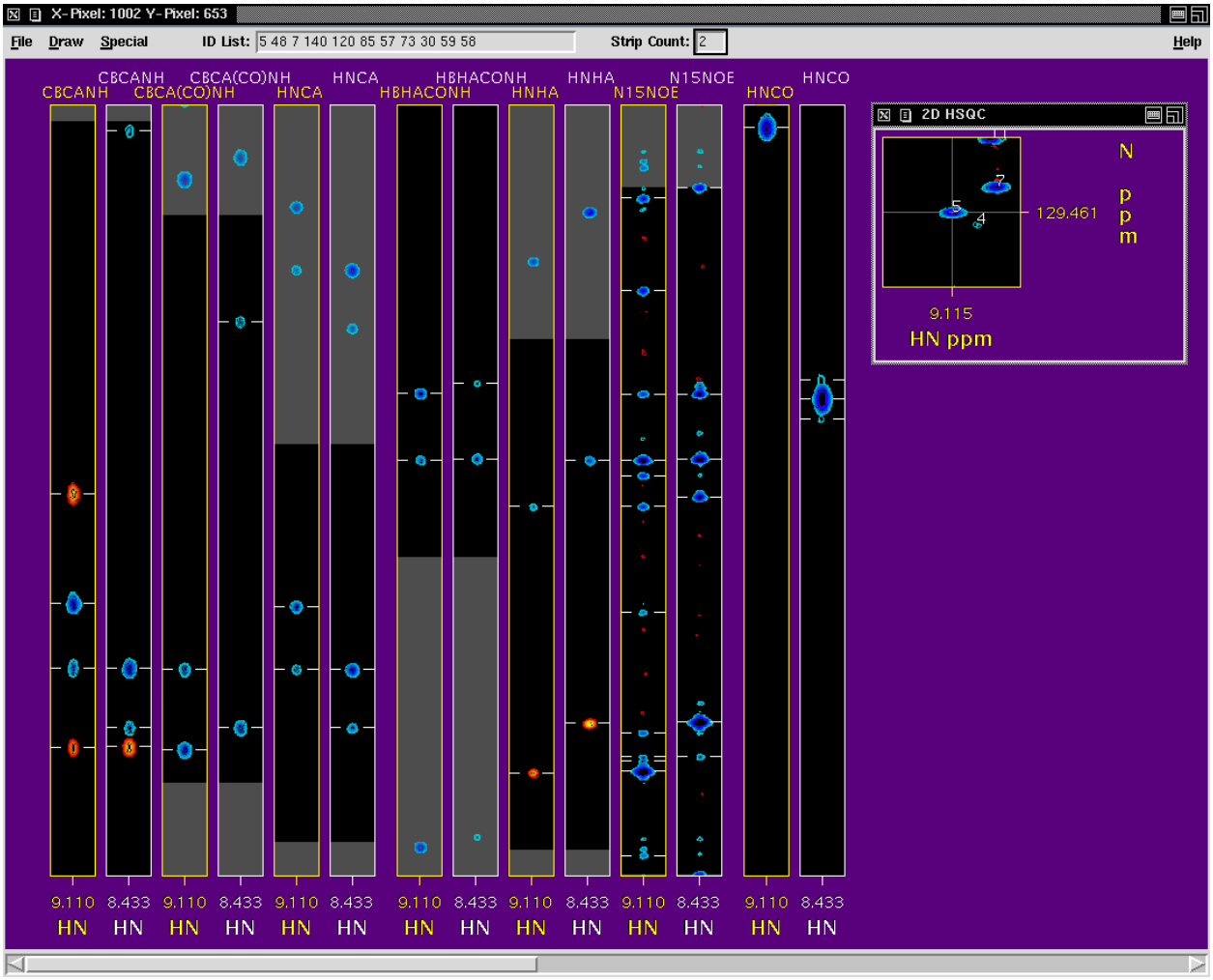


Figure 1.8. A prototype NMRWish TCL application for browsing through strips from related amide-detected 3D experiments, and automatically suggesting which strips might be the next or previous ones in the sequential assignment. Strips from comparable spectra are extended automatically by refolding the data so they can be displayed with uniform chemical shift ranges where needed; the portions of the data which have been extended redundantly this way are shaded in gray. In the example show here, a series of related strips from two different $^{1}H/^{15}N$ positions are displayed. The inset at upper right displays the corresponding location from a 2D $^{1}H/^{15}N$ correlated spectrum for the first set of strips being viewed.

New Capabilities and Data Formats. One of the special advantages to the pipeline approach is the ease and flexibility with which new capabilities and data formats can be implemented. The primary data format of the NMRPipe system consists of one or more 2D file planes, each with a 2048-byte header followed by four-byte floating-point spectral data values in a sequential order. Other multidimensional data formats can be adapted simply by use of alternative programs to read or write data at the head or tail of a pipeline; the submatrix formats of the powerful spectral analysis programs NMRView (Johnson and Blevins, 1994) and ANSIG (Kraulis, 1989; Kraulis et al., 1994) have been accommodated by their authors in this way. To facilitate work of this kind, the standard NMRPipe installation includes C source code for the spectrometer format

conversion programs, file header interpretation and general I/O utilities, and the multidimensional I/O programs **xyz2pipe** and **pipe2xyz**.

New processing functions can be implemented as simple UNIX filter programs which can be inserted directly in the pipeline data stream, without the need to alter the **nmrPipe** program itself. As an alternative to writing a complete program, nmrPipe includes the MAC function, a macro interpreter which implements a sub-set of the C programming language, augmented with a variety of vector processing commands. The interpreter was implemented primarily for development purposes, using the UNIX compiler generator Yacc (Johnson, 1986). The macro language allows direct manipulation of the data points, and the possibility to control the details of file I/O during processing. In its default mode, the MAC function will apply the contents of a user-written macro to every 1D vector in the given dimension, so that new functions can be implemented simply by placing a list of vector functions or other processing steps in a text file. This provides a convenient way to prototype new processing applications. For example, special processing steps for drift correction, gradient-enhanced data (Cavanagh et al., 1991; Palmer et al., 1991; Kay et al., 1992) and Bruker DMX digitally oversampled data have been developed this way.

Parallel Processing. Many possible approaches can be envisioned for performing a multidimensional processing task in parallel over a network of computers or on a multi-CPU machine. By modifying only the multidimensional I/O programs (**xyz2pipe** and **pipe2xyz**), we have implemented one simple but broadly applicable approach, which relies only on standard UNIX network file sharing, and avoids the need for special machine-specific parallel compiling or configuration of software. This particular implementation uses static load balancing, which means that the amount of data to be processed by each computer is fixed at the outset of a task, and therefore there is no compensation for possible changes in CPU performance during the course of a calculation. In practice, the user performs parallel processing by creating a single script which processes a complementary subset of a complete spectrum depending on which computer is used to execute it; the same script is then executed simultaneously on all CPUs involved. The division of data is performed automatically according to a user-supplied list of computers and their approximate relative speeds, so that only minor modification of an ordinary scheme is needed to convert it to a parallel scheme.

Graphical Interface. As noted by Güntert et al. (1992), it is a difficult task to create and maintain a single, integrated spectral graphics and processing program. Nevertheless, in our experience we have found it essential to be able to graphically inspect the FID data, to interactively choose processing parameters, and to examine intermediate processing results on the workstation screen or in hard copy. In an attempt to meet these needs, we have developed a supplemental graphics interface called NMRDraw, using the X11 network graphics library and the XView graphical interface toolkit (Heller and Van Raalte, 1993). The program, shown in Figure 1.7, currently runs on Sun, SGI, DEC Alpha, IBM RS6000 UNIX, and Linux PC computers.

The NMRDraw program provides facilities for inspecting raw and processed data via 1D and 2D slices or projections from all dimensions, as well as a macro editor for creating and executing

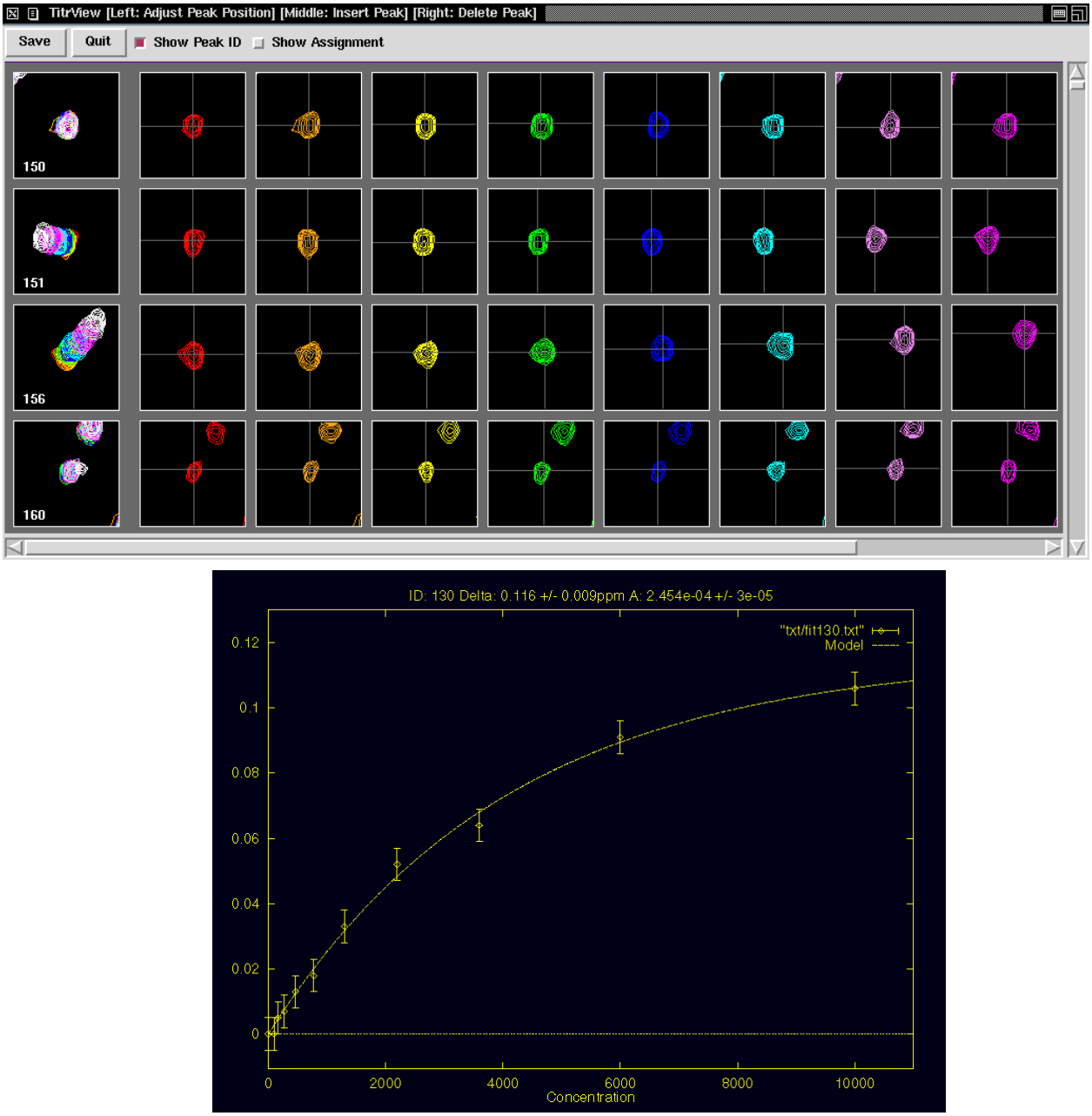


Figure 1.9. An NMRWish TCL application for interactive analysis of HSQC titration curves (Zhou et al., 1996; Johnson et al., 1996). The upper window shows the interactive graphical interface for following a peak's change in position over the spectra in the series. A given row follows a peak change in position over the series, as indicated by cursor lines; the first entry in each row shows the given region drawn in overlay for all spectra in the series. The peak positions are determined by automated peak detection, correlated by database manipulations, and can be adjusted manually. The lower window shows the results of a corresponding binding affinity analysis also performed by the application.

complete multidimensional processing scripts. NMRDraw also allows real-time display and interactive phasing of an arbitrary number of 1D slices selected from any dimension of the spectrum and displayed simultaneously. Interactive 1D processing is performed via program-

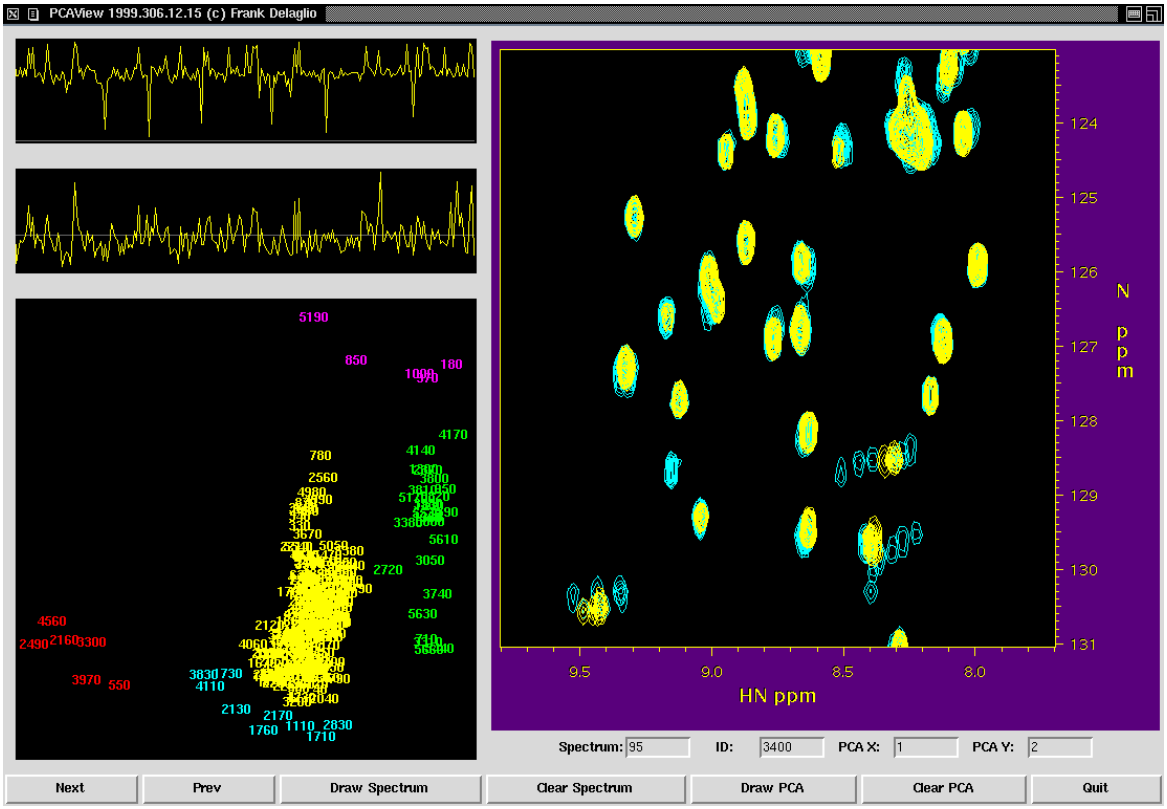


Figure 1.10. An NMRWish TCL application implementing the method of Ross and coworkers for analyzing HSQC drug screening series (Shuker et. al., 1996) by Principal Component Analysis (PCA) (Ross et al., 2000). The application uses multivariate statistics to provide a graphical summary of the similarities and differences in a collection of related spectra, in this case a series of automatically processed HSQC spectra of roughly 200 samples of a target protein mixed with various small molecules. Each number in the scatter plot at lower left represents an entire HSQC spectrum in the series. The distance between entries in the scatter plot relates to the degree of similarity between spectra. The spectral window on the right allows one or more spectra or regions from the series to be viewed in overlay.

---

controlled pipelines to **nmrPipe**, providing the functionality of both graphics and processingwithout the need to incorporate the two in a single program. In keeping with this philosophy of well-separated applications, the data extraction and display facilities of NMRDraw can also be operated remotely by two-way pipelines to other programs, in order to construct graphical spectral analysis schemes. In addition, NMRDraw makes use of this facility so that more complicated tasks such as automated peak detection can be performed in background by other programs.

During the initial development of NMRPipe, and independently of our graphics interface development, spectroscopists at a test site for the NMRPipe system used the TCL/TK graphics command language to create interactive **nmrPipe** schemes (N. Tjandra, private communication). TCL provides a method to build graphics applications using shell-scripts alone, without the need to write, compile, and link a complete program (Ousterhaut, 1994). TCL/TK was first introduced to the NMR community as the basis of the powerful and widely used NMRView spectral analysis package (Johnson and Blevins, 1994). Since TCL/TK provides an easy method for building graphical applications at the UNIX shell-script level, it is ideal for use with

NMRPipe schemes, which also operate at the shell-script level. Using this approach, it was possible to create a graphical interface that provides routine format conversion and processing without the requirement for users to edit shell-scripts directly.

NMRWish. A major feature of the TCL/TK language is that it was intentionally designed to allow additional functions to be added to the command language, so that it could be customized for particular scientific and engineering tasks. A key program in the current implementation of NMRPipe is NMRWish, our version of the TCL/TL interpreter program "wish". NMRWish has been extensively customized to include facilities for extracting and displaying spectral regions, strips, and projections, with multi-window interactive graphics and PostScript output. It also includes 1D-4D peak detection with discrimination of random noise and truncation artifacts, and a simple relational database engine for manipulation of peak tables, assignments, and other spectral information. This customized interpreter, based on a standard command language, makes it possible to develop a variety of targeted NMR applications quickly and efficiently.

Some examples include a prototype backbone assignment tool (Figure 1.8), a system for analysis of HSQC titrations (Figure 1.9), and an implementation of an HSQC screening method based on multivariate statistics (Figure 1.10). Examples of PostScript output produced by NMRWish scripts are also shown in the Appendix. Other users have independently implemented systems for both backbone and sidechain assignment using NMRWish (D. Kohda, private communication).

Companion Software. In addition to the processing, display, and analysis facilities described above, the NMRPipe system includes several other applications, such as algebraic combination of spectra, simulation of time-domain or frequency-domain data from peak tables, automated multidimensional Non-Linear Least-Squares modeling of spectral line shapes, general-purpose functional fitting with Monte Carlo error estimation, and Principal Component Analysis. Stand-alone functions for examining and adjusting spectral header parameters are also included. Processed data from the NMRPipe system can be used directly with the PIPP/CAPP system for computer-assisted spectral analysis (Garrett et al., 1991). In combination with other software systems, NMRPipe has been used to help generate roughly 10-20% of the NMR structures deposited in the Brookhaven Protein Databank since the beginning of 1994.

Results and Discussion

The NMRPipe system has been applied in over 300 laboratories, and has proven easy to use, robust, and thorough in its capabilities. In our direct experience, as a processing engine it is also more efficient than previous approaches we have tried, and it has successfully been adapted to new data formats and acquisition modes. Because of its design principles, it has been easy to port and maintain this system on several different computer platforms, and to use it as a platform to develop and coordinate a variety of graphics and analysis applications.

Processing times on various computers for a typical 3D application are given in Table 1.2, and times for some other applications are given in Figures 1.3, 1.4, 1.5, and 1.6. The main source of performance overhead in these examples is due to the multi-plane data format and to pipeline communication. We decided to use the multi-plane format in order to accommodate preexisting

Table 1.2. 3D Processing times on various computer systems[a]

Computer Type	Time, sec
Linux PII 400MHz Laptop PC	48
SGI R10000 CPU	73
SGI Challenge, 4 R4400 CPUs [b]	154
SGI Challenge, 4 R4400 CPUs [c]	187
HP 9000/755	239
SGI Indigo	408
DEC Alpha 3000 [d]	487
SGI Challenge, 1 R4400 CPU [e]	525
Sun Sparc 10	644
IBM RS6000/530	1128
Sun Sparc 2	1208
Sun Sparc 1	1864
Convex C3830 [f]	2146

[a] Processing of a (512*)(64*)(32*) Point HNCO FID using the script given in Figure 2. Times reported are actual times elapsed. No special attempt was made to vectorize or parallelize the code; only ordinary optimizing compilers were used. During processing, each axis size was doubled by zero filling, yielding a spectrum of (417)(128)(64) points after extraction of the amide-proton region and deletion of imaginary data. [b] This time is based on a distributed version of the processing script, which divides each processing task into four equal parts, one for each CPU. [c] This time is based on an ordinary version of the processing script, whose components are distributed automatically between CPUs by the operating system because they are separate programs. [d] This version of the software was compiled with a four-byte floating point compatibility mode, which is roughly half as fast as the best speed of the CPU. [e] This time is based on execution of the script on a single CPU. [f] This time was measured under heavy loading (44 users).

software which also used this format. While this format has the advantage of simplicity, it is not necessarily the best choice in all respects, especially for 4D data, since the number of file planes can become very large and relatively inefficient to manipulate. But, since the source and destination formats are independent of the processing pipeline itself, other formats could easily be implemented, for instance by substituting the programs which read and write multi-plane format data by programs which read and write submatrix format data. In this respect, the processing pipeline can be thought of as a format-independent processing engine. The overhead due to data format, while measurable, is not important in many cases. For example, consider the processing times for two versions of 4D processing given in Figures 1.3 and 1.4. The version in Figure 1.4 is 25 minutes faster than the version in Figure 1.3 because it avoids one intermediate read/write of the 4D data. But, this improvement amounts to only a 5% decrease in the overall processing time. This also suggests that an all-in-memory approach such as the one employed by PROSA (Güntert et al., 1992) is not always an advantage, since the performance gain will often be small, but the physical memory requirements (> 1024 megabytes in this case) may constitute a

Table 1.3. Network-distributed parallel processing times[a]

Number of Processors	Time, min	Parallel Efficiency [b]
1	119	100%
2	59	99%
3	40	99%
4	30	99%
5	26	91%

[a] Processing times for a Z-axis Linear Prediction Application on a Network of SGI Indigo Computers. An interferogram of (512)(128)(32*) points was extended to (512)(128)(64*) points by forward-backward LP with 8 complex coefficients, and the result was doubled by zero filling and Fourier processed. The processing task was divided equally on each computer involved. [b] The parallel efficiency is computed assuming that the ideal increase in processing speed is proportional to the number of computers used.

serious obstacle. As noted by Levy et al. (1986), use of virtual memory does not provide an effective solution to this problem, although in years to come, computers with multi-gigabyte physical memory capacity may become commonplace.

Overhead due to pipeline communication and management is an intrinsic aspect of the NMRPipe system. This overhead is examined in Figure 1.11. As shown, the overhead time increases roughly linearly with the number of programs in the pipeline. For the Sun Sparc 10 workstation, this overhead contributes about 2 min. to a typical 3D processing scheme. This amounts to about 15% of the time used for ordinary Fourier processing, and an insubstantial percentage for linear prediction applications.

A distinct performance advantage to the NMRPipe system is the ease with which processing tasks can be distributed over more than one CPU or workstation. The processing scripts themselves are naturally parallel, since they consist of several programs running simultaneously. So, as shown in Table 1.2, an ordinary NMRPipe scheme can show speed improvements on a multi-CPU computer without the need for special machine-specific compiling or vectorization, since the various programs in the script will be distributed at the discretion of the operating system. In the case shown for the four-CPU SGI Challenge, this simple approach yielded a 70% parallel efficiency compared to the same scheme executed on one CPU. In addition, the facilities of the NMRPipe system allow a processing task to be explicitly distributed by the user, an approach which yields even better performance, and still avoids the need for machine-specific optimization. An example is given in Table 1.3, which shows the results of a network-distributed processing application, with an efficiency of over 90% on five SGI workstations.

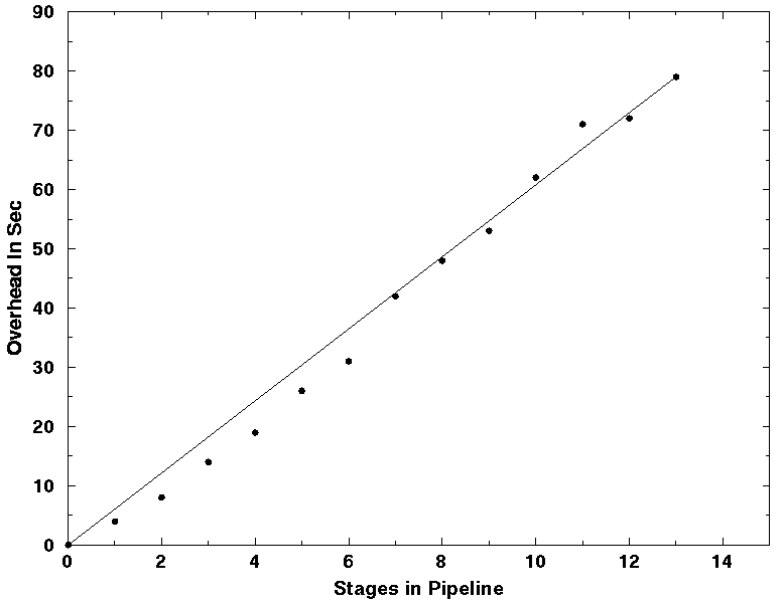


Figure 1.11. Overhead processing time due to pipeline communication and management for a 32 MB data set measured on a Sun Sparc 10 workstation. As shown, the overhead time increases roughly linearly with increasing numbers of functions in the pipeline. In this case, the best-fit least-squares line, also shown, represents an overhead of 0.19 sec/MB for each additional stage in the pipeline.

Concluding Remarks

The NMRPipe implementation of multidimensional spectral processing and analysis via UNIX pipes and scripts provides a solution which is comprehensive, easy to use, flexible, extensible, and efficient. It naturally accommodates parallel processing approaches, and encourages and supports use of well-separated applications for graphics and analysis. Since its inception, NMRPipe as served as the basis for a wide variety of NMR applications. As the NMRPipe approach is complementary to existing methods which rely on monolithic programs, its unique combination of advantages is likely to prove increasingly useful as biomolecular NMR continues to advance.

Acknowledgement

In the course of the past years, many people have assisted in the development, evaluation, and refinement of the NMRPipe software presented; for this invaluable assistance, thanks goes to S. Archer, D. Benjamin, R.A. Byrd, G.M. Clore, M. Donlan, N. Farrow, J. Forman-Kay, S. Gagne, D. Garrett, H. Grahn, A.M. Gronenborn, T. Harvey, H. Hatanaka, E. Henry, M. Ikura, Y. Ito, L.E. Kay, W. Klaus, D. Kohda, J. Kordel, A. LiWang, R. Martino, L. Nicholson, I. Pelczer, R. Powers, M. Shirakawa, S. Tate, N. Tjandra, H. Tsuda, T. Yamazaki, and T. Yamazaki.

2. Measurement of proton-proton couplings from regular 2D COSY spectra

Introduction

^{1}H-^{1}H J couplings provide important dihedral information, which is widely used during NMR structure determination. Numerous methods have been proposed for measuring such couplings, primarily aimed at cases where the ^{1}H multiplet is too overlapped or broad in the 1D spectrum to yield resolvable splittings. These include homonuclear J-spectroscopy (Aue et al., 1976) homonuclear E.COSY methods (Griesinger et al., 1987), heteronuclear E.COSY (Willker and Leibfritz, 1992), triple resonance E.COSY (Montelione and Wagner, 1989), quantitative J correlation (Vuister and Bax, 1993; Grzesiek et al., 1995), comparison of cross sections through in-phase and antiphase cross peaks (Titman and Keeler, 1990; Prasch et al., 1998), and the so-called DISCO method (Kessler et al., 1985), which relies on the same principle. Two new quantitative J correlation methods, based on constant-time COSY can also yield useful information (Tian et al., 2000, Wu and Bax, 2001). None of these approaches is fully satisfactory, however, when working with macromolecules at natural abundance.

Here, we describe a simple but effective method to directly obtain quantitative J-coupling information from cross peaks in regular phase-sensitive COSY spectra by amplitude-constrained multiplet evaluation (ACME). Problems with deriving J splittings from antiphase COSY spectra have long been recognized, (Neuhaus et al., 1985; Smith et al., 1991; Ludvigsen et al., 1991) and relate to the fact that the antiphase peak-to-peak splitting in a COSY type spectrum is a function of both the line width and the magnitude of the passive and active J couplings. Some methods derive the J couplings from least squares fitting of a cross peak to a convolution of an antiphase and several in-phase splittings, using the intensity of the cross peak as a variable, non-constrained parameter in the fit. However, an increase in line width results in a larger antiphase separation of lower intensity for the cross peaks, whereas an increase in active unresolved J coupling results in a similar increase in antiphase splitting but larger cross peak intensity. Without constraining the amplitude of the cross peak, this makes it difficult to separate the effects of line width and active coupling, resulting in poorly determined J values. Inherently better algorithms fit either a group of multiplets or the entire NMR spectrum simultaneously (Madi and Ernst, 1988; Yang and Havel, 1994; Yang et al., 1994; Schmidt, 1998). These algorithms implicitly use the fact that all fitted multiplets have the same intrinsic intensity (i.e., the same integrated intensity if all splittings were in-phase). The effectiveness of these methods is due to the fact that active coupling values in one multiplet appear as passive couplings in related multiplets. Typically, such methods require extensive definitions of the spin topologies and chemical shift assignments of all related multiplets, and may require some degree of multiplet fine structure. Furthermore, because of the potentially large numbers of spectral parameters to be fit, these methods usually require prior knowledge or tight constraints for many spectral parameters in order to make optimization tenable. This has so far limited the application of these methods.

Here, we demonstrate that by simply constraining the multiplet intensity, the ACME method makes it possible to extract couplings accurately and conveniently by fitting individual multiplets or small clusters of overlapping multiplets, without the need to consider all related multiplets simultaneously. To establish the multiplet intensity constraint, we use the information that the intrinsic intensity of all multiplets in the spectrum is the same, and identical to that of the in-

phase diagonal multiplets for the case where a 90˚ mixing pulse is used. This procedure is very simple from a user perspective and circumvents the convergence problem, while retaining a sharp and accurate minimum for the fitted active J coupling. This method does not require fine structure for the antiphase multiplets that need to be fitted, and therefore is also ideally suited to fitting the very complex unresolvable multiplets that are typically obtained in COSY spectra of macromolecules weakly oriented in a liquid crystalline phase.

Ignoring cross-correlated transverse relaxation, and assuming the weak coupling limit, the time domain signal of the A-spin diagonal, $S_{AA}(t_1,t_2)$, and AX cross peak, $S_{AX}(t_1,t_2)$, in a COSY experiment are given by:

$$S_{AA}(t_1,t_2) = S_o \Pi_k \cos(\pi J_{Ak} t_1) \cos(\Omega_A t_1) \exp(-t_1/T_{2A}) \times \Pi_k \cos(\pi J_{Ak} t_2) \exp(i\Omega_A t_2) \exp(-t_2/T_{2A}) \quad (2.1A)$$

$$S_{AX}(t_1,t_2) = S_o \sin(\pi J_{AX} t_1) \Pi_{k\neq X} \cos(\pi J_{Ak} t_1) \cos(\Omega_A t_1) \exp(-t_1/T_{2A}) \times \sin(\pi J_{AX} t_2) \Pi_{q\neq A} \cos(\pi J_{Xq} t_2) \exp(i\Omega_X t_2) \exp(-t_2/T_{2X}), \quad (2.1B)$$

where the products extend over all spins *k* coupled to A, and spins *q* coupled to X, Ω_A and Ω_X are angular chemical shifts of spins A and X, and T_{2A} and T_{2X} are the A- and X-spin transverse relaxation times. It is clear from Eq. 2.1B that the initial cross peak "buildup" in the (t_1,t_2) time domain is simply $S_o \sin(\pi J_{AX} t_1) \sin(\pi J_{AX} t_2) \approx S_o \pi^2 J_{AX}^2 t_1 t_2$. So, if S_o is known, the buildup of the time domain A-X cross peak signal, which simply can be obtained by inverse Fourier transformation of a resolved cross peak multiplet, provides a unique value for J_{AX}, with effects from passive couplings and T_2 only occurring at later times. As a result, J_{AX} is an orthogonal variable relative to both the passive couplings and T_2, but parallel to S_o. It is therefore critical that an accurate value for S_o is obtained prior to fitting the cross peaks. If the delay between scans is sufficiently long, the same S_o value applies for all multiplets within a given molecule and can be obtained from the initial ($t_1 = t_2 = 0$) time domain amplitude of either a diagonal multiplet (Eq. 2.1B), or the entire normalized time domain signal. The need for constraining the amplitude (scale factor) in the fit has been mentioned before (Madi and Ernst, 1988; Yang and Havel, 1994; Yang et al., 1994; Schmidt, 1998), but was less critical in the application to fitting of the fine structure of multiple related multiplets of a given spin system. So, the main difference relative to earlier fitting procedures is that in ACME fitting the intrinsic signal intensity, S_o, is held constant for all multiplets in the spectrum at a value determined experimentally from the diagonal (see below), and fine structure of the multiplet is not required for accurate measurement of a coupling. In contrast to most other fitting procedures, only the fitted value for the active coupling is meaningful. Fitted passive couplings and decay rates are parallel variables in parameter space and their optimum values generally do not provide useful coupling information. In practice, the actual fit is carried out in the frequency domain, using multiplet models generated by numerical Fourier transformation of the model time domain forms of Eq. 2.1. The numerical Fourier processing of the model function is performed automatically during the fit according to the zero filling and window functions that were applied to the experimental data.

Methods

Water suppression in phase-sensitive COSY spectra can be a problem, as solvent presaturation may unevenly affect the longitudinal ^1H magnetization through spin diffusion involving spatially proximate exchangeable protons, or protons resonating in the immediate vicinity of the H_2O signal. For this reason, we preferred to record the COSY spectrum used for this work in D_2O.

The ACME fitting procedure itself has been implemented via a graphical interface constructed using NMRWish, a companion package to the NMRPipe processing and analysis system (Delaglio et al., 1995; Cornilescu et al. 1999). NMRWish is a version of the TCL/TK script interpreter "wish" (Ousterhout, 1994; Johnson and Blevins, 1994), which has been augmented to include facilities for spectral display and manipulation, as well as relational database functions for manipulating spectral parameters, molecular structures, etc. Use of the graphical interface to extract couplings typically involves the following steps:

1. A cluster of one or more multiplets is selected interactively from a spectral display, and an expanded view of the selected region is shown.
2. The approximate center of each multiplet in the selected region is defined by manually positioning a cursor. Since the exact position of the multiplet center can be adjusted during the fit, its initial location is not critical.
3. An interactive parameter page is shown for each signal, and is used to specify the parameters to be included in the signal description, the initial values of variable parameters, and of values that are held constant during the fit. For example, one may specify how many passive couplings will be included in each dimension of the multiplet, and rough estimates for their initial values. In order to accommodate signals from equivalent spins such as CH_3 groups more easily, one can also specify an integer intensity scaling factor for the number of equivalent superimposed signals, or an integer exponent to a given modulation term for the number of equivalent couplings. As these passive couplings and decay rates are orthogonal in parameter space relative to the active coupling, their initial values are not critical and only become relevant when a cross peak displays fine structure, i.e., more than four components per multiplet..
4. Once the parameters of the signal models are defined, the fit is performed. In most cases, the signal positions, widths, and couplings are all allowed to vary, and only the intensity is held constant. The final multiplet model and residual is then displayed along with the experimental region, and the values shown in the parameter pages are updated to reflect the results of the fit.
5. The results are evaluated, either according to their χ^2 value or, for cases where not all signals in the selected region are included in the model, by inspection of the residual spectrum. If inspection of the results indicates that the model is reasonable, the fit is accepted and the results are automatically recorded in a table. If assignments are available, these can be entered and recorded along with the coupling values.

The fitting procedure itself is presently implemented via a macro interpreter that generates the model function at each iteration. This allows for substantial flexibility, for example by making it straightforward to adjust the model to account for dephasing delays inserted prior to the acquisition, or to extend the method to 3D data. However, use of an interpreted fitting function

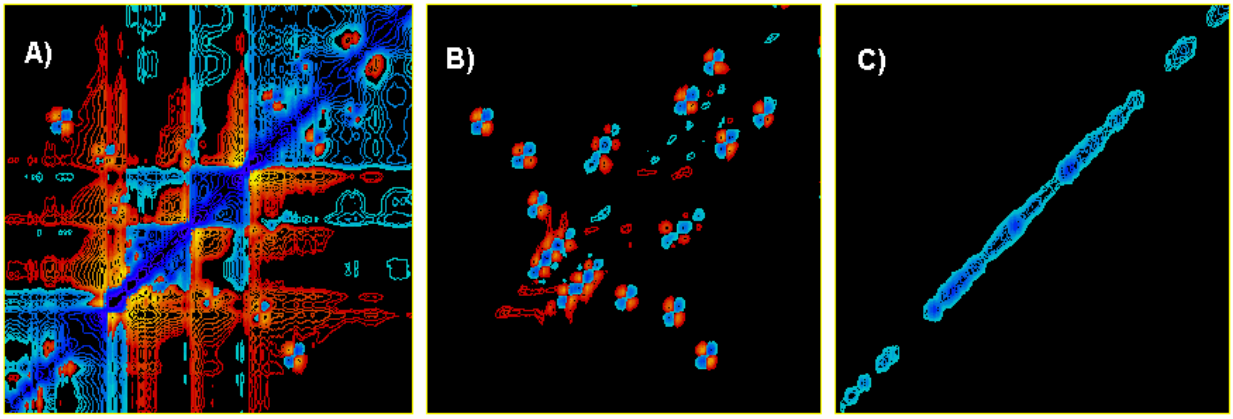


Figure 2.1. Diagonal region of the 800 MHz phase-sensitive COSY spectrum of human ubiquitin, recorded in D_2O, spanning 3.3 to 2.4ppm. (A) Result of ordinary Fourier processing. (B) Processed with the diagonal-suppression method described in the text. (C) The diagonal-only spectrum, which is the difference between spectra A and B, rephased to absorptive mode. The diagonal-only spectrum is displayed with contour levels ten times higher than the settings used for A and B.

makes this implementation relatively slow, with computation times of several seconds for fitting a single multiplet, and a minute or more for complicated clusters of multiplets. Nevertheless, this is still fast enough for the method to be convenient, especially in light of the fact that no complicated definition or set-up is required in order to extract couplings.

Prior to fitting the cross peaks, it is useful to remove the diagonal signals, which as a result of their long dispersive tails can interfere with the cross peaks. This can simply be done by subtracting the diagonal region of the COSY spectrum where the diagonal is phased to be absorptive, followed by Hilbert transformation and rephasing to make the cross peaks absorptive (Pelczer, 1991). An alternative method for doing this, which has the additional advantage of avoiding discontinuities near the edge of the cut-out diagonal region, has been implemented in the NMRPipe processing software. This procedure works by temporarily shifting the diagonal to the center of the spectrum so that it can be removed by traditional numerical solvent suppression methods (Marion et al., 1989; Callaghan et al., 1984). Variations on this scheme can also produce a complementary spectrum that contains only the absorptive diagonal signals, as described below.

Numerical diagonal suppression in COSY. In the case of traditional COSY spectra, in both frequency dimensions the diagonal is 90^o out of phase relative to the cross peaks. Therefore, when phasing the cross peaks to be absorptive, the tails from the dispersive diagonal peaks obscure many of the nearby cross peaks, as shown in Figure 2.1. Mueller suggested an elegant way of subtracting the diagonal by conducting a COSY experiment without a mixing pulse (Mueller, 1987). Alternatively, such a no-mixing pulse "COSY" spectrum can be generated from a single FID (Marion and Bax, 1988), although this is not effective for samples containing liquid crystal such as bicelles or phages. Instead, it is possible to attenuate the diagonal signals numerically using a scheme that combines frequency shifting with methods usually used for solvent signal subtraction, such as the time-domain convolution method (Marion et al., 1989) and time-domain polynomial subtraction (Callaghan et al., 1984). These solvent subtraction methods are designed to suppress low-frequency signals, i.e. those at the center of the spectrum.

Therefore, if we temporarily shift columns of a 2D spectrum in such a way that the diagonal signal is moved to the center, these low-frequency suppression methods can also be used for diagonal suppression. This requires an amount of shifting that varies according to the position of the diagonal signal in any given column (vector). Since frequency shifting can be achieved by a first-order phase correction in the time domain, the diagonal suppression steps can all be carried out in the t_1 time domain. In practice, after the t_2 Fourier transformation and absorptive phasing, the frequency shifting, diagonal subtraction, and frequency unshifting are applied to the t1 vectors of the 2D data matrix. An example UNIX processing pipeline script for this 2D COSY processing scheme is shown below. In the scheme, the directly-detected dimension is processed as usual, but with the signals of the first t_1 increment phased absorptively after the first FT. Then, each t_1 vector in the indirect dimension is shifted via phase correction so that its diagonal signal becomes on-resonance, a "solvent" filter is applied by subtracting a best-fit 4th or 5th order polynomial, and the vector is shifted back again. The solvent filter methods sometimes distort intensities at the head and tail of the time-domain data, leading to baseline curvature near the region of signal suppression in the spectrum. So in this case, since the signals of interest are sine-modulated, the leading points of the time-domain data can be attenuated with a suitable window without affecting the cross peak signals. Here, we used a custom roll-off function with a cosine-squared form. After these diagonal suppression steps, the t_1 vectors are then processed as usual, and finally, the direct dimension is rephased to dispersive mode. The script is annotated to describe the processing steps, with the functions applied to the indirect dimension given in bold:

```
nmrPipe -in test.fid                                \ Read FID
| nmrPipe -fn SP -off 0.5 -pow 2 -end 0.95 -c 0.5   \ Window, 1st Point Scale
| nmrPipe -fn ZF -size 2048                         \ Zero Fill
| nmrPipe -fn FT -verb                              \ Fourier Transform
| nmrPipe -fn PS -p0 138.7 -p1 18.1 -di             \ F2 Phase Correction
| nmrPipe -fn TP                                    \ 2D Transpose
| nmrPipe -fn MAC -macro diagShift.M                \ Shift Diagonal to Center
| nmrPipe -fn POLY -time                            \ Polynomial Solvent Filter
| nmrPipe -fn MAC -macro diagUnShift.M              \ Shift Diagonal Back
| nmrPipe -fn MAC -macro csr.M -var wide 10         \ Attenuate Head of FID
| nmrPipe -fn SP -off 0.5 -pow 2 -end 0.95 -c 1.0   \ Window
| nmrPipe -fn ZF -size 2048                         \ Zero Fill
| nmrPipe -fn FT -verb                              \ Fourier Transform
| nmrPipe -fn PS -p0 -110 -p1 220 -di               \ F1 Phase Correction
| nmrPipe -fn POLY -auto                            \ Auto Baseline Correction
| nmrPipe -fn TP                                    \ 2D Transpose
| nmrPipe -fn POLY -auto                            \ Auto Baseline Correction
| nmrPipe -fn PS -p0 90 -ht -di                     \ Rephase F2 to Dispersive
   -out test.ft2 -verb -ov                            Write Spectrum
```

The shifting operations were implemented using the macro interpreter facility of NMRPipe, which allows custom functions to be included in processing schemes without the need to compile new versions of the software. The macro language is a subset of the C programming language, augmented with a number of vector processing functions. The shifting macro `diagShift.M` is shown below; it is invoked once for each t1 vector in the interferogram. The macro language provides many automatically defined variables that describe the current data. In this case, the variables `xSize` and `ySize` give the number of points in t1 and f2 respectively, and `yLoc` gives the index number of the t1 vector being processed, in the range of 1 to `ySize`. The

variables `rdata` and `idata` are arrays containing the real and imaginary parts of the current vector. Finally, phase( ) is a vector processing function which applies a zero and first order correction as given in degrees:

```
xMid   = 1 + xSize/2;
yMid   = 1 + ySize/2;

slope  = (1 - xMid)/(yMid - 1);
offset = -slope*yMid;
shift  = slope*yLoc + offset;

p0     = 0.0;
p1     = -360.0*shift;

(void) phase( rdata, idata, xSize, p0, p1 );
```

The cosine-squared roll-off function was also implemented as a macro, `csr.M`, shown here. This function multiplies the data by a function that increases from zero to one over the first several points in the data, and leaves the remainder of the data unchanged. The number of points attenuated is specified by the variable `wide`, which is taken from the command-line of the processing script:

```
for( i = 0; i < wide; i++ )
   {
    c = cos( 0.5*PI*i/wide );
    w = 1.0 - c*c;

    rdata[i] *= w;
    idata[i] *= w;
   }
```

Figure 2.1 shows a comparison of processing results for a diagonal region in the 2D COSY of ubiquitin. The contour plot 2.1A shows the result of conventional processing, which predictably is dominated by the diagonal, making the crosspeaks very hard to see or measure. The contour plot 2.1B shows the same data, as processed with the diagonal suppression scheme. The crosspeaks here are easily seen, and in this case even those close to the diagonal can be quantified. The contour plot 2.1C shows the diagonal signal alone, as generated by taking the difference between the original and diagonal suppressed data, then rephasing both dimensions to absorptive mode. In this case, the diagonal is well isolated and undistorted, and can also be quantified. A more extensive view of the diagonal suppressed COSY spectrum is given in Figure 2.2.

Prior to ACME multiplet analysis, the diagonal-only spectrum is analyzed. Fitting of a single isolated, in-phase diagonal multiplet is rather insensitive to initial line width or multiplicity, and can be used to obtain the intrinsic signal amplitude. This fit can be repeated for several diagonal signals in order to establish reproducibility. In the ubiquitin case, fitting six different diagonal

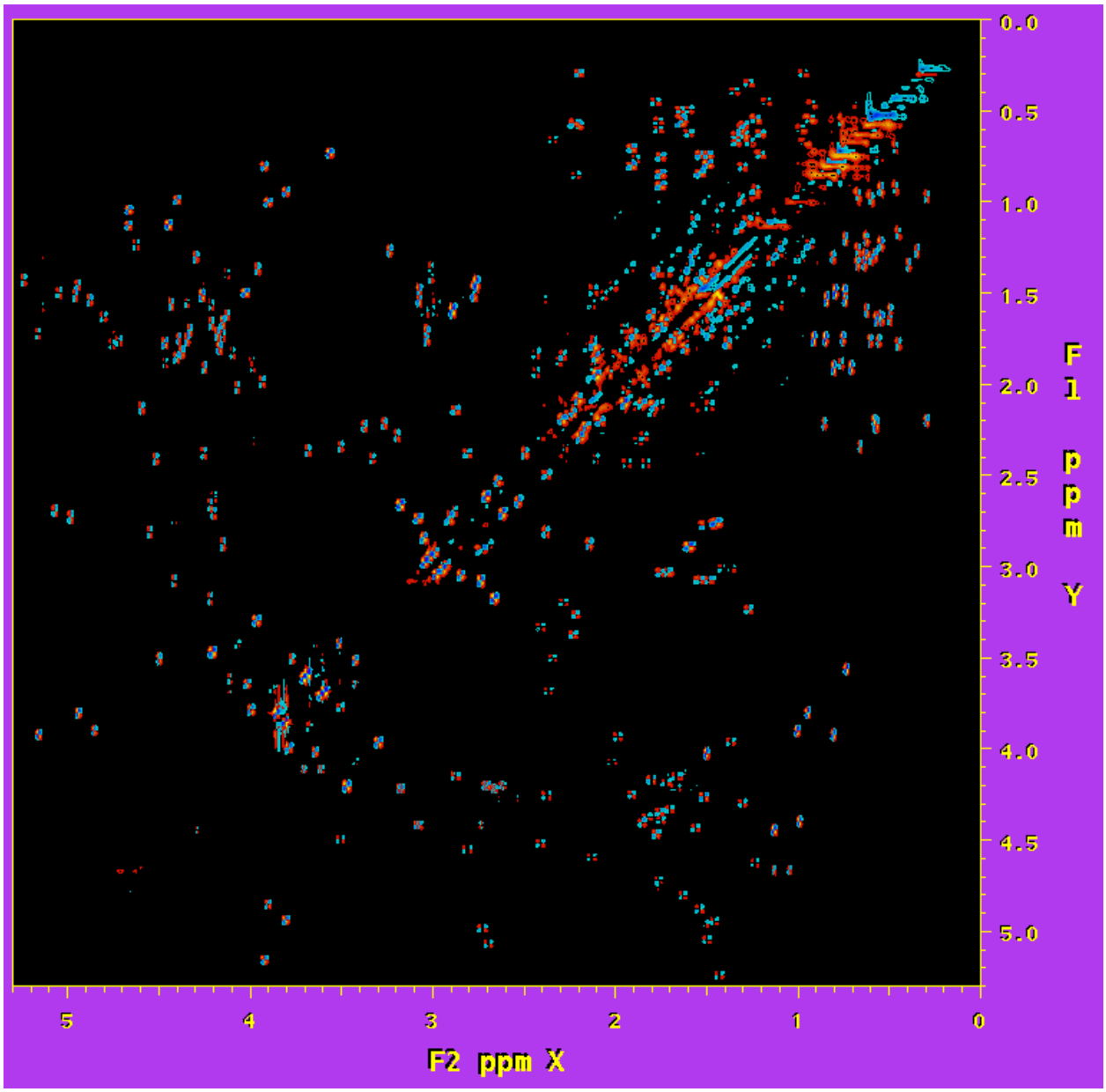


Figure 2.2. Region of the 800 MHz phase-sensitive COSY spectrum of human ubiquitin, recorded in D_2O, from which the dispersive diagonal signals have been removed using the procedure described in the text.

signals indicates that the amplitude could be determined to an accuracy of better than 5%. Alternatively, the integrated intensity of the entire diagonal subspectrum, or a fraction thereof, divided by the number of spins contributing to this diagonal can be used.

Figure 2.3 shows how the best-fit active J coupling for the ubiquitin Lys^6 $H^{\beta''}$-H^{α} COSY cross peak depends strongly on this intrinsic intensity. Figure 2.3A is the experimental multiplet, whereas Figures 2.3B-D are best-fitted simulated multiplets, where the intrinsic amplitude has been at its true value (Figure 2.3B) and at 10 and 100 times larger values (Figure 2.3C,D). The

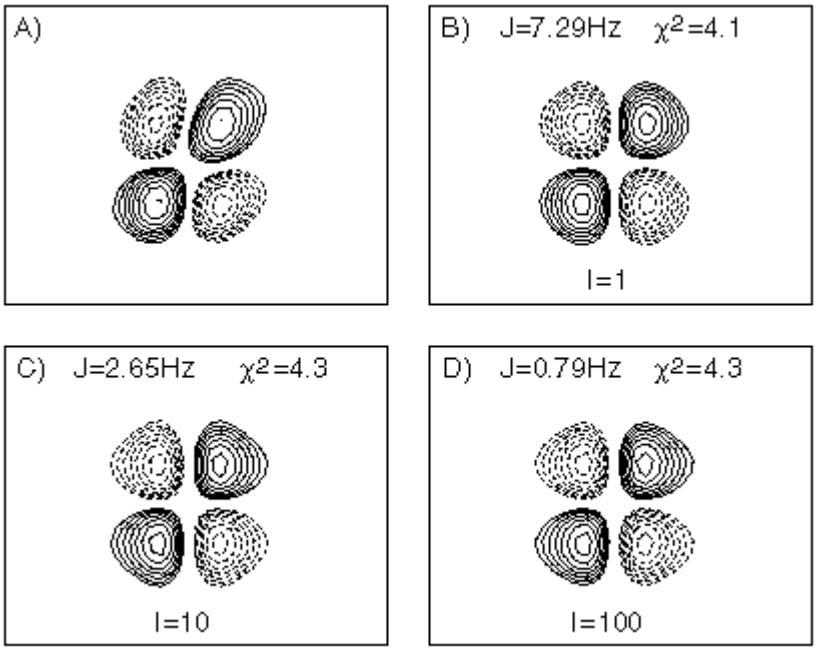


Figure 2.3. The H^{β}-H^{α} cross peak of ubiquitin Glu^{16} in the 800 MHz COSY spectrum. (A) experimental data, (B) best-fit simulated data using the correct intensity factor (I=1), and best fits when the intensity was constrained to be ten-fold (C) or 100-fold (D) larger. Note that the goodness of the fit (χ^2) is essentially the same for the three simulated spectra.

goodness of the fit (χ^2) for the multiplet shown is nearly indistinguishable, yet the magnitude of the active coupling decreases by almost an order of magnitude when the intrinsic intensity, I, is increased from 1 to 100. To a good approximation, in the limit where the line width is larger than the active J coupling, the best fitted coupling scales with the square root of the intensity, whereas the goodness of the fit remains comparable (Figure 2.3). This confirms that in the absence of amplitude information it is not possible to obtain an accurate J coupling from fitting a single antiphase COSY cross peak.

So, passive couplings can be kept either as fixed or as adjustable parameters during the fitting procedure. If kept variable, the accuracy of the resulting best-fitted *passive* couplings is poor, however, as the effect of an unresolved passive coupling is similar to that of the fitted natural line width parameter. Convergence of the least-squares minimizer is fastest when estimated approximate values for the passive couplings are entered as fixed non-variable parameters. This is typically the method in which we use the fitting procedure when no (partially) resolved passive splittings are observed in the cross peak.

The graphical interface for coupling extraction is shown in Figure 2.4. The window in Figure 2.4A shows a small region of the COSY spectrum of ubiquitin. The boxed region shows the group of multiplets manually selected for analysis. The window in Figure 2.4B contains the parameters of a fitted model for the first of five interactively selected signals. Parameters that are variable during the fit are marked by black checkboxes. The selected spectral region,

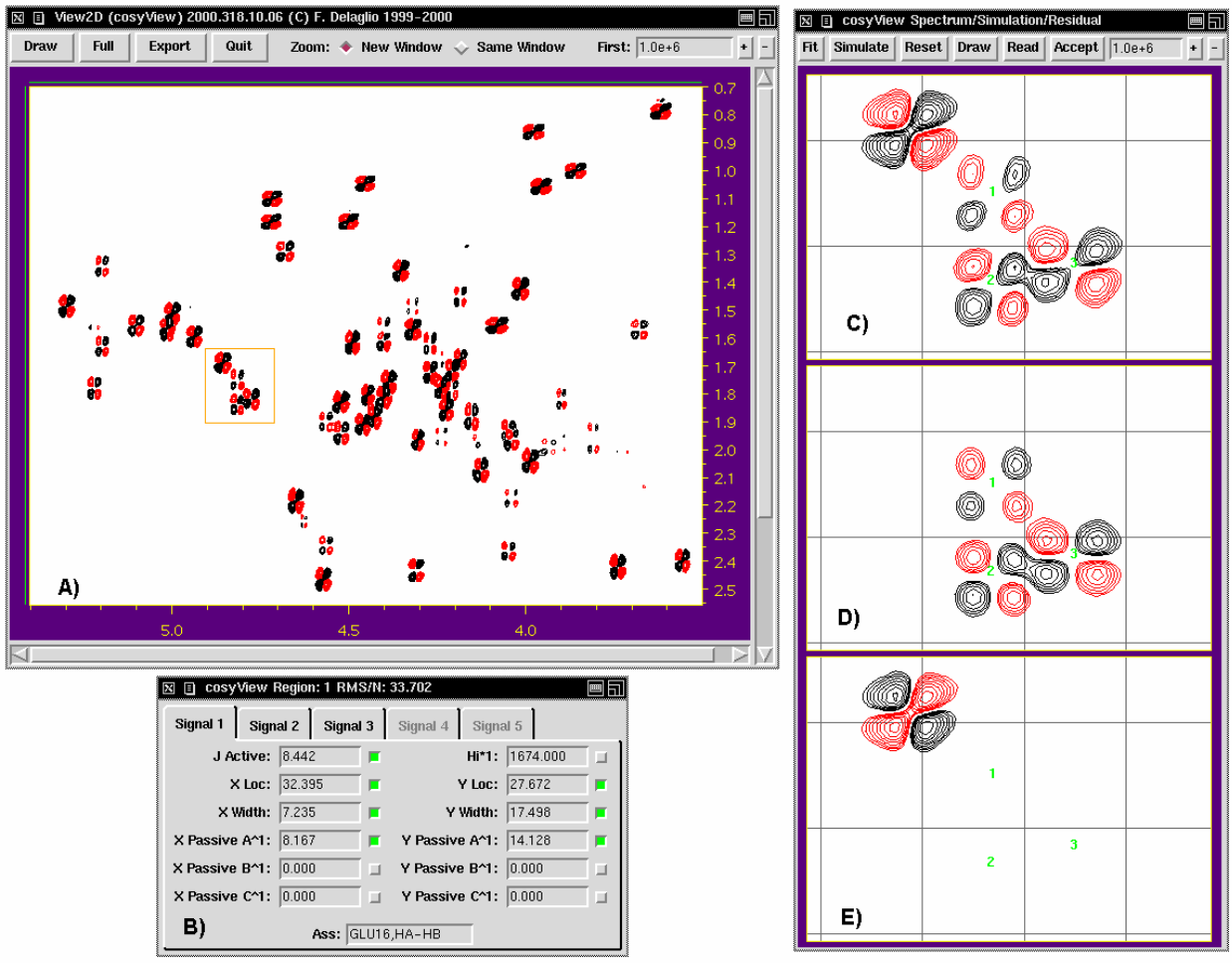


Figure 2.4. The ACME graphical interface for coupling extraction. (A) Region of the COSY spectrum with a zoom-box marking the spectral region on which multiplet fitting is to take place. (B) Parameter window that defines adjustable (green check box) and fixed parameters to be used in the fit for the first of up to five cross peaks. For passive couplings, the 'A^1' mark indicates the number of passive couplings of this size to be included in the simulation; i.e., for passive coupling to a methyl group this value is changed to 'A^3'. Similarly, the intensity parameter 'Hi*1' can be adjusted to 'Hi*3' for cross peaks involving a methyl group. Experimental data (C), best fitted data (D) and difference spectrum (E) when only the signals of the five rightmost cross peaks are entered in the parameter window and included in the fit optimization.

corresponding model, and residual spectrum are shown together in a third window (Figure 2.4C, 2.4D, and 2.4E). If multiple identical passive couplings are present, as for example in the case of the three H^{β}-H couplings in a H^{β}-H cross peaks of threonine residues, the multiplicity for a single passive coupling can be set to '3' in the parameter window, rather than defining three independent couplings. Similarly, when fitting a cross peak involving a methyl group, the intensity can be multiplied by three. The user interface currently allows for up to five multiplets with up to three independent passive couplings per dimension to be fit simultaneously. For larger numbers of cross peaks, the accuracy and convergence of fitted parameters decrease. As shown in Figure 2.4, not all cross peaks present in the selected window region (Figure 2.4C) need to be included in the fit. To illustrate this feature, only five of the eight cross peaks present

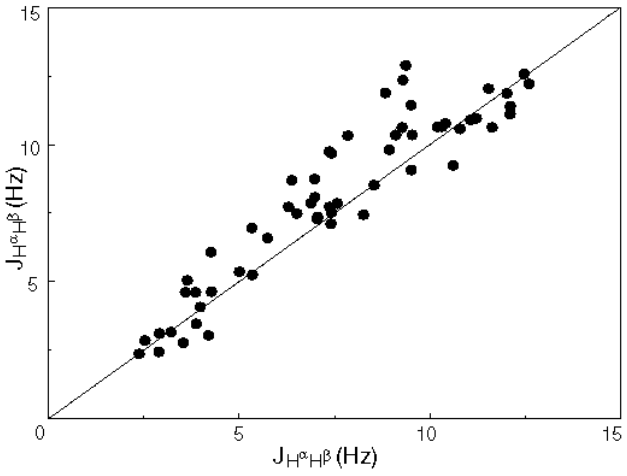


Figure 2.5. Comparison of 58 $^3J(H^\alpha,H^\beta)$ couplings in human ubiquitin, measured with the new fitting procedure (vertical axis) *versus* those measured previously with the HA(CA)HB experiment. The systematically smaller value for the HA(CA)HB derived couplings is attributed to the effect of passive spin-flips, which do not affect the result of the multiplet fitting procedure.

in Figure 2.4C were included, and the additional cross peaks remain present in the difference spectrum (Figure 2.4E).

When fitting spectra recorded in a dilute liquid crystalline phase (Tjandra and Bax, 1997), frequently the number of passive couplings will be much larger than three. However, from a practical perspective, only the largest passive couplings play a role in the fitting procedure and three passive couplings are more than sufficient for the wide variety of spin systems we have studied so far.

Results and Discussion

Figure 2.5 compares values for all 58 $^3J(H^\alpha,H^\beta)$ couplings previously measured with the HA(CA)HB experiment (Grzesiek et al., 1995) with those obtained from the new fitting procedure. On average, J values derived from the HA(CA)HB spectrum underestimate the true coupling if no correction is made for the finite life time of the spin state of the coupling partner. Because the new fitting procedure is not affected by spin-flips of the passive spin, which merely affect the fitted line-width, fitted J values tend to be larger. Overall, agreement is good (Pearson's correlation coefficient R=0.93) and comparable in quality to that between HA(CA)HB data and previous heteronuclear E.COSY measurements for ubiquitin. This indicates that the present approach for measuring these couplings is equally robust and therefore quite accurate. Any given coupling can be measured twice, from each of the two corresponding cross peaks. Reproducibility is invariably found to be quite high, with an rmsd of 0.7 Hz over the entire set of H^α/H^β cross peaks (N=78), indicating a random uncertainty of ±0.5 Hz in individual fits.

One detail that may require particular attention is the assumption of uniform intrinsic intensity. In order for this assumption to be valid, the spin system must be fully relaxed at the start of the COSY pulse sequence. Alternatively, two spectra with different interscan delays may be used, such that incomplete T_1 relaxation rates can be accounted for. As mentioned above, solvent presaturation may be problematic for exchangeable protons or signals close to the H_2O signal. In principle, double quantum filtering may also be used to attenuate the H_2O signal (Piantini et al., 1982). However, this decreases the inherent sensitivity of the COSY experiment two-fold and does not solve the dynamic range problem because the water signal remains present in individual transients. Also, it removes the amplitude information contained in the diagonal and therefore makes our fitting procedure less straightforward. In nucleic acids, partial exchange of base protons with solvent deuterons can result in erroneous amplitudes and thereby introduce errors in the derived couplings. The same, of course, applies to cross peaks to amides for proteins dissolved in D_2O.

The fitting procedure is based on the use of Eq. 2.1, i.e. on the assumption of first order, weakly coupled spectra. Fitting of cross peaks very close to the diagonal is affected by the diagonal removal routine described above, and therefore does not yield reliable results. In contrast, simulated results indicate that fitting of A-X or B-X cross peaks in an ABX spin system yields reliable results provided that $|\delta_A - \delta_B| > \sim 2\ J_{AB}$. So, although this approach is less rigorous than that developed by Schmidt (Schmidt, 1998) which takes the strong coupling into account in the model function, ACME is fully adequate for most cases of practical interest.

Concluding Remarks

The method described here is particularly useful for accurate and convenient measurement of ^1H-^1H couplings in molecules that are weakly aligned in a dilute liquid crystalline phase. These frequently give rise to completely unresolvable cross peak multiplets with more than half a dozen passive couplings, that are difficult to analyze accurately by the E.COSY method. The ACME program also contains the option to include the effect of delays preceding the t_1 and/or t_2 evolution period. Such delays may be desirable to allow dephasing of the rapidly decaying signals of bicelle or phage liquid crystal contributions. The principal disadvantage of the ACME method is the absence of sign information for the coupling involved. Several novel heteronuclear E.COSY-like methods have been presented recently that permit experimental measurement of both the sign and magnitude of ^1H-^1H couplings (Otting et al., 2000; Peti and Griesinger, 2000). However, without isotopic enrichment, such experiments are generally not applicable to macromolecules. If a reasonably accurate initial structure is available, this frequently can be used to determine the sign of the coupling. Alternatively, the structure calculation can use the absolute values of ^1H-^1H dipolar couplings as input restraints (Tjandra et al., 2000).

It can also be noted that the ACME is an excellent illustration of the unique philosophy and capabilities of the NMRPipe system, as it is a synthesis of methods for spectral processing, quantification, and interactive presentation.

3. Chemical shift database methods and protein backbone angles

Introduction

The strong dependence of isotropic chemical shifts on protein structure has long been recognized. In particular, the striking correlation between $^1H^\alpha$ chemical shift and secondary structure has been studied extensively (Pastore and Saudek, 1990; Williamson, 1990; Wishart et al., 1991; Ösapay K. and Case, 1993) and the $^1H^N$ shift was found to be sensitive to both hydrogen bonding and secondary structure (Pardi et al., 1983; Williamson, 1990; Wishart et al., 1991). The periodicity of the H^N shifts observed in many α-helical structures, in conjunction with the well-established relation between H^N chemical shift and hydrogen bond length (Pardi et al., 1983), suggests that they also contain information on helix bending (Kuntz et al., 1991). Similar correlations between the backbone torsion angles ϕ and ψ with the $^1H^\alpha$ and $^1H^\beta$ chemical shifts have been identified, which appear particularly useful for characterization of turns (Ösapay K. and Case, 1994).

Although most of the earlier reports on the relation between chemical shift and protein structure focus on $^1H^\alpha$ and $^1H^N$, with the advent of heteronuclear isotopic enrichment additional chemical shifts have become accessible and offer the potential to make the relation between chemical shift and structure quantitative. The secondary $^{13}C^\alpha$ and $^{13}C^\beta$ chemical shifts of a given residue were found to correlate closely with its ϕ and ψ torsion angles (Ando et al., 1984; Saito, 1986; Spera and Bax, 1991), and thereby also with secondary structure, as shown in Figure 3.1. Methods have been developed to obtain backbone torsion angle restraints and secondary structure information from either $^1H^\alpha$ and $^{13}C^\alpha$ (Luginbühl et al., 1995), or $^{13}C^\alpha$, $^{13}C^\beta$, $^{13}C'$, and $^1H^\alpha$ (Wishart and Sykes, 1994). The empirical correlation between ϕ and ψ backbone torsion angles and the $^{13}C^\alpha$ and $^{13}C^\beta$ chemical shifts also was found useful for identification of N-terminal helix-capping boxes (Gronenborn and Clore, 1994). This same group also introduced an effective method for incorporating the empirical secondary $^{13}C^\alpha$ and $^{13}C^\beta$ chemical shift profiles into the structure calculation protocol (Kuszewski et al., 1995; Celda et al., 1995). *Ab initio* calculations (de Dios and Oledfield, 1993) confirm that the backbone ϕ and ψ torsion angles strongly affect $^{13}C^\alpha$ and $^{13}C^\beta$ shielding, and the use of experimental $^{13}C^\alpha$, $^{13}C^\beta$ and $^1H^\alpha$ shifts, in conjunction with residue-specific chemical shift surfaces from *ab initio* methods, has been proposed as a tool for structure refinement (Pearson et al., 1995). Beger and Bolton proposed an approach to obtain the most probable ϕ and ψ angles from correlation maps between backbone chemical shifts of $^{13}C^\alpha$, $^{13}C^\beta$, $^1H^\alpha$, $^1H^N$ and ^{15}N of a given residue and its backbone torsion angles (Beger and Bolton, 1997). They also showed that this information considerably improves structural quality when used in cases where only a very small number of NOE restraints are available.

The similarity in secondary chemical shifts in homologous proteins has also been well recognized (Redfield and Robertson, 1991). Wishart et al. developed an elegant approach to utilize this similarity during the resonance assignment process (Wishart et al., 1997). However, a minimum of *ca* 30% sequence identity is quoted as the requirement for making this procedure reliable.

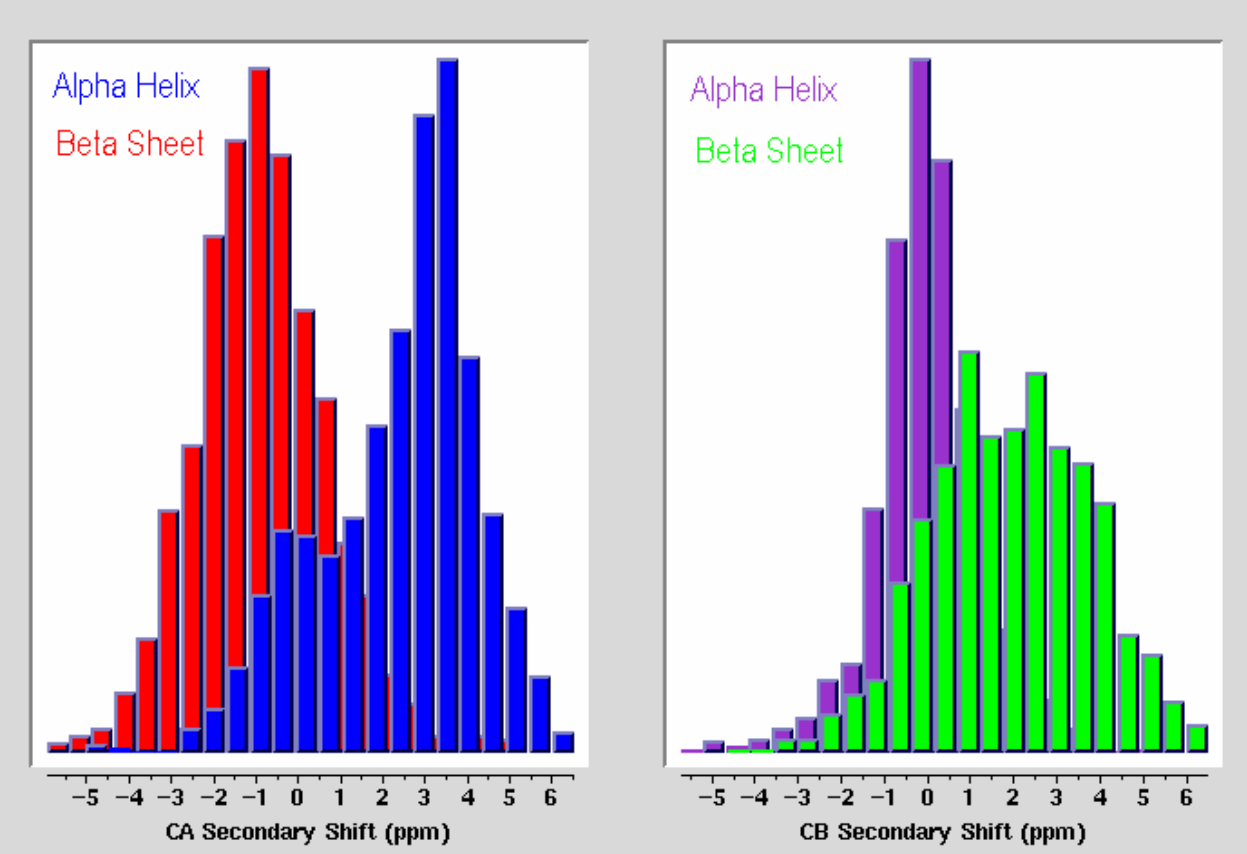


Figure 3.1. Relationship between distribution of observed secondary $^{13}C^{\alpha}$ and $^{13}C^{\beta}$ shifts for α-helix (blue and purple) and β-sheet (red and green) motifs. As shown, the two motifs have distinct but overlapping ranges of secondary shifts. Adapted after Spera and Bax (1991).

Here, we describe a hybrid approach that utilizes both sequence and chemical shift homology to predict the most likely backbone angles for a given residue. The idea is based on the notion that if a string of adjacent amino acids shows high similarity in secondary chemical shifts with a string of amino acids in a database, the central residues in the two strings are likely to have similar backbone torsion angles. In particular, when qualitative similarity in the residue types of the two strings is used as an additional criterion, the approach becomes remarkably robust. In essence, this is a generalization of the idea that helix-capping boxes can be identified best by combined use of their characteristic patterns of chemical shifts and the residue types involved.

As a useful extension to this work, we also describe how the contents of the database can be used to construct prediction surfaces for the simulation of chemical shifts based on secondary structure.

Methods

A database was created which contains nearly complete $^{13}C^{\alpha}$, $^{13}C^{\beta}$, $^{13}C'$, $^{1}H^{\alpha}$ and ^{15}N chemical shifts assignments of 20 proteins (Table 3.1), together with the backbone torsion angles ϕ and ψ, derived from crystal structures solved at a resolution ≤ 2.2 Å (nearly 3,000 residues, 14,000

chemical shifts). The format is such that the database can easily be extended by adding new structures for which at least four of the five chemical shifts are available per residue, and for which the structure is known accurately. The structural data follows the Brookhaven Protein Databank (PDB) format and the chemical shifts are in form similar to the original BioMagResBank format (Seavey et al., 1991). Residues with missing crystallographic coordinates (e.g. residues 1-17 of cutinase and the amino- and carboxy-terminal residues) as well as residues with multiple conformations in the X-ray structure have been excluded. Residues with high temperature (*B*) factors for the backbone atoms, exceeding 1.5 times the average *B*-factor for that protein, were also excluded. This includes the vast majority of cases where differences between crystal and solution structures previously have been noted.

The table lists the references describing the chemical shifts, the X-ray structure, the accession codes for data deposited in the BMRB and PDB databases, the resolution at which the crystal structure was solved, and the types of nuclei for which chemical shifts are available. When using collections of chemical shifts of proteins reported by different groups, it is critical to ensure that the same chemical shift referencing convention is used for all these proteins. This is particularly important for ^{13}C and ^{15}N, where a wide variety of direct and indirect referencing methods have been used.

Rather than relying on the information supplied with the deposited chemical shift data, we evaluate the need for applying a correction to ^{13}C shifts by calculating how much, on average, the secondary shifts (calculated by subtracting the random coil shifts of Spera and Bax) deviate from the corresponding secondary chemical shifts predicted by the (ϕ,ψ)-surfaces of Spera and Bax. These averages are conveniently calculated with a routine added to the X-PLOR program (Brünger, 1993) by Kuszewski et al (Kuszewski et al., 1997), and intended for use of the secondary $^{13}C^{\alpha}$ and $^{13}C^{\beta}$ shifts during structure calculation. We apply a chemical shift correction only if the average deviation for a given protein exceeds by more than a factor of three the expected random variation in this average (i.e., the standard error, *ca* 1 ppm, suggested by Spera and Bax, divided by the square root of the number of shifts used). This manner of correcting the deposited chemical shifts ensures that all secondary shifts are defined in the same manner, and corresponds to subtraction of the random coil $^{13}C^{\alpha}$ and $^{13}C^{\beta}$ shifts of Spera and Bax and random coil $^{13}C'$ shifts (Wishart et al., 1995) from experimentally determined shifts relative to internal trimethylsilyl propionate (TSP). Note that TSP resonates upfield from the IUPAC-recommended standard (Markley et al., 1998), dimethylsilapentane-5-sulfonic acid or DSS, by an insignificant amount (0.12 ppm at pH 7) (Wishart et al., 1995B). The same correction procedure must be used for all other new proteins added to the database. Only a small fraction of the proteins required the above correction procedure. For ^{15}N, the chemical shift reference standard is liquid ammonia at 25° C, and the need for application of a correction was evaluated by calculating the average ^{15}N chemical shifts for all non-Gly, non-Ser, non-Thr residues in α-helical and β-strand regions of the protein and comparing them with the database averages (119.47 ppm for α-helices and 122.38 ppm for β-strands). In addition, later versions of the database also made use of the suggested corrections to ^{15}N random coil values based on previous residue type (Braun et al., 1994). Whenever the average of the α-helix and β-strand ^{15}N chemical shift deviations (weighted according to the number of residues used for each type of secondary structure) is larger than 1 ppm, a correction to the chemical shifts needs to be applied. Alphalytic protease was the only

Table 3.1. Proteins contained in the TALOS chemical shift and structure database.

Protein Reference (*BioMagResBank no.)	Residues	X-ray structure ref. (*PDB code)	Resol. (Å)	Shifts
Alpha-lytic protease Davis et al., 1997	198	Fujinaga et al. 1985 (*2alp)	1.7	H^{α}, C^{α}, C^{β}, C', N
Basic pancreatic trypsin inhibitor Hansen, 1991	58	Wlodawer et al. 1984 (*5pti)	1.1	H^{α}, C^{α}, C^{β}, C', N
Calbindin, (*390) Drakenberg et al., 1989.	76	Svensson et al. 1992 (*4icb)	1.6	H^{α}, C^{α}, C^{β}, N
Calmodulin, (*547) Ikura et al., 1990	148	Chattopadhyaya et al.1992, (*1cll)	1.7	H^{α}, C^{α}, C^{β}, C', N
Calmodulin/M13, (*1634) Ikura et al., 1991	147	Meador et al.1992, (*1cdl)	2.2	H^{α}, C^{α}, C^{β}, C', N
Cutinase, (*4101) Prompers et al., 1997	214	Longhi et al.1997, (*1cex)	1.0	H^{α}, C^{α}, C^{β}, C', N
Cyclophilin Ottiger et al., 1997	165	Ke et al.1991, (*2cpl)	1.63	H^{α}, C^{α}, C^{β}, N
Cyanovirin-N Bewley et al., 1998	101	Yang et al., 1999 (*3ezm)	1.5	H^{α}, C^{α}, C^{β}, C', N
Dehydrase Copie et al., 1996	171	Leesong et al., 1996 (*1mka)	2.0	H^{α}, C^{α}, C^{β}, C', N
D-maltodextrin-binding protein Gardner et al., 1998	370	Sharff et al., 1993 (*1dmb)	1.8	H^{α}, C^{α}, C^{β}, C', N
HIV-1 protease Yamazaki et al., 1996	99	Lam et al., 1994	1.8	H^{α}, C^{α}, C^{β}, C', N
Human carbonic anhydrase I, (*4022) Sethson et al., 1996	260	Kumar and Kannan, 1994 (*1hcb)	1.6	H^{α}, C^{α}, C^{β}, C', N
Human thioredoxin in reduced form Qin et al., 1996	105	Weichsel et al., 1996 (*1ert)	1.7	H^{α}, C^{α}, C^{β}, N
III-glc Pelton et al., 1991	168	Worthylake et al., 1991 (*1f3g)	2.1	H^{α}, C^{α}, C^{β}, C', N
Interleukin-1, (*1061) Clore et al., 1990	153	Veerapandian et al., 1992 (*4i1b)	2.0	H^{α}, C^{α}, C^{β}, N
Metallo-β-lactamase, (*4102) Scrofani et al., 1998	232	Concha et al.,1996 (*1znb)	1.85	H^{α}, C^{α}, C^{β}, C', N
Profilin Archer et al., 1994	125	Fedorov et al., 1994 (*1acf)	2.0	H^{α}, C^{α}, C^{β}, C', N
Serine protease PB 92 Fogh et al., 1995	269	Betzel et al., 1992 (*1svn)	1.4	H^{α}, C^{α}, C^{β}, C', N
Staph nuclease D. Torcia, private communication	141	Loll and Lattman, 1989 (*1snc)	1.65	H^{α}, C^{α}, C^{β}, C', N
Ubiquitin Wang et al, 1998	76	Vijay-Kumar et al., 1987 (*1ubq)	1.8	H^{α}, C^{α}, C^{β}, C', N

protein for which such ^{15}N chemical shift adjustment (by -2.26 ppm) needed to be used. For ^{1}H, where historically chemical shift referencing has been much less of a problem, no such corrections were applied.

To investigate whether the $^{13}C'$ chemical shift is strongly influenced by the hydrogen bond length, hydrogens were added to the 1.1 Å crystal structure of basic pancreatic trypsin inhibitor (Wlodawer et al., 1984) with the program X-PLOR. For the 24 carbonyls involved in stable backbone-backbone hydrogen bonds, no significant correlation was found between the lengths of the backbone-backbone hydrogen bonds, calculated from this structure, and the corresponding $^{13}C'$ secondary shifts. This result suggests that the $^{13}C'$ secondary shift is primarily a function of the backbone geometry, in agreement with its previously reported correlation with secondary structure (Kricheldorf and Muller, 1983; Wishart et al., 1991). Therefore, we decided to include the $^{13}C'$ shift information in the evaluation, even while for several proteins no $^{13}C'$ shifts have been reported in the database.

Although the ^{15}N chemical shift is known to be influenced by hydrogen bonding (de Dios et al., 1993), it is also influenced by backbone geometry and therefore is included as an input parameter in the torsion angle prediction procedure. However, as discussed below, optimization of the torsion angle prediction program results in a relatively low weighting factor for this chemical shift.

Results and Discussion

Description of the search procedure. The backbone torsion angle prediction package TALOS (Torsion Angle Likelihood Obtained from Shifts and sequence similarity) is written in the TCL/TK language (Ousterhout, 1994) and uses NMRWish, a companion package to the NMRPipe processing and analysis system (Delaglio et al., 1995). NMRWish is a version of the TCL/TK script interpreter "wish", which has been customized to include a relational database engine for manipulation of spectral information and molecular coordinates. An outline of the prediction method used by TALOS is presented in Figure 3.2.

TALOS reads the experimental protein chemical shift tables and converts them to secondary chemical shifts before entering them in the database. In its current implementation, TALOS evaluates the similarity in amino acid sequence and secondary shifts for a string of three sequential amino acids relative to all triplets of sequential residues contained in the database. Although we expect further improvement in performance might be attainable for string lengths longer than three, the number of residues in the database is presently too small to yield a sufficient sampling for such longer strings.

It should be noted however, as we will describe below, the contents of the TALOS database can be used to generate chemical shift simulation surfaces, and these can be applied to fragments of arbitrary length.

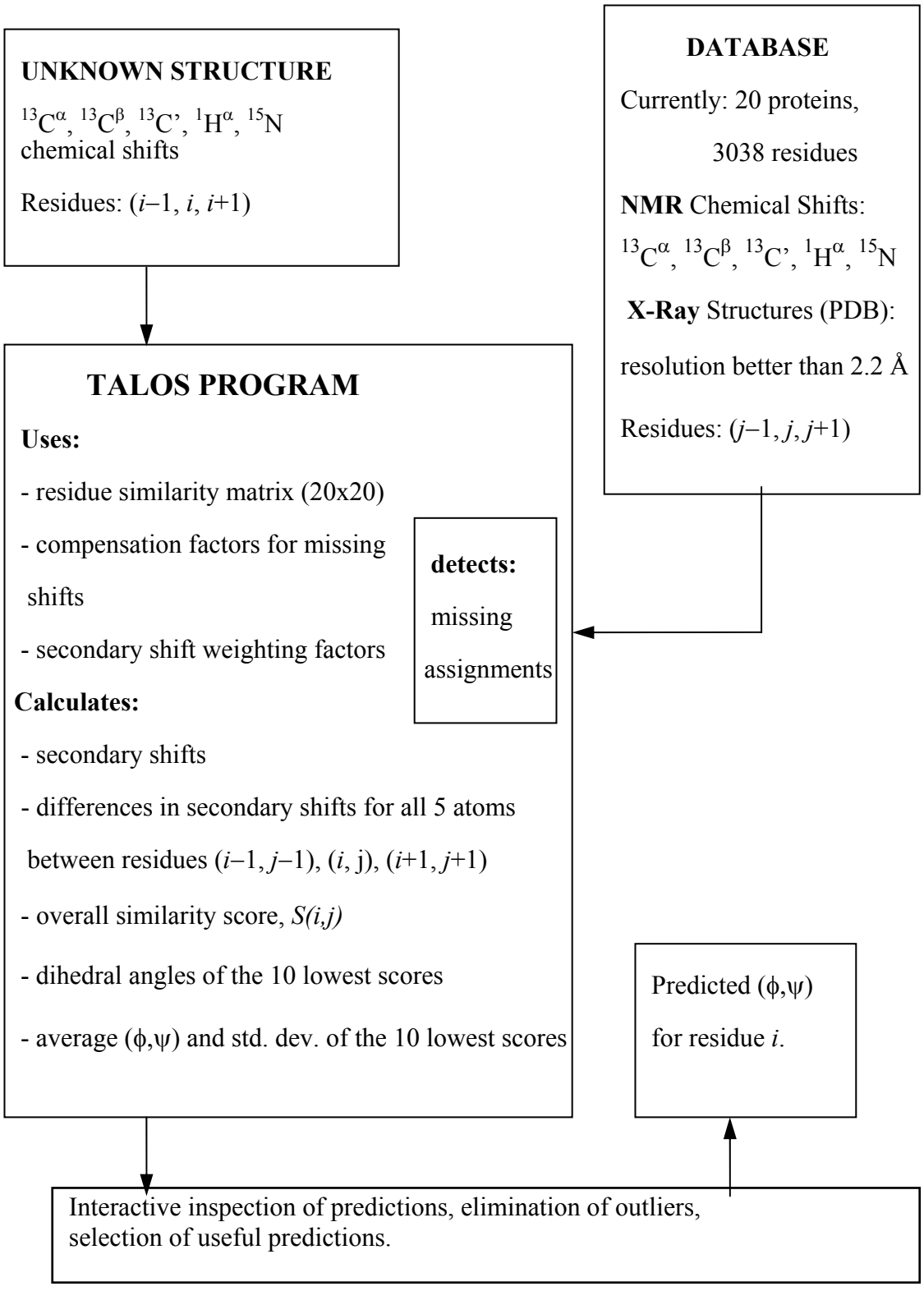


Figure 3.2. Flow chart of the TALOS program**.**

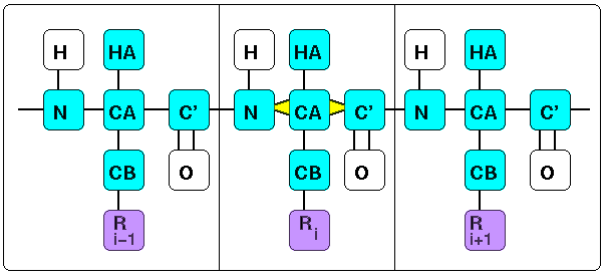


Figure 3.3. Schematic of residue triplet used in the TALOS approach. The 15 chemical shifts used to predict the backbone angles of residue *i* are highlighted in blue.

Similarity Score. For each query triplet of consecutive residues, as shown in Figure 3.3, the similarity to a triplet with center-residue *j* in the database is evaluated by computing a similarity factor, $S(i,j)$, given by:

$$S(i,j) = \sum_{n=-1}^{+1} \left[k_n^0 \Delta^2_{\text{Re sType}} + k_n^1 \left(\Delta\delta C^{\alpha}_{i+n} - \Delta\delta C^{\alpha}_{j+n} \right)^2 + k_n^2 \left(\Delta\delta N_{i+n} - \Delta\delta N_{j+n} \right)^2 + \right.$$
$$\left. + k_n^3 \left(\Delta\delta C^{\beta}_{i+n} - \Delta\delta C^{\beta}_{j+n} \right)^2 + k_n^4 \left(\Delta\delta C'_{i+n} - \Delta\delta C'_{j+n} \right)^2 + k_n^5 \left(\Delta\delta H^{\alpha}_{i+n} - \Delta\delta H^{\alpha}_{j+n} \right)^2 \right] \quad (3.1)$$

The value of $S(i,j)$ is evaluated for all triplets *j* in the database. Δδ denotes the secondary shifts of the $^{13}C^{\alpha}$, $^{13}C^{\beta}$, $^{13}C'$, $^{1}H^{\alpha}$ and ^{15}N nuclei. For Gly residues, $^{1}H^{\alpha}$ shifts are calculated as the average of $^{1}H^{\alpha 2}$ and $^{1}H^{\alpha 3}$. Values for the weighting factors, k^0_n through k^5_n are optimized as described below and are given in Table 3.2; the residue-type similarity matrix ascribes a number to how similar two types of amino acids are and this 20 × 20 matrix is shown in Table 3.3. The composition of this similarity matrix is largely based on empirical knowledge that, for example, Gly frequently has a positive ϕ angle, Pro has a very restricted range of ϕ angles, and C^{β}-branched residues are frequently found in β-sheets.

There has been some empirical adjustment of the similarity matrix during the process of optimizing the performance of the TALOS program, but results were not found to be particularly sensitive to small changes (by ±1) in the Table 3.3 matrix elements. Using the empirical *k* values of Table 3.2, and Δ_{ResType} of Table 3.3, $S(i,j)$ values typically range from 5 to 600.

For all database triplets, *j*, that yield a $S(i,j)$ value lower than an adjustable threshold (typically ~150), TALOS reports the corresponding X-ray crystal structure ϕ and ψ angles of residue *j*, together with the $S(i,j)$ value. The threshold is set sufficiently large to obtain a minimum of at least 10 matches for each residue *i*. Optimization of the 15 chemical shift weighting factors made use of a scheme which finds all triplets of residues in the database for which the central residue has ϕ/ψ angles within 15° of those of a query residue. We then calculate the average and the standard deviation of the secondary chemical shifts for each of the 15 types of chemical shifts (5 nuclei for residue *i*-1, *i*, and *i*+1) over this ensemble of triplets.

Table 3.2. Empirically optimized k factors, k^m_n (m : homology, C^α, N, C^β, C', H^α; n = -1, 0, 1), for weighting the relative importance of a given chemical shift or residue type in determining the similarity score, S(i,j) of Eq. 3.1.

Res.	Homology	^{15}N	$^{1}H^\alpha$	$^{13}C'$	$^{13}C^\alpha$	$^{13}C^\beta$
n = -1	0.74	0.16	14.66	1.15	0.72	0.76
n = 0	1.48	0.18	17.54	1.21	0.99	0.91
n = 1	0.74	0.20	15.25	1.04	0.72	0.70

The rms value of all database secondary chemical shifts of a given type of nucleus, divided by the standard deviation derived in the above described manner, provides a measure for how useful a given type of secondary chemical shift (e.g., $\Delta\delta N_{i-1}$) is at providing information on the ϕ/ψ angles of residue *i*. This ratio was calculated 183 times, each time using a different cutinase residue as the query residue. The chemical shift weighting factors listed in Table 3.2 are derived from the averages of these respective ratios, after scaling to compensate for the intrinsically different widths of the secondary shift distributions of the types of atoms involved (i.e., using the root-mean-square (rms) values of the ^{15}N, $^{1}H^\alpha$, $^{13}C^\alpha$, $^{13}C^\beta$ and $^{13}C'$ secondary chemical shift values in the entire database).

The relative weight of the residue type homology versus secondary shifts in the *S*(*i,j)* formula (k^0_n factors in Eq. 3.1) was optimized empirically, by searching for k^0_n factors that minimize the number of erroneous predictions, using all residues present in the database for test purposes. If a particular chemical shift is missing, the corresponding secondary chemical shift difference between the query and the corresponding database chemical shift is set to 1.5 times the rms value of the corresponding secondary chemical shift (rms values are 4.56 ppm for ^{15}N, 2.49 ppm for $^{13}C^\alpha$, 0.51 ppm for $^{1}H^\alpha$, 2.01 ppm for $^{13}C^\beta$, and 2.02 ppm for $^{13}C'$). This way of dealing with incomplete assignments decreases the likelihood that database residues with incomplete assignments contribute to the (ϕ,ψ) output of TALOS, but does not exclude them altogether.

Torsion angle prediction. To date, the database used by TALOS contains only 20 structures for which both a high-resolution X-ray structure and nearly complete resonance assignments are available. The reason we felt it is not warranted to include proteins for which a high-resolution NMR structure but no crystal structure is available is that, as discussed below, the agreement between the ϕ and ψ angles of most NMR structures and the output of TALOS is considerably lower than for the high-resolution crystal structures in the database.

The TALOS output for the ϕ and ψ backbone angles of the center residue in each string consists of the average of the corresponding angles in the 10 strings in the database with the highest degree of similarity (*cf.* Eq. 3.1). In a first, fully automated but very conservative mode of analysis, the program classifies only those predictions for which at least 9 out of 10 predictions fall in the same populated (gray shaded) region of the Ramachandran map (Figure 3.4), and none of the center residues in the 10 strings has a positive ϕ angle. If a single residue falls well outside the Ramachandran region in which the remaining 9 residues are located, its ϕ/ψ values are

Table 3.3. Residue similarity factors, $\Delta_{ResType}$, as used by TALOS in Eq. 3.1.

Res.	A	R	N D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W Y	V
A	0	1	1	1	1	1	2	1	2	1	1	1	2	3	1	2	2	2
R	1	0	1	1	1	1	2	1	2	1	0	1	1	3	1	2	1	2
N/D 1	1	0	1	1	1	2	1	2	1	1	1	1	3	1	2	1	2	
C	1	1	1	0	1	1	2	1	2	1	1	1	1	3	1	2	1	2
Q	1	1	1	1	0	1	2	1	2	1	1	1	1	3	1	2	1	2
E	1	1	1	1	1	0	2	1	2	2	2	1	1	3	1	2	1	2
G	2	2	2	2	2	2	0	3	3	3	3	3	3	3	3	3	3	3
H	1	1	1	1	1	1	3	0	2	1	2	2	1	3	2	2	1	2
I	2	2	2	2	2	2	3	2	0	1	2	2	2	3	2	1	2	0
L	1	1	1	1	1	2	3	1	1	0	1	1	1	3	2	2	1	2
K	1	0	1	1	1	2	3	2	2	1	0	1	2	3	1	2	2	2
M	1	1	1	1	1	1	3	2	2	1	1	0	2	3	1	2	2	2
F	2	1	1	1	1	1	3	1	2	1	2	2	0	3	2	2	0	1
P	3	3	3	3	3	3	3	3	3	3	3	3	3	0	3	3	3	3
S	1	1	1	1	1	1	3	2	2	2	1	1	2	3	0	1	2	2
T	2	2	2	2	2	2	3	2	1	2	2	2	2	3	1	0	1	1
W/Y	2	1	1	1	1	1	3	1	2	1	2	2	0	3	2	1	0	1
V	2	2	2	2	2	2	3	2	0	2	2	2	1	3	2	1	1	0

excluded from calculating the average and rmsd. This procedure typically results in predictions for only about 40% of the residues.

A subsequent interactive inspection of the results, using the graphical interface described below, permits additional predictions to be made. For example, if several predictions fall just outside the most populated region of the Ramachandran map, but generally cluster well with the other ϕ/ψ predictions, the prediction should be accepted. In some cases, there is one center-residue in the ensemble of 10 most similar triplets for which either ϕ or ψ deviates by more than 2 standard deviations from the average value for that angle. Empirical testing indicates that it is safe to remove (at most) one such triplet from the ensemble of 10 (TALOS then recalculates the new average ϕ and ψ angles and their rmsd), provided that the outlier does not have its ϕ angle in the $0° < \phi < +150°$ range, and the average $S(i,j)$ value is less than 80. When the TALOS output for a given query residue yields a cluster where at least 9 residues have positive ϕ angles, this prediction also should be accepted.

The standard deviations and the range of (ϕ,ψ) values in the 10 (or 9) most similar database strings provide a measure for the uncertainty in these averages. When this standard deviation exceeds 45°, the prediction must be deemed "ambiguous", and it is recommended that the result of the prediction not be used without careful further inspection of other data, such as the $d_{\alpha N}(i-1,i)/d_{\alpha N}(i,i)$ NOE intensity ratio (which provides information on the ψ angle), the $^3J_{HN H\alpha}$ coupling (ϕ angle), and $^1J_{C\alpha H\alpha}$ (primarily for identifying positive ϕ angles (Vuister et al., 1991, Vuister et al., 1993). Not including such cases where NOE or J coupling information is needed,

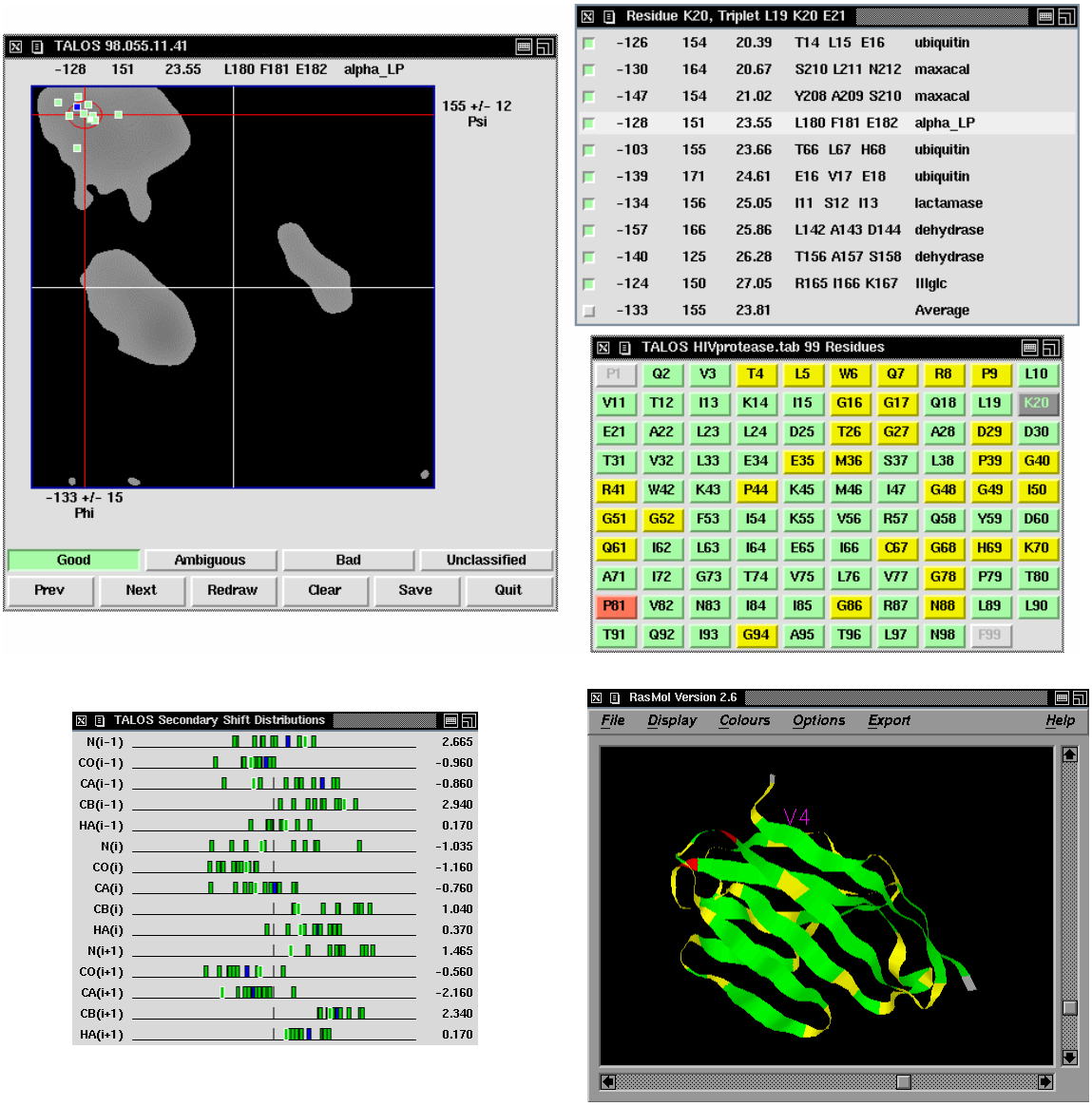


Figure 3.4. Graphical display of TALOS output for HIV protease. The center right window shows the amino acid sequence, with predictions for each residue classified as "good" (green), "ambiguous" (yellow), or "bad" (red). The prediction data for the selected residue, K20, are listed in the prediction display (top right) and graphed in the Ramachandran display (top left). The 10 individual matches from the database are indicated as small green squares in the Ramachandran display, and for reference purposes, the known ϕ/ψ position from the HIV protease X-ray structure (blue square) is also shown. The lower left window shows the secondary shift values for the 10 best matches. TALOS can be interfaced with the RasMol (Sayle and Milner-White, 1995) molecular viewer (bottom right) to view a corresponding reference structure if one is available, colorized according to the current prediction classifications. Clicking on any of the items in the interface highlights the corresponding items in the other windows.

---

the above described protocol typically allows a definitive prediction of the ϕ and ψ angles to be made for about two thirds of the residues.

Because the number of proteins for which complete NMR assignments and high resolution crystal structures are available is still very limited, the TALOS database usually contains insufficient entries for unambiguous identification of residues with positive ϕ angles. However, testing indicates that if the center-residue of a query triplet has a positive ϕ angle, this frequently results in a significant fraction of center-residues that also have positive ϕ angles in the ten most similar database triplets. These positive ϕ angle triplets typically yield the lowest S(*i,j*) values; this initially suggested that the program will successfully predict more of the positive ϕ angles once the database becomes sufficiently large. However, subsequent examination of the chemical shift prediction surfaces described below indicates that there may be a certain unavoidable degree of ambiguity in the relationship between secondary shift and (ϕ,ψ). For now, unambiguous identification of such positive ϕ angles in most cases requires additional experimental data, such as a very small $^1J_{C\alpha H\alpha}$ (<136 Hz) (Vuister et al., 1991, Vuister et al., 1993), or the presence of an exceptionally strong intraresidue H^N-H^α NOE.

Display of TALOS output. A graphical interface for inspecting and interactively updating the TALOS output is available. An example of its use is shown in Figure 3.4 for HIV protease. The interface consists of three windows: the sequence display, the prediction display, and the Ramachandran display. The sequence display lists the residues in the protein whose backbone angles are being predicted. The residues are color-coded according to whether the overall prediction for a given residue was designated as good, ambiguous, or bad. In the initial display, before interactive analysis, residues are color-coded as green (prediction accepted in automated mode) and gray (requires inspection). If the true ϕ/ψ angles are known, residues for which a wrong prediction was accepted can be classified as bad (red), which is convenient for testing purposes. All residues for which TALOS has made predictions that meet the criteria listed above, are highlighted in green. Residues shaded in yellow are those for which no firm prediction can be made, but which nevertheless may contain useful information. For example, if for a given residue 5 out of the 10 triplets show a positive ϕ angle, this suggests that there is a high likelihood that the center residue of the query triplet has a positive ϕ angle.

When a given residue is selected in the sequence display (K20 in Figure 3.4), the ϕ, ψ, and *S*(*i,j*) parameters are listed in the prediction display, together with the residue numbers and the names of the proteins from which the triplets were taken. The ten ϕ/ψ pairs are graphed in the Ramachandran display, which also shows the most populated areas of the entire database, shaded in gray. If a reference or trial structure for the query protein is available, its ϕ/ψ angles will also be graphed on the Ramachandran display (blue square). By clicking on an individual match in the Ramachandran display, it is possible to include or remove this entry from the overall prediction, which is based on the average and standard deviations of the selected matches.

The final results are summarized in an ASCII text table which gives the average ϕ/ψ angles and their standard deviations for each residue. Versions of the TALOS program are available for most types of UNIX platforms.

Accuracy of TALOS-predicted angles. Figure 3.5 plots the predicted ϕ and ψ angles of ubiquitin versus those of the high resolution crystal structure. As can be seen from this plot, TALOS does considerably more than classifying residues by their type of secondary structure,

and there is a good correlation between predicted and crystallographic torsion angles, even when considering only the residues with a positive ψ angle, for example.

Figure 3.6 shows the predicted ϕ and ψ angles as a function of residue number, together with the corresponding crystallographically determined angles. The error bars correspond to the standard deviation from the average angle for the center-residue of the 10 (or 9) best fitting triplets in the database. No result is shown if this standard deviation exceeds 45°, or if any (but less than 9) of the ϕ angles of the center-residues have a positive ϕ angle.

Cross-validation tests of the accuracy of TALOS predictions were made by eliminating each protein from the database and using the program with the remaining information in the database to predict its backbone angles (Table 3.4). We found that for about 2% of the residues in the database (i.e., 3% of the predictions made) TALOS predicts the wrong torsion angles (angles that are in a different region of the Ramachandran plot relative to the crystal structure). Some examples are:

- Thr^{45} in cutinase: Predicted ψ = -4 ± 10°; X-ray ψ = 163°. Although the B factor is not unusually high, ^{15}N relaxation data indicate this residue is located in the middle of a flexible loop that differs in conformation relative to the crystal structure (Prompers et al., 1997).
- Asp^{159} of beta-hydroxydecanoyl thiol ester dehydrase: Predicted ϕ = -57 ± 7°, ψ = -36 ±10°; X-ray ϕ = 56°, ψ = 52°.
- Asp^{19} of staphylococcal nuclease: Predicted ϕ = -90 ± 12°; ψ = 8 ±11°; X-ray ϕ = -156°, ψ= -166°.

Both for Asp^{159} and Asp^{19} there are no doubt regarding the similarity in backbone angles in solution and in the crystalline state, but TALOS fails to predict the unusual backbone angles of these residues. The user therefore should be aware that a small fraction of the TALOS predictions may be in error. However, as shown below, for the majority of cases, the output of TALOS is highly accurate. When listing the rms differences between the predicted ϕ/ψ angles and those of the crystal structure, the small fraction of erroneous predictions are not included.

For ubiquitin, TALOS yields 53 ϕ/ψ angle predictions (76 % of its database residues) and the rms differences between the predicted ϕ/ψ angles and those of the crystal structure are 12°/9°. Similarly, for cutinase ϕ/ψ predictions are made for 127 residues (69%, including 5 bad predictions, but excluding the disordered N-terminal tail), with rmsds of 12°/12° relative to the crystallographically determined ϕ and ψ angles.

BPTI yielded the worst performance of all proteins tested. Only 32 ϕ/ψ predictions (65%, 4 bad predictions) were made, which agree to within rmsds of 16° and 17° with the 1.1 Å crystal structure. Differences relative to the solution structure (Berndt et al., 1992) are slightly larger (18/19°). For the same set of ϕ and ψ angles, the rms differences between the average solution structure and crystal structure are 14° and 12°, respectively.

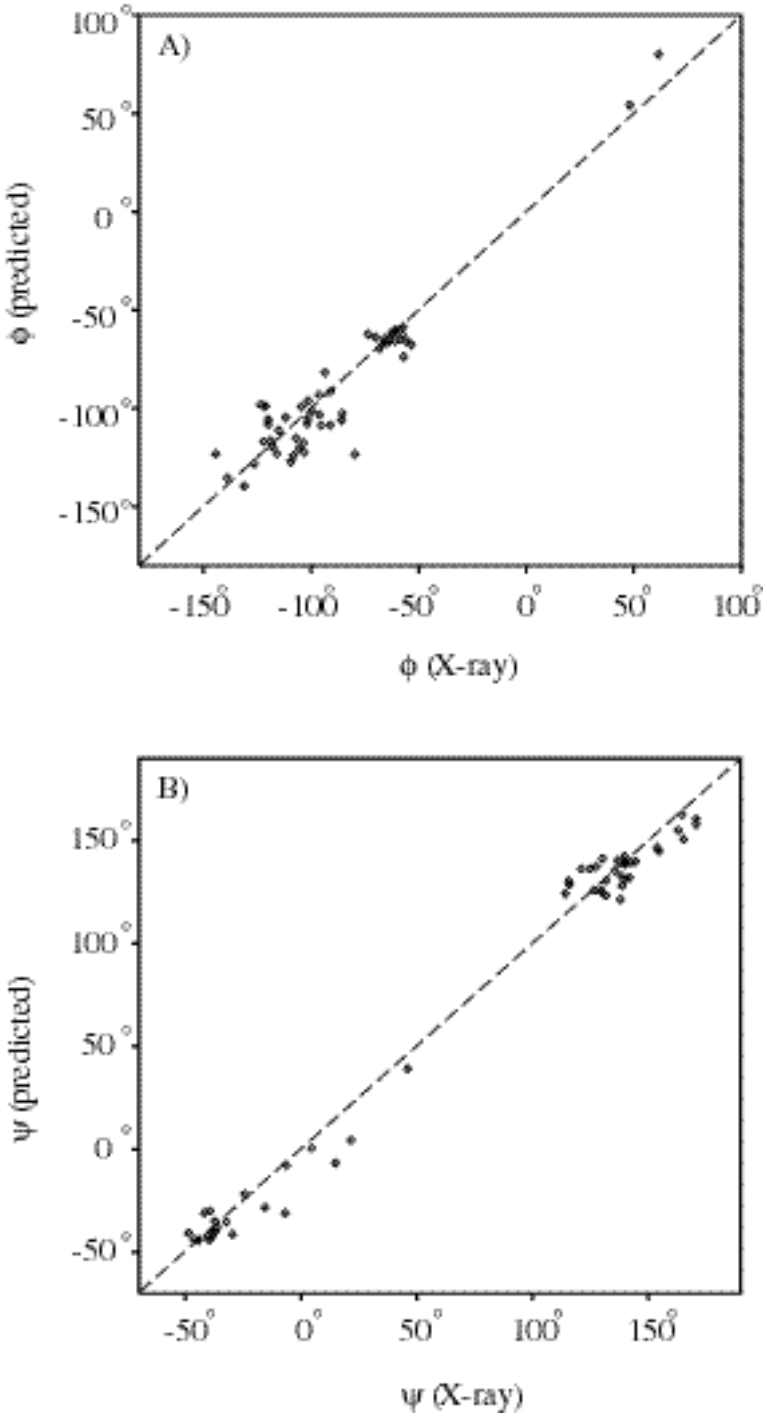


Figure 3.5. Plots of the backbone angles (A) ϕ, and (B) ψ predicted by TALOS, versus those observed in the crystal structure, for ubiquitin.

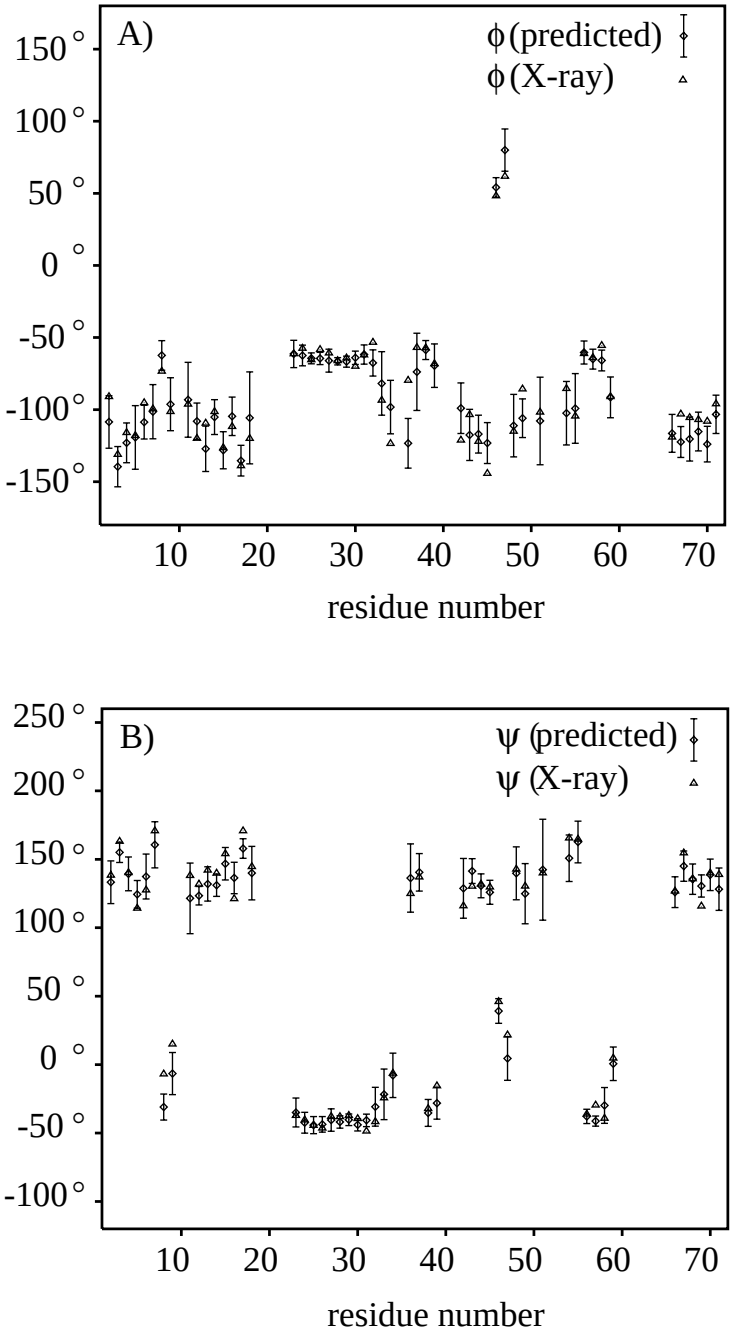


Figure 3.6. Predicted backbone angles (A) ϕ, and (B) ψ of ubiquitin. The length of the error bars represents the standard deviation from the average of the dihedral angles of the 10 residues from the database having the highest chemical shift and sequence similarity with the query residues. Triangles correspond to the angles observed in the crystal structure.

Table 3.4. Summary of TALOS backbone angle prediction for proteins included in the database.

Protein Name	Good (%)	Bad (%)	Ambig. (%)	Avail.	All
HIV-1protease	65 67.0	1 1.0	31 32.0	97	99
III-glc	87 61.7	4 2.8	50 35.5	141	168
Alpha-lytic protease	101 54.6	3 1.6	81 43.8	185	198
BPTI	32 58.2	4 7.3	19 34.5	55	58
Calbindin	48 72.7	0 0.0	18 27.3	66	75
Calmodulin	107 84.9	0 0.0	19 15.1	126	148
Calmodulin/M13	98 80.3	0 0.0	24 19.7	122	148
Cutinase	122 66.7	5 2.7	56 30.6	183	214
Cyanovirin-N	55 61.1	1 1.1	34 37.8	90	101
Cyclophilin	87 54.0	5 3.1	69 42.9	161	165
Dehydrase	91 62.7	3 2.1	51 35.2	145	171
HCA I	149 60.3	7 2.8	91 36.9	247	260
Interleukin-1β	75 61.0	3 2.4	45 36.6	123	153
Lactamase	137 66.2	4 1.9	66 31.9	207	232
Serine protease PB 92	161 62.7	8 3.1	88 34.2	257	269
D-MBP	217 62.0	5 1.4	128 36.6	350	370
Profilin	72 67.9	0 0.0	34 32.1	106	125
Staphilococcal nuclease	81 67.5	4 3.3	35 29.2	120	141
Human thioredoxin	79 76.7	1 1.0	23 22.3	103	105
Ubiquitin	53 75.7	0 0.0	17 24.3	70	76

Total:		
predictions:	2910	
correct:	1920	(65.3%)
incorrect:	58	(2.0%)

Listed are the number of "Good" predictions, and the percentage relative to the total number of residues with acceptable B factors (Avail.), the number of "Bad" predictions, and the number of residues for which no predictions could be made (Ambig.), plus the total number of residues (All).

For human thioredoxin, the NMR data have been derived for a mutant that differs from the sequence used for the crystal structure. The ϕ angles predicted by TALOS are nevertheless in very good agreement with those of the crystal structure, with 80 (78%) ϕ/ψ predictions (rmsds of 15° and 12° from the X-ray structure, respectively), including one erroneous prediction. For reference, the rmsds relative to the solution structure for the same group of ϕ and ψ angles are 20° and 22°, respectively. The pairwise rmsd between the crystal structure and solution structure angles is 16° (ϕ) and 20° (ψ).

Use of TALOS output in structure calculation. The dihedral constraints for the backbone torsion angles obtained from TALOS are available immediately after completion of the resonance assignment and therefore can be used at the very early stages of structure calculation. It is, however, important to realize that a small fraction of the TALOS predictions is likely to be in error. Preliminary testing on the effect of inclusion of TALOS constraints in calculation of a protein structure was carried out for ubiquitin.

Three sets of calculations were performed: (A) using only 273 NOEs, randomly taken from the total set of 2727 NOE cross peaks, peak-picked from 3D and 4D NOESY spectra (J.L. Marquardt, unpublished results); (B) additionally using TALOS-ϕ/ψ constraints for the 53 residues for which a (correct) prediction had been made; (C) as B, but deliberately introducing two serious errors in the ϕ/ψ constraints by interchanging the TALOS-derived angles of Ala[46] (TALOS: $\phi = 54\pm7°$, $\psi = 39\pm9°$; X-ray: $\phi = 48°$, $\psi = 46°$) with those of Arg[54] (TALOS: $\phi = -102\pm22°$, $\psi = 150\pm17°$; X-ray: $\phi = -85°$, $\psi = 165°$). Starting from a fully extended strand and using an X-PLOR based simulated annealing protocol[111], set A yielded convergence for 9 out of 30 calculated structures. The backbone rmsd (residues 2-70) from the average was 1.52 Å, and the backbone rmsd displacement between the average of these NMR structures and the crystal structure was 1.36 Å. For set B, ϕ- and ψ-constraints were included as "harmonic-well" potentials with zero energy over the range $\phi_{TALOS} \pm SD$ and $\psi_{TALOS} \pm SD$, where SD is the standard deviation in the set of 10 (or 9) residues from which ϕ_{TALOS} and ψ_{TALOS} were derived. Outside the well, the energy increased quadratically with 200 kcal/rad^2. With 13 out of 30 calculations converging, the yield was 50% higher than in the absence of TALOS constraints. Moreover, the rmsd from the average was also considerably lower (0.75 Å), as was the difference relative to the X-ray structure (0.89 Å). For set C, which includes the erroneous backbone constraints, convergence was worst (7 out of 30), but the rmsd from the average (0.87 Å) and between the averaged NMR and crystal structure (1.04 Å) were intermediate. The errors introduced in the NMR structure by the wrong TALOS constraints were highly localized.

Although preliminary, the above results for ubiquitin are quite encouraging. They suggest that a substantial improvement in quality of the structure can be obtained by including the TALOS-derived ϕ/ψ-restraints, particularly when the number of NOEs per residue is low. The introduction of two serious errors in the TALOS-derived torsion angle restraints decreases the quality of the structure, but it remains better than in the absence of the TALOS-derived constraints. Nevertheless, it is recommended that the constraints are used with care, keeping in mind that they may contain errors. Thus, if either a TALOS- or NOE-constraint (or both) are violated consistently during structure calculations, it is essential to recheck the quality of the constraint(s) involved. In this respect, an erroneous TALOS-derived restraint is no different from a wrongly assigned NOE connectivity.

Use of the TALOS database to simulate chemical shifts from backbone angles. Using the method of Spera and Bax, the TALOS database was used to prepare secondary shift prediction surfaces with respect to (ϕ,ψ) (Spera and Bax, 1991) for each of the five types of chemical shifts $^1H^\alpha$, $^{13}C^\alpha$, $^{13}C^\beta$, $^{13}C'$, and ^{15}N. The surfaces were prepared iteratively, based on our initial observations that the secondary shift predictions generated from the surfaces had larger than usual errors for certain residue types, in particular Cys and His. Furthermore, Gly, Pro, and residues preceding a Pro (xPro) were treated separately, as these residues have Ramachandran distributions which are different from the others, and so residue-specific surfaces were prepared for each of these three cases. Subsequent tests indicated that no substantial improvement was gained if other residues were treated separately as well, or grouped by chemical similarity. This resulted in the creation of surfaces four residue classes: (1) Gly, (2) Pro, (3) xPro, and (4) all others (the *generic* class). In the case of Gly, two kinds of surfaces for H^α were prepared; one for $\min(H^{\alpha3}, H^{\alpha3})$ and another for $\max(H^{\alpha2}, H^{\alpha3})$.

Table 3.5. Empirical adjustments to random coil shifts as suggested by initial chemical shift prediction surfaces.

ResidueType	Shift Type	Adjustment to Random Coil Shift (ppm)
Asn	$^{13}C^{\alpha}$	0.25
Cys-reduced	$^{13}C^{\alpha}$	1.49
Cys-reduced	$^{13}C^{\beta}$	-0.49
Cys-oxidized	$^{13}C^{\alpha}$	2.01
Cys-oxidized	$^{13}C^{\beta}$	0.68
Cys-oxidized	^{15}N	0.81
His	$^{13}C^{\alpha}$	1.13
His	$^{13}C^{\beta}$	1.34
His	$^{13}C'$	1.55
Leu	$^{13}C^{\alpha}$	-0.26
Leu	$^{13}C'$	-0.50
Trp	$^{13}C^{\beta}$	-0.84
Trp	^{15}N	-0.93
Ala	^{15}N	-0.98

Generic surfaces were prepared as follows. In the first iteration, surfaces were prepared using all shifts except those for Cys, His, Gly, Pro, and xPro. In the second iteration, in an effort to minimize the effect of outliers influenced by factors independent of secondary structure (metal or ligand binding, dynamics, possible misassignments, etc.), new surfaces were generated using the 90% of the TALOS database with the best match to the predictions. Predictions made from these second-iteration surfaces were then used to compute adjustments to the random coil values for cases where systematic offsets in the prediction were greater than one third of the standard deviation in the prediction error. These empirical adjustments are shown in Table 3.5. As listed, both Cys and His results called for adjustments in their random coil values, helping to explain the systematic errors in initial predictions for these residues.

In the third iteration, surfaces were recomputed by including these adjustments in the random coil values, again using the 90% of the database with the best agreement between observed and predicted shifts. In addition to the secondary chemical shift prediction surfaces, prediction error surfaces were computed as the rms differences between observed and predicted shifts. These error surfaces estimate the standard deviation of the secondary shift prediction associated with a given (ϕ,ψ) location. The two kinds of surface are used together; the shift prediction surface is used to simulate a chemical shift as the sum of the predicted secondary shift and random coil value, while the prediction error surface provides an error estimate which can be used to normalize the prediction to account for the variation in chemical shift which is not directly dependent on (ϕ,ψ). The collection of generic surfaces is shown in Figure 3.7, and predictions generated from these surfaces for the chemical shifts of ubiquitin are shown in Figure 3.8.

Table 3.6 provides a summary of prediction results for all proteins in the TALOS database, as well as for two other proteins, using the NMR structure of the 81-residue DinI (Ramirez et al.,

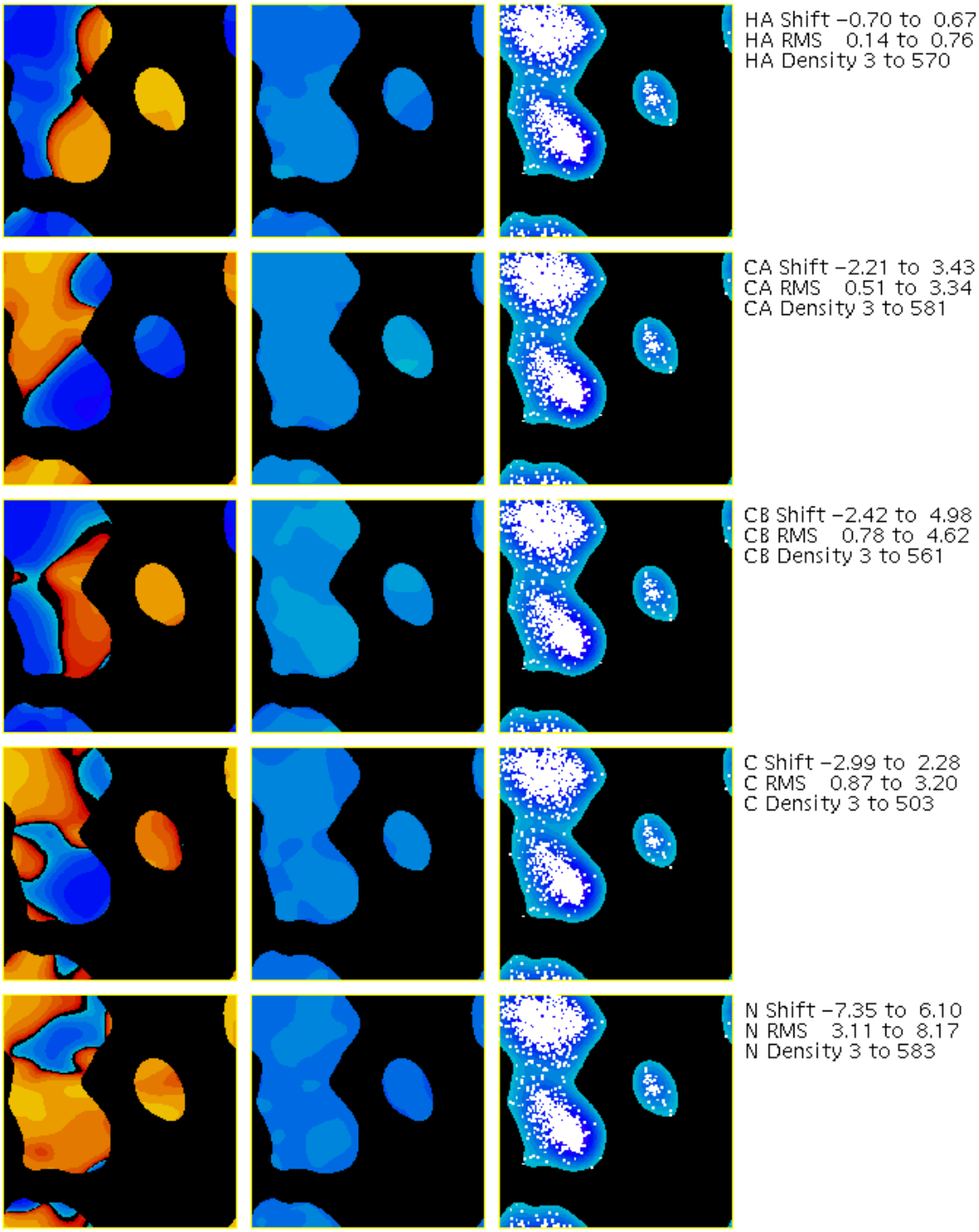


Figure 3.7. Shift prediction surfaces for the generic class of residues (those not Gly, Pro, or xPro). The five rows of surfaces represent data for the five types of secondary chemical shifts $^{1}H^{\alpha}$, $^{13}C^{\alpha}$, $^{13}C^{\beta}$, $^{13}C'$ and ^{15}N. In a given row, the leftmost surface shows the predicted secondary chemical shift with respect to ϕ and ψ, and the central surface shows the corresponding estimated standard deviation in the prediction. The rightmost surface shows the density of observations, with individual (ϕ,ψ) locations from the database superimposed as white points. The ranges for each surface are given to the right of each row; red shades of the surface are negative values, blue shades are positive.

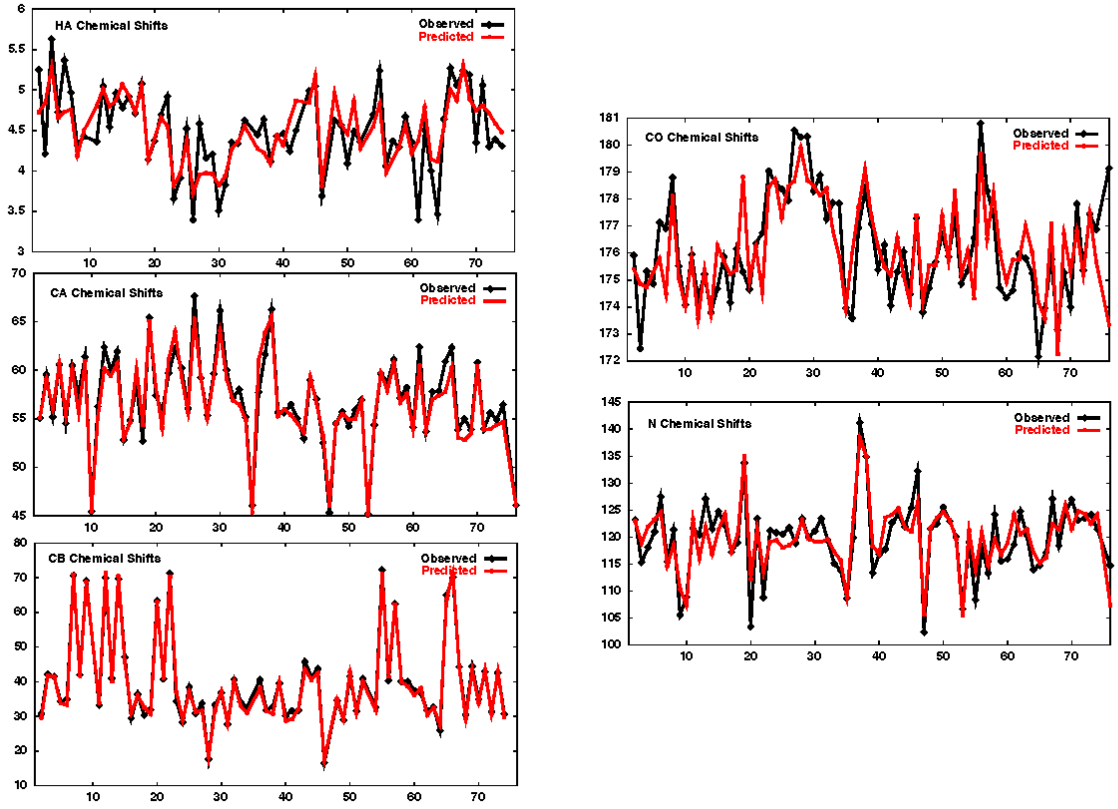


Figure 3.8. Observed (black) and predicted (red) chemical shifts in ppm versus residue number. The simulations are based on the backbone angles from the X-ray structure of ubiquitin. The ppm rms values between the observed and predicted shifts shown are: $^{1}H^{\alpha}$ 0.31; $^{13}C^{\alpha}$ 0.95; $^{13}C^{\beta}$ 1.19; $^{13}C'$ 1.35; ^{15}N 3.31.

2000) and the crystal structure of the 213-residue HAV (Hepatitis A virus 3C proteinase, PDB code *1qa7) (Bergmann et al., 1999). In the case of DinI, results from Ramirez et al. for a reliable NMR structure are reported, along with results for an erroneous structure produced by automated NOE assignment methods in another lab (Y. Ito, private communication). The two versions of DinI differ by 6.2 Å, although they have similar secondary structure. The statistic χ^2/N, where N is the number of chemical shifts simulated, should ideally approach unity.

As shown, the results for the good structure of DinI are somewhat better than average (χ^2/N of 0.63), while the results for HAV are worse (χ^2/N of 1.49) than any result from the database. It should be noted that since the TALOS database is restricted to the most well-defined regions of the proteins via the B-factor criterion, the simulation results for the database probably underestimate the prediction errors. Also, the HAV chemical shifts are derived from a sample with a bound inhibitor, and this could effect the structure or shifts. In the case of DinI, the erroneous structure has worse agreement with the chemical shifts than the good structure (χ^2/N of 0.91 vs 0.63), although the agreement in both cases is within the expected range. However, the erroneous structure has 13% of its shifts not simulated, because many of its backbone angles are in unrealistic parts of the Ramachandran map.

Table 3.6. Chemical shift prediction statistics listed by protein.

Protein Name	RMSD (ppm)	χ^2/N	N = Number of Shifts Simulated	NS = Number of Shifts not Simulated
HIV-1protease	1.83	0.84	450	5
III-glc	2.12	1.05	679	3
Alpha-lytic-Protease	2.12	1.01	897	0
BPTI	2.19	1.39	279	0
Calbindin	2.06	1.12	209	0
Calmodulin	1.59	0.58	617	0
Calmodulin/M13	1.75	0.71	600	0
Cutinase	1.97	1.02	896	2
Cyanovirin-N	2.33	1.17	436	0
Cyclophilin	2.14	0.96	634	0
Dehydrase	2.03	1.13	696	10
HCA I	2.07	1.07	1156	0
Interleukin-1β	2.03	0.96	483	4
Lactamase	2.36	1.28	950	32
Serine Protease	2.18	1.39	726	15
D-MPB	1.90	0.88	1646	13
Profilin	2.10	1.06	524	4
Staph Nuclease	2.40	1.06	578	5
Human Thioredoxin	1.83	0.76	416	0
Ubiquitin	1.67	0.69	358	0
Average	**2.03**	**1.01**	**661**	**5**
DinI (Reliable Structure)	1.53	0.63	375	13
DinI (Erroneous Structure)	1.99	0.91	337	51
HAV 3C Protease	2.32	1.49	1001	20

The chemical shift simulation results are also summarized by residue type and atom type in Table 3.7. As Table 3.6 and 3.7 both show, a small number of shifts cannot be simulated by this method, because the corresponding (ϕ,ψ) values fall in unpopulated regions of the prediction surfaces. As can be seen in Table 3.7, results for Cys still yield larger prediction errors than all other residue types. Since only a limited number of entries (< 35 for each atom type) were available for Cys in the database, it was not possible to characterize this behavior in detail.

Concluding Remarks

The TALOS approach described here is the first to combine both chemical shift and residue type information for predicting the backbone torsion angles. Also, instead of using the chemical shift information of only a single residue, it considers the chemical shifts and residue types of a string (of length 3, in the present case) to obtain this information.

The weight of a particular secondary shift was adjusted by considering the width of its distribution over a narrow range of backbone torsion angles relative to the entire range of

Table 3.7. Chemical shift prediction statistics listed by residue and atom type.

Res Type	$^{1}H^{\alpha}$ RMSD (ppm) and count	$^{13}C^{\alpha}$ RMSD (ppm) and count	$^{13}C^{\beta}$ RMSD (ppm) and count	$^{13}C'$ RMSD (ppm) and count	^{15}N RMSD (ppm) and count
Ala	0.35 254	0.96 265	1.15 263	1.51 235	4.00 263
Arg	0.38 93	1.20 95	1.26 89	1.17 86	3.63 95
Asn	0.38 144	0.89 149	1.46 146	1.59 127	3.75 145
Asp	0.33 159	1.05 162	1.30 159	1.40 143	4.03 162
Cys[a]	0.41 14	2.10 14	1.32 14	1.80 6	3.93 14
cys[b]	0.75 19	1.69 19	2.48 18	1.62 19	2.55 19
Gln	0.31 107	1.01 106	1.26 104	1.21 83	3.63 106
Glu	0.28 183	1.09 183	1.27 168	1.16 142	3.20 186
Gly	0.39 509	0.93 277	-- 0	1.62 244	4.00 277
His	0.55 51	1.65 52	2.24 52	1.92 46	4.76 50
Ile	0.39 167	1.43 171	1.57 170	1.42 152	4.09 170
Leu	0.38 238	0.97 242	1.39 236	1.35 204	3.84 244
Lys	0.32 191	1.20 192	1.13 184	1.25 147	3.44 194
Met	0.42 58	1.40 59	1.91 56	1.25 49	4.03 57
Phe	0.45 131	1.42 133	1.39 130	1.44 93	3.43 131
Pro	0.40 109	0.95 115	1.10 107	1.67 92	1.56 6
Ser	0.38 173	1.41 175	1.09 169	1.62 145	3.93 174
Thr	0.36 198	1.46 201	1.34 199	1.34 175	4.84 198
Trp	0.43 31	1.92 35	1.45 35	1.52 29	3.97 34
Tyr	0.46 79	1.63 83	1.39 82	1.31 75	4.00 81
Val	0.42 216	1.38 216	1.04 215	1.45 184	4.46 216
xPro	0.35 115	1.42 122	1.59 98	1.51 71	4.08 122
All	**0.38 3239**	**1.23 3066**	**1.34 2694**	**1.45 2547**	**3.95 2944**

[a]Reduced form. [b]Oxidized form.

secondary chemical shifts in the database. The relative importance of the chemical shifts versus residue homology has been adjusted empirically to yield the most reliable predictions for proteins of known structure. Remarkably, the weighting factors for the center-residue in the string of 3 residues in Table 3.2 is only slightly higher than for its two flanking residues, indicating that they are of comparable value when predicting a residue's ϕ/ψ angles. The contribution from the residue type homology to the similarity factor *S* is rather modest, typically about 25%**.** Nevertheless, reliability of TALOS predictions is considerably improved when including this residue type homology.

At the outset of developing this approach, we anticipated being able to obtain χ_1 angle predictions too. However, these χ_1 results so far appear insufficiently reliable for general use. Three possible reasons for this are that (1) chemical shifts of the backbone nuclei are not sufficiently sensitive to χ_1, (2) in the crystal structures it is not possible to reliably and routinely separate residues with a single χ_1 conformation from those which undergo χ_1 rotameric averaging, and (3) there are practical difficulties in comparing χ_1 angles for residues with different types of sidechains, i.e., a C^{β}-branched residue such as Thr with a non-branched residue.

Although it may be feasible to develop criteria which yield useful TALOS χ_1 predictions, it is expected that it will be difficult to make predictions that are more reliable than those based on residue type and a residue's own backbone angles, as implemented by Kuszewski et al. (Kuszewski et al., 1997). Our results indicate that concerted use of ^{15}N, $^{13}C^{\alpha}$, $^{1}H^{\alpha}$, $^{13}C^{\beta}$ and $^{13}C'$ chemical shifts of triplets of adjacent residues can be used to predict the backbone torsion angles for the majority of residues in assigned proteins.

When using the crystal structure as the standard, the accuracy of the TALOS prediction appears to exceed that of even some of the best solution structures calculated on the basis of NOEs and J couplings. In principle, one could possibly argue that, as the angles in the database are all derived from crystal structures, one might expect the TALOS output to be closer to the crystal structure than to the solution structure. However, this argument is clearly invalid as it would require a systematic (as opposed to a random) difference between torsion angles in crystal structures and in solution. Second, when comparing the TALOS output for ubiquitin with a solution structure calculated by including a large number of $^{13}C^{\alpha}$-$^{1}H^{\alpha}$, $^{13}C^{\alpha}$-$^{13}C'$, ^{1}H-^{15}N, $^{13}C'$-^{15}N and $^{13}C^{\alpha}$-$^{13}C^{\beta}$ dipolar couplings (Bax and Tjandra, 1997, Marquardt et al., private communication), the agreement of the TALOS-predicted angles with the solution structure is actually better than with the crystal structure, with rmsd's of 10° (solution) and 12° (X-ray) for ϕ and 8° (solution) and 9° (X-ray) for ψ. The rmsd between crystal structure and solution structure torsion angles is 7° for both ϕ and ψ.

The 3% fraction of TALOS predictions which are found to be in disagreement with the crystal structure includes residues which may adopt a different conformation in the solution and crystal structures (e.g., Thr^{45} in cutinase, discussed above), although most of these regions where differences occur are excluded by the B-factor criterion (see Materials and Methods). For most proteins used in our database, no high resolution solution structure is available, and it therefore was not possible to exclude these residues from the database. A set of residues in the database for which the solution backbone angles differ strongly from those in the crystalline state does not increase the number of errors when TALOS is applied to a new protein. Instead, if their chemical shifts match those of the query triplet, they result in an outlier in the display of Figure 3.4. The same is true if a small fraction of residues in the database is wrongly assigned.

It also should be pointed out that a database approach such as the one described here tends to predict torsion angles that fall closer to the most commonly occupied regions of the Ramachandran map than the true value. This is a direct result of the fact that TALOS angles are derived from a set of triplets with the most similar chemical shifts: First, if the true backbone angles of a given center-residue position it somewhere on the edge of the most populated region of the Ramachandran map, there statistically will be a larger number of "hits" inside than outside the most populated region, simply because the density of residues is higher in the most populated region. This effect is visible in Figure 3.5B, for example, where for residues with X-ray ψ angles in the -25° to +25° range the predicted ψ angles are shifted in the direction of the α-helical region of the Ramachandran map. Similarly, for residues with unusually large ψ angles in the X-ray structure, the predicted values consistently are shifted slightly towards the more populated region near ψ = 130°. Second, in rare cases where residues are located far outside the populated

region of the Ramachandran map (such as Asp[19] in Staphylococcal nuclease), no other triplet with such unusual angles may be present in the database. If TALOS finds a cluster of triplets that accidentally match the shifts and residue types of the query triplet, it is likely that the torsion angles in this cluster fall in the highly populated region of the Ramachandran map. Both these types of problems may be alleviated when the database becomes larger.

It is important to realize that the TALOS-derived ϕ/ψ-values are empirical in nature. In a conservative approach, deviations between these ϕ/ψ-values and those in structures calculated on the basis of regular experimental restraints can be used for "trouble-shooting" purposes. Alternatively, in cases where an insufficient number of regular experimental constraints are available, preliminary results on ubiquitin suggest that incorporation of the TALOS-derived ϕ/ψ-values can enhance structural quality considerably. Collecting a large number of NOEs can be particularly difficult in larger proteins, which require extensive deuteration. It is expected that the use of TALOS-derived torsion angle restraints, when combined with one-bond dipolar couplings measured in dilute liquid crystalline media (Bax and Tjandra, 1997; Clore et al., 1998; Hansen et al., 1998; Bewley et al., 1998), will make it possible to obtain reliable backbone structures for such larger systems, even if only a limited number of NOEs is available.

As we have also shown, the contents of the TALOS database can also be used to create a chemical shift prediction scheme. The results also indicate the following typical ppm rmsd values for our method of chemical shift prediction based on secondary structure: $^{1}H^{\alpha}$ 0.38; $^{13}C^{\alpha}$ 1.23; $^{13}C^{\beta}$ 1.34; $^{13}C'$ 1.45; ^{15}N 3.95. These values may represent approximate limits on the precision of shift prediction which is possible based exclusively on secondary structure, without consideration of adjacent residue effects, tertiary structure, hydrogen bonding etc. Visual inspection of the chemical shift surfaces (as in Figure 3.7) also suggests that there may be a certain unavoidable degree of ambiguity in the relationship between combinations of secondary shift and (ϕ,ψ). For example, in all cases but $^{13}C'$, the typical sign and magnitude of secondary shifts in the positive ϕ region is similar to that found in parts of the corresponding alpha-helical region of a given surface. In future, it should be possible to quantify this degree of ambiguity, so that a particular residue can be classified according to the complete set of (ϕ,ψ) whose predicted shifts match the observed ones.

Like the TALOS (ϕ,ψ) predictions, the chemical shift predictions can also be used for simple trouble-shooting as well, for example to help identify misfolded shifts. It should be noted though, as for the two cases of DinI in Table 3.6, that the backbone-based chemical shift predictions alone will generally not distinguish between proteins with similar secondary structure but differing tertiary structure. As we will show in the next section, this prediction method is enhanced when combined with information obtained from dipolar couplings.

Acknowledgement

For assistance in constructing and evaluating TALOS, thanks goes to S. Archer, V. Basus, R. Boelens, W. Chazin, G.M. Clore, B. Farmer, S. Fesik, K. Gardner, P. Hansen, M. Ikura, M. Ottiger, J. Prompers, S. Scrofani, and D. Torchia for providing chemical shift assignments included in database, and J. Marquardt, M. Ottiger, and J.-S. Hu for useful discussions.

4. Protein structure modeling using dipolar coupling and chemical shift homology

Introduction

Determination of the three-dimensional (3D) structure of a protein in solution from NMR data has relied primarily on the measurement of a large number of interproton distances (NOEs), supplemented by torsion angle restraints derived from J couplings and chemical shifts (Wüthrich, 1986; Edison et al., 1994; Gronenborn and Clore, 1995). Recently developed methods for measurement of dipolar couplings (Tolman et al., 1995; King et al., 1995; Tjandra et al., 1996; Bax and Tjandra, 1997; Tjandra and Bax, 1997) provide additional structural information which can be used to improve the accuracy of the NMR-derived protein structure (Tjandra et al, 1997; Ottifer et al, 1997; Bewley et al., 1998; Drohat et al., 1999; Clore et al., 1999). Here, we describe a novel approach for determining the backbone structure of a protein solely from dipolar couplings. The first stage of our method, which we refer to as molecular fragment replacement (MFR), is analogous to a method described by Kraulis and Jones for determining local fragment structures from NOE patterns (Kraulis and Jones, 1987), and also is similar to the commonly used database approach for fitting the main chain electron density of protein X-ray structures (Jones and Thirup, 1986). It also bears some similarity to a recently described approach for identifying the fold of a protein from its dipolar couplings by searching a database (Annila et al., 1999), but this latter method requires a very similar structure to be present in the database, and therefore is not a *de novo* method.

The MFR method is demonstrated for the protein ubiquitin, for which a 1.8 Å X-ray crystal structure is available (Vijay-Kumar et al., 1987), and which has been studied extensively by NMR (DiStefano and Wand, 1987; Weber et al., 1987; Schneider et al., 1992; Tjandra et al., 1995; Hu and Bax, 1997). The ordered part of its NMR structure (residues 2-72) is in excellent agreement with the X-ray structure, with an rmsd of 0.35 Å (Cornilescu et al., 1998). Four backbone couplings (N-H; C'-N; C'-H^N; $C^\alpha H^\alpha$) have previously been measured for most residues in ubiquitin, but less for Pro, residues preceding Pro, and residues with broadened or missing amide resonances Ottiger and Bax, 1998). The couplings were measured in two different liquid crystalline phases, yielding information on the internuclear vector orientations relative to two different axes systems (Ramirez and Bax, 1998).

A thorough discussion of dipolar couplings in macromolecular structure determination is provided by Bax and coworkers (Bax et al., 2001). Some details are reproduced here.

Alignment methods. If a molecule tumbles isotropically in solution, dipolar couplings will be averaged to zero. However, if some influence interferes with the isotropic tumbling to introduce a degree of alignment, dipolar couplings will no longer average to zero, and can be measured. Tolman et al. were the first to apply the magnetic alignment approach to a protein (Tolman et al., 1995). Using paramagnetic myoglobin, uniformly enriched in ^{15}N, they showed a clear field dependence of the one-bond ^1H-^{15}N splitting correlated that with the orientation of the corresponding internuclear vector relative to the calculated magnetic susceptibility anisotropy tensor. Similar measurements were reported independently and simultaneously by Bolton and co-workers for an unlabeled DNA dodecamer (Kung et al., 1995). In both myoglobin and the DNA dodecamer, changes of several Hertz in one-bond dipolar coupling between low and high-field measurements were reported, corresponding to alignment factors on the order of 10^{-4}.

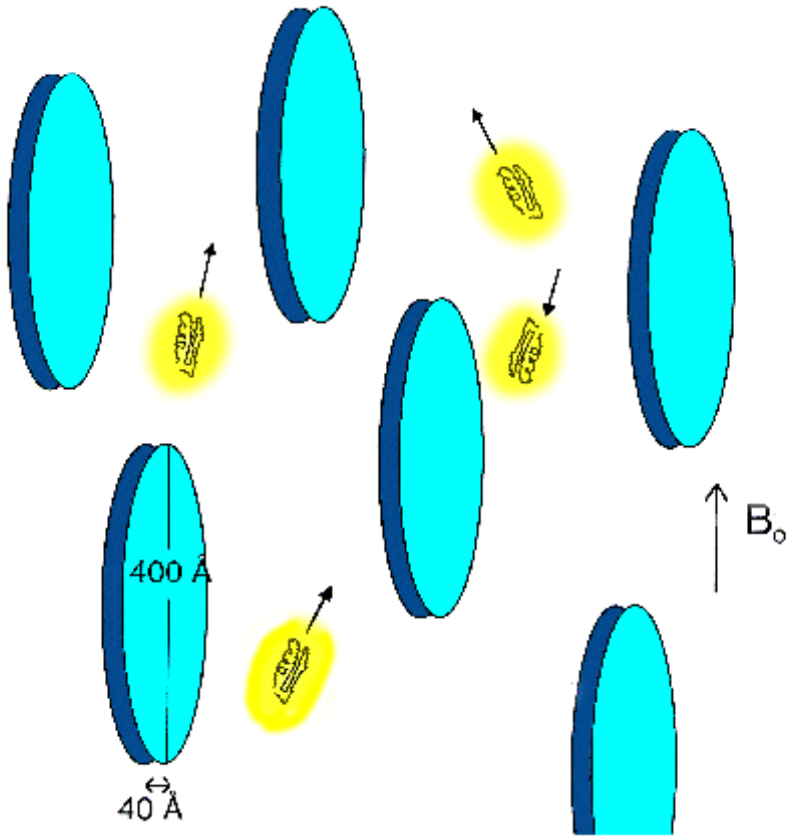


Figure 4.1. Cartoon of proteins in a bicelle liquid crystal. The disk-shaped bicelles (blue) align with the magnetic field B_0. Occasional collisions between the protein and bicelle prevent the protein from tumbling isotropically, inducing partial alignment of the protein.

Much smaller effects were observed for a small diamagnetic model protein, ubiquitin, where the largest dipolar contribution to the change in ^{1}H-^{15}N splitting between 360 and 600 MHz was only 0.2 Hz (Tjandra et al., 1996).

Liquid crystal media were later employed successfully to introduce more substantial alignment, relying on occasional steric collisions or electrostatic interactions between the biomolecule and the ordered particles in the liquid crystal to impart a degree of alignment, as diagrammed in Figure 4.1. Bicelles were the first liquid crystalline medium used for weak alignment of proteins and DNA (Bax and Tjandra, 1997; Tjandra and Bax, 1997). These are planar micelle particles usually consisting of regular saturated phospholipids. Most commonly, a mixture of dimyristoyl phosphatidyl choline (DMPC) and dihexanoyl phosphatidyl choline (DHPC) is used. DMPC makes up the bilayer, which constitutes the plane of the micelle, and the DHPC detergent, mostly covers the rim. Bicellar liquid crystals were originally developed by Prestegard, Sanders and co-workers for the purpose of studying lipophilic molecules, anchored in these highly ordered membranes (Sanders and Prestegard 1990; Sanders and Prestegard, 1991). The DMPC/DHPC combination was found to be particularly robust. When raising the temperature above 25 ˚C, the

system switches from isotropic to a nematic liquid crystalline phase (Sanders and Schwonek, 1992).

Use of a liquid crystalline solution consisting of filamentous phages for aligning macromolecules was introduced simultaneously and independently by Clore et al. and by Pardi et al. (Clore et al., 1998; Hansen et al, 1998A,B). Clore et al. used a medium consisting of the filamentous phage *fd*, and also demonstrated the potential of tobacco mosaic virus (TMV). The Pardi group used the phage Pf1, which is similar to *fd*, but which at a contour length of *ca* 2 μm is nearly twice as long. Both phages have a diameter of 6.5 nm, and a persistence length of *ca* 1 μm. As a result of the very high aspect ratios of these particles, solutions can remain liquid crystalline down to very low concentrations, as low as a few mg/ml.

Bacteriorhodopsin is an integral membrane protein with seven transmembrane helices. It is present at extremely high density (75% w/w) in the cell membrane of *Halobacterium salinarium* (Oesterhelt and Stoeckenius, 1974). Fragments of this so-called purple membrane (PM) are easily prepared in large quantities and have been used for a wide array of optical and biophysical experiments. The PM fragments have an average diameter of about 1 μm, and this is sufficiently large that the total magnetic susceptibility anisotropy of such a particle causes it to align nearly 100% when placed in a strong magnetic field (≥11 T), with the membrane plane orthogonal to the direction of the magnetic field (Lewis et al, 1985). So, in contrast to the bicelle, phage and related systems, PM does not need to form a liquid crystalline phase for obtaining alignment.

Very recently, Tycko et al. proposed the use of a different type of medium for aligning proteins: the use of a compressed gel (Tycko et al., 2000). This gel is highly (>90%) hydrated but is relative stiff due to the presence of extensively cross-linked polyacrylamide. Protein can be soaked into the gel, and when the gel is compressed, pore shape inside the gel becomes non-random with respect to the direction in which the pressure is applied, inducing preferred alignment of the protein. The potential advantages of this system are that it is very inert, carries virtually no charge, and can be done over a wide range of polymer density, temperature, pH, and ionic strength. Also, it generally will be relatively easy to extract the solute from the gel. Potential disadvantages may be that it can be difficult to make the system sufficiently homogeneous for obtaining very high-resolution spectra.

Mathematical description of dipolar couplings. The relation between the internuclear vector and the dipolar coupling between two spins, *A* and *B* can be found in numerous textbooks. For the purpose of deriving the resonance frequencies (i.e., dipolar splittings) only the *z* component of the local field of one nuclear dipole at the position of the second nucleus is relevant (secular approximation):

$$H_{dd} = D^{AB}_{max} < I_{Az}\, I_{Bz}\, (3\cos^2\theta - 1) > \qquad (4.1A)$$

where the <> brackets refer to the time or ensemble average, which are equivalent for isotropic and liquid crystalline solution, θ is the angle between the *A-B* internuclear vector and the magnetic field (Figure 4.2), and:

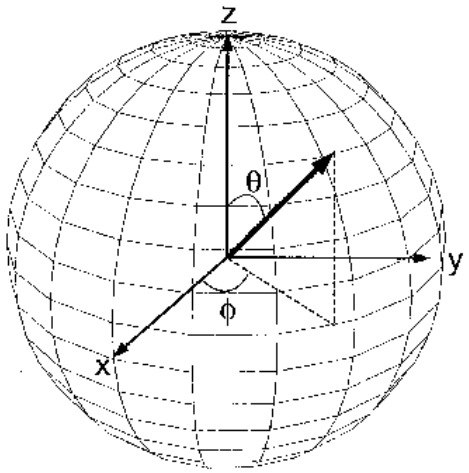


Figure 4.2. Internuclear coupling vector D^{AB}, which forms angle θ with respect to the principal axis.

$$D^{AB}_{max} = -\mu_o(h/2\pi)\gamma_A\gamma_B/(4\pi^2 r_{AB}^3) \quad (4.1B)$$

is the static dipolar coupling, in SI units, which equals 21.7 kHz for H-N pairs, assuming an internuclear distance r_{NH} = 1.04 Å. The constant, $-\mu_o$, is the magnetic permittivity of vacuum, h is Planck's constant, γ_X is the magnetogyric ratio of spin X, and r_{AB} is the distance between nuclei *A* and *B*. The residual dipolar splitting between spins *A* and *B* equals

$$D^{AB} = D^{AB}_{max} \langle P_2(\cos\theta)\rangle \quad (4.1C)$$

with $P_2(x) = {}^1/_2(3x^2 - 1)$.

If the molecule is rigid, the orientation of the internuclear vector, r_{AB}, in an arbitrary molecular coordinate system can be described by the angles α_x, α_y, and α_z between the vector and the *x, y*, and *z* axis of the coordinate system. The angles β_x, β_y, and β_z define the instantaneous orientations of each of these axes relative to the static magnetic field. With cosθ being the scalar product between a unit vector in the internuclear direction and a unit vector parallel to B_o, $P_2(\cos\theta)$ can be rewritten as:

$$\langle P_2(\cos\theta)\rangle = {}^3/_2 \langle (\cos\beta_x\cos\alpha_x + \cos\beta_y\cos\alpha_y + \cos\beta_z\cos\alpha_z)^2 \rangle - {}^1/_2 \quad (4.2A)$$

With $C_i = \cos\beta_i$ and $c_i = \cos\alpha_i$, this can be rewritten as:

$$\langle P_2(\cos\theta)\rangle = {}^3/_2 [\ \langle C_x\rangle^2 c_x^2 + \langle C_y\rangle^2 c_y^2 + \langle C_z\rangle^2 c_z^2 + 2\langle C_x C_y\rangle c_x c_y + 2\langle C_x C_z\rangle c_x c_z + 2\langle C_y C_z\rangle c_y c_z\ ] - {}^1/_2 \quad (4.2B)$$

By writing $S_{ij} = {}^3/_2 \langle C_i C_j\rangle - {}^1/_2 \delta_{ij}$, where δ_{ij} is the Kronecker delta function, we obtain:

$$<P_2(\cos\theta)> = \Sigma_{i,j=\{x,y,z\}}\, S_{ij} \cos\alpha_i \cos\alpha_j \qquad (4.2C)$$

The 3x3 matrix *S* is commonly referred to as the Saupe matrix, the Saupe order matrix, or simply the order matrix. As $<C_x>^2 + <C_y>^2 + <C_z>^2 = 1$, the matrix *S* is traceless, and with $<C_i\, C_j> = <C_j\, C_i>$, *S* is also symmetric, and therefore only contains five independent elements.

If the structure of the molecule is known, *i.e.* $\cos\alpha_i$ factors in Eq. 4.2C are known, the five independent elements of this matrix generally can be solved provided that dipolar couplings for at least five internuclear vectors are available. However, if any pair of internuclear vectors is parallel, and for other special cases such as a set that includes three mutually orthogonal interactions, more measured couplings are required. For macromolecules, many more dipolar couplings are frequently measured, and *S* is overdetermined. Its elements are then best determined using singular value decomposition (Losonczi et al., 1999).

The order matrix is real and symmetric, and it therefore is always possible to define a molecular axis system where *S* becomes diagonal. As will become clear below, in a number of applications it can be advantageous to work in this principal axis frame, where Eq. 4.2B now simplifies to:

$$D^{AB}(\alpha_x, \alpha_y, \alpha_z) = {}^3/_2\, D^{AB}_{max} \{[<C_x>^2 c_x^2 + <C_y>^2 c_y^2 + <C_z>^2 c_z^2] - 1\} \qquad (4.3A)$$

where $<C_i>^2$ corresponds to the probability of finding the *i*-th axis parallel to the magnetic field. Only the relative differences in the $<C_i>^2$ values contribute to the residual dipolar coupling. So, writing $<C_i>^2 = {}^1/_3 + A_{ii}$, Eq. 4.3A can be expressed in polar coordinates ($\theta = \alpha_z$; $c_z = \cos\theta$; $c_x = \sin\theta\cos\phi$; $c_y = \sin\theta\sin\phi$) to yield:

$$D^{AB}(\theta,\phi) = {}^3/_2\, D^{AB}_{max} [\cos^2\theta\, A_{zz} + \sin^2\theta\, \cos^2\phi\, A_{xx} + \sin^2\theta\, \sin^2\phi\, A_{yy}] \qquad (4.3B)$$

Defining $|A_{zz}| > |A_{yy}| > |A_{xx}|$, and using $A_{yy} + A_{xx} = -A_{zz}$; $2\sin^2\phi = 1 - \cos2\phi$; and $2\cos^2\phi = 1 + \cos2\phi$, this can be rewritten as:

$$D^{AB}(\theta,\phi) = {}^3/_2\, D^{AB}_{max} [P_2(\cos\theta)\, A_{zz} + {}^1/_2 \sin^2\theta\, \cos2\phi\, (A_{xx} - A_{yy})] \qquad (4.3C)$$

Defining an axial component of the alignment tensor $A_a = {}^3/_2 A_{zz}$, and a rhombic component, $A_r = (A_{xx} - A_{yy})$, then results in:

$$D^{AB}(\theta,\phi) = D^{AB}_{max} [P_2(\cos\theta)\, A_a + {}^3/_4\, A_r \sin^2\theta\, \cos2\phi] \qquad (4.3D)$$

Note that the maximum value for $<C_i>^2$ is one, i.e., the maximum for A_{zz} equals 2/3, and the maximum value for A_a becomes one when the *z* axis of the principal alignment tensor becomes fully aligned with the static field. In practice, the dilute liquid crystal work discussed in this chapter concerns A_a values on the order of 10^{-3}. Eq. 4.3D is sometimes rewritten as:

$$D^{AB}(\theta,\phi) = D^{AB}_a [(3\cos^2\theta - 1) + {}^3/_2\, R \sin^2\theta\, \cos2\phi] \qquad (4.3E)$$

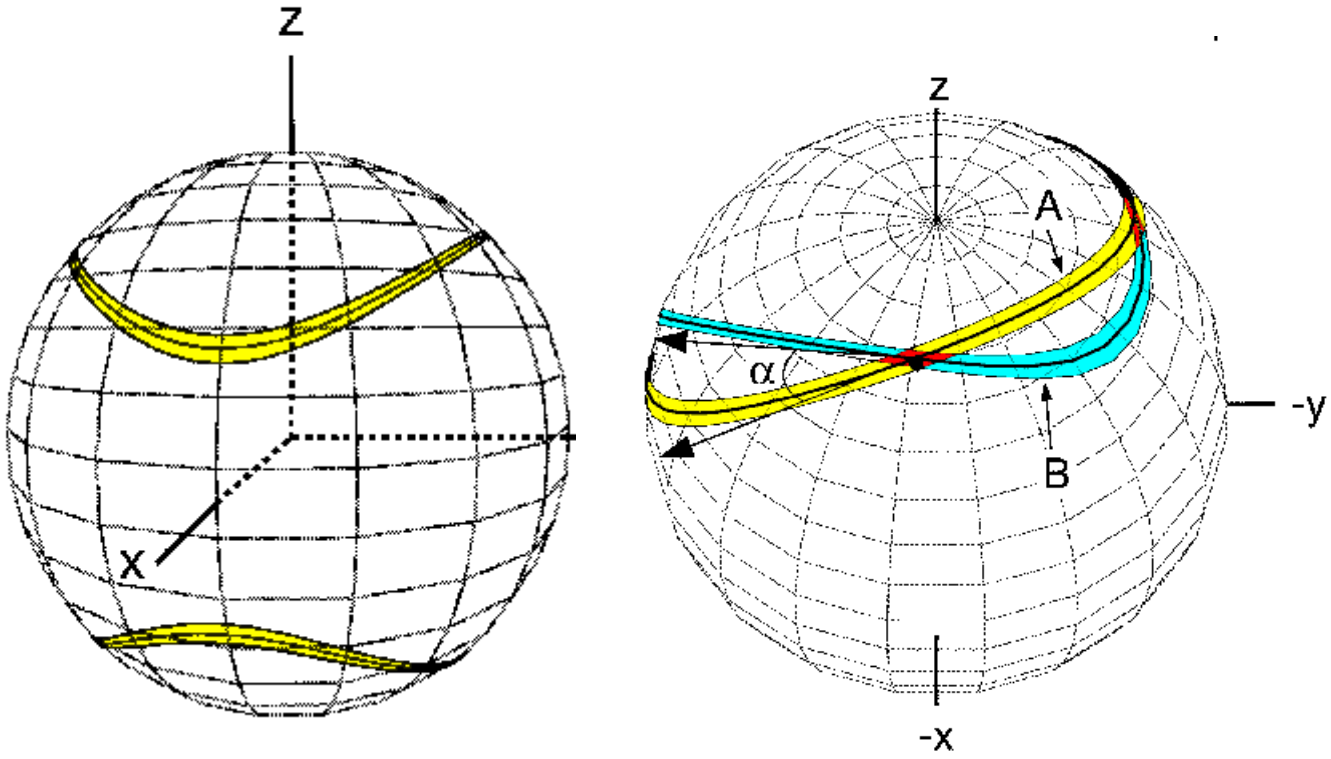


Figure 4.3. Orientations of an internuclear coupling vector D^{AB} which are consistent with a given dipolar coupling. A dipolar coupling measured at a single alignment tensor corresponds to two continuous ranges of orientations (A, yellow), which are mirror images of each other. By introducing data for a second alignment tensor, with a different set of corresponding orientations (B, blue) the allowed orientations are restricted to the intersections (red).

---

where $D^{AB}{}_a = {}^1/_2 D^{AB}{}_{max}$ is referred to as the magnitude of the dipolar coupling tensor, frequently normalized to the N-H dipolar interaction, and $R = A_r/A_a$ is the rhombicity.

A critical aspect of the dipolar couplings is their dependence on $\cos^2\theta$, which in practice means that there are two continuous ranges of orientation for the internuclear coupling vector which are consistent with a given coupling value, and they are mirror images of each other (Figure 4.3). A simple way to reduce this ambiguity is to introduce couplings measured at another alignment tensor, which restricts the orientations to only those positions which are consistent with both alignment tensors simultaneously. Then, only the intersecting orientations will be consistent with coupling values from both samples (Figure 4.3). Within the context of a protein, residues are arranged in preferred orientations relative to each other, and in most cases, only one of these will be consistent with the collection of dipolar couplings. So, one way to reduce the impact of ambiguity is to only consider physically realistic protein conformations. However, there will still generally be cases that more than one conformation is consistent with the dipolar data. To help resolve potential ambiguity still further, we can employ secondary structure information from chemical shifts, using the methods described in Section 3.

As a simple example of the extreme sensitivity of dipolar couplings to structural details, we consider the 81-residue protein DinI, discussed in the previous section (Ramirez et al., 2000). As shown in Table 3.6, there is only a small difference in backbone chemical shift simulation statistics for two versions of the structure which differ by 6.2 Å. However, there is a remarkable difference in the agreement of the two structures with respect to the measured dipolar couplings.

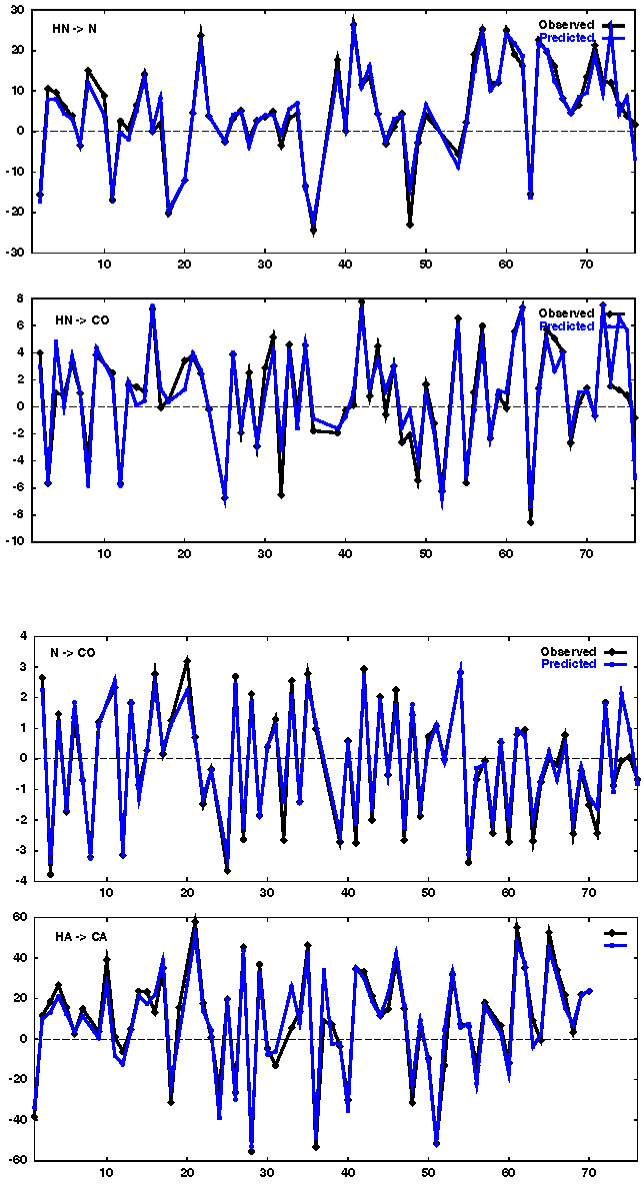


Figure 4.4. Observed and predicted dipolar couplings in Hz vs residue number using the crystal structure of ubiquitin.

For the bicelle data, the rmsd agreement of the "good" structure to the dipolar couplings is 3.98 Hz relative to H-N, while the erroneous structure has an rmsd agreement of 9.04 Hz. For the data measured in phage, the difference is even more dramatic: 3.65 Hz compared to 15.31 Hz. In this case, the dipolar couplings unquestionably reveal the inaccuracy of the structure.

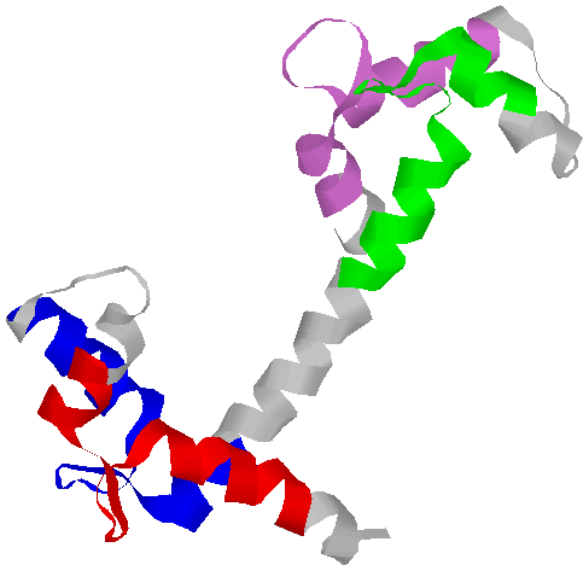


Figure 4.5. Crystal structure of calmodulin. The 25-residue ranges spanning the four calcium binding loops are highlighted: residues 10-34 (the query fragment), red; 46-50 blue; 83-107 green; 119-143 purple.

Methods

Measurement details of the ubiquitin dipolar coupling used in this work has been presented previously (Ottiger and Bax, 1998; Ramirez and Bax, 1998). An example of this dipolar coupling data measured for ubiquitin in one of two alignment tensors in shown in Figure 4.4, which shows the four types of observed couplings (N-H; C'-N; C'-H^N; $C^\alpha H^\alpha$) plotted with the corresponding best-fit predicted values, simulated by the SVD method from the crystal structure. The overall rmsd between observed and predicted couplings is 3.69 Hz, scaled relative to H-N.

Our first experiments with a homology-based scheme were performed using couplings and observed chemical shifts (Ikura et al., 1990) for the protein calmodulin. Measured dipolar couplings for this molecule were not available at the time, so example couplings were simulated from the crystal structure (Chattopadhyaya et al.1992), assuming similar alignment tensor parameters as for ubiquitin in Figure 4.4. The test case focused on the four calcium binding loops in calmodulin, which are all structurally similar; a diagram of the molecule with the four loops highlighted is given in Figure 4.5. Because of their structural similarity, the simulated patterns of chemical shifts and couplings for one calcium binding loop should be also similar to that of the other three.

The test of this assumption is shown in Figure 4.6. In this figure, data for a 25-residue fragment spanning the first calcium binding loop (residues 10-34) was used as the query fragment. Then, for each possible 25-residue segment in calmodulin, 1-25, 2-26, 3-27, etc., we simulated its chemical shifts using the methods described in the previous section, and also computed the best-fit of the given segment to the dipolar couplings of residues 10-34. In each case we tallied the differences of the query fragment's measured shifts and couplings and those simulated for each segment in calmodulin. These are plotted with respect to the segment's offset in the sequence.

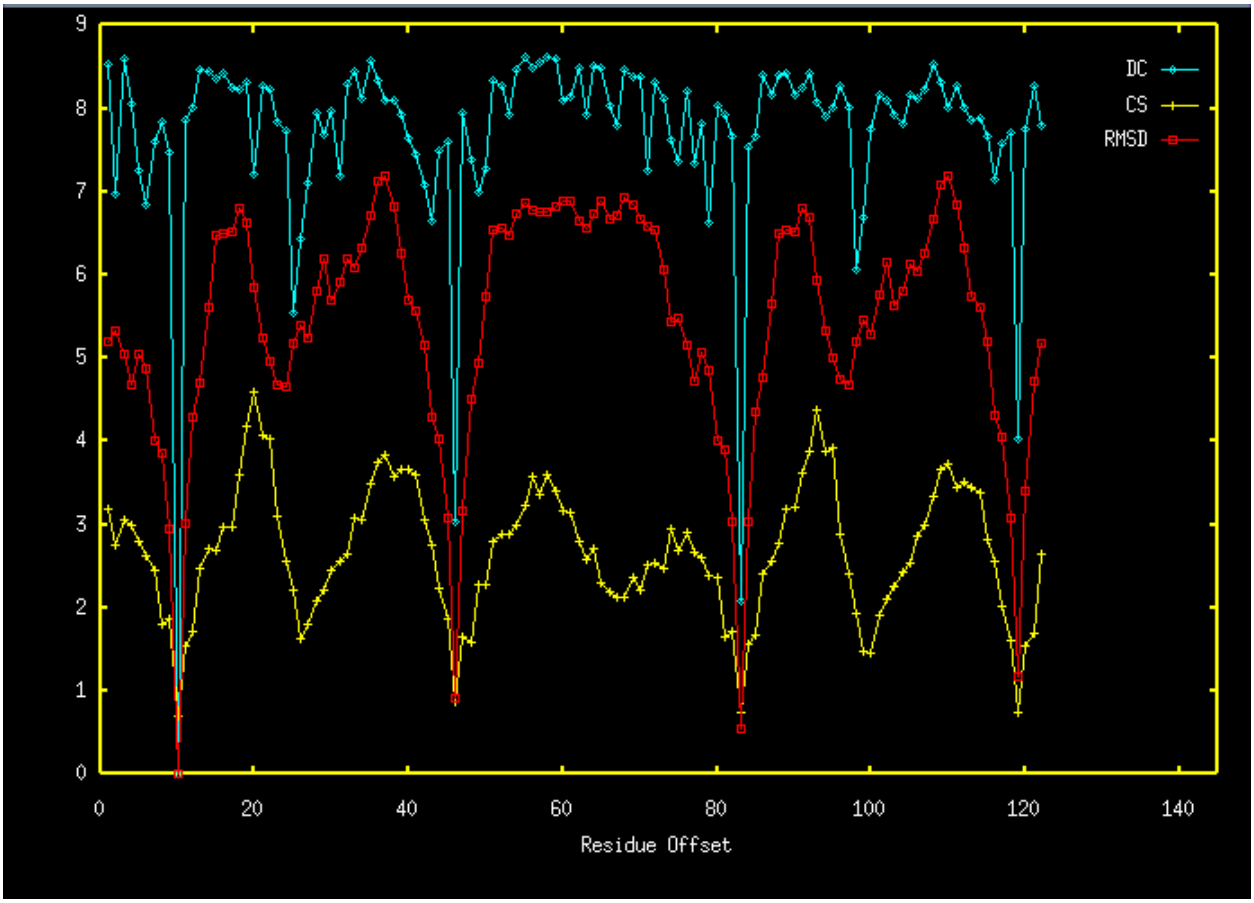


Figure 4.6. Dipolar coupling and chemical shift homology statistics within calmodulin, using the first calcium binding loop as the query segment. Blue: rms agreement between observed and predicted couplings in Hz. Yellow: χ^2/N agreement between observed and predicted chemical shifts. Red: backbone rmsd (Å) between the calmodulin fragment starting at the given residue offset and the calcium binding loop spanning residues 10-34. As shown, deep simultaneous minima in both the coupling and shift statistics corresponds to good agreement in the backbone rmsd.

The yellow line shows the χ^2 between observed and predicted chemical shifts, and the blue line shows the rmsd in Hz for the dipolar couplings. For reference purposes, the backbone rmsd between the test segment and the first calcium binding loop is shown in red. As expected, both the chemical shift and dipolar coupling statistic lines show a strong minimum at an offset of 10, which corresponds to the exact overlap of the query fragment with itself. There are also strong minima in the statistics of both the shifts and couplings at offsets of 46, 83, and 119, and these correspond to the positions of the other three calcium binding loops. Of these, it is the minimum at offset 83 that is the most pronounced. Importantly, this corresponds to the calcium binding loop which has the most similar structure to the query fragment, with an rmsd of 0.54 Å (compared to 0.92 Å and 1.16 Å for the other two segments).

The first test of this homology approach using measured dipolar couplings was performed for a 16-residue fragment in ubiquitin, a beta-sheet hairpin from residues 2-17. In the test, we simulated chemical shifts and dipolar couplings for a collection of 16-residue fragments drawn from the protein structures in the Brookhaven Protein Data Bank (PDB), and found the

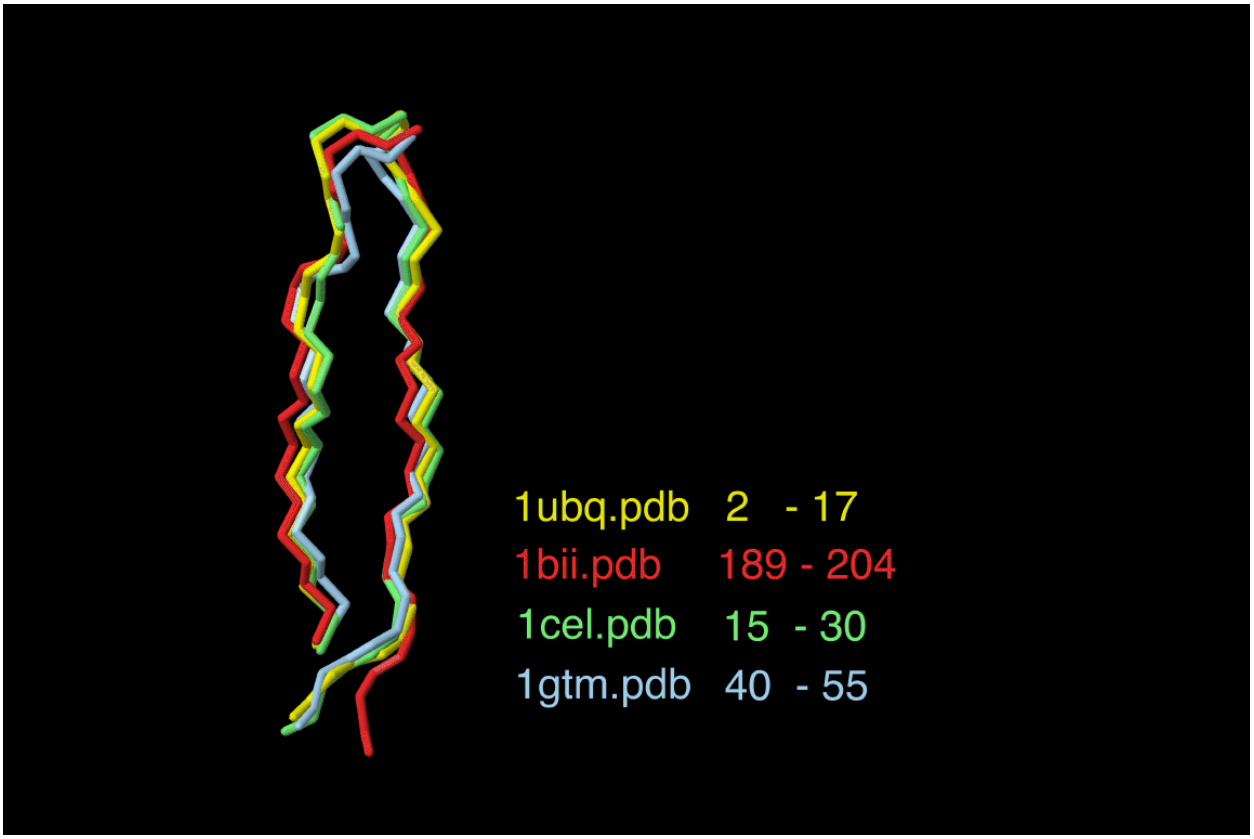


Figure 4.7. Ubiquitin backbone, residues 2-17, drawn with the three best matching fragments from the PDB search based on diplolar coupling and chemical shift homology. The rmsd of the fragments shown relative to the ubiquitin crystal structure is: *1bbi 0.83 Å; *1cel 0.45 Å; *1gtm 0.91 Å.

fragments which gave a best match to the measured shifts and couplings for ubiquitin. To expedite this search, a reduced version of the PDB was created, containing only 1560 proteins, of which two thirds are of a resolution of 2.2 Å or better. The three best fragments from the search, shown in Figure 4.7, all match the ubiquitin backbone to better than 1 Å. This is a critical result, indicating that a database homology search approach can provide quantitatively meaningful structure information.

This immediately suggests the basis for a general approach to the determination of quantitative elements of protein structure, which can be implemented in many possible ways. The approach is called the molecular fragment replacement method (MFR). The basic idea is simple: search through a series of known protein fragments in the database for those whose simulated NMR parameters (in this case, dipolar couplings and shifts) give a best match to those observed for the target protein. Then, assemble the collection of fragments to synthesize larger structural elements.

Results and Discussion

In its present implementation, the MFR method uses a fragment size of 7 residues. We found this to be a reasonable compromise in an attempt to make the fragments as large as possible (so that their patterns of couplings and shifts are as distinctive as possible), but still small enough to

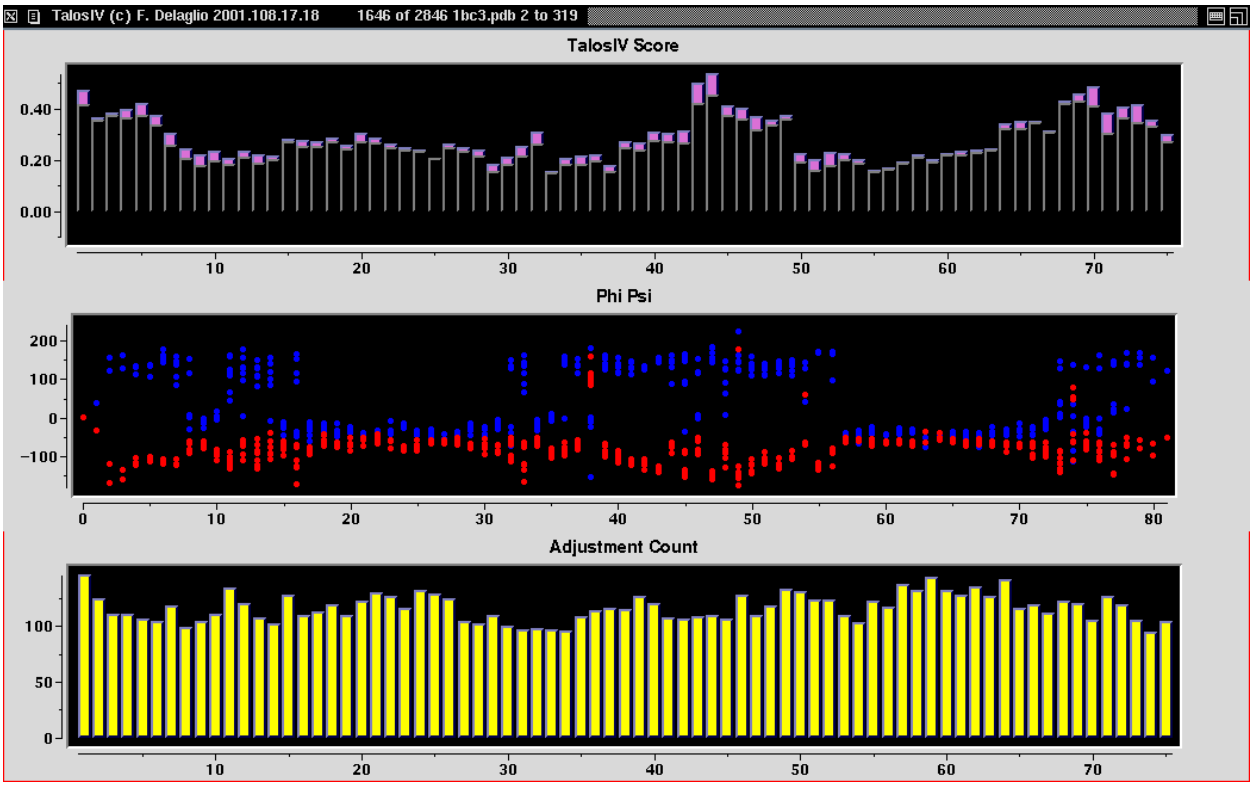


Figure 4.8. Graphical interface showing the status of an NMR Homology search. The bars in the upper graph show the range of NMR homology scores for the collection of best fragments found so far at each residue offset in the query protein ubiquitin. The circles in the central graph show the corresponding distribution of ϕ (red) and ψ (blue) angles for these fragments. The graph at bottom tallies how many times a new fragment from the database search resulted in an improvement over the best fragments found so far.

guarantee that many fragments in the database have a similar shape. In practice, we found that this called for fragments in the range of 5 to 15 residues long. At the 7-residue size, the subset of the PDB used for the search contains 350,000 fragments.

Homology search procedure. For a given 7-residue fragment in the query protein, the entire ensemble of 350,000 PDB fragments are searched to select the 20 best matches on the basis of the lowest χ^2 between measured and best-fitted dipolar couplings and, to weaker degree, the χ^2 between experimental and predicted chemical shifts. This procedure is repeated by shifting the 7-residue fragment by one residue at a time; i.e., for a N-residue query protein the search is carried out N-6 times. A graphical interface, shown in Figure 4.8, displays the progress of the search, which takes roughly 30 sec for each protein in the database.

Ignoring the torsion angles of the first and last residue of each database fragment, the overlapping bundle of 7x20 best-fitting fragments provides 5x20 pairs of ϕ and ψ angles at each residue of the query protein (less for the first and last five residues of the protein). In favorable cases, as typically found near the center of α helices, all "hits" for a given residue cluster in the same region of the Ramachandran map; this is similar to the TALOS approach described in the previous section. As with that approach, frequently there will be outliers, as the dipolar

couplings may not define uniquely the conformation of each individual 7-residue stretch (see below) so that more than one type of 7-residue peptide conformation in the database matches the experimental dipolar couplings. Empirically, we find that if the largest cluster is more than twice as large as the next largest cluster, the median ϕ and ψ angles of the largest cluster provide reliable estimates for the ϕ and ψ angles of the query residue (rmsd of 11˚ and 17˚ relative to angles measured from the ubiquitin X-ray structure). When no single cluster is dominantly populated, the median angles of the most populated cluster are deemed ambiguous and should be used with caution in subsequent structure calculation.

Alternatively, we find that graphical inspection of the (ϕ,ψ) trajectories of the bundle of fragments often resolves ambiguities, for example by highlighting "dead ends", terminal residues of fragments that are not part of a continuous path defining the backbone structure. An example of this visualization is shown in Figure 4.9. It is also useful to create and distinguish a collection of fragments selected solely on the basis of chemical shifts, since many ambiguous cases can be resolved by looking for consensus between fragments selected using dipolar couplings, and fragments selected using chemical shifts alone.

In the case of ubiquitin, using dipolar couplings from two alignment tensors along with the chemical shifts, the 70 best-scoring search fragments for each range of 7 residues had rmsd values of 0.11 to 2.71 Å relative to the crystal structure, with an average value of 0.82; the distribution of backbone rmsd with respect to dipolar coupling homology for these fragments is shown in Figure 4.10. Furthermore, the estimated alignment tensor parameters of magnitude and rhombicity for these 70 best fragments match those determined for the entire protein to better than 5%. So, not only does the search procedure provide substantial numbers of fragments whose conformation is quantitatively similar to those in the query protein, but it also provides a convenient and accurate way to estimate the parameters of the alignment tensor.

The data of Figure 4.10 highlights an important challenge in the use of the dipolar coupling scheme. As shown, the fragments with very good agreement to the crystal structure (< 1 Å) tend to have good agreement with the dipolar couplings (rmsd 1 – 6 Hz). However, there are still several fragments with good agreement to the couplings which are nevertheless different in structure from the known backbone conformation. This relates to the ambiguity in the couplings as discussed above.

Creation of starting structures. Starting structures are built for the contiguous segments with unambiguous ϕ and ψ values, derived in the manner described above. Calculations for ubiquitin and several other proteins yield segment lengths in the 10-50 residue range. Remarkably, a

---

Figure 4.9 (Overleaf). Graphical interface showing the distribution of backbone angles from a dipolar coupling and chemical shift fragment homology search for ubiquitin. In the upper display, a series of Ramachandran maps for residues 2-19 are shown; the ϕ and ψ angles of the database fragments are marked in yellow, and the ϕ and ψ angles for the crystal structure are marked in red. In the lower display, the ϕ,ψ trajectories for the best 10 out of 20 fragments per range are displayed instead, as lines connecting the backbone angles for a given fragment. Inspection of this trajectory display clarifies ambiguities present in the angular distributions for individual residues. For example, the trajectories make it clear that the correct conformations for residues Val^{17} and Glu^{18} are in the beta sheet region, as the trajectories in the alpha helical region are "dead ends", i.e. not part of a continuous trajectory.

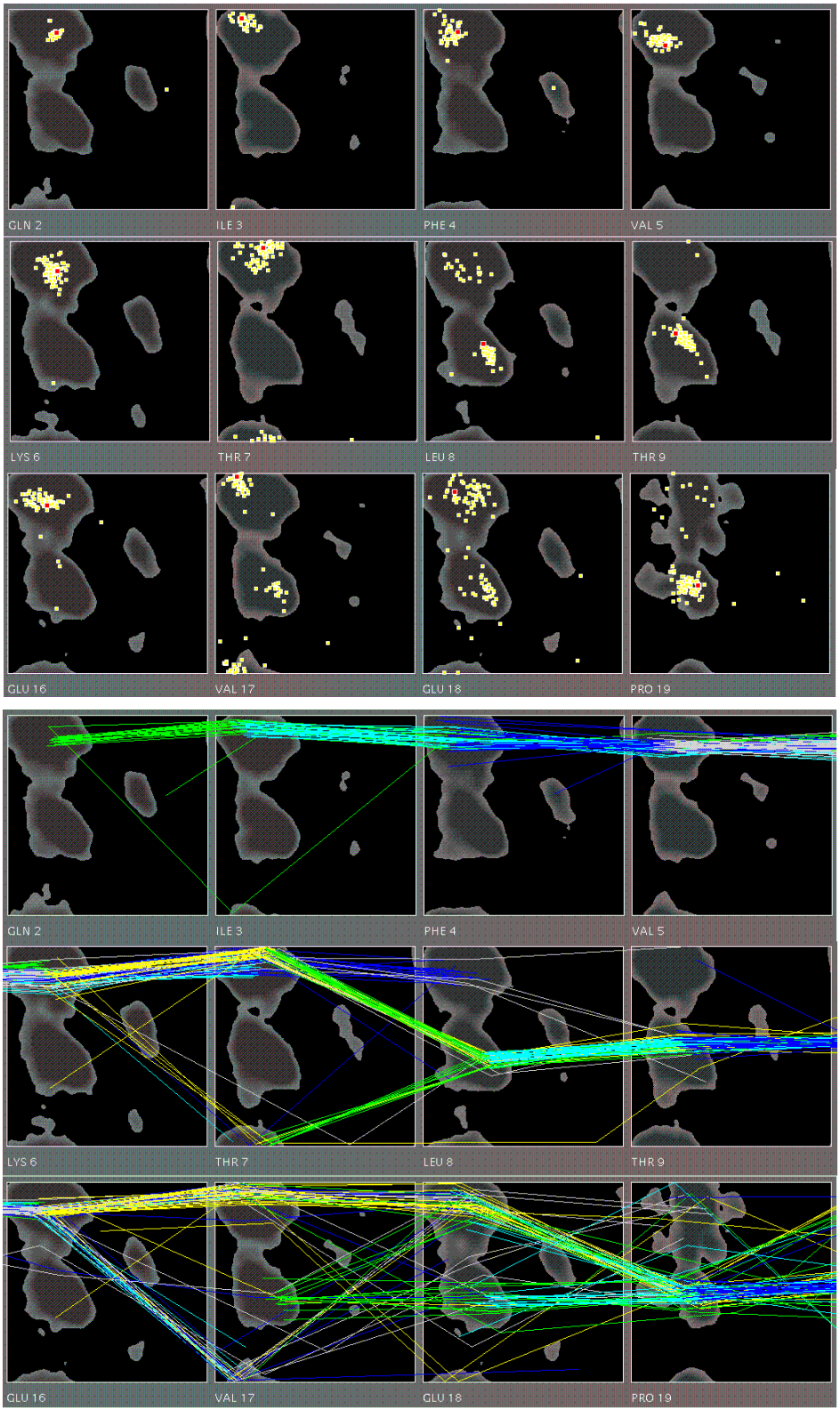


Figure 4.9 (See previous page). Graphical interface showing distribution of fragment backbone angles.

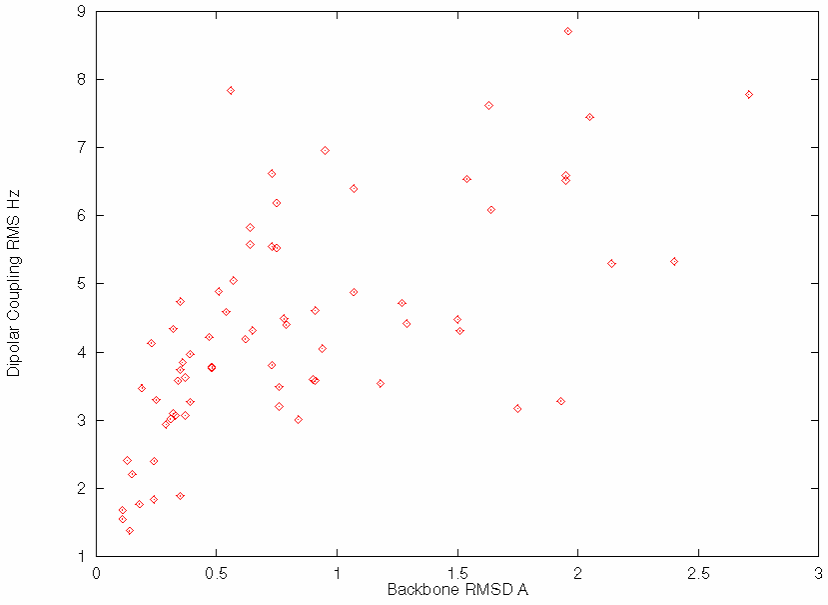


Figure 4.10. Dipolar coupling score versus backbone rmsd (Å) relative to the crystal structure for the 70 best 7-residue fragments found by the database search for ubiquitin.

---

higher rhombicity of the alignment tensor results in less ϕ/ψ ambiguity and longer segment lengths. This is easily understood considering that if for any 7-residue segment either its N-C^{α} or C^{α}-C' bond is parallel to one of the principal axes of the alignment tensor, a database fragment which differs by 180° in the corresponding ϕ or ψ angle will yield couplings for all its residues that are identical to a segment with the correct backbone angles. For an axially symmetric tensor, this type of degeneracy occurs each time a N-C^{α} or C^{α}-C' is parallel or orthogonal to the unique axis of the alignment tensor, i.e., for any orientation in the x-y plane. Most of such 180°-flipped bonds are not populated in the database, because they result in steric clashes, but nevertheless this ambiguity constitutes the main barrier to building longer segment lengths.

A starting structure built for ubiquitin using search data from two alignment tensors is shown in Figure 4.11. It is easily seen that at this stage errors in the MFR derived backbone angles accumulate when building the initial model because the angular estimates are made independently at each residue, and therefore the long range information contained in the dipolar couplings is not yet used. However, even though the starting structure is physically unrealistic, nevertheless most of its backbone angles are close to their "ideal" values.

Refinement procedure. Because only small adjustments to the backbone angles of the starting structure are required, a special refinement procedure was employed. This refinement is based on a simple iterative gradient approach, which adjusts ϕ/ψ to minimize the χ^2 between the complete set of measured dipolar and chemical shifts and those predicted for the entire model structure at the current iteration.

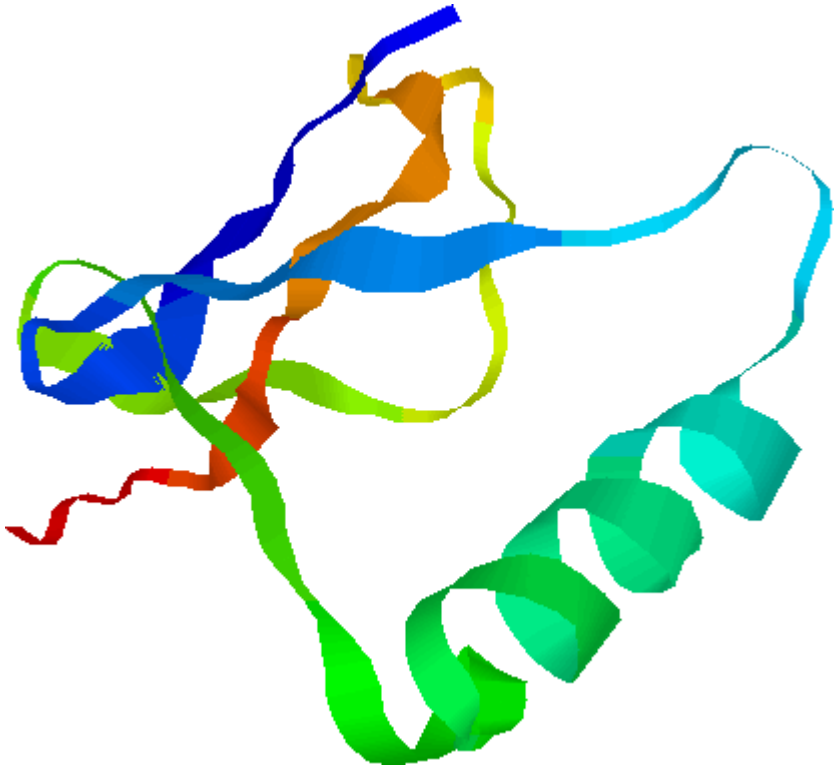


Figure 4.11. Starting structure for ubiquitin, produced from angular distributions of the ensemble of 20x70 best fragments found in the database search. The search was based on chemical shifts and couplings from two alignment tensors. Although this starting structure has an rmsd of 6.95 Å relative to the crystal structure, its backbone angles are close to those in the crystal structure (rmsd of 11˚ and 17˚ for ϕ and ψ).

At each iteration, a residue is selected at random, the gradient of χ^2 with respect to $\phi$and ψ is calculated numerically, and new optimal values for ϕ and ψ are predicted. If the new angles improve the overall agreement between observed and predicted values for the protein, they are accepted and replace the prior angles. In this particular implementation, vanderWaals forces are not considered, allowing unrestricted motion of the backbone during refinement. Figure 4.12 shows how this procedure results in rapid convergence; when the model's fit to the experimental dipolar couplings and chemical shifts improves, so does the fit to the X-ray (and NMR) structure.

Table 4.1 shows that the approximate backbone angles in ubiquitin can be defined unambiguously for two contiguous segments (1-52 and 54-76) when using 67 N-H, 69 C'-N, 69 $C'\text{-}H^N$, and 66 $C^\alpha\text{-}H^\alpha$ dipolar couplings, measured in two liquid crystalline media. The "ambiguous" backbone angles of residue Gly^{53} are fortuitously correct, and building the polypeptide using all MFR angles yields an initial fold with roughly the correct topology, differing by 6.95 Å from the X-ray structure. For shorter segments this rmsd is correspondingly smaller. When using dipolar couplings measured in only a single liquid crystal medium, there is a larger number of residues for which the approximate backbone angles cannot be determined using the MFR approach (Asp^{52}, Gly^{53} and Glu^{64} for the medium with the more rhombic alignment tensor; Pro^{19}, Asp^{52}, Gly^{53}, and Thr^{55} for the less rhombic tensor).

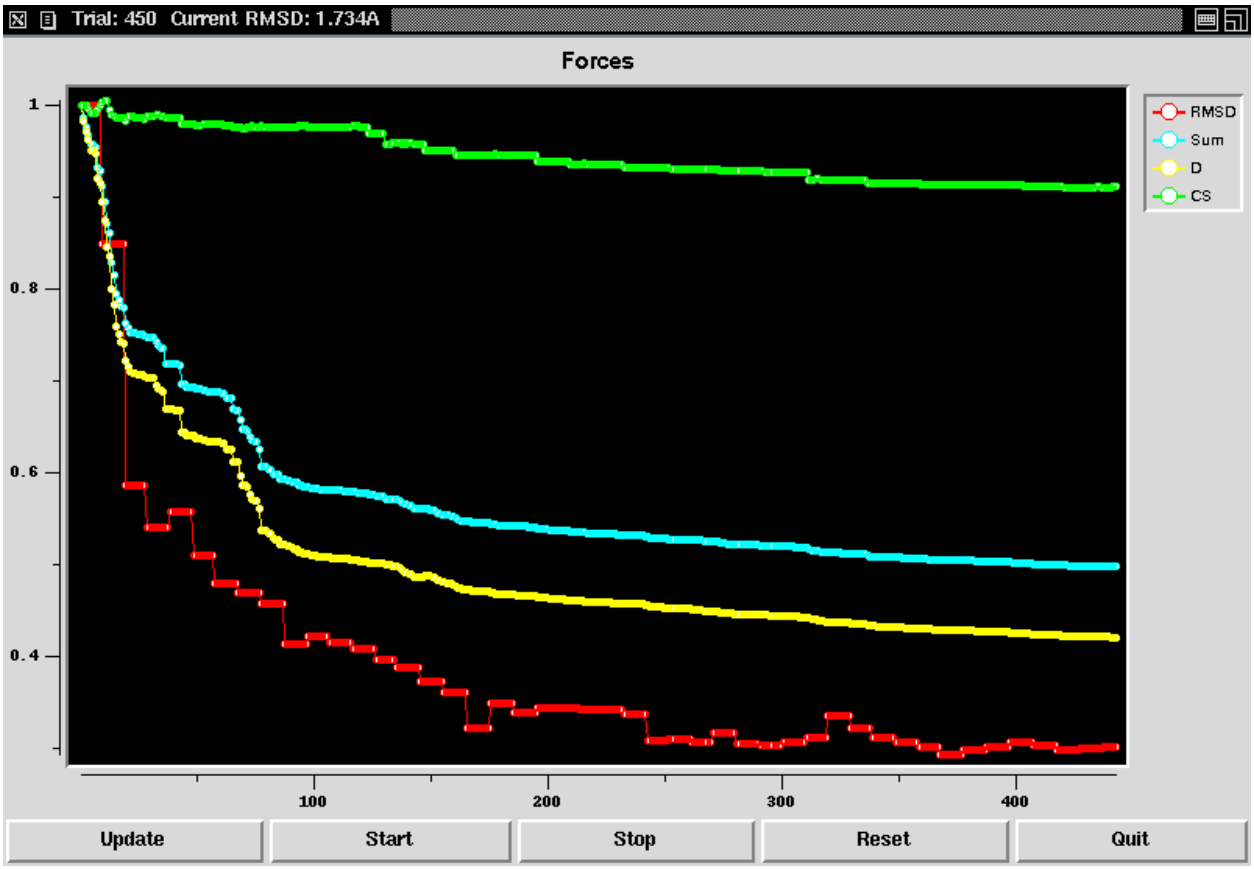


Figure 4.12. Progress of the backbone refinement scheme. χ^2 between predicted and measured chemical shifts (green) and rmsd between best-fitted and experimental dipolar couplings (blue) as is shown a function of iteration number during the minimization protocol, which aims to minimize these values. The chemical shift χ^2 drops from 0.79 to 0.63 during refinement. The dipolar coupling rmsd is calculated after normalizing all couplings relative to N-H, and decreases from 6.52 to 2.48 Hz during refinement. The red line corresponds to the backbone rmsd between the model and the ubiquitin crystal structure and decreases from 6.95 Å for the initial model to 1.21 Å (residues 1-76), shown for reference purposes. Each curve is normalized by its initial value, so that all curves start at 1.0. The entire refinement takes 10 min on a 400 MHz Linux PC.

Generally, each segment, as defined by ranges of residues with unambiguous backbone angle conformations, is refined separately. As mentioned above, in the case of ubiquitin it was possible to refine the entire structure at once, using both alignment tensors simultaneously. The result of this refinement is given in Figure 4.13, which shows a superposition of the ubiquitin X-ray structure. The structure generated with shifts and couplings alone agrees with the crystal structure to a backbone rmsd of 1.21 Å; when excluding the flexible C-terminus this rmsd drops to 0.88 Å.

Concluding Remarks

This work demonstrates that it is possible to calculate the three-dimensional structure of large protein backbone segments, and in favorable cases an entire small protein, exclusively from dipolar couplings and chemical shifts. Dipolar couplings can be rapidly measured once a protein

Table 4.1. Ubiquitin segments calculated from dipolar couplings and chemical shifts.[a]

Segment	Initial model[b] RMSD (Å)[c]	Single Alignment Tensor Refined model RMSD (Å)[c]	Two Alignment Tensors Refined model RMSD (Å)[c]
1-52	5.44	0.87	0.82
53-76	3.79	1.17	0.42
1-76	6.95	1.21	0.88

[a]Using N, C^{α}, H^{α}, C^{β} and C' chemical shifts and the two sets of experimental dipolar couplings measured with charged and uncharged bicelles. The amide of Gly^{53} is conformational-exchange broadened beyond detection. [b]Initial model built from MFR derived backbone angles. [c]rmsd vs X-ray excluding residues 1 and 73-76.

assignment is completed and the approach described here obviates the time-consuming NOE analysis. Furthermore, this approach can also be extended to incorporate other NMR observables, in particular J couplings and sequential NOEs, which are easily obtained, and can be readily simulated from secondary structure. Our protocol can easily integrate long range NOE and hydrogen bond information too, and even a handful of such long-range constraints may be sufficient to correctly define the relative position of oriented fragments relative to one another. Alternatively, packing the fragments using molecular modeling is expected to be relatively rapid and straightforward.

The approach described above is just one of many possible schemes for reconstituting a protein structure from database fragments using dipolar coupling homology, and numerous such variations are being explored. This new approach to structure determination calls for a combination of NMR parameter simulation, database search, visualization, and molecular manipulation and refinement. We found it natural to extend the capabilities of our TCL/TK interpreter NMRWish to include these facilities. We call this system of structure-oriented tools DYNAMO; it was used to implement all of the calculations and interfaces presented in this section. An example application of this system is shown in Figure 4.14. In addition to the homology search and refinement methods presented, DYNAMO also includes many of the traditional facilities for refinement by restrained molecular dynamics (Brünger, 1993), as shown. As we have described in the previous sections, the interpreter also includes complete facilities for manipulating spectral data and NMR observables. Therefore, DYNAMO helps extend the groundwork for a unique integrated system of NMR analysis and structure calculation, all based on a single standard, easy to use command language.

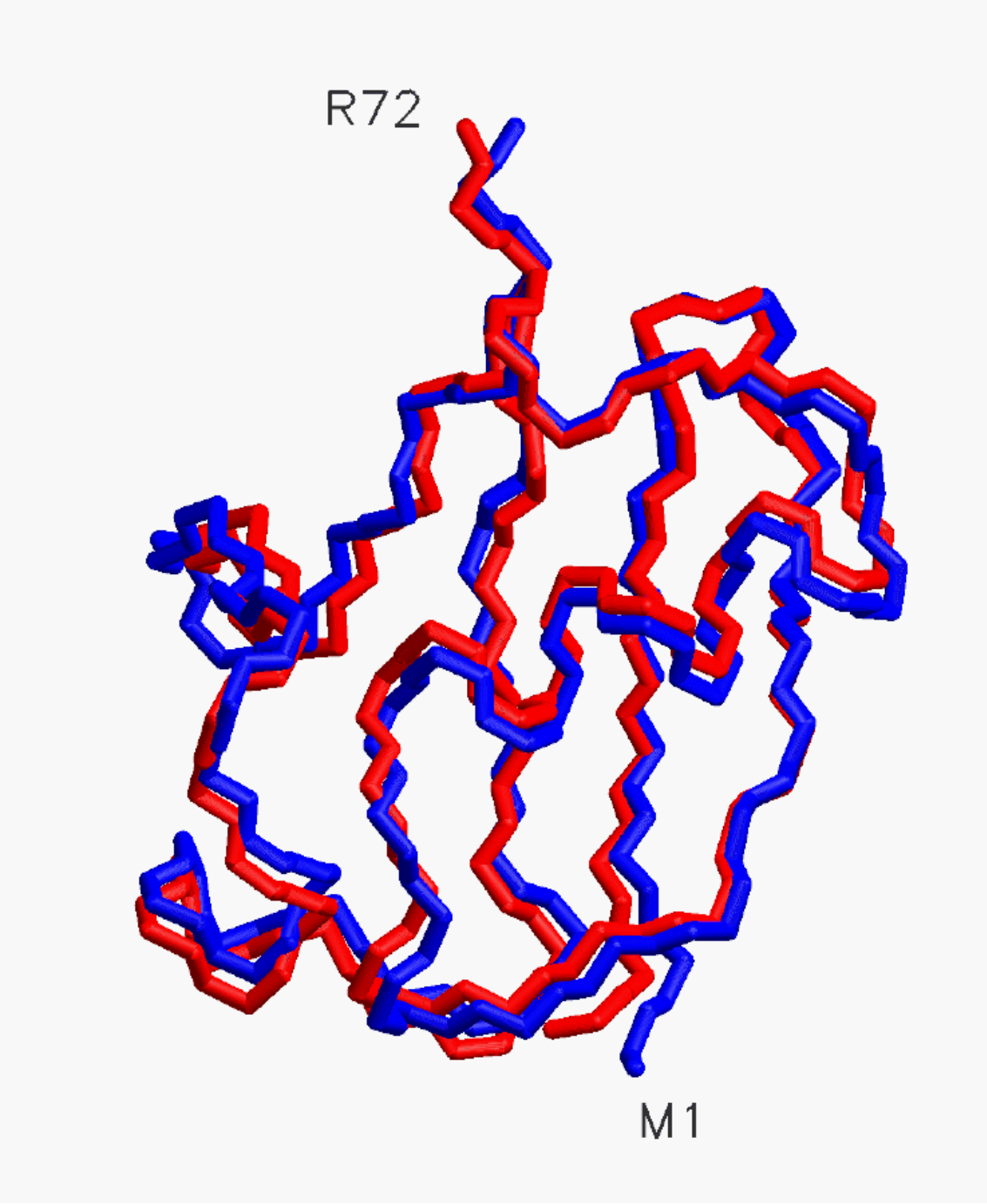


Figure 4.13. Backbone representations of the X-ray crystal structure (blue) of ubiquitin residues 1-72, and the refined model (red) obtained from dipolar couplings in two different liquid crystalline media, together with isotropic chemical shifts. The rmsd is 0.88 Å. Planar, trans peptide bonds, standard covalent geometry, and bond lengths were used to calculate this model, but no vanderWaals radii or other terms.

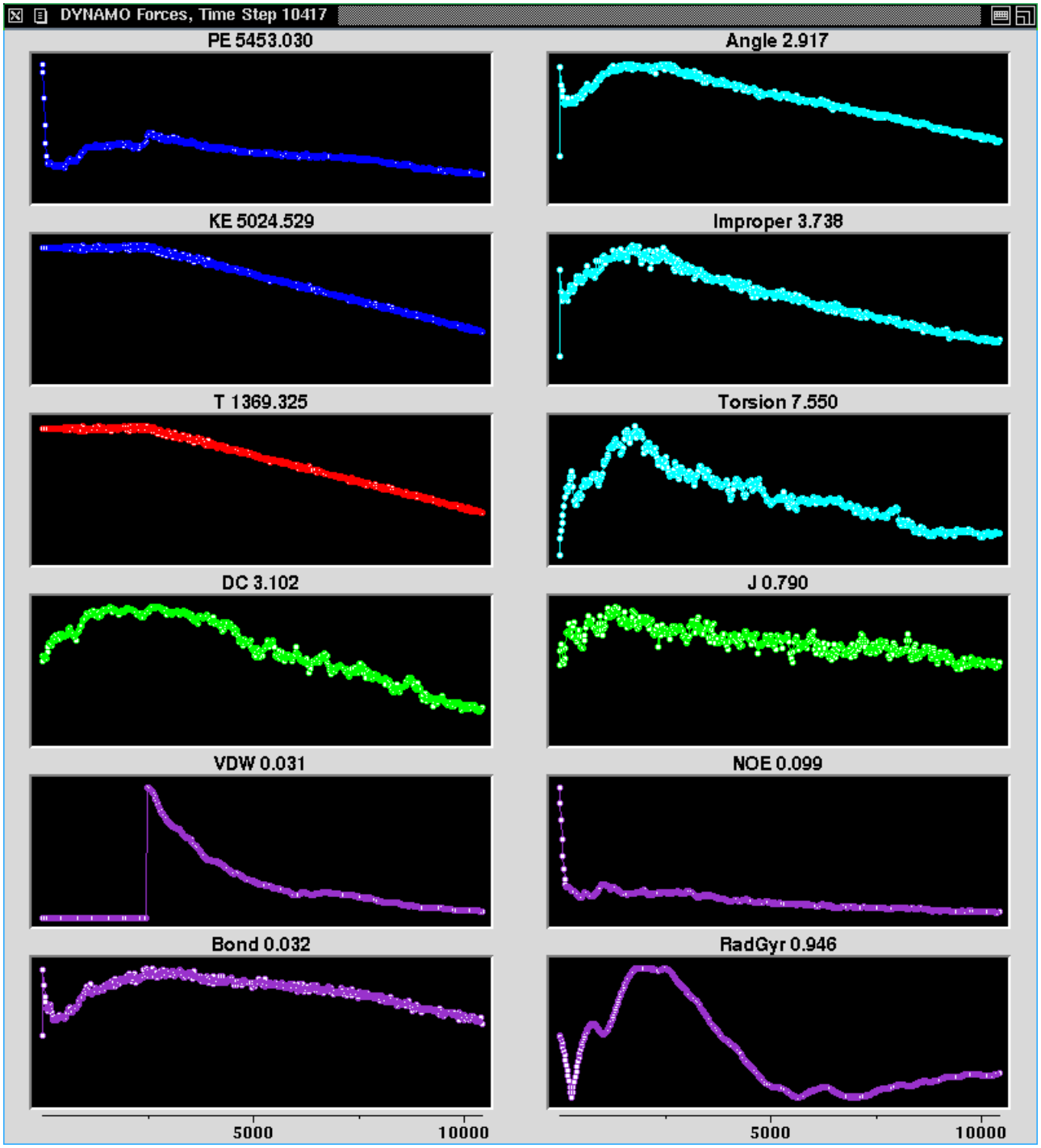


Figure 4.14. The DYNAMO interface for conventional restrained molecular dynamics by simulated annealing in cartesian space (Brünger, 1993), displaying parameters with respect to time step for an annealing schedule on ubiquitin. The graphs in dark blue show the potential energy (PE) and kinetic energy (KE) of the system. The red graph shows the annealing temperature at the given time step. The purple curves show the rmsd in Å for the vanderWaals (VDW), bond length, NOE distance, and radius of gyration (RadGyr) restraints (Kuszewski et al., 1999). The light blue graphs show the rmsd in degrees for the various angular restraints. The green curves show the rmsd in Hz for dipolar coupling (DC) (Tjandra et al, 1997) and J coupling (J) restraints.

Summary and Concluding Remarks

The software methods and tools presented in this work have made substantial progress in answering the needs of multidimensional NMR processing and analysis, and as a result, many aspects of this work are already widely used throughout the world. In particular, the software has kept pace with the evolution of multidimensional NMR from a specialized research technique to a routine method. The software has also served as a platform to implement the latest approaches, and to develop and characterize new ones. As such, the computational tools presented promise to keep pace with further advances in the field, including new strategies for protein structure determination.

In **Section 1**, we introduced the NMRPipe approach for NMR processing and analysis coordinated by UNIX pipelines and scripts. Pipeline based processing schemes have proved to be both more concise and faster than other methods, are also naturally parallel, and so can easily distribute computations over multiple computers effectively. The software also provides for rigorous inverse processing methods, to take the best advantage of enhanced processing techniques such as Linear Prediction and Maximum Entropy Method. A critical component of the NMRPipe system is the customized script interpreter nmrWish, an implementation of the standard TCL/TK graphical scripting language, which we have augmented with over 150 commands for manipulating, analyzing, and visualizing spectral and structural data. This script interpreter serves as our primary means for developing and integrating the various aspects of an NMR analysis task. Use of this script-based approach allows applications to be extensively customized, highly interactive, or completely automated. Numerous examples of such applications have been presented in this work, including ACME, TALOS, and DYNAMO, as well as applications for spectral assignment, automated processing, and NMR-based drug screening, providing excellent examples of the adaptability and usefulness of the NMRPipe approach.

In **Section 2**, we described one particular application of spectral quantification in detail, the long-standing but often still difficult task of extracting biomolecular proton-proton couplings from COSY spectra. The application, called ACME, takes advantage of the little-used fact that antiphase multiplet fitting, which is normally unstable because of counter-compensating effects of amplitude and linewidth, becomes straightforward if the amplitude of the signal can be measured experimentally and held constant. In the method, a given multiplet is identified interactively, the number of couplings associated with the multiplet is specified, and the multiplet is modeled automatically using an amplitude constraint established from the diagonal signals. The method was validated by comparing ACME-derived couplings to those measured independently by heteronuclear methods, indicating that accuracies of *ca.* 0.7 Hz can be achieved for proteins and DNA. This application is a good overall example of the NMRPipe approach, because it requires integration of special processing schemes, visualization, and optimization in one application. Part of the ease of use of ACME is due to small but important details in the various aspects of NMRPipe. For example, the ACME method creates its model cross peak signals by numeric Fourier transform of synthetic time-domain J-modulated signals. In order for this reconstruction to be performed correctly, the details of spectral processing (original time-domain size, window function and zero fill used, etc) must be automatically accessible. Since the processing components of NMRPipe keep rigorous records of acquisition and processing

parameters, this information is readily available to other programs. The overall result is an application that is both accurate and convenient.

In **Section 3**, we described the groundwork of NMR database search methods for structure determination. The initial work is based on the well-known observation that protein chemical shift is correlated with protein secondary structure. This empirical relationship can be exploited quantitatively. In the TALOS approach, we compare the secondary chemical shifts of residue triplets in a database to those in a query protein, under the assumption that if the patterns of chemical shifts are a good match, then the ϕ and ψ of the central residue in the database triplet will be a good predictor for the corresponding value in the query protein. In practice, a (ϕ,ψ) prediction is only made if there is a (ϕ,ψ) consensus in at least 9 out of the 10 best matching triplets in the database. Using a small database derived from 20 proteins with complete NMR assignments ($^{1}H^{\alpha}$, $^{13}C^{\alpha}$, $^{13}C^{\beta}$, $^{13}C'$ and ^{15}N) and high-resolution crystal structures, this approach can provide consensus angular estimates for roughly 65% of the residues in a given protein, to an rms of better than 15^{o}, although of these, roughly 3% are in error.

The contents of the TALOS database can also be used to characterize secondary shift distributions with respect to (ϕ,ψ). In turn, surfaces of these distributions can be used to predict the chemical shifts associated with a given backbone angle and residue type, with typical ppm precisions of: $^{1}H^{\alpha}$ 0.38, $^{13}C^{\alpha}$ 1.23, $^{13}C^{\beta}$ 1.34, $^{13}C'$ 1.45, and ^{15}N 3.95.

In **Section 4**, we introduced a new method for protein structure prediction based on homology of NMR observables, and demonstrated for the first time the possibility of determining a protein backbone fold exclusively from dipolar couplings and chemical shifts. In the method, called Molecular Fragment Replacement (MFR), we compose a protein structure based on a collection of small (*ca*. 7-residue) fragments selected by a database search according to the best matches between observed and predicted dipolar couplings and chemical shifts. In practice, we search through a subset of about 1500 proteins from the Brookhaven Protein Databank (PDB). In the case of ubiquitin, for which dipolar coupling data has been measured at two different alignment tensors, this yields a collection of overlapping fragments whose rmsd ranges from 0.1 to 3.0 Å relative to the crystal structure, with an average of better than 1.0 Å. In the next stage of the scheme, the (ϕ,ψ) values for the collection of fragments are used to derive consensus estimates of the backbone angles in a starting structure, and to identify the endpoints of segments in the structure where the backbone angles cannot be clearly determined. In practice, segment sizes in the range of 10-50 residues can be constructed in this way, although for the ideal case of ubiquitin, where the two residues with ambiguous angular estimates are fortuitously correct, the entire small protein could be treated.

A segment constructed in this way from consensus angles has a starting structure which is spatially different from the ideal one, but very close in terms of its (ϕ,ψ) values. As such, these segments can be refined by a series of small adjustments to the backbone angles, to recover the long-range agreement between the observed and predicted dipolar couplings. In the case of ubiquitin, this remarkably resulted in a backbone structure with an rmsd of 1.2 Å relative to the X-ray structure.

Future Directions. Our initial results show that the NMR homology search approach to structure calculation is a promising and exciting one. They also suggest particular areas of development to make the MFR method robust and routine. There are two principal challenges.

The first challenge concerns the degree of degeneracy between dipolar couplings and structure. As we have shown, while most fragments with a good match between observed and predicted dipolar couplings also have similar backbone conformations, there are also small but significant numbers of fragments with good dipolar coupling homology but very different structures. So, methods for discriminating against these "bad" fragments will be critical. As mentioned, one possible approach is to include other NMR parameters, such as J couplings (which must be measured in any case to determine dipolar couplings) and sequential NOEs. Both of these parameters are easily measured. Also, since they are readily estimated on the basis of backbone structure, they can be incorporated into the existing search scheme with no difficultly.

The second challenge concerns the methods used to assemble the fragments into a refined protein structure. The simple consensus angle approach presented here has the drawback of using a physically unrealistic starting structure, thus losing contact with the information in the original collection of fragments. This suggests that new structure refinement methods which use the fragments directly should be employed, for instance by fitting a structure to a bundle of fragments, or building a structure by rotating fragments into orientations that are mutually consistent with respect to the dipolar couplings, and then translating them into position within the overall protein chain. This work is a primary motivation for our development of the new structure calculation software DYNAMO, and will be a primary focus of our future efforts.

References

1. Ando, I., Saito, H., Tabeta, R., Shoji, A. and Ozaki, T. (1984) *Macromolecules*, **17**, 457-461.

2. Annila, A., Aitio, H., Thulin, E. and Drakenberg, T. (1999) *J. Biomol. NMR*, ***14***, 223-230.

3. Archer, S.J., Vinson, V.K., Pollard T.D. and Torchia, D.A. (1994) *FEBS Lett.*, **337**, 145-151.

4. Barkhuijsen, H., De Beer, R., Bovée, W.M.M.J. and Van Ormondt, D. (1985) *J. Magn. Reson.*, **61**, 465-481.

5. Barkhuijsen, H., De Beer, R. and Van Ormondt, D. (1987) *J. Magn. Reson.*, **73**, 553-557.

6. Bax, A. and Grzesiek, S. (1993) *Acc. Chem. Res.*, **26**, 131-138.

7. Bax, A. and Tjandra, N. (1997) *J. Biomol. NMR,* **10**, 289-292.

8. Bax, A., Ikura, M., Kay, L.E., Barbato, G. and Spera, S. (1991) *Protein Conformation* Wiley, New York (Ciba Foundation Symposium 161) 108-135.

9. Bax, A., Kontaxis, G. and Tjandra, N. (2001, in press) *Methods Enzymol.*

10. Bax, A. and Tjandra, N. (1997) *J. Biomol. NMR,* **10**, 289-292.

11. Bax, A., Vuister, G.W., Grzesiek, S., Delaglio, F., Wang, A.C., Tschudin, R. and Zhu, G. (1994). *Nuclear Magnetic Resonance, Pt C.,* **239**, 79-105.

12. Beger, D.B. and Bolton, P.H. (1997) *J. Biomol. NMR*, **10**, 129-142.

13. Bergmann, E.M., Cherney, M.M., Mckendrick, J., Vederas, J.C. and James, M.N.G. (1999) *Virology,* **265**, 153.

14. Berndt, K.D., Guntert, P., Orbons, L.P. and Wüthrich, K. (1992) *J. Mol. Biol.*, **227**, 757-775.

15. Betzel, C., Klupsch, S., Papendorf, G., Hastrup S., Branner, S. and Wilson, K.S. (1992) *J. Mol. Biol.*, **223**, 427-445.

16. Bewley, C.A., Gustafson, K.R., Boyd, M.R., Covell, D.G., Bax, A., Clore, G.M. and Gronenborn, A.M. (1998) *Nature, Struct. Biol.,* **5**, 571-578.

17. Biamonti, C., Rios, C.B., Lyons, B.A. and Montelione, G.T. (1994) *Advances in Biophysical Chemistry,* **4**, 51-120.

18. Braun, D., Wider, G. and Wüthrich, K. (1994) *J. Am. Chem. Soc*, **116**, 8466-8469.

19. Brünger, A.T. (1993) *XPLOR Manual Version 3.1*, Yale University, New Haven, CT.

20. Bystrov, V.F. (1976) *Progress in NMR Spectroscopy,* **10**, 41-81.

21. Callaghan, P.T., MacKay, A.L., Pauls, K.P., Soderman, O. and Bloom, M. (1984) *J. Magn. Reson*., **56**, 101-109.

22. Cavanagh, J., Palmer, A.G., Wright, P.E. and Rance, M. (1991) *J. Magn. Reson.*, **91**, 429-436.

23. Celda, B., Biamonti, C., Arnau, M.J., Tejero, R. and Montelione, G.T. (1995) *J. Biomol. NMR*, **5**, 161-172.

24. Chattopadhyaya, R., Meador, W.E., Means, A.R. and Quiocho, F.A. (1992) *J. Mol. Biol.*, **228**, 1177-1192.

25. Clore, G. M. and Garrett, D. S. (1999) *J. Am. Chem. Soc.*, **121**, 9008-9012.

26. Clore, G. M., Starich, M. R. and Gronenborn, A. M. (1998) *J. Am. Chem. Soc.*, **120**, 10571-10572.

27. Clore, G.M. and Gronenborn, A.M. (1991) *Progr. NMR Spectrosc*., **23**, 43-92.

28. Clore, G.M., Bax, A., Driscoll, P.C., Wingfield, P. and Gronenborn, A. (1990) *Biochemistry*, **29**, 8172-8184.

29. Clore, G.M., Starich, M.R. and Gronenborn, A.M. (1998) *J. Am. Chem. Soc.,* **120**, 10571-10572.

30. Clore, G.M., Starich, M.R., Bewley, C.A., Cai, M.L. and Kuszewski, J. (1999) J. *J. Am. Chem. Soc.*, **121**, 6513-6514.

31. Concha, N.O., Rasmussen, B.A., Bush, K. and Herzberg, O. (1996) *Structure,* **4**, 823-836.

32. Copie, V., Battles, J.A., Schwab, J.M. and Torchia, D.A. (1996) *J. Biomol. NMR*, **7**, 335-340.

33. Cornilescu, G., Delaglio, F. and Bax, A.(1999) *J. Biomol. NMR,* **13**, 289-302.

34. Cornilescu, G., Marquardt, J. L., Ottiger, M. and Bax, A. (1998) *J. Am. Chem. Soc.*, **120**, 6836-6837. PDB entry 1D3Z.

35. Davis, J.H., Agard, D.A., Handel, T.M. and Basus, V.J. (1997) *J. Biomol. NMR*, **10**, 21-27.

36. de Dios, A.C. and Oldfield, E. (1993) *J. Am. Chem. Soc.*, **116**, 5307-5314.

37. de Dios, A.C., Pearson, J.G. and Oldfield, E. (1993) *Science*, **260**, 1491-1495.

38. Delaglio, F., Grzesiek, S., Vuister, G., Zhu, G., Pfeifer, J. and Bax, A. (1995) *J. Biomol. NMR*, **6**, 277-293.

39. Delaglio, F., Kontaxis, G. and Bax, A.(2000) *J. Am. Chem. Soc.*, **122**, 2142-2143.

40. Delsuc, M.A. (1989) *Maximum Entropy and Bayesian Methods*, Kluwer Academic, Amsterdam.

41. Delsuc, M.A., Ni, F. and Levy, G.C. (1987) *J. Magn. Reson.*, **73**, 548-552.

42. DiStefano, D.L. and Wand, A.J. (1987) *Biochemistry*, **26**, 7272-7281.

43. Weber, P.L., Brown, S.C. and Mueller, L. (1987) *Biochemistry*, **26**, 7282-7290.

44. Drakenberg, T., Hofman, T. and Chazin, W.J. (1989) *Biochemistry*, **28**, 5946-5954.

45. Drohat, A.C., Tjandra, N., Baldisseri, D.M. and Weber, D.J. (1999) *Prot. Science*, **8**, 800-809.

46. Edison, A.S., Abildgaard, F., Westler, W.M., Mooberry, E.S. and Markley, J.L. (1994) *Meth. Enzymol.*, **239**, 3-79

47. Fedorov, A.A., Magnus, K.A., Graupe, M.H., Lattman, E.E., Pollard, T.D. and Almo, S.C. (1994) *Proc. Natl. Acad. Sci. U.S.A.*, **30**, 8636-8640.

48. Fesik, S.W. and Zuiderweg, E.R. (1988) *J. Magn. Reson*., **78**, 588-593.

49. Fleming, K., Gray, D., Prasannan, S. and Matthews, S. (2000) *J. Am. Chem. Soc.*, **122**, 5224-5225.

50. Fogh, R.H., Schipper, D., Boelens, R. and Kaptein R. (1995) *J. Biomol. NMR*, **5**, 259-270.

*51. Friedrichs, M.S. (1995) J. Biomol. NMR., **5**, 147-153.*

52. Fujinaga, M., Delbaere, L.T.J., Brayer, G.D. and James, M.N.G. (1985) *J. Mol. Biol*., **184**, 479-502.

53. Gardner, K.H., Zhang, X., Gehring, K. and Kay, L.E. (1998) *J. Am. Chem. Soc*, **120**, 11738-11748.

54. Garrett, D.S., Powers, R., Gronenborn, A.M. and Clore G.M. (1991) *J. Magn. Reson.*, **94**, 214-220.

55. Gronenborn, A.M. and Clore, G.M. (1994) *J. Biomol. NMR*, **4**, 455-458.

56. Gronenborn, A.M. and Clore, G.M. (1995) *Crit. Rev. Biochem. Mol.*, **30**, 351-385.

57. Grzesiek, S. and Bax, A. (1992A) *J. Magn. Reson*., **96**, 432-440.

58. Grzesiek, S. and Bax, A. (1992B) *J. Am. Chem. Soc.*, **114**, 6291-6293.

59. Grzesiek, S. and Bax, A. (1992C) *J. Magn. Reson*., **99**, 201-207.

60. Gull, S.F. and Daniell, G.J. (1978) *Nature*, **272**, 686-690.

61. Güntert, P., Doetsch, V., Wider, G. and Wüthrich, K. (1992) *J. Biomol. NMR*, **2**, 619-629.

62. Hansen, M.R., Mueller, L. and Pardi, A. (1998A) *Nat. Struct. Biol.,* **5**, 1065-1074.

63. Hansen, M.R., Rance, M. and Pardi, A. (1998B) *J. Am. Chem. Soc.,* **120**, 11210-11211.

64. Hansen, P.E. (1991) *Biochemistry*, **30**, 10457-10466.

65. Heller, D. and Van Raalte T. (1993) *XView Programming Manual*, O'Reilly & Assoc., Inc.

66. Hoch, J.C. (1985) Rowland Institute for Science Technical Memorandum RIS-18t.

67. Hoch, J.C. (1989) *Methods Enzymol.*, **176**, 216-241.

68. Hoch, J.C., Stern, A.S., Donoho, D.L. and Johnstone, I.M (1990) *J. Magn. Reson.*, **86**, 236-246.

*69. Hore, P.J. (1985) J. Magn. Reson., **62**, 561-567.*

70. Hu, J.-S. and Bax, A. (1997) *J. Am. Chem. Soc*., **119**, 6360-6368.

71. Hus, J. C., Marion, D. and Blackledge, M. (2000) *J. Mol. Biol.*, **298**, 927-936.

72. Ikura, M., Kay, L.E. and Bax, A. (1990) *Biochemistry*, **29**, 4659-4667.

73. Ikura, M., Kay, L.E., Krinks, M. and Bax, A. (1991) *Biochemistry*, **30**, 5498-5504.

74. Johnson, B. and Blevins, R.A. (1994) *J. Biomol. NMR.*, **4**, 603-614.

75. Johnson, P.E., Tomme, P., Joshi, M.D., and McIntosh, L.P. (1996) *Biochemistry*, **35**, 13895-13906.

76. Johnson, S. (1986) in *UNIX Programmer's Manual: Supplementary Documents 1*, University of California, Berkeley.

77. Jones, T.A. and Thirup, S. (1986) *EMBO J.*, **5**, 819-822.

78. Karplus, M. (1959) J. Phys. Chem., **30**, 11-15.

79. Kauppinen, J. and Saario, E. K. (1993) *Appl. Spectrosc.*, **47**, 1123-1127.

80. Kay, L.E, Keifer, P. and Saarinen, T. (1992) *J. Am. Chem. Soc.*, **114**, 10663-10666.

81. Kay, L.E., Ikura, M., Zhu, G. and Bax, A. (1991) *J. Magn. Reson.*, **91**, 422-428.

82. Kay, L.E., Marion, D. and Bax, A (1989) *J. Magn. Reson.*, **84**, 72-84.

83. Ke, H.M., Zydowsky, L.D., Liu, J. and Walsh, C.T. (1991) *Proc. Nat. Acad. Sci. USA* **88**, 9483-9487.

84. Kernighan, B.W. and Pike, R. (1984) *The UNIX Programming Environment*, Prentice-Hall, Englewood Cliffs NJ.

85. Kernighan, B.W. and Ritchie, D.M. (1988) *The C Programming Language*, Prentice-Hall, Englewood Cliffs NJ.

86. King, H.C., Wang, K.Y., Goljer, I. and Bolton, P.H. (1995) *J. Magn. Reson. Ser. B*, **109**, 323-325.

87. Kjaer, M., Ansersen, K.V. and Poulsen, F.M. (1994) *Methods Enymol.*, **239**, 288-307.

88. Kraulis, P.J., (1989) J. Magn. Reson., ***84****, 627-633.*

89. Kraulis, P.J., Domaille, P.J., Campbell-Burk S.L., Van Aken, T. and Laue, E.D. (1994) *Biochemistry*, **33**, 3515-3531.

90. Kraulis, P.J. and Jones, T.A. (1987) *Prot. Struct. Func. Genet.*, **2**, 188-201.

91. Kricheldorf, H.R. and Muller, D. (1983) *Macromolecules*, **16**, 615-623.

92. Kumar, V. and Kannan, K.K. (1994) *J. Mol. Biol*., **241**, 226-232.

93. Kumaresan, R. and Tufts, D.W. (1982) *IEEE Trans.*, **ASSP-30**, 833-840.

94. Kung, H.C., Goljer, I,. and Bolton, P.H. (1995) *J. Magn. Res. Ser. B*, **109**, 323-325.

95. Kuntz, I.D., Kosen, P.A. and Craig, E.C. (1991) *J. Am. Chem. Soc.*, **113**, 1406-1408.

96. Kuszewski, J., Gronenborn A.M. and Clore, G.M. (1997) *J. Magn. Reson.,* **125**, 171-177.

97. Kuszewski, J., Gronenborn A.M. and Clore, G.M. (1999) *J. Am. Chem. Soc J.,* **121**, 2337-2338.

98. Kuszewski, J., Qin, J., Gronenborn A.M. and Clore, G.M. (1995) *J. Magn. Reson. Ser. B*, **106**, 92-96.

99. Lam, P.Y.S., Jadhav, P.K., Eyerman, C.J., Hodge, C.N., Ru, Y., Bacheler, L.T., Meek, J.L., Otto, M.J., Rayner, M.M., Wong, Y.N., Chang, C.-H., Weber, P.C., Jackson, D.A., Sharpe, T.R. and Erickson-Viitanen, S. (1994) *Science*, **263**, 380-384.

100. Laue, E.D., Mayger, M.R., Skilling, J. and Staunton, J. (1986) *J. Magn. Reson.*, **68**, 14-29.

101. Laue, E.D., Skilling, J. and Staunton, J. (1985) *J. Magn. Reson.*, **63**, 418-424.

102. Laue, E.D., Skilling, J., Staunton, J., Sibisi, S. and Brereton, R. (1985) *J. Magn Reson.*, **62**, 437-452.

103. Leesong, M., Henderson, B.S., Gillig, J.R., Schwab, J.M. and Smith, J.L. (1996) *Structure*, **4**, 253-256.

104. Levy, G.C., Delaglio, F., Macur, A. and Begemann, J. (1986) *Comput. Enhanced Spectrosc.*, **3**, 1-12.

105. Lewis, B.A., Rosenblatt, C, Griffin, R.G., Courtemanche, J., and Herzfeld, J. (1985) *Biophys. J.,* **47**, 143-150.

106. Loll, P.J. and Lattman, E.E. (1989) *Proteins. Struct., Funct.*, **5**, 183-201.

107. Longhi, S., Czjzek, M., Lamzin, V., Nicolas, A. and Cambillau, C. (1997) *J. Mol. Biol.*, **268**, 779-799.

108. Losonczi, J.A., Andrec, M., M. Fischer, W.F. and Prestegard, J.H. (1999) *J. Magn. Reson.,* **138**, 334-342.

109. Luginbühl P., Szyperski T. and Wüthrich, K. (1995) *J. Magn. Reson.*, **109**, 229-233.

110. Marion D., Ikura, M. and Bax, A. (1989), *J. Magn. Reson.,* **84**, 425-430.

111. Marion, D. and Bax, A. (1988) *J. Magn. Reson*., **80**, 528-533.

112. Marion, D. and Wüthrich, K. (1983) *Biochem. Biophys. Res. Commun.*, **113**, 967-974.

113. Marion, D., Ikura, M., Tschudin R and Bax, A. (1989) *J. Magn. Reson*., **85**, 393-399.

114. Marion, D., Kay, L.E., Sparks, S.W., Torchia, D.A. and Bax, A. (1989) *J. Am. Chem. Soc.*, **111**, 1515.

115. Markley, J.L., Bax, A., Arata, Y., Hilbers, C.W., Kaptein, R., Sykes, B.D., Wright, P.E., Wüthrich, K. (1998) *J. Biomol. NMR,* **12**, 1-23.

116. Mazzeo, A.R., Delsuc, M.A., Kumar, A. and Levy, G.C., (1989) *J. Magn. Reson.*, **81**, 512-519.

117. Meador, W.E., Means, A.R. and Quiocho, F.A. (1992) *Science*, **257**, 1251-1255.

118. Meadows, R.P., Olejniczak, E.T. and Fesik, S.W. (1994) *J. Biomol. NMR*, **4**, 79-96.

119. Mueller, L. (1987) J. Magn. Reson., 72, 191.

120. Ni, F. and Scheraga, H.A. (1986) *J. Magn. Reson.*, **70**, 506-511.

121. Ni, F., Levy G.C. and Scheraga, H.A. (1986) *J. Magn. Reson.*, **66**, 385-390.

122. Nilges, M., Gronenborn, A.M., Brünger, A.T. and Clore, G.M. (1988) *Protein Engineering* **2**, 27-38.

123. Oesterhelt, D. and Stoeckenius, W. (1974) *Meth. Enzymol.,* **31**, 667-678.

124. Oh, B.H., Westler, W.M., Derba, P. and Markley, J.L. (1988) *Science*, **240**, 908-911.

125. Olejniczak, E.T. and Eaton, H.L. (1990) *J. Magn. Reson.*, **87**, 628-632.

126. Ösapay K. and Case, D.A. (1994) *J. Biomol. NMR*, **4**, 215-230.

127. Ottiger, M., Zerbe, O., Güntert, P. and Wüthrich, K. (1997) *J. Mol. Biol.*, **272**, 64-81.

128. Ottiger, M. and Bax, A. (1998) *J. Am. Chem. Soc.,* **120***,* 12334-12341.

129. Ottiger, M., Tjandra, N. and Bax, A. (1997) *J. Am. Chem. Soc.*, **119**, 9825-9830.

130. Ousterhout, J.K., (1994) *Tcl and the Tk Toolkit*, Addison-Wesley, Reading MA.

131. Palmer III, A.G., Fairbrother, W. J., Cavanagh, J., Wright, P.E. and Rance, M. (1992) *J. Biomol. NMR*, **2**, 103-108.

132. Palmer, A.G., Cavanagh, J., Wright, P.E. and Rance, M. (1991) *J. Magn. Reson.*, **93**, 151-170.

133. Pardi, A., Wagner, G. and Wüthrich K. (1983) *Eur. J. Biochem.*, **137**, 445-454.

134. Parks, S.I. and Johannesen, R.B. (1976) *J. Magn. Reson.*, **22**, 265-267.

135. Pastore A. and Saudek V. (1990) *J. Magn. Reson.*, **90**, 165-176.

136. Pearson J.G., Wang J., Markley J.L., Le H. and Oldfield, E. (1995) *J. Am. Chem. Soc.*, **117**, 8823-8829.

137. Pelczer, I. and Szalma, S. (1991) *Chemical Reviews*, **91**, 1507-1524.

138. Pelton, J.G., Torchia, D.A., Meadow, N.D., Wong, C. and Roseman, S. (1991) *Biochemistry*, **30**, 10043-10057.

139. Prompers, J.J., Groenewegen, A., van Schaik, R.C., Pepermans, H.A.M. and Hilbers, C.W. (1997) *Protein Sci.*, **6**, 2375-2384.

140. Qin, J., Clore, G.C. and Gronenborn, A.M. (1996) *Biochemistry*, **35**, 7-13.

141. Ramirez B.E., Voloshin O.N., Camerini-Otero R.D. and Bax A. (2000) *Protein Science*, **11**, 2161-2169.

142. Ramirez, B.E. and Bax, A. (1998) *J. Am. Chem. Soc*., **120**, 9106-9107.

143. Redfield, A.G. and Kunz, S.D. (1975) *J. Magn. Reson.*, **19**, 250-254.

144. Redfield, C. and Robertson, J. (1991) Proceedings of a NATO Advanced Research Workshop on Computational Aspects of the Study of Biological Macromolecules By NMR, Plenum Press, New York NY.

145. Ross, A., Schlotterbeck, G., Klaus, W. and Senn, H. (2000) *J. Biomol. NMR*., **16**, 139-146.

146. Saito, H. (1986) *Magn. Reson. Chem,* **24**, 835-852.

147. Sanders, C.R. and Prestegard, J.H. (1990) *Biophys. J.,* **58**, 447-460.

148. Sanders, C.R. and Prestegard, J.H. (1991) *J. Am. Chem. Soc.,* **113**, 1987-1996.

149. Sanders, C.R. and Schwonek, J.P. (1992) *Biochemistry,* **31**, 8898-8905.

150. Sayle, R. and Milner-White E.J. (1995) *Trends in Biochemical Sciences*, **20**, 374.

151. Schmieder, P., Stern, A.S., Wagner, G. and Hoch, J.C. (1994) *J. Biomol. NMR*, **4**, 483-490.

152. Schneider, D.M., Dellwo, M.J. and Wand, A.J. (1992) *Biochemistry*, **31**, 3645-3652.

153. Scrofani, S.D.B., Wright, P.E. and Dyson, J.H. (1998) *J. Biomol. NMR*, **12**, 201-202.

154. Seavey, B.R., Farr, E.A., Westler, W.M. and Markley, L. (1991) *J. Biomol. NMR*, **1**, 217-236.

155. Sethson, I., Edlund, U., Holak, T.A., Ross, A. and Johnson, B-H. (1996) *J. Biomol. NMR*, **8**, 417-428.

156. Sharff, A.J., Rodseth, L.E. and Quiocho, F.A. (1993) *Biochemistry,* **32**, 10553-10559.

157. Shuker, S.B, Hajduk, P.J., Meadows, R.P. and Fesik, S.W. (1996) *Science*, **274**, 1531-1534.

158. Sibisi, S. (1983) *Nature*, **301**, 134-136.

159. Sitkoff, D. and Case D.A.(1998) *Prog. Nucl. Magn. Reson. Spectrosc*., **32**, 165-190.

160. Skilling, J. and Bryan, R.K. (1984) *Mon. Not. R. Astr. Soc.*, **211**, 111-124.

161. Spera S. and Bax A. (1991) *J. Am. Chem. Soc.*, **113**, 5491-5492.

162. States, D.J., Haberkorn, R.A. and Ruben, D.J. (1982) *J. Magn. Reson.*, **48**, 286-292.

163. Stephenson, M. (1988) *Prog. NMR. Spectrosc.*, **20**, 515-626.

164. Stevens, W.R. (1992) *Advanced Programming in the UNIX Environment*, Addison-Wesley Pub. Co., Reading MA, 428-434.

165. Svensson, L.A., Thulin, E. and Forsen, S. (1992) *J. Mol. Biol.*, **223**, 601-606.

166. Tjandra, N. and Bax, A.(1997) *Science*, **278**, 1111-1114.

167. Tjandra, N., Grzesiek, S. and Bax, A. (1996) *J. Am. Chem. Soc.,* **118**, 6264-6272.

168. Tjandra, N. and Bax, A. (1997) *Science*, **278**, 1111-1114.

169. Tjandra, N., Feller, S.E., Pastor, R.W. and Bax, A. (1995) *J. Am. Chem. Soc.*, **117**, 12562-12566.

170. Tjandra, N., Grzesiek, S. and Bax, A. (1996) *J. Am. Chem. Soc.*, **118**, 6264-6272.

171. Tjandra, N., Omichinski, J. G., Gronenborn, A.M., Clore, G.M. and Bax, (1997) A. *Nature Struct. Biol.*, **4**, 732-738.

172. Tolman, J.R., Flanagan, J.M., Kennedy, M.A. and Prestegard, J.H. (1995) *Proc. Natl. Acad. Sci. U. S. A.,* **92**, 9279-9283.

173. Tycko, R., Blanco, F.J. and Ishii, Y. (2000) *J. Am. Chem. Soc.,* **122**, 9340-9341.

174. Veerapandian, B., Gilliland, G.L., Raag, R., Svensson, L.A., Masui, Y. and Hirai, Y., Poulos, T.L. (1992) *Proteins. Struct., Funct.*, **12**, 10-23.

175. Vijay-Kumar, S., Bugg, C.E. and Cook, W.J. (1987) *J. Mol. Biol.*, **194**, 531-544.

176. Vuister, G.W., Delaglio, F. and Bax, A. (1992) *J. Am. Chem. Soc.*, **114**, 9674-9675.

177. Vuister, G.W., Delaglio, F. and Bax, A. (1993) *J. Biomol. NMR,* **3**, 67-80.

178. Vuister, G.W., Tessari, M., Karimi-Nejad,Y. and Whitehead, B. Eds. (1999). Pulse sequences for measuring coupling constants. *Biological Magnetic Resonance*. Dordrecht, The Netherlands, Kluwer.

179. Wang, A.C., Grzesiek, S., Tschudin, R., Lodi, P.J. and Bax, A. (1995) *J. Biomol. NMR*, **5**, 376-382.

180. Wang, Y.-X., Marquardt, J.L., Wingfield, P., Stahl, S.J., Lee-Huang, S., Torchia, D.A. and Bax, A. (1998) *J. Am. Chem. Soc.,* **120**, 7385-7386.

181. Weichsel, A., Gasdaska, J.R., Powis, G. and Montfort, W.R. (1996) *Structure*, **15**, 735-751.

182. Williamson, M. (1990) *Biopolymers*, **29**, 1423-1431.

183. Wishart, D.S , Watson, M.S., Boyko, R.F. and Sykes, B.D. (1997) *J. Biomol. NMR,* **10**, 329-336.

184. Wishart, D.S. and Sykes, B.D. (1994) *J. Biomol. NMR*, **4**, 171-180.

185. Wishart, D.S., Colin, G.B., Holm, A., Hodges, R.S. and Sykes, B.D. (1995A) *J. Biomol. NMR*, **5**, 67-81.

186. Wishart, D.S., Colin, G.B., Yao, J., Abildgaard, F., Dyson, H.J., Oldfield, E., Markley, J.L. and Sykes, B.D. (1995B) *J. Biomol. NMR*, **6**, 135-140.

187. Wishart, D.S., Sykes, B.D. and Richards, F. M. (1991) *J. Mol. Biol.*, **222**, 311-333.

188. Wlodawer, A., Walter, J., Huber, R. and Sjolin, L. (1984) *J. Mol. Biol.*, **198**, 469-480.

189. Worthylake, D., Meadow, N.D., Roseman, S., Liao, D.-I., Herzberg, O. and Remington, S.J. (1991) *Proc. Nat. Acad. Sci. USA*, **88**, 10382-10386.

190. Wu, N.L. (1984) *Astron. Astrophys.*, **139**, 555-557.

191. Wüthrich, K. (1986) *NMR of proteins and nucleic acids*, Wiley, New York.

192. Yamazaki, T., Hinck, A.P., Wang, Y.-X., Nicholson, L.K., Torchia, D.A., Wingfield, P.T., Stahl, S.J., Kaufman, J.D., Chang, C.-H., Domaille, P.J. and Lam, P.Y.S. (1996) *Protein Science*, **5**, 495-506.

193. Yang, F., Bewley, C. A., Louis, J. M., Gustafson, K. R., Boyd, M. R., Gronenborn, A. M., Clore, G. M. and Wlodawer (1999), A *J .Mol. Biol.,* **288**, 403-412.

194. Zhou M., Harlan J.E., Wade, W.S., Crosby, S., Ravichandran, K.S., Burakoof, S.J. and Fesik, S,W. (1996) *J. Biol. Chem*., **271**, 31119-31123.

195. Zhu, G. and Bax, A. (1990) *J. Magn. Reson.*, **90**, 405-410.

196. Zhu, G. and Bax, A. (1992A) *J. Magn. Reson.*, **98**, 192-199.

197. Zhu, G. and Bax, A. (1992B) *J. Magn. Reson.*, **100**, 202-207.

Appendix 1 - Generic Arguments of NMRPipe

Generic Arguments: the following is a list of arguments used by more than one program or function in the examples and Figures.

-di deletes imaginary data from the current dimension after the given processing function is performed.

-hdr extracts parameters recorded during previous processing from spectral header rather than the command line.

-in specifies the input file or file template (see "Input and Output Templates" below).

-inPlace permits in-place processing (replacement of the input data by the output result).

-inv activates the inverse mode of a given function; function PS will apply inverse (negative) phase correction; function FT will perform an inverse Fourier transform; function ZF will undo any previous zero-filling; function SP will apply the inverse window function and first point scaling.

-out specifies the output file or file template (see "Input and Output Templates" below).

-ov permits over-writing of any preexisting files.

-sw updates the sweep width and other PPM calibration information to accommodate an extraction or shift function.

-verb performs processing in verbose mode, with status messages.

Appendix 2 - Selected Processing Functions of NMRPipe

Processing Functions: the following is an alphabetical list of the **nmrPipe** processing functions used in the examples and figures. The functions and arguments described are not complete lists, but rather only those which are used in the examples.

EXT extracts a region from the current dimension with limits specified by the arguments **-x1** and **-xn**; the limits can be labeled in points, percent, Hz, or PPM. Alternatively, the left or right half of the data can be extracted with the arguments **-left** and **-right**.

FT applies a real or complex forward or inverse Fourier transform, with sign alternation or complex conjugation, as indicated by spectral parameters or command-line arguments.
HT performs a Hilbert transform to reconstruct imaginary data, choosing between ordinary and mirror-image mode if the argument **-auto** is used.

LP extends the data to twice its original size by default, using a complex prediction polynomial whose order is specified by argument **-ord**. Mixed forward-backward LP is performed if the **-fb** argument is used. Mirror-image LP for data with no acquisition delay is performed if the argument **-ps0-0** is used; mirror-image LP for data with a half-dwell acquisition delay is performed if the argument **-ps90-180** is used.

MEM applies Maximum Entropy reconstruction according to the method of Gull and Daniell: argument **-ndim** specifies the number of dimensions to reconstruct, argument **-neg** activates two-channel mode, for reconstruction of data with both positive and negative signals, argument **-zero** corrects the zero-order offset introduced during reconstruction, argument **-alpha** specifies the fraction of a given iterate which will be added to the current MEM spectrum, argument **-sigma** specifies the estimated standard deviation of the noise in the time-domain, argument **-freq** produces the final MEM result in the frequency-domain, arguments **-xconv** and **-yconv** specify the line-sharpening function, which in Figure 1.6 is EM (Exponential Multiply) for both dimensions, and arguments **-xcQ1** and **-ycQ1** specify the corresponding line-sharpening parameters, which in Figure 1.6 are 20 Hz and 15 Hz for the 15N and 1H dimensions respectively. Other arguments can be used to optimize convergence speed, or to increase stability for reconstruction of data with high dynamic range.

POLY (frequency-domain) applies polynomial baseline correction of the order specified by argument **-ord**, via an automated baseline detection method when used with argument **-auto**. The default is a fourth order polynomial. The automated baseline mode works as follows: a copy of a given vector is divided into a series of adjacent sections, typically 8 points wide. The average value of each section is subtracted from all points in that section, to generate a "centered" vector. The intensities of the entire centered vector are sorted, and the standard deviation of the noise is estimated under the assumption that a given fraction (typically about 30%) of the smallest intensities belong to the baseline, and that the noise is normally distributed. This noise estimate is multiplied by a constant, typically about 1.5, to yield a classification threshold. Then, each section in the centered vector is classified as baseline only if none of the points in that section exceeds the threshold. These classifications are used to correct the original vector.

POLY (time-domain) when used with the argument **-time**, fits all data points to a polynomial, which is then subtracted from the original data. It is intended to fit and subtract low-frequency solvent signal in the FID, a procedure which often causes less distortion than time-domain convolution methods. By default, a fourth order polynomial is used. For speed, successive averages of regions are usually fit, rather than fitting all of the data points.

PS applies the zero and first order phase corrections as specified in degrees by the arguments **-p0** and **-p1**. The PS function applies no processing if these values are both zero; for this reason, a zero,zero phase correction step is commonly kept in a processing scheme for completeness, so that the scheme can be copied and reused more easily.

RS, when used in the time-domain, applies a right-shift by the number of points specified by argument **-rs**, and updates the recorded time-domain size if the argument **-sw** is used.

SOL uses time-domain convolution and polynomial extrapolation to suppress solvent signal with a default moving average window of +/- 16 points.

SP applies a sine-bell window extending from $\sin^r(a\pi)$ to $\sin^r(b\pi)$ with offset **a**, endpoint **b**, and exponent **r** specified by arguments **-off**, **-end**, and **-pow**, first-point scaling specified by argument **-c**. The default length is taken from the recorded time-domain size of the current dimension. By default, **a** = 0.0, **b** = 1.0, **r** = 1.0 (sine bell), and the first point scale factor is 1.0 (no scaling).

TP exchanges vectors from the X-axis and Y-axis of the data stream, so that the resultant data stream consists of vectors from the Y-axis of the original data. It is identical to YTP.

YTP is another name for the TP transpose function, which exchanges vectors from the X-axis and the Y-axis of the data stream. The alternative name is provided for contrast with the other transpose functions **ZTP** (X-axis/Z-axis Transpose) and **ATP** (X-axis/A-axis Transpose).

ZF pads the data with zeros; the amount of padding can be specified by argument **-zf**, which defines the number of times to double the data size, or by the argument **-size**, which specifies the desired complex size after zero filling. By default, the data size is doubled by zero filling. Use of the argument **-auto** will cause the zero-fill size to be rounded up to the nearest power of two.

ZTP exchanges vectors from the X-axis and Z-axis of the data stream, so that the resultant data stream consists of vectors from the Z-axis of the original data.

Appendix 3 - Data Input/Output Programs and Arguments of NMRPipe

Input and Output Templates: the following describes the method used to specify input and output data in the multi-file 2D plane format as well as those programs used along with **nmrPipe** in the examples and figures. The arguments described are not complete lists, but rather only those which are used in the examples.

3D File Name Templates: 3D data in the multi-file 2D plane format is specified as a template, a single name which stands for a series of 2D file planes. The template includes a format specification, usually "%03d", which is substituted by the Z-axis plane number in the actual file names. The format specification is interpreted by rules of the C programming language; the "03d" in the template means the plane number will be included as a zero-padded three-digit number, to give a series of names such as fid/noe001.fid, fid/noe002.fid, fid/noe003.fid, etc.

4D File Name Templates: 4D data in the multi-file 2D plane format is specified as a template, a single name which stands for a series of 2D file planes. The template includes a format specification, usually "%02d%03d", which is substituted by the A-axis and Z-axis plane numbers in the actual file names. The format specification is interpreted by rules of the C programming language; the "02d" and "03d" in the template means the A-axis plane number will be included as a zero-padded two-digit number, followed by the Z-axis plane number as a zero-padded three-digit number.

bruk2pipe converts binary data from various types of Bruker spectrometers to the nmrPipe data format. The related programs **var2pipe** and **bin2pipe** perform Varian Unity conversions, and general-purpose binary conversions, respectively. The programs take as input a file or data stream in the binary spectrometer format, and produce a file, file series, or data stream in the NMRPipe format. The programs require a collection of arguments defining the acquisition parameters for each dimension, prefixed by **-x**, **-y**, **-z**, and **-a**; the commonly required arguments follow: arguments **-xN** etc. define the total number of points saved in the input file for a given dimension; arguments **-xT** etc. define the number of valid complex points actually acquired, in case this differs from the number of points saved in the input file; arguments **-xMODE** etc. define the quadrature detection mode of the given dimension; arguments **-xSW** etc. define the full spectral with in Hz for the given dimension; arguments **-xOBS** etc. define the observe frequency in MHz for a given dimension, while arguments **-xCAR** etc. define the carrier position in PPM; arguments **-xLAB** etc. define unique axis labels; argument **-ndim** defines the number of dimensions in the input; argument **-aq2D** defines the type of 2D output file planes produced as either magnitude mode, States/States-TPPI, or TPPI.

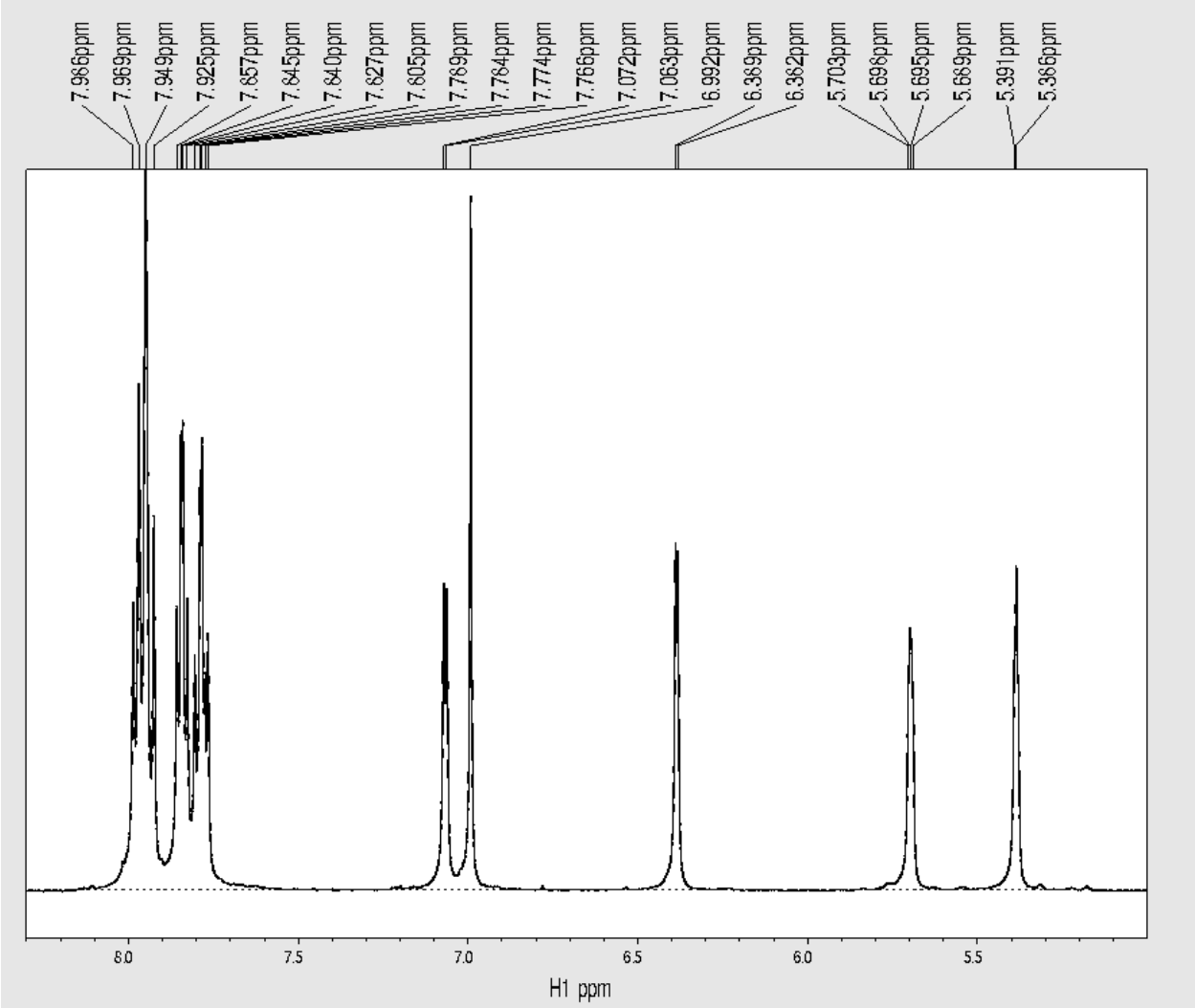
xyz2pipe creates a data stream for multidimensional processing via pipeline by reading vectors from the selected axis of nD data in the multi-plane format. The arguments **-x**, **-y**, **-z**, and **-a** select the axis, and the argument **-in** is used to specify the input file series as a template (see "Input and Output Templates" above). Depending on the dimension selected, the other dimensions are reordered by a multidimensional rotation, which is similar, but not always identical, to a transpose. If the original order of dimensions is described as XYZA..., the relative reordering of data can be summarized as follows:

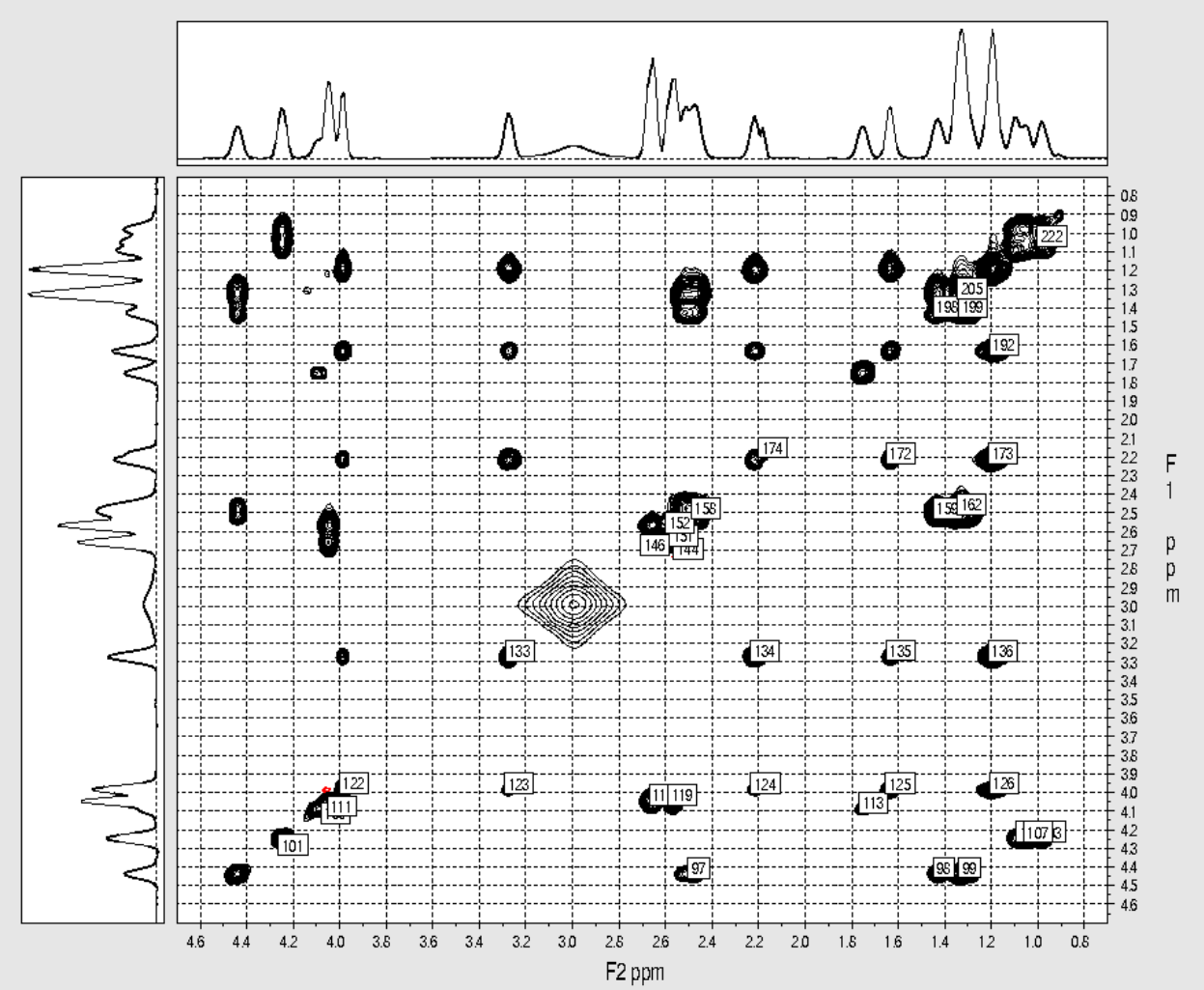
nmrPipe -fn TP	Exchange of the first two dimensions:	XYZA... to YXZA...
nmrPipe -fn ZTP	Exchange of the first and third dimensions:	XYZA... to ZYXA...
nmrPipe -fn ATP	Exchange of the first and fourth dimensions:	XYZA... to AYZX...
xyz2pipe –x	No change to data order:	XYZA... to XYZA...
xyz2pipe –y	Rotation of the first two dimensions (same as TP):	XYZA... to YXZA...
xyz2pipe –z	Rotation of the first three dimensions:	XYZA... to ZXYA...
xyz2pipe –a	Rotation of the first four dimensions:	XYZA... to AXYZ...

pipe2xyz writes vectors from a data stream to the selected axis of ND data in the multi-plane format. The arguments **-x**, **-y**, **-z**, and **-a** select the axis, and the argument **-out** is used to specify the output file series as a template (see "Input and Output Templates" above). In order to write to a given axis, the program **pipe 2xyz** performs rotations of the data which are complementary to those performed by **xyz2pipe**. This means that a pipeline which begins with **xyz2pipe** reading from a given dimension and ends with **pipe2xyz** writing to the same dimension will conserve the original data order if no transpose steps are included in between.

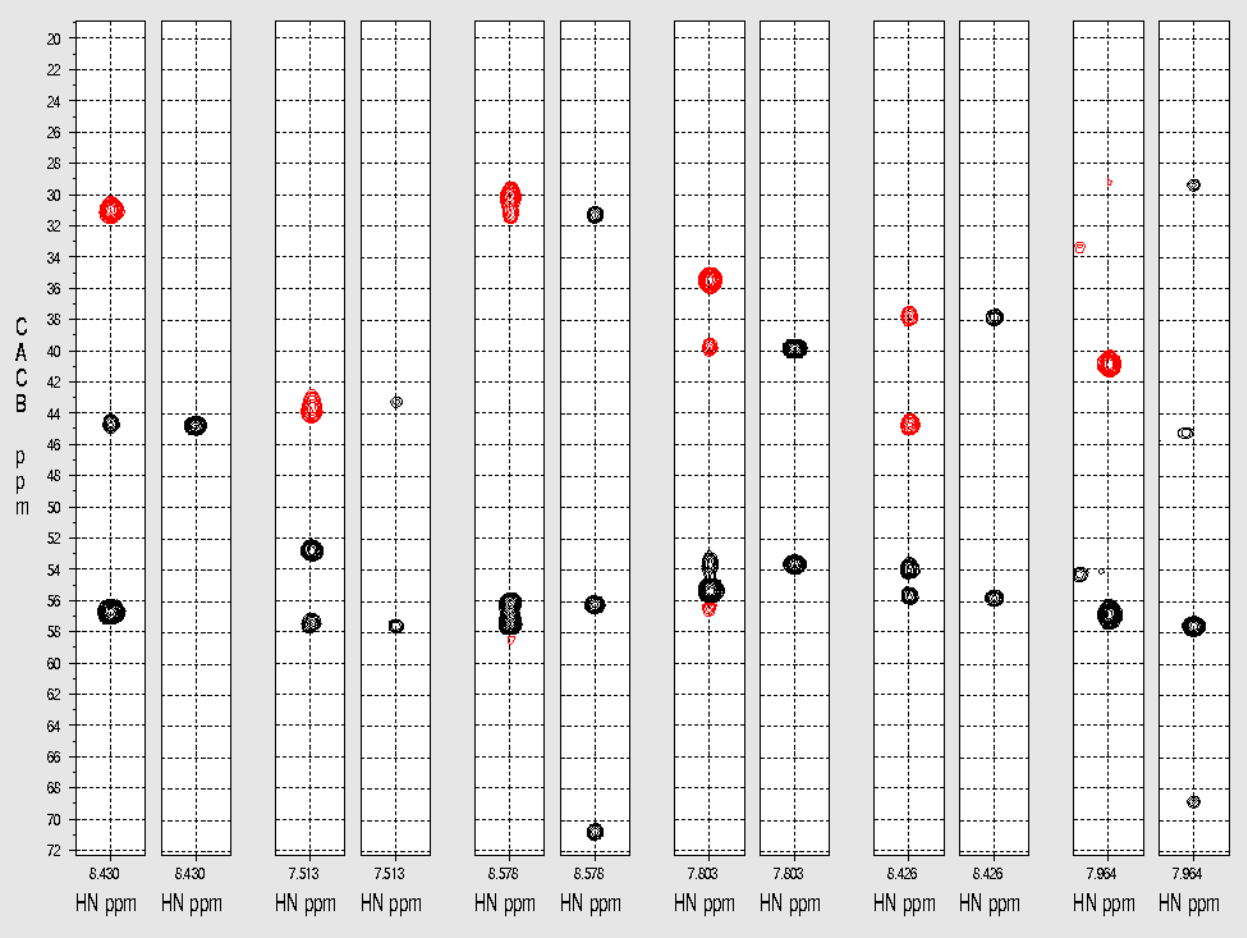
Appendix 4 – Example PostScript Output from NMRPipe

Example PostScript Output: A 1D ^{1}H Spectrum with Peak Labels.

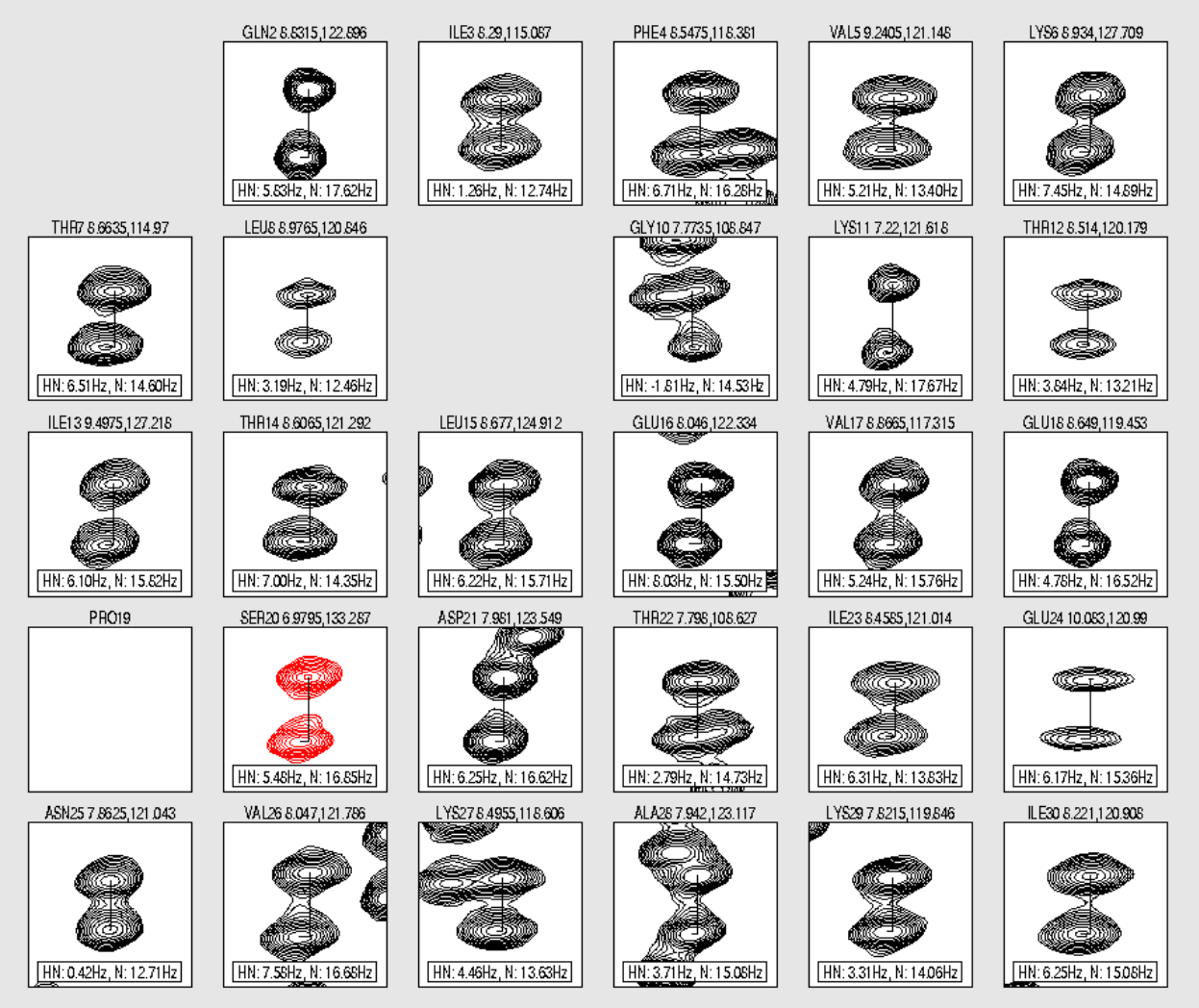




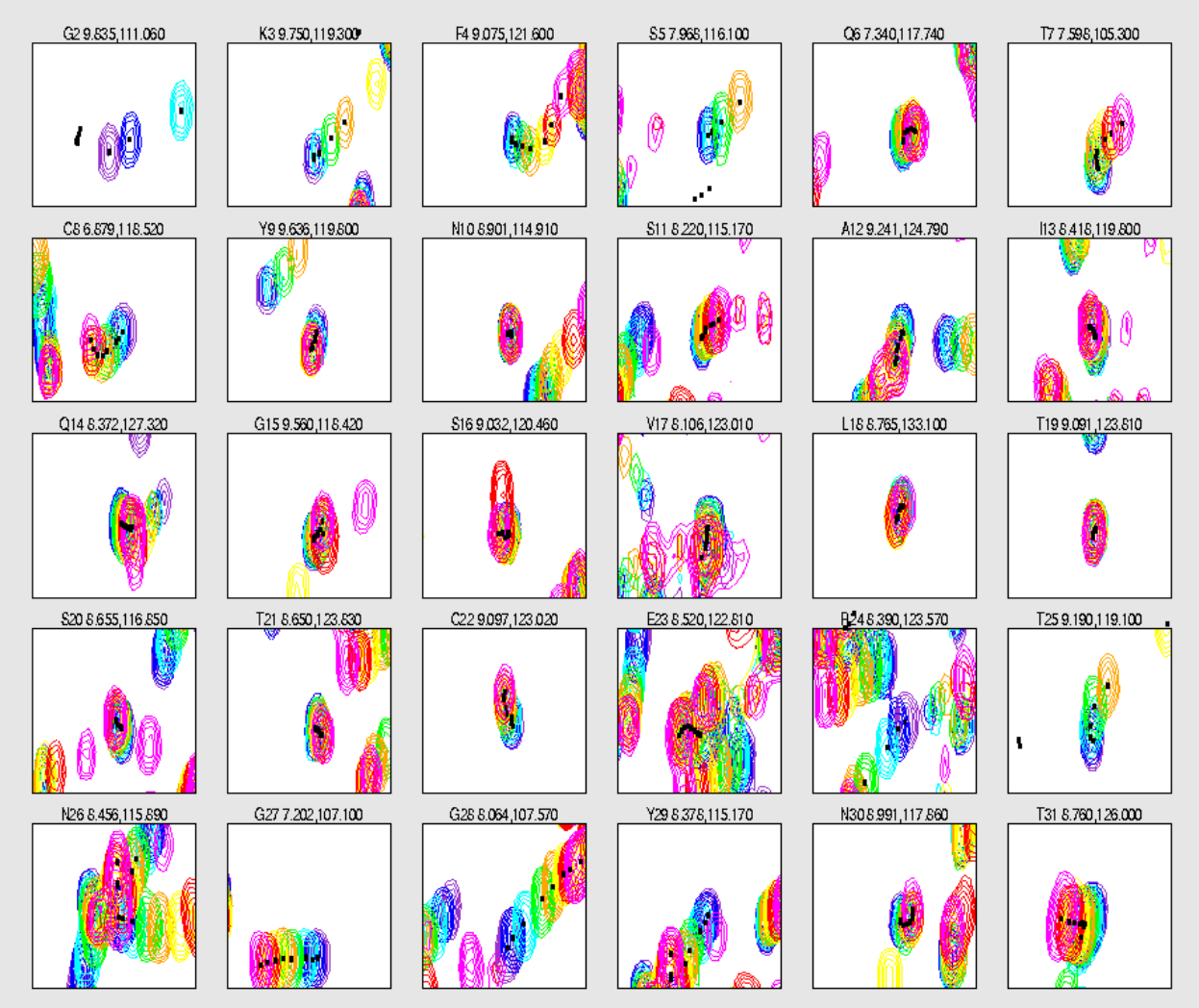
Example PostScript Output: 2DNOE Spectrum, Peak Labels and Projections.



Example PostScript Output: Strip Plots from a 3D CBCANH Spectrum.



Example PostScript Output: 2D Region Plots from an HSQC Coupling Analysis.



Example PostScript Output: Region Plots from a 2D HSQC Titration Analysis.

Dedication and Acknowledgements

This thesis is dedicated to the memory of Prof. Harry Brumberger, an outstanding and patient teacher of science, who guided my first attempts at research, and encouraged my first efforts in scientific software development.

Likewise, grateful thanks goes to his long-time collaborator, Prof. Jerry Goodisman, who was my mentor, and whose generous devotion of time and lively intellect set the example for all my work in computational methods.

These acknowledgements would not be complete without mention of Prof. George Levy, who by unlikely chance provided an introduction to the field of NMR.

I would also like to acknowledge the kindness, expertise, and tireless assistance of Anthony and Edna Delaglio in the preparation and design of this thesis document.

This work was supported in part by the AIDS Targeted Anti-Viral Program of the Office of the Director of the National Institutes of Health.

Along these lines, I gratefully thank Dr. Ad Bax for his unwavering patience and support, and for making it possible to join the NIH. His amazing creativity, exuberance, powers of observation and scientific skill made all of this work possible.

Very special thanks to Mr. Masaaki Akutsu, whose energy, guidance, and generous support has allowed wide-spread use of this work in the NMR community of Japan, which in turn has greatly stimulated this work's further progress. And, as importantly, I thank Akutsu-san for his many years of friendship, and for his thoughtful introductions to many aspects of Japanese culture, which has been a treasure.

Finally, very warmest thanks to the wonderful Prof. Yuji Kobayashi, my first scientific collaborator in Japan. He has encouraged many aspects of this work for the past 15 years, and made presentation of this thesis possible.