Graphical User Interface for Decomposition-Based Analysis of NMR Projection Data

Contents

- Installation and Running
- Files in the archive
  1. ProjTools
  2. ExampleData
  3. Results
- How to use the ‘NMR Projection Data Analysis’ GUI
  1. Merge definitions
  2. Generate Intervals
  3. PRODECOMP
  4. SHABBA
  5. Help
  6. Quit
- Protocol for assignments with PRODECOMP and SHABBA
- Further reading

| GUI for the Analysis of NMR Projection Data |

! Tested under Linux (Fedora); requires Python and the following modules: numpy, Tkinter, matplotlib.

Installation and Running

Unzip the archive in a directory of your choice, go to the **ProjTools** folder and run the program by typing in a terminal:

./ProjAnalys.py

A graphical user interface will open (see the Instructions below how to proceed with the analysis of the projections).

Files in the archive

1. **ProjTools** directory consisting of Graphical User Interface modules:
   1.1. **/merge/MergeDefs.py**: merging files that describe experimental data sets (see 2.2 and 2.6).
   1.2. **/peak2intls/***: generating lists of intervals defining individual decompositions in PRODECOMP.
   1.3. **/prodecomp/***: projection decompositions using PRODECOMP.
   1.4. **/shabba/***: backbone assignment (SHABBA) using PRODECOMP results.
2. **ExampleData** directory
   2.1. **./ProdecompInput/s*.ft2** spectra[1] in nmrPipe format after Fourier transformation (visualize with *nmrDraw*; see http://spin.niddk.nih.gov/NMRPipe).
   2.2. **./ProdecompInput/prodecomp.txt**: File with definitions of input data[2]. It includes a header with descriptions of the data format, nuclei occurring in the direct and indirect dimensions, experimental parameters related to each evolving dimension (spectral widths and center of the range), size of the data and one line per projection with filename and definition. The projection data format is defined with the keyword 'FORMAT' and can be a) 'Bruker': Fourier transformed TopSpin files, namely '2rr'; in this case the data paths are defined up to 'PROCNO', i.e. '1= PATH: /opt/topspin/data/dory/nmr/backbone/101/pdata/1/ \'; b) 'FT2': '*.ft2' format of NMRPipe where a spectrum is stored as a single row of binary data, starting with a header; c) 'CEP': A binary format similar to the one of *.ft2 (NMRpipe) but without header; d) 'ASCII': A general ascii format. Except a) all the data are described with their complete paths.
   2.3. **./ShabbaInput/*c***: Directories with decomposition results obtained for ubiquitin.
   2.4. **./ShabbaInput/CompNewFormat.txt**: File with definition of components in a format[3] similar to the one for the input data (see 2.2).
   2.5. **./ShabbaInput/ubiq_format***: Ascii files with the ubiquitin sequence in formats accepted by SHABBA.
   2.6. **./MergeDefs/prodecomp*.txt:** Files similar to the one described in 2.2 that correspond to two sets of backbone experiments run for Histone G1.
   2.7. **./Peak2Intl/pk*.txt**: Files with HSQC peak lists in NMRPipe (*nmrpipe.txt) and TopSpin (*Topspin.txt) formats.

---

[1] Note: The FNNLS algorithm of PRODECOMP ignores negative input data. Projections with both positive and negative signals should be copied and one copy should be inverted. Both copies should be fed to PRODECOMP. This is actually a way to differentiate Cα from Cβ in certain spectra.

[2]
```
FORMAT= FT2
NUCLEI= HN, N, CO, Cxi-1
SW_ppm= (4.500), 41.106, 16.564, 82.820
SW_hz= (2700.000), 2500.000, 2500.000, 12500.000
O1_ppm= (7.750), 115.523, 173.059, 43.174
SIZE= (517), 121
1= PATH: /home/dory/WORK/PROGRAMS/WebSoftware/ExampleData/ProdecompInput/s101.ft2 \
   DEFINITION: 1, 1, 0, 0
2= PATH: /home/dory/WORK/PROGRAMS/WebSoftware/ExampleData/ProdecompInput/s112.ft2 \
   DEFINITION: 1, 1, 1, 0
3= PATH: home/dory/WORK/PROGRAMS/WebSoftware/ExampleData/ProdecompInput/s113.ft2 \
   DEFINITION: 1, 1, -1, 0
```

[3]
```
NUCLEI= HN, N, CO, Cxi-1
SW_ppm= (4.5), 41.106, 16.564, 82.820
O1_ppm= (7.75), 115.523, 173.059, 43.174
SIZE= (517), 121
1= PATH: /home/dory/WORK/PROGRAMS/WebSoftware/ExampleData/ShabbaInput/25_33c1/ \
   COMPONENT: 0
2= PATH: /home/dory/WORK/PROGRAMS/WebSoftware/ExampleData/ShabbaInput/50_58c1/ \
   COMPONENT: 0
3= PATH: /home/dory/WORK/PROGRAMS/WebSoftware/ExampleData/ShabbaInput/87_109c4/ \
   COMPONENT: 1
```

3. **RESULTS** directory for the above example
   3.1. **prodecomp_merged.txt**: File with results of merging the files from 2.6 above.
   3.2. Plot (**Decomp270_275.png**) with illustrations of the resulting components and shapes.
   3.3. Plot (**comp32.png**) of shapes characteristic for Gly component.
   3.4. Plot (**BBA.png**) showing a backbone assignment of ubiquitin using projection data.
   3.5. **nmrpipe_intls.txt:** File with list of intervals and components (see 3.3 below).
   3.6. Directory '**ProdecompOutput**' with ascii files of resulting components and shapes
   3.7. Directory '**ShabbaOutput**' with files generated during run of SHABBA, namely:
      - **BBAssignment**.txt: results of ubiquitin backbone assignment output from SHABBA
      - ***.log**: files with additional information output during the run
      - **rawcorr.xls**: file with results of correlating shapes of components
      - **08_corr.xls**: file with results of correlations after processing them with a factor of 0.8
4. This manual.

How to use the 'NMR Projection Data Analysis' GUI (*demo examples*[4])

1. **'Merge definitions'** – for merging definitions of multiple projection experiments into a common definition file.

1.1 A new window asking for a definition file is open. Navigate through the directory tree *(e.g. ../ExampleData/MergeDefs/)* to the first file to be merged and select it *(e.g. prodecomp1.txt)*. A pop-up message will ask if you want to add another file, press **'OK'** to continue selecting a file *(e.g. prodecomp2.txt)* to be merged with the previous one(s). Repeat selecting definition files until all are done and press **'Cancel'** in the message window that pops-up after the last selection. A new folder tree for output of the merged file will open; choose a location and enter the name of the file to be saved *(e.g. ../Results/prodecomp_merged.txt)*. The output file can be examined with a text editor. !Note that for correct merging of definition files they must have consistent parameters, namely 'FORMAT', 'NUCLEI' definitions (e.g. equivalent names refer to equivalent nuclei types), acquisition parameters and 'SIZE'.

2. **'Generate Intervals' –** for generating a list of intervals and components for automatic running of PRODECOMP. It uses as input a 2D-peak list and gives as output a list where each entry describes one interval corresponding to one peak.

2.1 When pressed, it will open a new window for the input peak list of a ^{15}N-HSQC-type (at present only TopSpin and NMRPipe formats are tested). Press **'Open peak list'** and choose the file with the peak list to be loaded (*e.g. ExampleData/Peak2Intl/pk101nmrpipe.txt)*. Edit the file with a text editor and check the numbers of the columns described in the GUI as 'Parameters from the peak list'. Fill in the column numbers starting from 0 *(e.g. 5,6,17 for HN and N shifts and Intensity columns in pk101nmrpipe.txt, respectively).*

2.2 The rest of the entries labelled as 'Parameters for interval calculation' have default values that normally don't need to be changed. For each peak in the peak list they define as follows: a) choice of the number of points giving the size of decomposition interval; b) parameter for confining the peaks that will be considered when calculating corresponding number of components for this interval; c) the function that is used to model HN shapes (Gaussian is assumed as default one); d) Half Width at Half Maximum (HWHM) that is used together with c) to calculate peaks' intensities within the decomposition interval; e) minimum intensity for accounting a neighbouring peak as a component (a treshold of 30% of the intensity of the central peak is suggested). Press **'Output results'** to save the list with obtained intervals for subsequent decompositions *(e.g. ../Results/nmrpipe_intls.txt)*.

3. 'PRODECOMP' – for decomposition of 2D projections recorded for high-dimensional NMR data.

3.1 Press **'Load Definition file'** in the window that opens and select the file with data descriptions *(e.g. prodecomp.txt in ../ExampleData/ProdecompInput/ or MergeOut.txt from ../Results, see 1. above. !Note that the 'prodecomp.txt' file has to be modified to describe the data paths correctly according to the local settings).*

3.2 For each data file a line giving its location and definition will be printed in the Listbox widget. Use *Shift* and *Ctrl* keys or *Shift* and arrows to make a data selection in a way similar to selecting files in a windows folder. Press **'Take Selection'** and **'Change Selection'** buttons to input or cancel the data selection, respectively.

3.3 Now **'INPUT->PRODECOMP->OUTPUT'** button is active and when pressed a window will pop-up with a question 'Do you want to open FILE with parameters?'. If you have a file with intervals and components that was

[4] Italic font in this manual indicates input used in the demo of the program.

output from **'Generate Intervals'** (see 2. above) you can press **'OK'** and load it *(e.g. ../Results/nmrpipe_intls.txt)*. If you don't have such a file, press **'Cancel'**. A new window will show decomposition parameters as a table with columns referring to intervals ('Start, points' and 'End, points') and components ('Number of components'). A single row with values of '-1' will appear if no file with intervals is loaded. The last column labelled as 'Select comp No' is used in further analysis (see 4. below) and its values are not considered by PRODECOMP. In addition 'Regularisation factor' and 'Number of iterations' are given and need usually not be changed from their defaults. Check if the decomposition parameters are complete and start the decomposition.

3.4 There are two ways to run decompositions: a) as individual runs using **'Run ONE'** function that opens an entry widget for typing the interval number to be run (1st column of the parameters' table); or b) as a set of sequential runs (listed in the table) that will be executed one after the other when **'Run/Save ALL'** is pressed. The decomposition of selected sets of projections according to the input parameters yields as output components described by one-dimensional shapes. The results after each run will be automatically saved in a directory 'ProdecompOutput' opened where the program is started. Information about the running decompositions (the number of currently run interval and the time each iteration takes) is printed in the terminal where the program was started. A message 'Decomposition is done!' follows individual runs while a message 'Finished!' would indicate the end of a set of runs. *For testing, run an interval along HN with points 270-275 and 3 components that is the 10th interval in nmrpipe_intls.txt, i.e. press 'Run ONE' and type '9' for interval No in the entry widget. You should see messages printed on the terminal while the decomposition is running. Results of this decompositions are given in the **RESULTS** directory as a plot (Decomp270_275.png; Thr9 cyan, Ser65 red, Gln62 green), as well as the stored ascii in 'ProdecompOutput'.*

3.5 If you are using option 'a)' in 3.4 press **'Save CURRENT'** to store the decomposition results for further analysis. The shapes are written in a default directory called 'ProdecompOutput' in ascii format; in the latter the name of each file contains information for (a) direct ('fdir') or indirect ('f') dimension, (b) the interval in points along the direct dimension used in the decomposition, (c) 's' and 'c' followed by the number of the shape and component, respectively *(e.g. press 'Save CURRENT' and the decomposition results from the run above are output as ascii files in ../ProjTools/ProdecompOutput/.../)*. Press **'Plot ONE'** to visualize selected decomposition results (!Note that only saved results can be plotted). It will open an entry widget where to specify the number of the row in the table (see the column labelled 'Interval No') with corresponding parameters. Press 'Select' and a new window will show parameters for the nuclei present in the decomposition (as read from the input to PRODECOMP, see 3.1. above) and colours for plotting the individual components. Check the nuclei parameters and modify with proper values if needed (!Note that if zeros are filled in, the corresponding shape plots will be in points). Press **'Plot ALL'** or **'Plot ONE-BY-ONE'** to show the shapes for all components in one or individual graphs, respectively (in ppm if information is given, in points otherwise). The plotting window offers some useful options like zooming or saving the plots. *A plot of the results of the example decomposition above are given in ../Results/ (*.png; Thr9 red, Ser65 cyan, Gln62 blue).*

3.6 For changing the selection of input projections use the **'Change Selection'** and **'Take Selection'** buttons in PRODECOMP window. To change the parameters for the decompositions edit corresponding values in the table widget. To add or remove a complete line with parameters press **'Add ONE'** or **'Remove ONE'**, respectively (a pop-up window asks for a number of interval (entry) in the table). Then PRODECOMP can be run with another set of projections and/or parameters using **'Run ONE'** button (!Note that if you don't press **'Save CURRENT'** the results of the previously run decomposition will be lost). Also if you press **'Run/Save ALL'** the results will be stored in the same directory ('ProdecompOutput'), i.e. decomposition results with the same interval and number of components will be overwritten.

3.7 Each entry in the last column labelled as 'Select, comp No' is filled manually with a number of a component that refers to a ^{15}N-HSQC peak used to define the corresponding decomposition interval (see 2. above). To select unambiguously a component per interval, e.g. to avoid repeating components for overlapped decomposition intervals when filling the entries, plot each decomposition result (row in the table). Then look in the plot for a component that has HN- and N- shifts as the ones describing the peak in the HSQC spectra (and given in the peak list), and find the component number in the legend. Then check if this component is resolved according to 1.1 in Protocol for assignments with PRODECOMP and SHABBA (!Note that the number of peaks in the shapes may indicate if a component describes noise or side-chains in the HSQC-spectra. In this case remove the interval from the table using **'Remove ONE'**). To help selecting the correct component you may use **'Plot ONE-BY-ONE'** function producing a separate plot of its shapes or press **'Correlate TWO'** to examine a table with correlations (0-100%) between components of two intervals indicating which of them (values > 50%) most likely describe the same spin system. Press **'Save LIST with ALL'** to store the table with all parameters in a text file that can be either used as a backup file of the table with intervals and components to be loaded again (to continue decomposition analysis, plot decomposition results and etc.) or as an input for SHABBA (!Note that the latter would require filled in ALL values in the last column, i.e. NO values of '-1' are allowed).

4. **'SHABBA' –** for analysis of 'PRODECOMP' results and backbone assignment.

4.1 Press **'Load selected PRODECOMP results'** in the new window and select the file describing the decomposition results, i.e. the output from **'Save LIST with ALL** in 3.7 *(e.g. ../ExampleData/ShabbaInput/CompNewFormat.txt*

!Note that the latter has to be modified to describe the data paths correctly according to the local settings). You first need to confirm that the proper nuclei (Cα/β and Hα/β of residues *i-1* and *i*) are selected in the loaded file; change the corresponding entries if necessary. A default directory 'ShabbaOutput' for automatic output of intermediate and final results from SHABBA analysis will open where the program is initiated. Information about the executed steps and their results will be stored in a file 'shabba.log' that can be opened with a text editor. Press **'Check for Gly'** to get a list of the components that satisfy criteria for being Gly (see 2.1 in Protocol for assignments with PRODECOMP and SHABBA). Press **'Plot Comp No'** for a visual check of a component that can be entered on the right *(e.g. type in 53)*. A new table with parameters (see **'Plot ONE'** in 3.5) will pop-up, press **'Plot'** and the C- and H- shapes of the component will be shown, check if they belong to a Gly according to the criteria above and close the plotting window *(e.g. see comp32.png in /Results that is later assigned to 35Gly, !Note the intensities and number of peaks in the shapes)*. If needed modify the corresponding entry in the last column of the table ('Correct compNo') to '1' if the visual check confirms that the component is a Gly or '0' otherwise. Check in this way all listed components and then press **'Confirm Gly components'** to correct the C- and H- shapes of the ONLY those confirmed as Gly (with '1' in 'Correct compNo'). This will close the current window and disable **'Check for Gly'** option, i.e. it can be executed only once.

4.2 Press **'Correlate shapes'** to calculate the correlations among the C- and H- shapes of ALL input components. As a result the current window will expand opening new entries for analysis of the shapes. First you are asked to check the peak picking parameters. Enter a 'Factor' parameter *(e.g. 0.8)* and press **'Process'** to analyse the correlations using this factor (see 2.2 in Protocol for assignments with PRODECOMP and SHABBA). Chains of sequentially connected components will be printed in the Listbox widget and shabba.log file (see above). Type different values for 'Factor' parameter and press **'Process'** to see how this parameter influences the output chains.

4.3 Below the Listbox widget you can find the parameters used for peak picking the C- shapes, namely 'C-shift(pt)' - giving the relative shift (in points) that is allowed for a signal present in different data sets *(e.g. Cα/Cβ signals from the two example data sets showed a maximum difference of 1pt)*; 'Tresh(%)' – peaks with intensity below this value are considered as noise and are excluded when comparing shifts in *i* and *i-1* shapes. These parameters have default values that might not need to be modified. !Note that the acquisition parameters for C-nuclei are read from the definition file (see the file in 3.1 above) and used for points-to-ppm conversion, so their values have to be correct.

4.4 Press **'Position chains in the sequence'** and a pop-up window will ask you to load a file with protein sequence *(e.g. ../ExampleData/ubiq_format1.txt)*. There are 5 different ascii formats that are supported so far for the input protein sequence *(see 'ubiq_format*.txt' in ../ExampleData)*. Once the sequence is loaded a new window will show a plot of the suggested backbone assignment according to 2.4 in Protocol for assignments with PRODECOMP and SHABBA *(e.g. see BBA.png)*.

4.5 A parameter 'H-shift(pt)' similar to the one in 4.3 describes the maximum relative shift for H-shapes *(e.g. Hα/Hβ showed no difference in the two data sets and a default value of 0 is proposed)*. Additional parameter that describe the number of expected signals depending on nuclei types are given in a file 'ShabbaPar.txt' and used for selecting peaks from the shapes. Press **'Peak pick ALL shapes'** to continue with peak picking the rest of the shapes and collect the chemical shifts for the already assigned components. A widget for filtering out the noise from the H-nuclei shifts using statistics of shapes' intensities will pop-up. Press **'Histogram plot'** to see the statistical distribution as a histogram. Examine the plot that will show a threshold value for peak intensities and close it. This will update the single entry in the Statistics window, which also can be manually changed. Press **'Take treshold'** and a message box will indicate the place where the complete backbone assignment is stored. Press **'Ok'** to terminate the program and go to the directory with the assignment results ('ShabbaOutput'). !Note that running SHABBA will automatically save intermediate results of correlations and log files in the same directory and they will be overwritten if the program is run again.'

5. **'Help' –** for opening this manual.

6. **'Quit' –** for exiting this program.

Protocol for assignments with PRODECOMP and SHABBA

1. PRODECOMP: Decomposition of projections

   For decompositions of projections to a set of components a series of intervals along the direct dimension are chosen based on a 15N-HSQC spectrum. PRODECOMP can then be applied to each of the intervals using the following run-time parameters: the selection of an interval along the direct dimension in data points, the number of resonances (number of components) expected in this interval, a regularization factor and the number of iterations to be performed. Default values for the latter two entries need normally not be changed. After running PRODECOMP one obtains shapes describing each of the components within the selected interval. The number of peaks found in the shapes, typically one or a few depending on the nuclei involved, is an indication of a successful decomposition.

When decomposing different intervals along the direct dimension the only parameters that need to be changed are the interval borders and the number of components (obtained from inspection of a 15N-HSQC spectrum).

Once the decompositions along the direct dimension are completed, the resulting components and the corresponding shapes are examined visually: Only components that exhibit clear peaks or that are not identical to others (due to overlapping intervals) are selected for further analysis (i.e. SHABBA).

2. SHABBA: Processing of the decomposition results (All steps of SHABBA described below are automated and take a few seconds of processor time).

   The following steps are implemented in the software library SHABBA. As in many triple resonance spectra, the Cα resonances of glycine have the same phase as resonances involving Cβ in other residues. Consequently, glycine components exhibit a peak in the shape for Cβ while the shape for Cα contains only noise. The same holds for the shapes with Hα and Hβ. Observation in a shape of more than three local intensity maxima that exceed 50% of the global maximum is a clear sign for the presence of only noise in this shape; in an α-shape this indicates the presence of a glycine. Alternatively, for glycine shapes the maximal intensity observed in the α-shapes is distinctly lower than the one in the β-shapes. The ratios of the shape maxima, max(Cα)/max(Cβ) and max(Hα)/max(Hβ), are typically one to two orders of magnitude smaller in glycine components compared to other residues. Following the automatic identification of glycine components, the α-shapes of these components were replaced by the β-shapes, and the β-shapes were set to zero.

   The identification of glycine components is followed by an automatic calculation of correlations between all pairs of components: From the first component in a pair the shapes for Cα and Cβ are added up, as are the shapes for Hα and Hβ. These are then compared to the Cα/β(i-1) and Hα/β(i-1) shapes of the second component. The resulting correlations for all component pairs are collected in a table, which is then analyzed in order to obtain chains of connected components. This step uses three simple rules that depend on a single parameter called 'factor': 1) All diagonal entries are removed: no spin system is sequential to itself. (2) For pairs of entries that are symmetric with respect to the diagonal, the smaller is removed if it is less than the symmetric entry multiplied by 'factor': if spin system *i* precedes spin system *j*, then *j* cannot precede *i*. (3) For each row (column), entries smaller than the row (column) maximum multiplied by 'factor' are eliminated: each entry can have only one successor (predecessor). The parameter called 'factor' is typically set to a value between 0.8 and 1.0.

   An automated tool was designed to peak pick the shapes using couple of parameters that have default values and normally do not need to be modified by the user (see 4.3). The HN shape of each component, which contains only a small number of points according to the interval selection, consists of a single peak that was fitted using one Gaussian function. All other shapes extend over the entire spectral width along the indirect dimension, and many of them contain more than one signal. The program analyses each shape and peak picks all local maxima. Peak picking parameters define the number of strongest peaks to be output while the rest are considered as noise.

   For each chain of components resulting from the above analysis of the correlation table, the Cα and Cβ chemical shifts were plotted as a function of the sequence of components, providing a specific pattern. Next, the BioMagResBank (http://www.bmrb.wisc.edu/ref_info/statsel.htm) is consulted for statistical chemical shifts values for each type of amino acid residue. These values were plotted versus the protein sequence, providing patterns in a similar fashion as for the components. The patterns from the components were then matched to the statistical expected ones for the sequence by "gliding" of the former along the latter. For each alignment an rmsd match among the corresponding chemical shifts was determined. A penalty term of 20 ppm was added whenever a glycine was compared to a non-glycine or whenever a proline is aligned to a component that is not at the beginning of a chain. The alignment with the lowest rmsd provides the final assignment of each chain of components.

For further reading please visit:

http://www.lundberg.gu.se/nmr/

http://www.extend-nmr.eu/prodecomp.html

and look at the References:

D. Malmodin and M. Billeter, *J. Am. Chem. Soc.*, 2005, **127**, 13486.
D. Malmodin and M. Billeter, *Magn. Reson. Chem.*, 2006, **44**, S185.
D. K. Staykova, J. Fredriksson, W. Bermel, and M. Billeter, *J. Biomol. NMR*, 2008, **42**, 87.
D. K. Staykova, J. Fredriksson, and M. Billeter, *Bioinformatics*, 2008, **24**, 2258.