Chapter 6

Rapid Prediction of Multi-dimensional NMR Data Sets Using FANDAS

Siddarth Narasimhan, Deni Mance, Cecilia Pinto, Markus Weingarth, Alexandre M.J.J. Bonvin, and Marc Baldus

Abstract

Solid-state NMR (ssNMR) can provide structural information at the most detailed level and, at the same time, is applicable in highly heterogeneous and complex molecular environments. In the last few years, ssNMR has made significant progress in uncovering structure and dynamics of proteins in their native cellular environments [1–4]. Additionally, ssNMR has proven to be useful in studying large biomolecular complexes as well as membrane proteins at the atomic level [5]. In such studies, innovative labeling schemes have become a powerful approach to tackle spectral crowding. In fact, selecting the appropriate isotope-labeling schemes and a careful choice of the ssNMR experiments to be conducted are critical for applications of ssNMR in complex biomolecular systems. Previously, we have introduced a software tool called FANDAS (Fast Analysis of multidimensional NMR DAta Sets) that supports such investigations from the early stages of sample preparation to the final data analysis [6]. Here, we present a new version of FANDAS, called FANDAS 2.0, with improved user interface and extended labeling scheme options allowing the user to rapidly predict and analyze ssNMR data sets for a given protein-based application. It provides flexible options for advanced users to customize the program for tailored applications. In addition, the list of ssNMR experiments that can be predicted now includes proton (^{1}H) detected pulse sequences. FANDAS 2.0, written in Python, is freely available through a user-friendly web interface at http://milou.science.uu.nl/services/FANDAS.

Key words Biomolecular NMR, Labeling schemes, Spectral prediction, Spectral analysis and proton detection

1 Introduction

NMR represents a powerful tool for studying protein structure and dynamics. Thus, there is a growing need to make it more accessible to the community by providing analysis tools from an early level of the project to the final data analysis stage. We previously introduced a web application termed Fast Analysis of multidimensional NMR DAta Sets (FANDAS) [6] which aids in spectral analysis by producing peak lists for a variety of multidimensional solid-state NMR (ssNMR) experiments. FANDAS has the unique ability to

Ranajeet Ghose (ed.), *Protein NMR: Methods and Protocols*, Methods in Molecular Biology, vol. 1688,
DOI 10.1007/978-1-4939-7386-6_6,

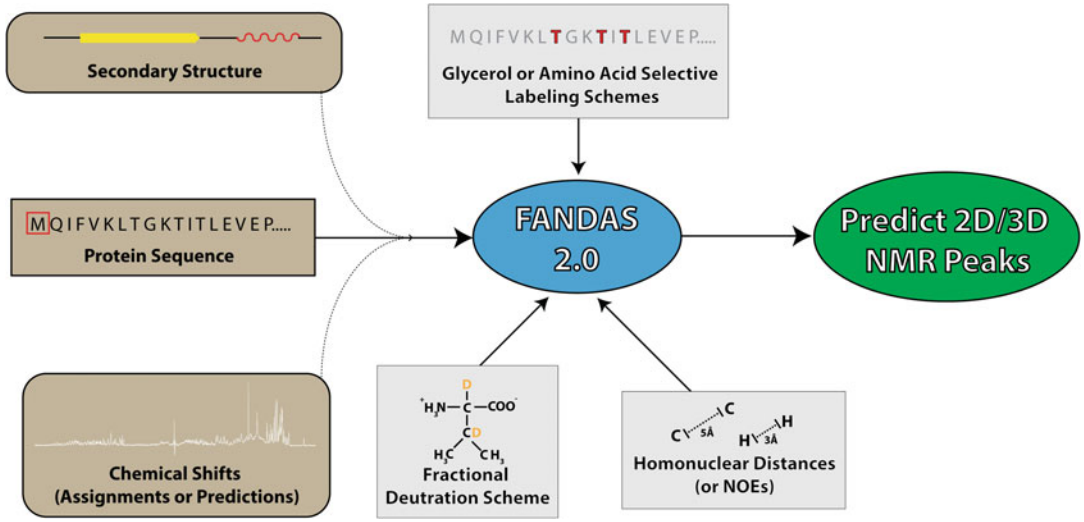

Fig. 1 Overall scheme of FANDAS, describing the different types of data used to refine the final prediction of peaks for multi-dimensional NMR experiments. The different types of inputs that FANDAS accepts are given in *brown boxes* and input modification options such as labeling schemes and distances are given in *grey boxes*

accommodate a variety of inputs and input modifications to accurately predict the peak pattern for given NMR experiments (Fig. 1). This property has been particularly useful in predicting NMR peaks in multidimensional NMR experiments and has, for example, enabled rapid identification of the most suitable labeling schemes even for studies of proteins in native cellular environments [3–5, 7–9].

Here, we present an improved version of FANDAS, called "**FANDAS 2.0**" (simply referred to as FANDAS in this chapter). FANDAS is now built on Python, and extended to a web interface using the Flask micro-framework (http://flask.pocoo.org). Therefore, vast improvements could be done to the user interface by introducing a user-friendly web-form that integrates all inputs at once. Another major improvement aids in choosing labeling schemes by offering improved choices for forward labeling (*see*, ref. 10 for a further discussion of forward and reverse isotope labeling) such as specific ^{13}C-only or ^{15}N-only labeling which were not available before. These new options speed up the process of predicting NMR correlation data for different forward labeling schemes, in particular with respect to inter-residue sequential correlation experiments such as N-Co-Cx [11]. Additionally, it is now possible to predict correlations for proton-detected ssNMR experiments that are of increasing use in biological ssNMR [12, 13]. Finally, additional features have been implemented, which will be discussed in detail in the following sections. For the users with command line experience, we provide a standalone python application (https://github.com/siddarthnarasimhan/

FANDAS_2.0), which is also the engine for the webserver and can be run on a local personal computer. This version provides more flexibility to the user allowing customization of the program for advanced studies such as those involving unique or unnatural amino acids, or to extend the program to predict additional ssNMR experiments currently not implemented in FANDAS 2.0.

This chapter provides protocols describing the use of both the webserver and command line versions of FANDAS. In the following sections, a step-by-step description is given for each feature in FANDAS, using a sample protein as an example.

2 Materials and Prior Knowledge

2.1 Introduction to SPARKY Peak Lists

FANDAS has been geared toward users of the NMR data analysis program, SPARKY [14]. Thus, a prior knowledge of SPARKY is highly recommended. SPARKY allows for easy NMR spectral visualization, assignment, and analysis. The peaks identified in a spectrum are usually stored as space-delimited plain text files known as peak lists. Each line in the peak file, corresponding to a single peak, consists of three elements: (a) The peak label that appears as an annotation on the spectrum, (b) the peak coordinates that represent chemical shifts of the correlating nuclei, and (c) the notes section, where the user is free to add any notes regarding the peak (Fig. 2). The output of FANDAS is composed of peak files that the user can then superimpose on the spectra.

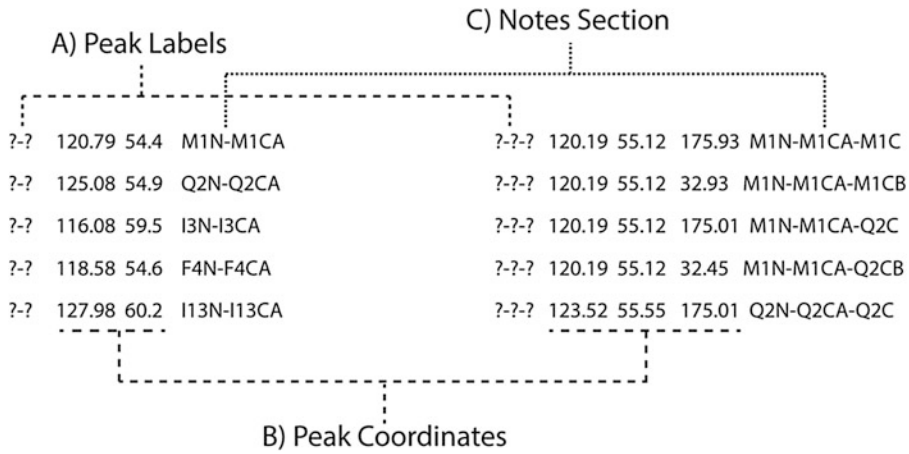

Fig. 2 Description of SPARKY 2D (*left*) and 3D (*right*) peak lists generated using FANDAS. (**a**) The peak labels, which would appear on the spectrum when the peak list is loaded. The default "? -?" for 2D peak list and "?-?-?" for 3D peak list imply that the peaks do not have any labels. (**b**) Peak coordinates, which are essentially the chemical shifts of the correlating nuclei. (**c**) The notes section- notes do not appear on the spectrum, but can be accessed too in the peak list. By default, FANDAS stores the peak information in the notes section

2.2 FANDAS Webserver

The FANDAS webserver has been made user friendly to enable its use in the wider scientific community. Thus, no prior computational knowledge is required to be able to operate it and there are no restrictions on the operating system or the browser. However, we recommend usage of any version of Google Chrome or Mozilla Firefox. The server is freely available at: http://milou.science.uu.nl/services/FANDAS.

2.3 FANDAS Local Installation

To use a local version of FANDAS, the user is required to have at least some elementary knowledge on working with the command line (bash environment is recommended) like navigating through folders and opening text files using editors. Thus, a basic knowledge of using text editors like vim or nano is recommended (alternatively GUI-based text editors like Notepad + or TextEdit could also be used). FANDAS has been created on a computer running MacOS X El Capitan (10.11) with Anaconda 2.4.1 running Python 2.7.12. However, FANDAS can be readily run on a different platform, provided that the following computational requirements are met:

1. Windows/Mac/Linux operating systems that can preferably run a BASH shell environment (the protocol we describe here uses this environment, but the user is free to use something else).
2. Python 2.7 (https://www.python.org/downloads/) or higher with pip (https://pypi.python.org/pypi/pip) to install python packages and the python package numpy (preferably 1.11.2) (https://www.scipy.org/scipylib/download.html).
3. git (https://git-scm.com/book/en/v2/Getting-Started-Installing-Git): to clone the FANDAS package onto the local desktop.
4. DSSP (http://swift.cmbi.ru.nl/gv/dssp/) or STRIDE (http://webclu.bio.wzw.tum.de/stride/) for secondary structure assignments (Optional).
5. SHIFTX2 (http://www.shiftx2.ca/) for estimating chemical shifts from existing structural models (Optional).

Detailed instructions to install the above (required) packages on your operating system of choice are given in the **readme.md** on the GitHub repository https://github.com/siddarthnarasimhan/FANDAS_2.0.

3 Methods

This section presents a general tutorial, with instructions to use the FANDAS webserver and the command line version of FANDAS. For this purpose, a small post translational modifier protein—Ubiquitin,

has been used as a test case. The test dataset used for this demonstration (**test_dataset.tar**) can be downloaded from the GitHub repository: https://github.com/siddarthnarasimhan/FANDAS_2.0. This demonstration will summarize a typical FANDAS workflow and give the user a glimpse of all the features that the program offers.

3.1 Using the FANDAS Webserver

To run the FANDAS webserver, open a web browser of your choice and navigate to: http://milou.science.uu.nl/services/FANDAS/. The webpage consists of a web form that is divided into blocks, containing fields corresponding to different types of input. Instructions to fill in particular fields are mentioned in the webpage and will be elucidated below in a step-by-step manner.

1. ***Input Sequence and Secondary Structure*** **Block:** This block contains three fields of which two are mandatory:
   (a) **Project Name (mandatory):** The user may choose any arbitrary name with more than four letters or numbers. No special characters (except underscore, "_") or spaces are permitted.
   (b) **Protein Sequence (mandatory):** The input sequence must be in single letter amino-acid code (not case sensitive). To this end, inputs are restricted to the 20 naturally occurring amino acids and any other input values will be changed to "A," i.e., alanine. If the user uses a raw FASTA sequence, all lines that do not contain the sequence need to be removed.
   (c) **Secondary Structure:** The secondary structure assignments will be used to accurately assign chemical shifts to the backbone heavy atoms and protons by using predetermined average chemical shift statistics for every amino acid existing in different distinct conformations [15]. If the protein structure is available, it is possible to obtain these assignments by using programs such as DSSP [16] (http://swift.cmbi.ru.nl/gv/dssp/) or STRIDE [17] (http://webclu.bio.wzw.tum.de/stride/) and if the structure is not known, prediction tools such as JPRED4 [18] (http://www.compbio.dundee.ac.uk/jpred/) or PSIPRED [19] (http://bioinf.cs.ucl.ac.uk/psipred/) may be used. The user must ensure that the secondary structure assignments/predictions are simplified to the assignments that are permitted, namely: alpha-helix **"a,"** beta-sheet **"b,"** and random coil **"c"** in addition to NMR calculated averages **"n"** which is assigned to the protein by default. In the example shown in Fig. 3, the secondary structure assignments were done using command line version of STRIDE on the PDB structure 1UBQ and

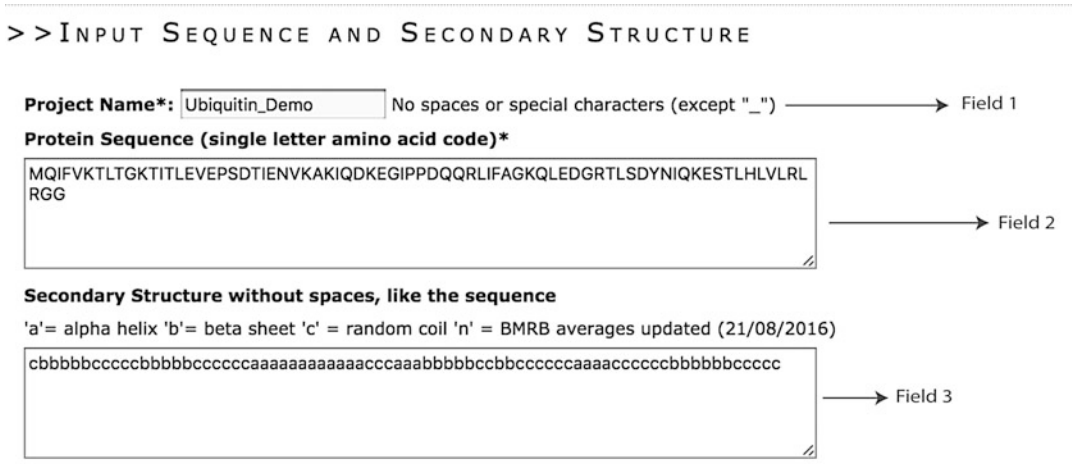

Fig. 3 Example input of the "*Input Sequence and Secondary Structure*" block

simplified to **a**, **b,** and **c**. If the secondary structure assignments are not available for a part of the protein, it is recommended that the user fills in **"c"** or **"n"** at the appropriate sequence sites. For example, if for the sequence "**APAPMLQSMVSLLQSLV"** the secondary structure of the first four residues is not known, the secondary structure input could be "**nnnn**" or "**cccc**" followed by the assignments for the remaining residues e.g.,: **nnnnaaaaaaaaaaaaa**.

An example of a completed form is given in Fig. 3.

2. ***Input Chemical Shifts as BMRB Table*** **Block:** FANDAS also accepts user defined chemical shift assignments that would be used in lieu of the default assignments discussed in the previous section. In the absence of chemical-shift assignments, it is possible to generate (predict) chemical shifts on the basis of a 3D protein structure to a reasonable level of accuracy for every atom using programs such as SHIFTX2 [20] (http://www.shiftx2.ca/cgi-bin/shiftx2.cgi). The following are the descriptions of the fields present in this input block:

   (a) **BMRB Tables**: It is recommended for the assignments to be in a tabular NMR-STAR format (for a detailed description, *see* http://www.bmrb.wisc.edu/formats.html). FANDAS also accepts any table-like text file with rows (lines) corresponding to each nucleus, with columns consisting of atom name, residue number, and chemical shift (Fig. 4). In the example (Fig. 5), the command line version of SHIFTX2 was used to generate the BMRB tables using the crystal structure of Ubiquitin (1UBQ).

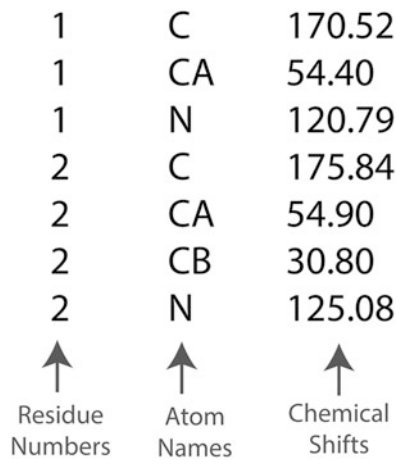

Fig. 4 An example of BMRB tables that is accepted in FANDAS

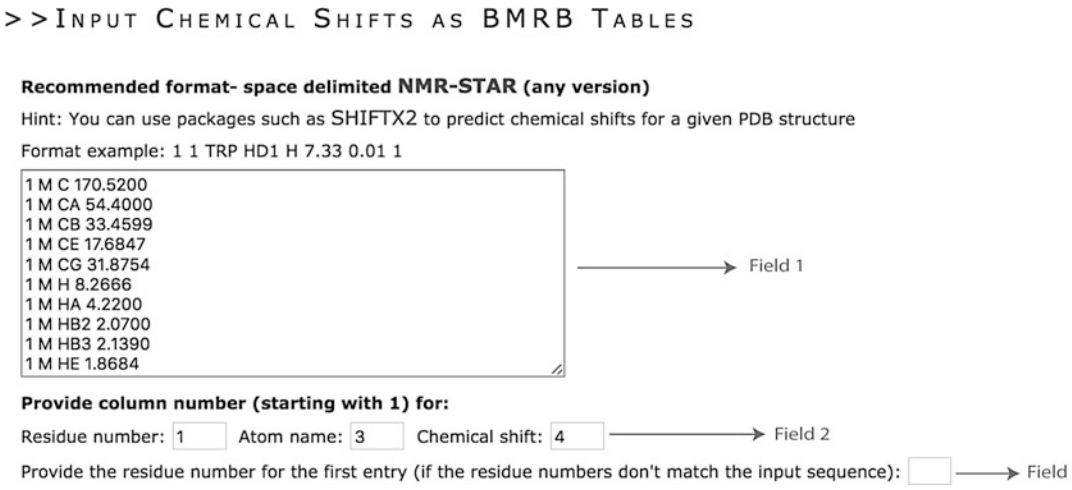

Fig. 5 Filled out sample of "*Input Chemical Shifts as BMRB Tables*" block

(b) **Provide column numbers for residue number, atom name, and chemical shift:** The user is required to provide the column numbers for residue number, atom name, and chemical shift corresponding to the tables input in the previous field.

(c) **Provide the residue number for the first entry:** This field should be left blank if the residue numbers in the BMRB tables match the residue numbers in the input sequence. If not, the residue number for the first entry (line) in the assignment table is to be entered in this field. This value is used to offset all residue numbers in the table to match the sequence.

3. ***Amino-Acid Selective Labeling Schemes*** **Block:** By default, it is assumed that the protein of interest is fully isotope labeled (i.e., 100% enriched at the ^{13}C and ^{15}N positions). The options provided in this section allow for the incorporation of amino-acid selective forward or reverse labeling schemes (Fig. 6). If forward or reverse labeling schemes are used, the user must specify a list of amino acids for the chosen labeling scheme. This section is particularly useful for assessing the result of different labeling schemes on the resulting ssNMR spectra. A typical application for features in this section is discussed section 4.
4. ***Glycerol Labeling Schemes and Fractional Deuteration:*** To reduce spectral crowding among the various carbon positions, glycerol-based labeling schemes have been introduced [21–23]. Such schemes have been implemented in FANDAS and choosing any of the glycerol labeling schemes would disable amino acid selective (forward or reverse) labeling schemes or fully labeled schemes. Additionally, this section is set up to include fractional deuteration (dashed box in Fig. 7) scheme, which is intended to fully deuterate $^{13}C\alpha$ nuclei as well as specific side chain carbons [13, 24].

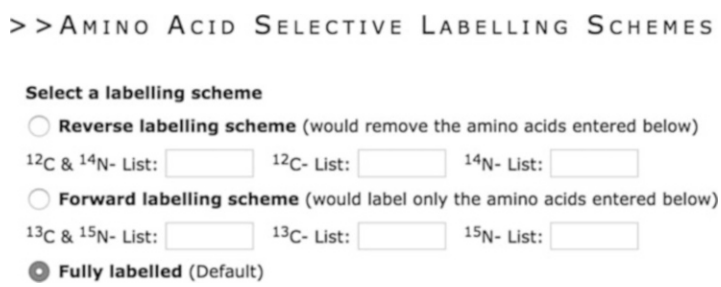

Fig. 6 "*Amino-Acid Selective Labeling Schemes*" block consists of options to choose the appropriate labeling scheme and a provision to enter the desired labeled amino acids in double or single labeled forms

Fig. 7 Highlighting how different labeling schemes are handled in FANDAS and the option for fractional deuteration (*green box*)

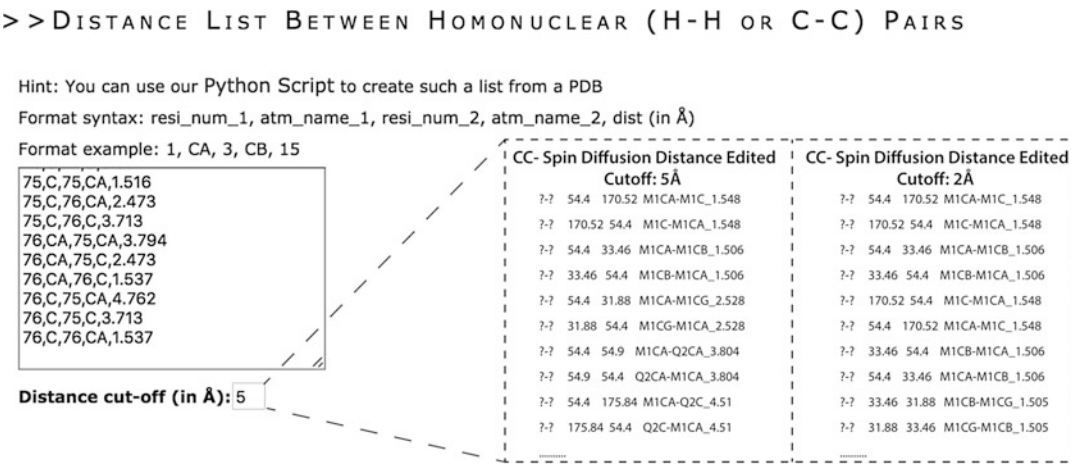

Fig. 8 Example outputs (*dashed boxes*) suited for the analysis of distance-dependent CC correlation experiments (such as protein-driven spin diffusion) for two different cut-off values. Distance lists were calculated using the crystal structure of Ubiquitin (PDB: 1UBQ)

5. ***Distance List Between Homonuclear (H-H or C-C) Pairs*** **Block:** Predicting peaks for distance-edited experiments requires a list of distances between H-H or C-C groups depending on the experiment. These distances can essentially be through-space distance restraints obtained previously from NMR experiments or they can be generated from PDB coordinates. To generate distances from PDB coordinates, we have generated a dedicated python script **"distance_calculator.py"** which can be accessed from the GitHub repository: https://github.com/siddarthnarasimhan/FANDAS_2.0. Along with the distances provided, the user must provide a cutoff distance (in Å). To exemplify such a distance cutoff, a sample output for a distance edited CC Spin diffusion experiment is shown for two cut-off values in (Fig. 8).
6. ***Predict Peaks for NMR Experiments (SPARKY Format)*** **Block:** This is the final section of FANDAS webserver where the experiments for which predictions are to be made, are listed. This section consists of a list of experiments that the user can simply select using the appropriate check boxes (Fig. 9).
   (a) Peak Labels: Along with the list of experiments, this section features an option to turn on peak labels (highlighted with a dashed box in Fig. 9). As described in Subheading 2.1, the default peak labels are null: "?-?" "?-?-?". By turning on the "Peak Labels" option, the peak list would now be incorporated with the labels. The labels would inform the user of the nuclei involved in the correlation observed (Fig. 10).

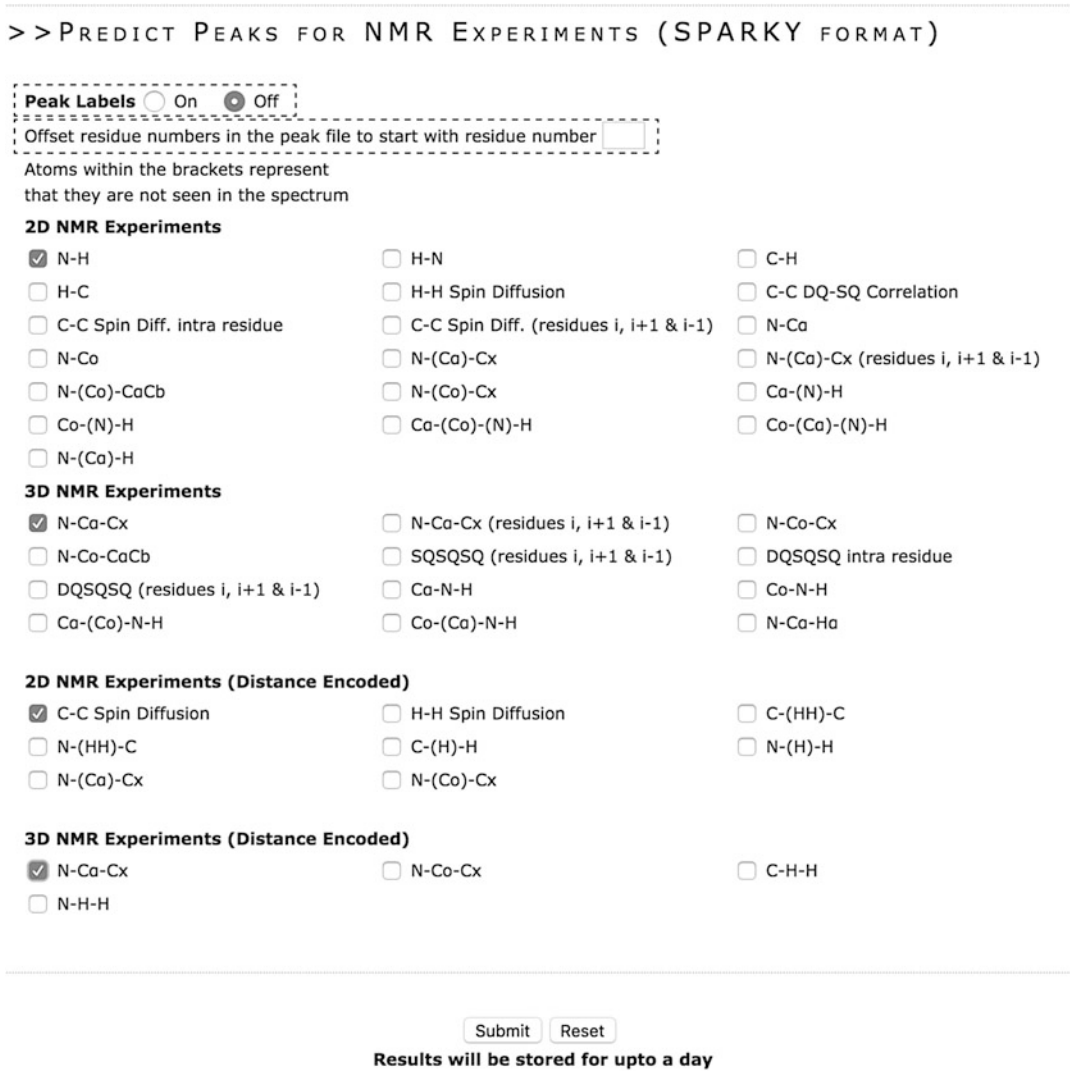

Fig. 9 The final input block, where the user can select experiments for which predictions are to be made, specify if peak labels are required and if the residue numbers need to be offset (highlighted using *dashed boxes*)

(b) Residue number offset: This field (highlighted with a dashed box in Fig. 9) takes in numerical values to offset the residue number in the final peak lists. If the user wishes that the residue numbers start from a different number than one, it can be specified here. This is particularly useful if the prediction is being done for a specific segment or a domain and not the whole protein, wherein the starting residue number may not be one.

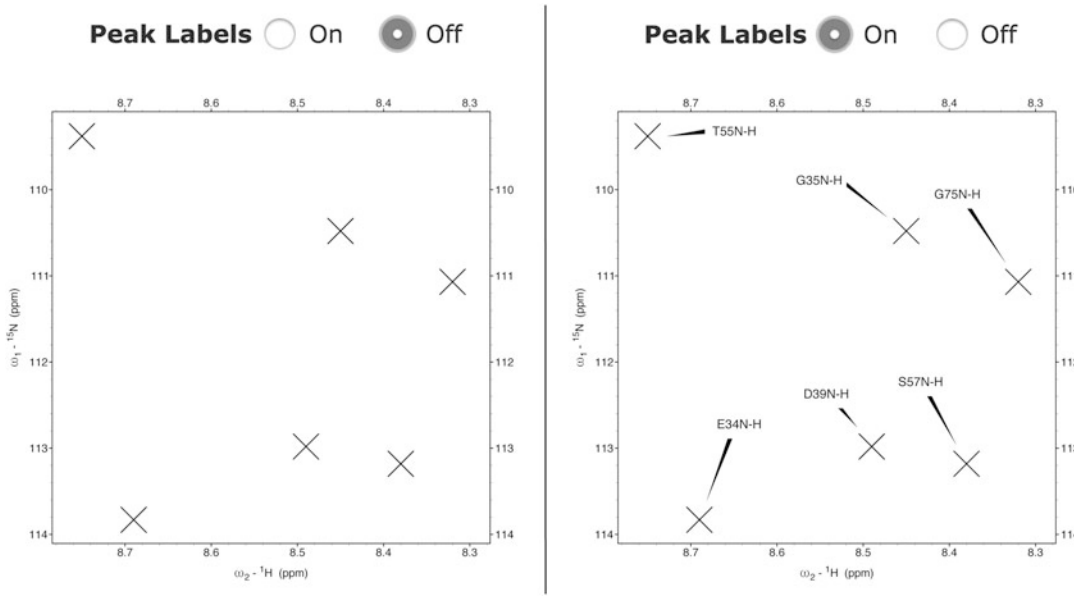

Fig. 10 Sample peak outputs overlaid on empty N-H Spectra in SPARKY. The left panel shows the default output when Peak Labels are inactive and the right panel shows the output when Peak Labels are turned on

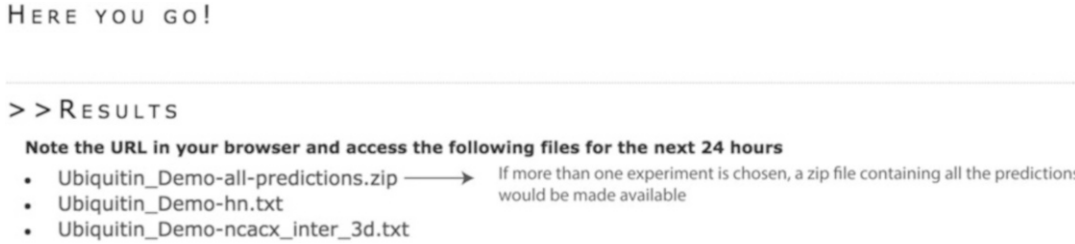

Fig. 11 Prediction results as seen on the results page

7. **Accessing the Results:** After the form is completed, the job would be submitted to the server and the user will be redirected to the results page where the predictions will be available for download (Fig. 11). The user can access the peak files by simply clicking on the appropriate file that is listed. Currently, results are stored for up to one day on the server and can be accessed by simply navigating to the URL of the results page that is shown in the browser window.

3.2 Running the FANDAS Application Locally

The features of this version are the same as the webserver, since the webserver uses this python application as an engine. Thus, this section will explain how to clone the application along with the supporting files onto a local computer and how to use the interface rather than discuss-specific features. Every command line input and output is shown in a black background; input lines begin with a **$** sign.

1. **Cloning the git repository onto the local computer:** For a quick introduction to git, we refer the interested reader to https://git-scm.com/book/en/v2/Getting-Started-Git-Basics. To start with, open a BASH environment with a terminal emulator of your choice and type the following:

```
$ git clone https://github.com/siddarthnarasimhan/FANDAS_2.0
```

   The user should see the following output if the git cloning was successful (if not, please check if git has been installed):

```
Cloning into 'FANDAS_2.0'...
remote: Counting objects: 7, done.
remote: Compressing objects: 100% (6/6), done.
remote: Total 7 (delta 0), reused 7 (delta 0), pack-reused 0
Unpacking objects: 100% (7/7), done.
```

   Navigate to the FANDAS_2.0 folder and list the files present. All files are required and the demo folder (ubiquitin_demo) is shown below:

```
distance_calculator.py fandas.py readme.md standard.dat test_-
dataset.tar
```

2. **Description of the contents in the FANDAS_2.0 Package:**
   (a) **distance_calculator.py**: This script is used to calculate the distance between homonuclear pairs from the supplied PDB coordinates.
   (b) **fandas.py**: This is the FANDAS_2.0 application.
   (c) **README.md**: A github mandated readme file for the program.
   (d) **standard.dat**: A file containing the standard chemical-shift tables for all amino acids.

(e) **test_dataset.tar**: Contains a test dataset that has been used to demonstrate the use of FANDAS in this manual. It contains a readme file named "**readme.txt**" that describes all the files in the folder. In the following sections, the files in this folder would be referred to for demonstration.

3. **Testing the application and the help flag:** This is important, as the user would already know if all the modules (*see* **readme.md** file for more information) necessary to run FANDAS are available. Besides, considering the number of arguments that FANDAS accepts, the user may have to invoke the help feature each time the program is used, to ensure that arguments are used correctly. Execute the script with the help "**-h**" flag:

```
$ python fandas.py -h
```

If the usage message followed by a list of arguments is printed (shown in the next section), then there are no compilation errors and thus the application is ready to use.

4. **Usage message and description of the arguments:** The usage message essentially lists all arguments that FANDAS takes as flags. The flags that have been listed within square brackets are optional arguments. The ones that are not listed within square brackets are the obligatory arguments, which in our case are only **-i** that corresponds to the input sequence:

```
usage: fandas.py [-h] -i I [-ss SS] [-bt BT] [-btc BTC BTC BTC] [-ls LS]
                 [-dl DL [DL ...]] [-cl CL [CL ...]] [-nl NL [NL ...]] [-fd]
                 [-exp_2d EXP_2D [EXP_2D ...]]
                 [-exp_2dd EXP_2DD [EXP_2DD ...]]
                 [-exp_3d EXP_3D [EXP_3D ...]]
                 [-exp_3dd EXP_3DD [EXP_3DD ...]] [-sl]
                 [-dlist DLIST [DLIST ...]] [-dlim DLIM] [-o O]
```

When the help option is invoked, a description of each argument is printed below the usage message (shown below). The description for each argument needs to be read carefully. Inputs for sequence, secondary structures, BMRB tables, and distance lists are to be supplied as plain text files.

```
optional arguments:
  -h, --help            show this help message and exit
  -i I                  input sequence as a text file (REQUIRED)
  -ss SS                secondary structures as a text file: 'a'- alpha helix,
                        'b'- beta sheet & 'c'- random coil; if unspecified,
                        will use BMRB averages: 'n'
  -bt BT                BMRB tables as a text file in NMR Star format
  -btc BTC BTC BTC      column numbers for residue number, atom name and
                        chemical shifts in the BMRB tables
  -ls LS                labelling scheme. Default is Fully Labelled. Other
                        options: fw = Forward Labelled, rv = Reverse Labelled,
                        gl13 = 1,3 Glycerol Labelling, gl2= 2 Glycerol
                        Labelling
  -dl DL [DL ...]       list of 13C & 15N (for forward labelling) or 12C & 14N
                        (for reverse labelling) amino acids
  -cl CL [CL ...]       list of 13C (for forward labelling) or 12C (for
                        reverse labelling) amino acids
  -nl NL [NL ...]       list of 15N (for forward labelling) or 14N (for
                        reverse labelling) amino acids
  -fd                   fractionally deuterated, if you use this flag, it
                        would be implemented automatically
  -exp_2d EXP_2D [EXP_2D ...]
                        list of 2D experiments: ['NH', 'HN', 'CH', 'HC', 'HH',
                        'DQSQ', 'CC_SPINDIFF_INTRA', 'CC_SPINDIFF_INTER',
                        'NCA', 'NCO', 'NCACX', 'NCACX_INTER', 'NCOCX',
                        'NCOCA_CB', 'CANH', 'CONH', 'CACONH', 'COCANH',
                        'NCAH']
  -exp_2dd EXP_2DD [EXP_2DD ...]
                        list of distance encoded (distance list, -dl and limit
                        -dlm must be provided) 2D experiments: ['CC_SPINDIFF',
                        'HH', 'CHHC', 'NHHC', 'CHH', 'NHH', 'HHC', 'NCACX',
                        'NCOCX']
  -exp_3d EXP_3D [EXP_3D ...]
                        list of 3D experiments: ['NCACX', 'NCACX_INTER',
                        'NCOCX', 'NCOCA_CB', 'SQSQSQ_INTER', 'DQSQSQ_INTRA',
                        'DQSQSQ_INTER', 'CANH', 'CONH', 'CACONH', 'COCANH',
                        'NCAH']
  -exp_3dd EXP_3DD [EXP_3DD ...]
                        list of distance encoded (distance list, -dl and limit
                        -dlm must be provided) 3D experiments: ['CHH', 'NHH',
                        'NCACX', 'NCOCX']
  -sl                   automatically assign peak labels in the sparky file
  -dlist DLIST [DLIST ...]
                        distance list (as a file) in angstorms as follows:
                        "resi_num,atm_nam,resi_num_2,atm_nam_2,dist".
                        EXAMPLE:"2,CA,4,CB,4.5"
  -dlim DLIM            distance limit in angstorms, default: 5 Angstorm
  -o O                  names for output, default: "fandas_output"
```

5. **Preparation of the input files:** To prepare the input files, we recommend creating a working directory to store all input text files such as sequence, secondary structure assignments, BMRB tables, and the distance lists. The files in the **"test_dataset.tar"** are used as an example below. The choice of text editor is not important, provided that the file created is a plain text file.
6. **Making peak predictions:** For easy use, it is recommended to create an alias for the path of the **"fandas.py"** script in the ***".bashrc"*** or ***".bash_profile"*** file so that the script is globally executable. To make the predictions, navigate to the working directory and type the following command (The description of each argument in the command input is given in Table 1):

```
$ python fandas.py -i 1ubq-seq.txt -ss 1ubq-ss.txt -bt 1ubq-bmrb-tables.txt -btc 1 3 4 -dlist 1ubq-dist.txt -dlim 5 -exp_2d hn -exp_2dd chh -exp_3d ncacx -exp_3dd ncacx -sl
```

7. **Guide to using labeling schemes:** By default, the protein is assumed to be fully ^{13}C & ^{15}N labeled unless arguments defining the labeling schemes are specified. Alternative labeling schemes available are shown in the table below (Table 2).

Table 1
Description of all parameters in a sample FANDAS input

Flag + Argument(s)	Description
`-i 1ubq-seq.txt`	input file (or path) is "**1ubq-seq.txt**"
`-ss 1ubq-ss.txt`	file (or path) containing secondary structure is "**1ubq-ss.txt**"
`-bt 1ubq-bmrb-tables.txt`	file (or path) containing the BMRB tables is "**1ubq-bmrb-tables.txt**"
`-btc 1 3 4`	column indices for residue number, atom name & the chemical shift in the BMRB tables are **1, 3 & 4**
`-dlist 1ubq-dist.txt`	file (or path) containing the distance list is "**1ubq-dist.txt**"
`-dlim 5`	distance limit between nuclei for them to be treated as neighbors for predicting peaks in distance-encoding NMR experiments
`-exp_2d cc_spindiff_intra`	predict peaks for a 2D intra residue CC spin diffusion experiment
`-exp_2dd chh`	predict peaks for a 2D distance-edited CHH experiment
`-exp_3d ncacx`	predict peaks for 3D N-Ca-Cx spectrum
`-exp_3d ncacx`	predict peaks for 3D distance edited N-Ca-Cx spectrum
`-sl`	assign peak labels that would be visible on the spectrum in SPARKY

Table 2
Alternative and supplementary labeling schemes to the default—uniformly ^{13}C & ^{15}N default labeling scheme

Flag + Argument(s)	Description
`-ls <labeling scheme>`	labeling scheme to be used; when left empty, the protein would be assumed to be fully ^{13}C and ^{15}N labeled
`-dl <residue list>`	list of forward $^{13}C + ^{15}N$ labeled amino acids or reverse $^{12}C + ^{14}N$ (unlabeled) amino acids as per the chosen labeling scheme (−**ls**)
`-cl <residue list>`	list of forward ^{13}C labeled or reverse ^{12}C (unlabeled) amino acids as per the chosen labeling scheme (−**ls**)
`-nl <residue list>`	list of forward ^{15}N labeled or reverse ^{14}N (unlabeled) amino acids as per the chosen labeling scheme (−**ls**)
`-fd`	include fractional deutration; this can be combined with other labeling schemes

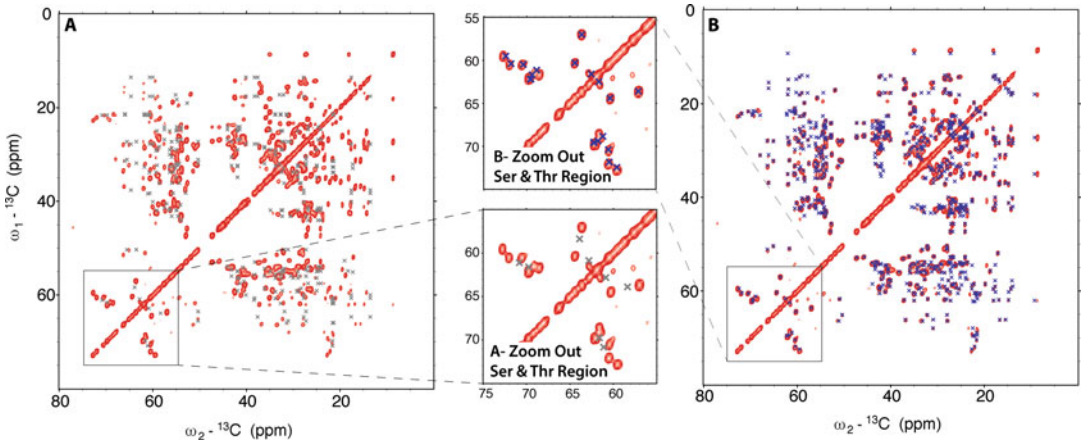

Fig. 12 FANDAS Predictions of CC (PARIS) experiment [25] when only secondary structures are supplied (Panel **A** with the *gray peaks*) and in an information-rich regime where chemical shifts of all atoms were predicted from the crystal structure using SHIFTX2 (Panel **B** with *blue peaks*). In the zoom-in of the serine and threonine region, one can clearly observe that the prediction accuracy is greatly improved when more information is supplied to FANDAS

4 Analysis and Case Study

As mentioned in the introduction section, FANDAS operates by integrating a variety of inputs to predict the peaks occurring in different experiments. Sample outputs for a CC Spin diffusion experiment produced by FANDAS in a data rich and data deprived cases are shown in Fig. 14. Thus, indicating that the quality of the predictions entirely relies on the quality of the input provided. Even when operating in a low information regime, it is possible to get a

substantial amount of preliminary information for a FANDAS-based spectral analysis (Fig. 12).

To highlight some of the areas where FANDAS could be useful besides spectral analysis, the following case studies are described.

1. **Case Study- Choosing Amino Acid Selective Labeling Schemes using FANDAS:** This is one of the most insightful features that FANDAS offers, particularly at the early stages of an NMR study. Tailored amino-acid selective labeling schemes can drastically reduce spectral crowding and allow the user to focus on selected protein regions. This section demonstrates how FANDAS can be used to rapidly assess the effect of changing labeling schemes. To illustrate the output generated for the different labeling schemes, two 2D experiments have been selected:
   (a) **N-Ca intra-residue correlations** that probe protein residues that are both ^{13}C and ^{15}N labeled.
   (b) **N-Co inter-residue correlations** that probe the polarization transfer between the backbone nitrogen atom of the (i)th residue to the carboxyl carbon atom of the (i-1)th residue. This requires the (i)th residue to be at least ^{15}N labeled and the (i-1)th residue to be at least ^{13}C labeled.

Case 1: **Fully Labeled (Default):** *The default option (Fig. 13a) assumes that all the residues are fully labeled. The output for such a labeling scheme would contain all peaks that could possibly occur in each experiment (Fig. 13b).*

Case 2: **^{13}C & ^{15}N Labeled:** *If specific residue types are ^{13}C & ^{15}N labeled, FANDAS retains the ^{13}C & ^{15}N (forward labeled Fig. 14a) chemical shifts or removes them and retains the remaining amino acids (reverse labeled Fig. 14b). When a forward labeling scheme of this nature is used, the spectral crowding is vastly reduced in both N-Co and N-Ca spectra, and sequential correlations can be observed at specific sites as shown in Fig. 14c.*

Case 3: **Using ^{13}C only labeled amino acids and ^{15}N only labeled amino acids:** *There exist labeling schemes where either ^{13}C or ^{15}N amino acids are labeled in combination to probe site selective sequential correlations on proteins* (see, *refs. 3, 4*). *FANDAS treats inputs for this labeling scheme in a similar fashion to the previous case. If residues are ^{13}C or ^{15}N labeled, FANDAS either retains only the ^{13}C or ^{15}N (forward labeled Fig. 15a) chemical shifts or removes them and retains the remaining amino acids (reverse labeled Fig. 15b) as shown in Fig. 15c.*

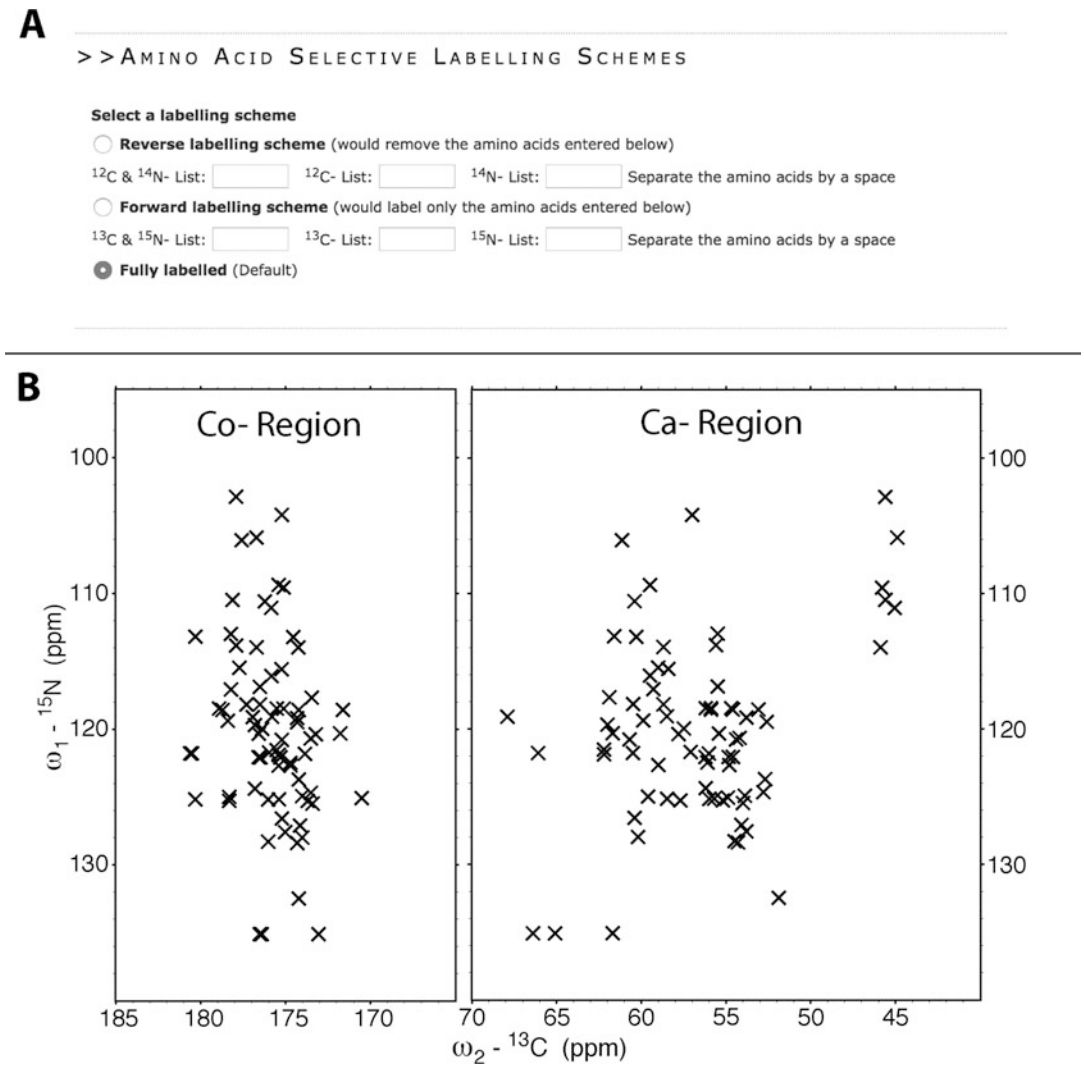

Fig. 13 The default fully labeled option and the resulting predictions for peaks in the N-Co and N-Ca spectra of Ubiquitin

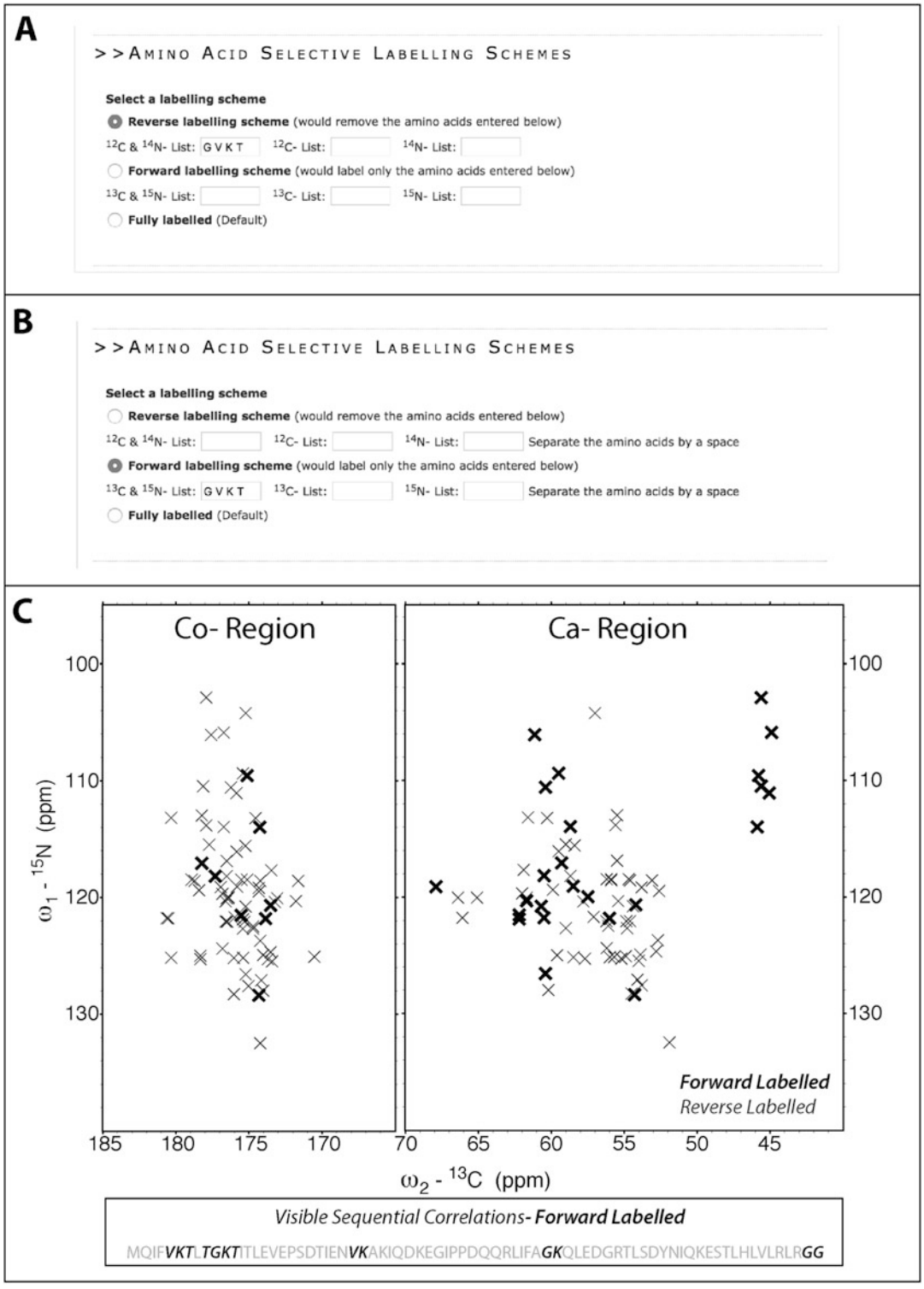

Fig. 14 Incorporating amino acid selective $^{13}C + {}^{15}N$ labeling scheme in FANDAS for glycine, valine, lysine, and threonine residues. As can be seen in the peak prediction, the two labeling schemes complement each other

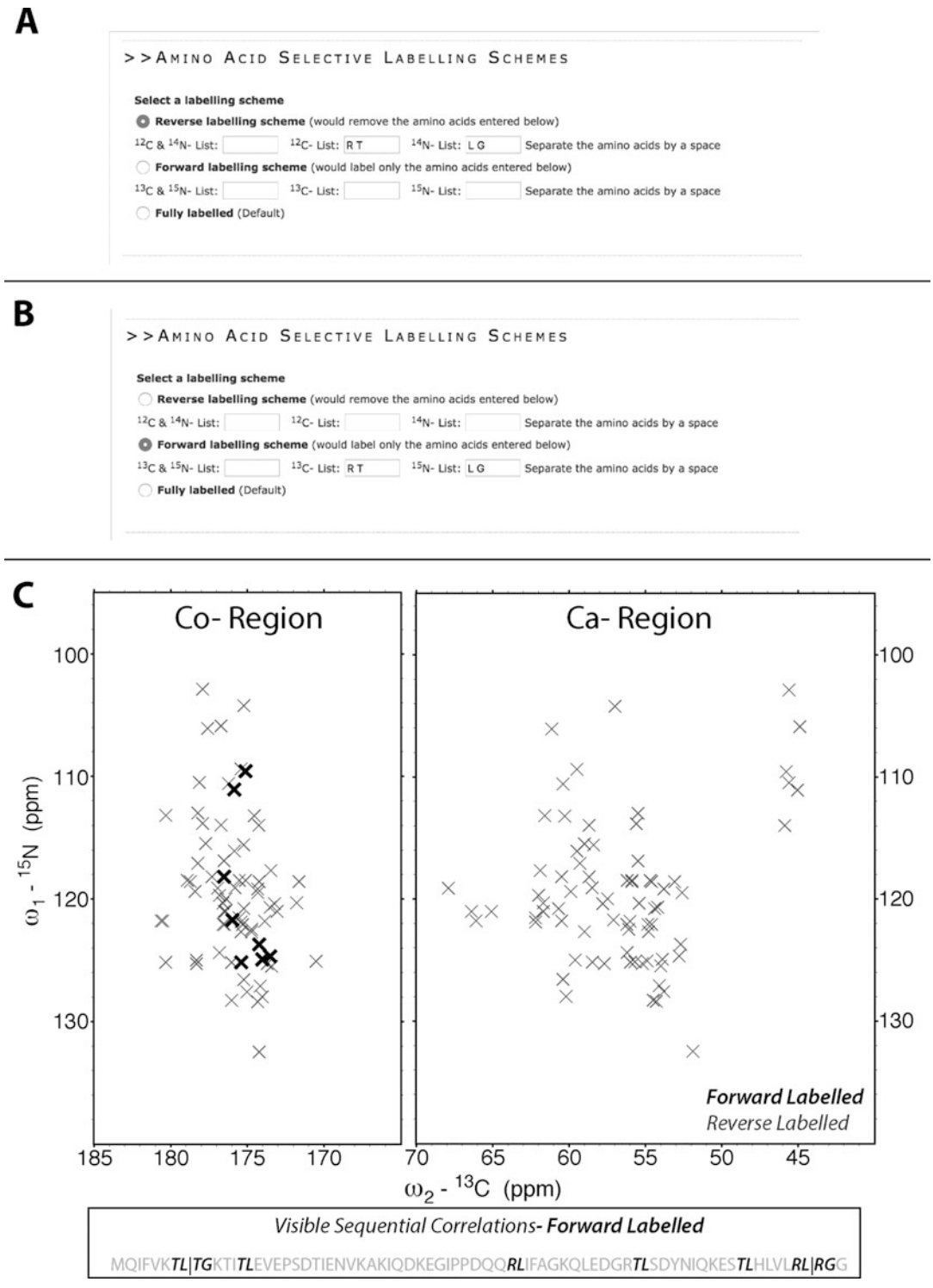

Fig. 15 Incorporating ^{13}C labeled arginine, threonine and ^{15}N labeled leucine, glycine amino acid selective labeling scheme in FANDAS. As can be seen in the peak prediction, sequential correlations can be observed for specific parts of the protein

Acknowledgments

This work was funded in part by the Netherlands Organization for Scientific Research (NWO) (grants 700.26.121 and 700.10.443 to M.B.). The development of the web portal was supported by a European H2020 e-Infrastructure grant West-Life (grant no. 675858 to A.B.). The authors would like to thank Panagiotis Koukos of the Computational Structural Biology Group for his humble assistance in hosting the webserver.

References

1. Renault M, Pawsey S, Bos MP, Koers EJ, Nand D, Tommassen-van Boxtel R, Rosay M, Tommassen J, Maas WE, Baldus M (2012) Solid-state NMR spectroscopy on cellular preparations enhanced by dynamic nuclear polarization. Angew Chem Int Ed Engl 51(12):2998–3001. https://doi.org/10.1002/anie.201105984
2. Renault M, Tommassen-van Boxtel R, Bos MP, Post JA, Tommassen J, Baldus M (2012) Cellular solid-state nuclear magnetic resonance spectroscopy. Proc Natl Acad Sci USA 109(13): 4863–4868 10.1073/pnas.1116478109
3. Kaplan M, Cukkemane A, van Zundert GC, Narasimhan S, Daniels M, Mance D, Waksman G, Bonvin AM, Fronzes R, Folkers GE, Baldus M (2015) Probing a cell-embedded megadalton protein complex by DNP-supported solid-state NMR. Nat Methods 12(7):649–652. https://doi.org/10.1038/nmeth.3406
4. Kaplan M, Narasimhan S, de Heus C, Mance D, van Doorn S, Houben K, Popov-Celeketic D, Damman R, Katrukha EA, Jain P, Geerts WJ, Heck AJ, Folkers GE, Kapitein LC, Lemeer S, van Bergen En Henegouwen PM, Baldus M (2016) EGFR dynamics change during activation in native membranes as revealed by NMR. Cell 167(5):1241–1251. e1211. https://doi.org/10.1016/j.cell.2016.10.038
5. Kaplan M, Pinto C, Houben K, Baldus M (2016) Nuclear magnetic resonance (NMR) applied to membrane-protein complexes. Q Rev Biophys 49:e15. https://doi.org/10.1017/S003358351600010X
6. Gradmann S, Ader C, Heinrich I, Nand D, Dittmann M, Cukkemane A, van Dijk M, Bonvin AM, Engelhard M, Baldus M (2012) Rapid prediction of multi-dimensional NMR data sets. J Biomol NMR 54(4):377–387. https://doi.org/10.1007/s10858-012-9681-y
7. Sinnige T, Weingarth M, Renault M, Baker L, Tommassen J, Baldus M (2014) Solid-state NMR studies of full-length BamA in lipid bilayers suggest limited overall POTRA mobility. J Mol Biol 426(9):2009–2021. https://doi.org/10.1016/j.jmb.2014.02.007
8. Sinnige T, Houben K, Pritisanac I, Renault M, Boelens R, Baldus M (2015) Insight into the conformational stability of membrane-embedded BamA using a combined solution and solid-state NMR approach. J Biomol NMR 61(3–4):321–332. https://doi.org/10.1007/s10858-014-9891-6
9. Baker LA, Daniels M, van der Cruijsen EAW, Folkers GE, Baldus M (2015) Efficient cellular solid-state NMR of membrane proteins by targeted protein labeling. J Biomol NMR 62(2):199–208. https://doi.org/10.1007/s10858-015-9936-5
10. Renault M, Cukkemane A, Baldus M (2010) Solid-state NMR spectroscopy on complex biomolecules. Angew Chem Int Ed Engl 49(45):8346–8357. https://doi.org/10.1002/anie.201002823
11. Pauli J, Baldus M, van Rossum B, de Groot H, Oschkinat H (2001) Backbone and side-chain ^{13}C and ^{15}N signal assignments of the alpha-spectrin SH3 domain by magic angle spinning solid-state NMR at 17.6 Tesla. Chembiochem 2(4):272–281
12. Sinnige T, Daniels M, Baldus M, Weingarth M (2014) Proton clouds to measure long-range contacts between nonexchangeable side chain protons in solid-state NMR. J Am Chem Soc 136(12):4452–4455. https://doi.org/10.1021/ja412870m
13. Mance D, Sinnige T, Kaplan M, Narasimhan S, Daniels M, Houben K, Baldus M, Weingarth M (2015) An Efficient labelling approach to harness backbone and side-chain protons in

^{1}H-detected solid-state NMR spectroscopy. Angew Chem Int Ed Engl 54(52):15799–15803 https://doi.org/10.1002/anie.201509170

14. Goddard TD, Kneller DG SPARKY 3. University of California, San Francisco
15. Wang Y, Jardetzky O (2002) Probability-based protein secondary structure identification using combined NMR chemical-shift data. Protein Sci 11(4):852–861. https://doi.org/10.1110/ps.3180102
16. Joosten RP, te Beek TA, Krieger E, Hekkelman ML, Hooft RW, Schneider R, Sander C, Vriend G (2011) A series of PDB related databases for everyday needs. Nucleic Acids Res 39(Database issue):D411–D419. https://doi.org/10.1093/nar/gkq1105
17. Frishman D, Argos P (1995) Knowledge-based protein secondary structure assignment. Proteins 23(4):566–579. https://doi.org/10.1002/prot.340230412
18. Drozdetskiy A, Cole C, Procter J, Barton GJ (2015) JPred4: a protein secondary structure prediction server. Nucleic Acids Res 43(W1):W389–W394. https://doi.org/10.1093/nar/gkv332
19. Jones DT (1999) Protein secondary structure prediction based on position-specific scoring matrices. J Mol Biol 292(2):195–202. https://doi.org/10.1006/jmbi.1999.3091
20. Han B, Liu Y, Ginzinger SW, Wishart DS (2011) SHIFTX2: significantly improved protein chemical shift prediction. J Biomol NMR 50(1):43–57. https://doi.org/10.1007/s10858-011-9478-4
21. LeMaster DM, Kushlan DM (1996) Dynamical mapping of E. coli thioredoxin via ^{13}C NMR relaxation analysis. J Am Chem Soc 118(39):9255–9264 doi:https://doi.org/10.1021/ja960877r
22. Hong M, Jakes K (1999) Selective and extensive ^{13}C labeling of a membrane protein for solid-state NMR investigations. J Biomol NMR 14(1):71–74
23. Castellani F, van Rossum B, Diehl A, Schubert M, Rehbein K, Oschkinat H (2002) Structure of a protein determined by solid-state magic-angle-spinning NMR spectroscopy. Nature 420(6911):98–102. https://doi.org/10.1038/nature01070
24. Nand D, Cukkemane A, Becker S, Baldus M (2012) Fractional deuteration applied to biomolecular solid-state NMR spectroscopy. J Biomol NMR 52(2):91–101. https://doi.org/10.1007/s10858-011-9585-2
25. Weingarth M, Demco DE, Bodenhausen G, Tekely P (2009) Improved magnetization transfer in solid-state NMR with fast magic angle spinning. Chem Phys Lett 469(4–6):342–348. https://doi.org/10.1016/j.cplett.2008.12.084