Assignment on-line Documentation

the Assignment Module Version 1.22

15 feb 1999

SET-UP

SET-UP

Presentation

This module implements a set of very simple assignment tools, which may however proof to be useful. It is almost completely written in the Gifa macro language (the command find_dist is based on a Perl script), and as such can be fully adapted to your needs. Right now, it is principally aimed toward protein and peptide assignment. Extending it to oligonucleotides and sugars is probably a simple matter of extending the basic residues data-bases. However, I have no idea whether it can be used to help in the assignment process of other kind of organic molecules or not.
You will not find here any fancy tools nor automatic assignment, the only help provided here consists in a set of tools permitting to visualise several spectra at the same time, to add notes to peaks, to draw lines to help for visual align search, and to store the information in several data-bases, one for assigned peaks, one for spins and one for spin systems (consisting simply in a set of spins). For the moment, this module works only for 2D data-sets and has little support for heteronuclear 3D. However, due to the Gifa versability (calling UNIX from Gifa, creating/reading files, etc..) it is quite easy to adapt this canvas to your proper need, for instance calling from within this set-up your favorite automatic assignment tools.

File set-up

A complete assignment is kept in a special directory, called a project. The project, which may reside anywhere on the disk, holds several files and directories used for storing informations, it may also contain any file that the user may wish to keep.
In the project directory, you will find typically two files : The file *parameters* is a macro which is executed when selecting the project. It contains all the definitions and some basic environment variables. The file *zoom_window* contains the zoom window coordinates for the 'multi-zoom' tool (see the ZOOM command in the documentation).
Five obliged directories reside also in the project.
The *db* directory holds the data-bases in dbm format as described below, the primary sequence is also

found in this directory in a file called *primary*. The format is as follows : one residue per line, coded in one-letter code.
The *spectra* and *PDB* directories hold respectively all the spectra and PDB files associated with the project. Typically, for space optimisation, links to actual experiment files will be stored here rather than the complete file.
The *processing* and *constraint* directories hold respectively the intensity curves and the constraint files generated by the *Integration* and *Dynamic* menu (see below).

Assignment Information Structure

Assignment information is stored as sets of peaks, spins and spin systems, with the following structure :

- each experiment is associated to a peak data base.
- each peak in a given experiment can be assigned to 2 spins (one for each spectral dimension). Along with the spin indexes, the coordinates of the peak are stored, as well as its amplitude.
- each spin is characterized by a chemical shift, a name and is associated to a spin system (if assigned)
- each spin system has an index in the primary sequence (when assigned), and a set of spins.
- for peak and spin entries a free field is also available for user notes (not for spin systems).
- all yet unknown piece of information can be set as unknown ("unk"), except chemical shifts.
- The peak, spin and spin system information is stored in three independent dbm data bases.

data-bases

Except for the primary structure file presented above, there are 3 kind of data-base files in the *db* directory, there are all in dbm format and each of them is thus composed of two files **.pag* and **.dir* which should not be modified directly. The dbm format is a generic UNIX format for flat data bases. For instance, these files can very easily be accessed with the *perl* language. Nevertheless, the dbm format is not fully compatible among all UNIX platforms, and you should be carefull with that (specially Linux users).
The *name_of_experiment.pag* and *name_of_experiment.dir* files hold the peak data base for a given experiment. An entry in the peak data-base stores all the pertinent information for a given peak. Peak entries can be created by copying them from the peak-picker, or during the assignment process. With a peak entry is stored two pointers to the spin databases, pointing to the parent spins of this peak.
Spin.pag and *Spin.dir* is the spin base, a spin is stored as a chemical shift, a name, and the spin-system to which it belongs.
Finally *Spin_sys.pag* and *Spin_sys.dir* is the spin-system base, for each entry, the spin-system type, the index in the primary sequence, as well as the list of the spins is stored.
In all the data-bases, entries are referred to by a numerical id that ranges from 1 to the highest value. The value of the highest id used is stored in a special entry indexed as "LARGEST". However, due to the very nature of the dbm format used for the file, (and of the associative arrays used internally in Gifa) there is no need for the id to be contiguous. So, if an entry is deleted, the numbering of the other entries is unaffected.

build list

The **build list** is the main tool for progressing in the assignment work. The idea is to make a list of all the peaks within a spin-system (in the TOCSY sense). Once this list is complete, it is possible to *promote* the list to a new spin-system which is then entered in the data-base.
The build list is managed with a set of tools found in the *graph tool* menu. The marker tool permits to detected peak alignments, and to create a spin for each alignment. The list can be listed or displayed directly on screen. And of course, the list can be promoted to a spin-system.

Previous Next Contents

MENUS DESCRIPTION

When entering the assignment mode, Gifa will set up an assignment environment with the basic menus, the Peak menus and 4 additional menus that give access to all the commands needed to performed spectral assignment. The macro `env_att.g` actually sets-up every thing for assignment..

Project menu

This menu permits to create or select a project, and more generally to realize all the operations global to the project.

Help (short)

Produces a short help of the assignement module as a 'recipe' to use it. The complete assignment help contained in the HTML documentation can be called from this short help.

New Project

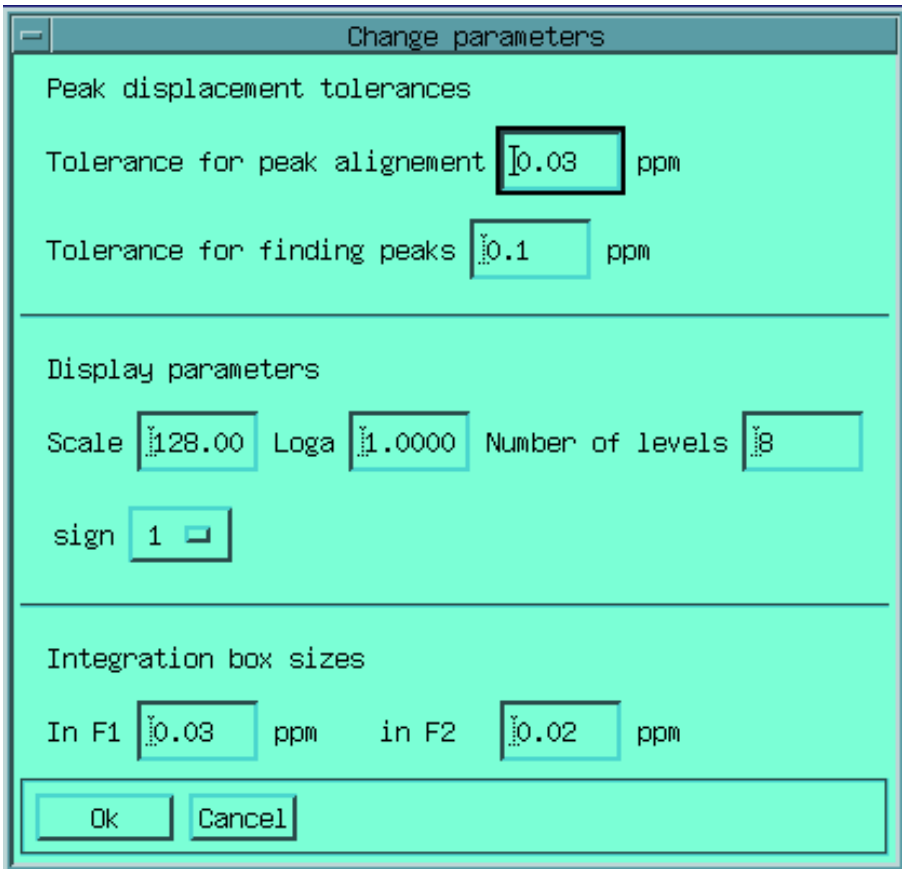
To create a new project. It will create a directory with all the empty data-bases. You will be also prompted for the primary sequence of the studied protein. The primary sequence can be given, either literally in 1 letter code, from a file (1 or 3 letter code), or from a PDB file.

Select Project

Permit to select any previously created project. Only one project can be used at a time.
When selecting a project, you have the choice, in the dialog box, to either create a backup of the current state of the project (simply a tar file), to recover from a previous backup (thus deleting the current state) or not to do any action.
After selection, the number of assigned systems is displayed on screen, then the **multi zoom** tool and the **File Selector** tool are opened.

Change param

A set of parameters is stored with the project, these parameters can be changed from here.



You can define :

- Peak alignment tolerance, used when searching for peaks alignment or spin/peak alignment
- Tolerance for finding peaks is the maximum distance allowed when searching for a peak by coordinates.
- Standard Display parameters, that will be set when entering the project : `SCALE` (display height as defined with the zoom box) `LOGA` (contour spacing) number of `LEVEL` and `SIGN` of display (1 : only positive, -1 : only negative, 0 : both).
- Integration box sizes define the size of the box, used either for display and for integration.

Save data-bases

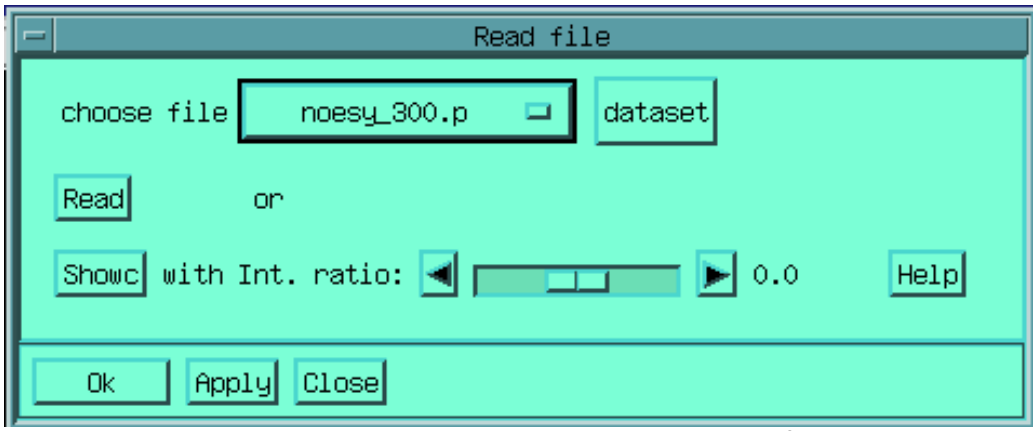
The assignment data-bases are permanently kept on file, however, in case of a program crash, the last modifications may be lost. Clicking here secures the very last entries.

Make backup

Copy the current backup file backup.tar to backup.tar.old, and store the current dbm files of the project in the archive file backup.tar, produced by the command tar.

File Selector

Start the `File Selector` box:



All the spectra which are used in the project should be accessed from here. A spectrum can be loaded in memory, in the same time the associated peak data-base is loaded. A spectrum can also be only showed (see the SHOWC command in the documentation), displaying it on screen, but not loading it in memory, in which case the intensity ratio cursor permits to adapt the intensity display to the current scale.
When loading a file in-memory, the related peak data-base is opened, and the peak data base of the previously displayed spectrum is closed. if the file loaded in-memory has no associated peak data base, a new one is created.
Note that you can also use the super2d tool (in the display menu) to display several spectra superimposed, by default, in super2d, you will be prompt for using the last dataset used with the Showc button.

Add spectra

From here, you can add a spectrum to the list of the currently used spectra. "Adding" a spectrum consists in either copying or linking it into the dedicated directory(see above, **File set-up**). Linking sets a UNIX *soft-link* which stores the address of the file only, thus permitting an important gain in disk space.

Add PDB file

Permit to add a PDB file to the list of the currently used PDB files, in a way analogous to the way used for spectra (see the previous command 'Add spectra'). This PDB file list is used in the 'Find distance' command (see the *utilities* menu below).

Stat on Primary seq

Produce statistics on the amino acids contained in the primary sequence.

Copy pk to peak db

This one permits to copy the content of the peak table (obtained with the Peak picking tool) to the peak data-base. Peak will be there but without assignment of course. This permits to load a first set of peaks, for instance the finger print region, from which the assignment work can proceed.
This command can be issued several time and at any moment during the assignment work, thus adding peaks into the assignment data-base.

Rem unassigned peaks

Remove the unassigned peaks from the peak data-base. The unassigned peaks are those for which no spin has been assigned in F1 or F2.

Copy the db from another data-set

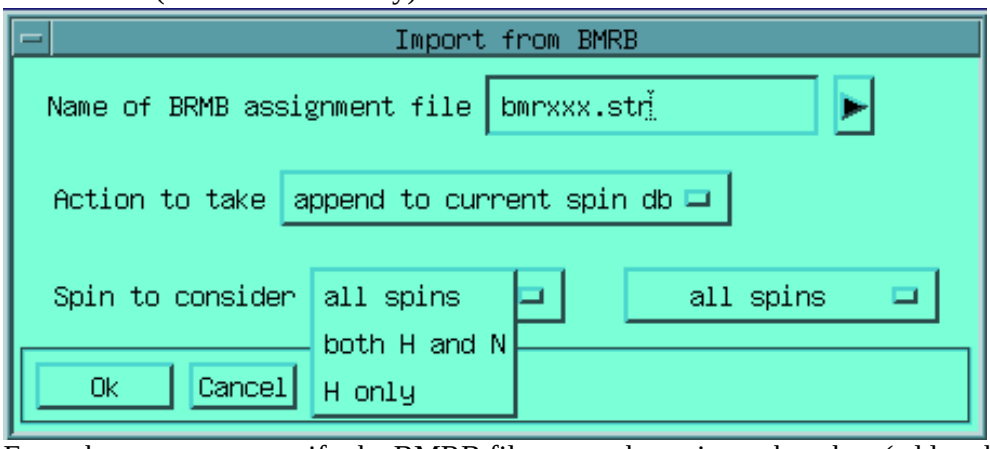
Copy the peak data-base of a data-set of the spectra list to the peak data-base of another data-set of the list. The command asks for the permission to erase an already existing data-base.

Merge the peak table into the db

Merge the peak table with the peak database of the assignment projet.
For each peak of the peak table, the command looks at the closest element of the database within the click tolerance and change the spectral coordinates and intensity of this element to the peak coordinates and intensity. The assignment information of the dbm element is unchanged. If no database element in found within the peak tolerance, the peak is added to the data-base as an unassigned element.

Import db from BMRB file

This tool can be used when starting an assignment project for a protein for which an assignment information is already available. The information can be either available in BioMagResBank data format (seehttp://www.bmrb.wisc.edu/) or in plain ascii format (see next menu entry)



From there, you can specify the BMRB file name, the action to be taken (add to the current db or start a new one), which spins should be considered : all, ^{1}H and ^{15}N or only ^{1}H; and finally, if a check on spin validity (checking if the spin name agrees with the names defined in the topology file) should be performed.
From the file, only the spins and the spin systems will be defined. The peak db will not be filled but will had to be filled independently (see below).

Import db from ascii file

Same as above, but in free ascii format. So far, only 2 file formats are handled, both are one line per spin system, with the following formats :
format 1 :

```
primary_number res_name 15N_chem_shift Ha_chem_shift Hb_chem_shift, etc...
```

format 2 :

```
res_name primary_number 15N_chem_shift Ha_chem_shift Hb_chem_shift, etc...
```

where etc.. represent the list of chemical shift given along the residue side chain.
You can adapt this macro to fit your specific needs by modifying the `ascii2spin` macro

Create peak db from current spin system definitions

This command takes the current spin-system and spin definitions, and create assignment peaks for each possible peaks for the current experiement.

... Integration module

This command will add the Integration menu which permits to quantify precisely one, or a set of experiments. Useful for quantifying a NOESY experiement, building build-up curve, and generating the constraints file for structure generation.

... Relaxation module

This command will add the Dynamic menu which permits to handle set of 2D experiment for relaxation measurements. Useful to realize the integration of a set of T_1 or T_2 heteronuclear or homonuclear experiments, and then to extract the relaxation parameters, with precise error estimate. Support for measuring heteronuclear J coupling is also available.

Quit Assignment

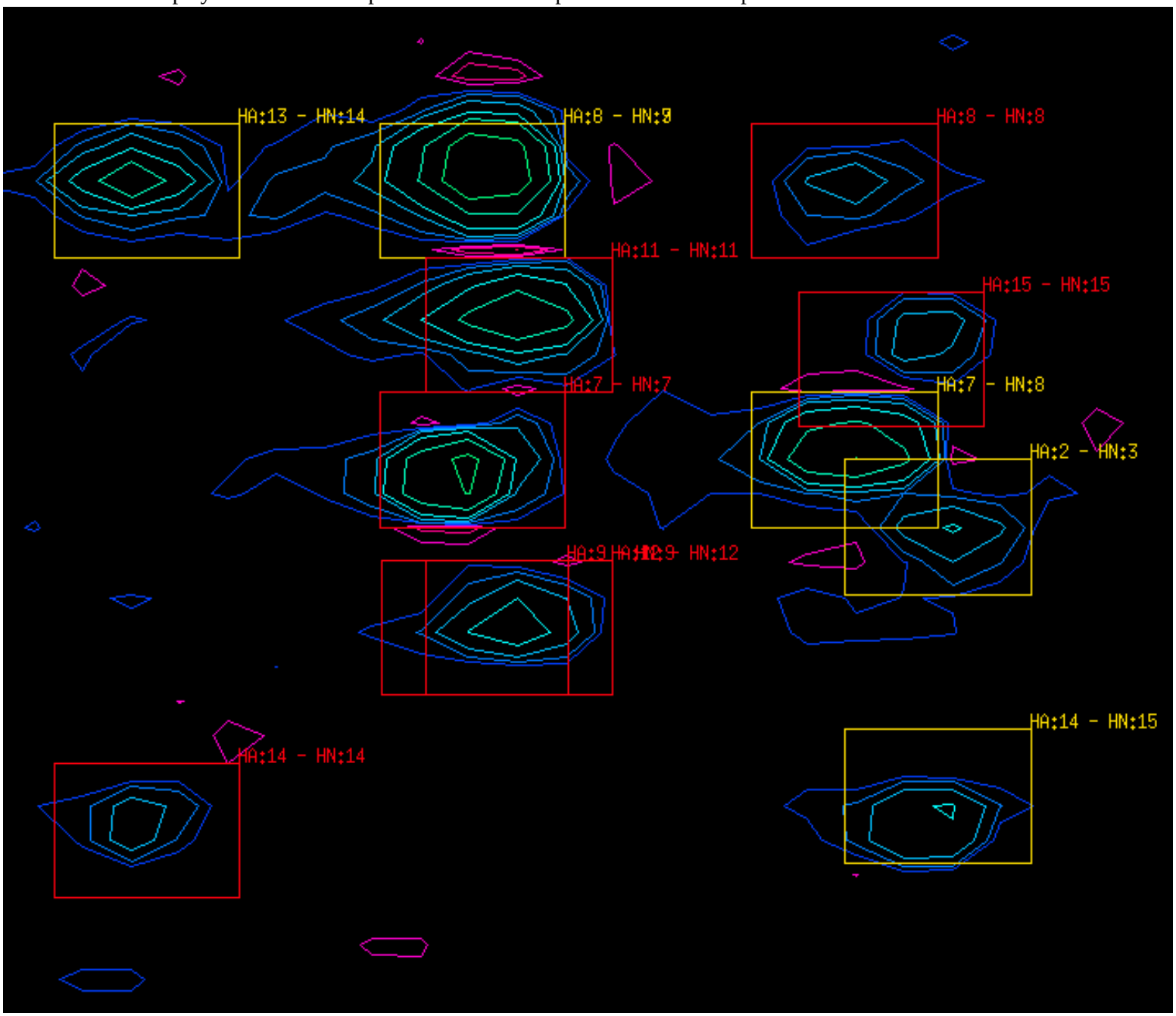
Use this entry if you want to quit the assignment module and restore a normal set-up

data-bases menu

This menu permits to graphically display and to modify the assignment data-bases (peak, spin or spin system data-bases)..

Show database

This command displays on the current spectral zoom all the peaks in the current peak data base.



in this exemple red sqaures are intra-residue peaks and yellow peaks are inter-residue peaks. Unassigned peaks are displayed as crosses.

Show all peaks

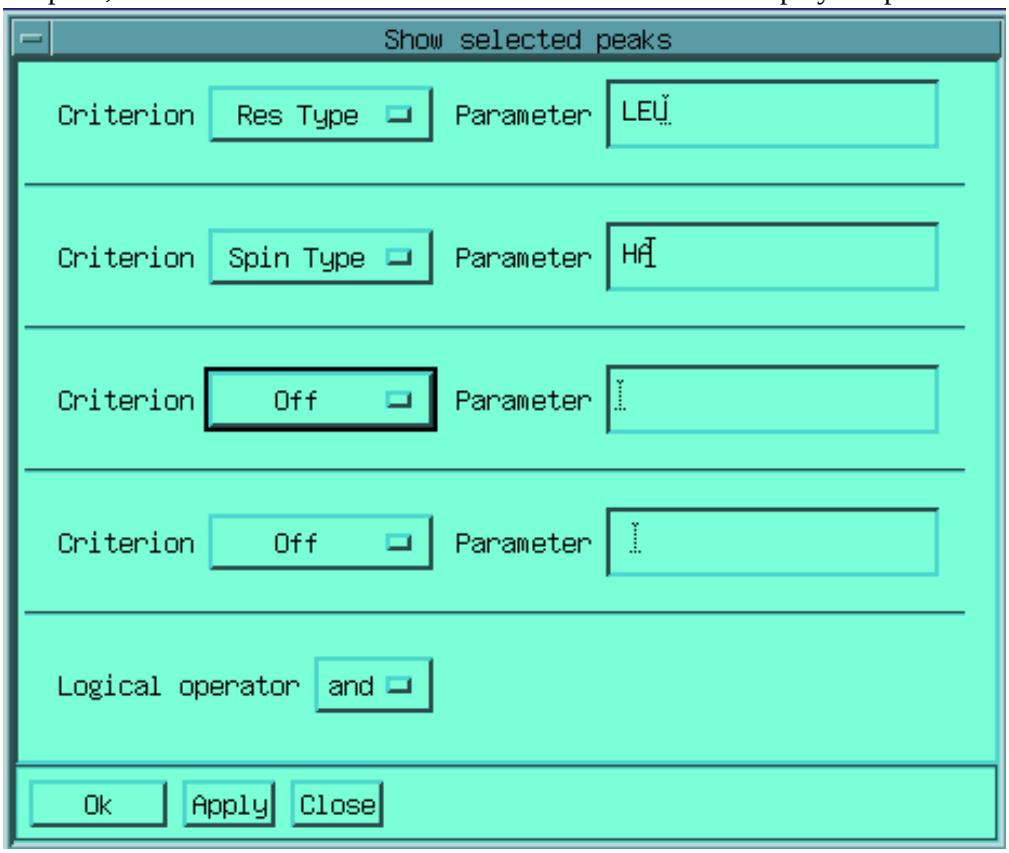
Is equivalent to the previous command except that only the square will be drawn, thus making a display easier to read in the case of dense spectra. No color coding will be used either.

Show unassigned peaks

This command displays on the current spectral zoom all the unassigned peaks in the current peak database. The unassigned peaks are those for which the F1 or F2 spin has not been assigned.

Show selected peaks

This command displays on the current spectral zoom all the peaks verifying different criteria given by the user. Peaks can be displayed depending on the value of the peak note, the spins notes, the residue numbers or types, the spin types and the peak intensity (maximum value or threshold). The criteria for peak and spin notes and spin types are tested as substrings of the peak.corresponding parameters.
The criteria can be applied according to a logical parameter: 'and' means that all the given criteria must be verified to display the peak, 'or' means that one verified criterion is sufficient to display the peak.



In the exemple above, only peaks were the H α of a Leucine is involved will be displayed

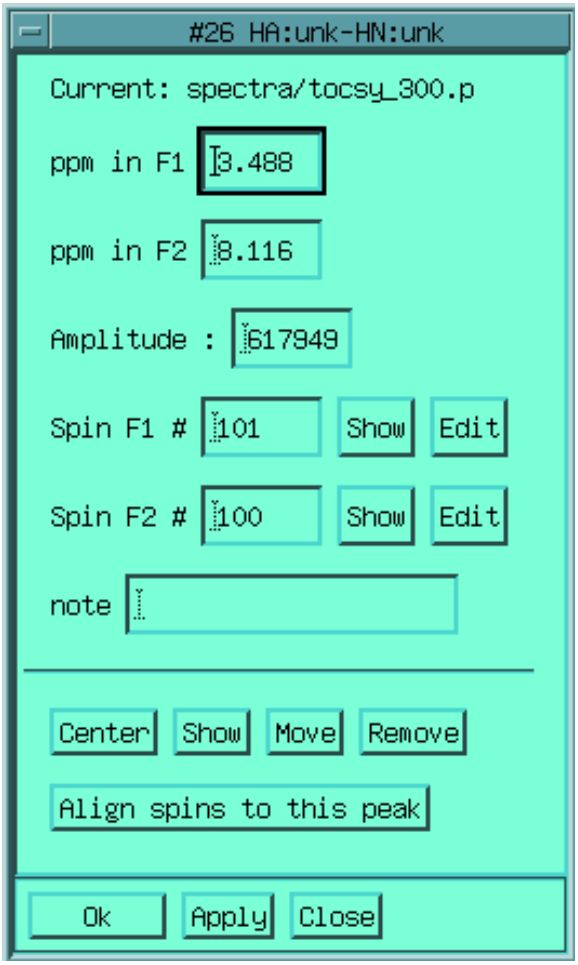
Find a peak

After selecting this command, the program will wait until you click on a peak on the spectrum, and will high-light the selected peak as well as print its id in the terminal screen.

Edit a peak

As the previous command, but an edition box for the selected peak will show up. You directly see the content of the assignment data-base, and can actually modify it. If the peak is already assigned, you will be able to see/edit the corresponding

spins.



You can also center the zoom window on that peak, move the peak (click on the new peak location), and align the associated spins to that peak (remember that chemical shifts are handled independantly in the peak db and in the peak db).

Create new peak

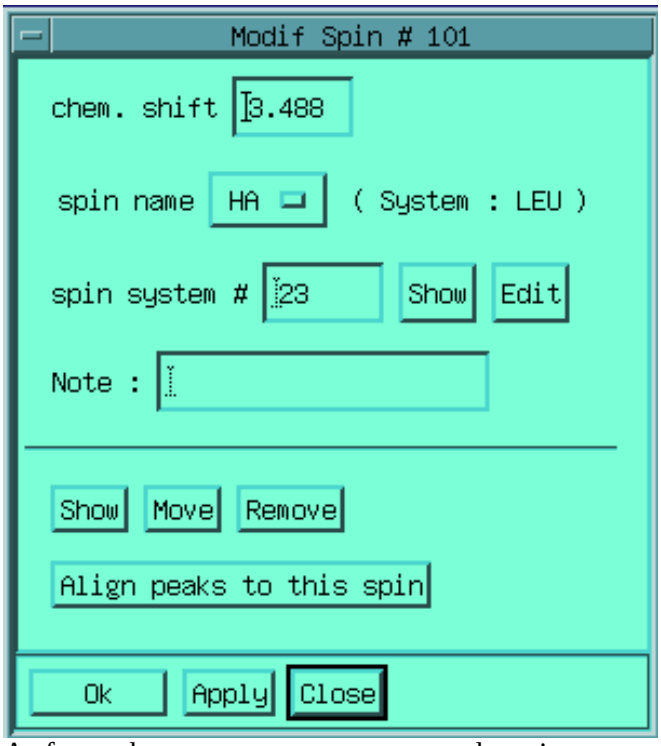
This command wait until you click on the spectral window, and creates a new entry in the peak data-base. If a peak already exists within the distance tolerance for the mouse clicking, the program asks to user a confirmation for creating the peak. The new peak is then edited.

Find a spin

After selecting this command, the program will wait until you click on the spectrum, and will propose spins close to the click points, indicated along which axis (F1/F2) they are found.

Edit a spin

As the previous command, but an edition box for the selected spin will show up. You directly see the content of the assignment data-base, and can actually modify it. Related peaks and spin system can also be edited.



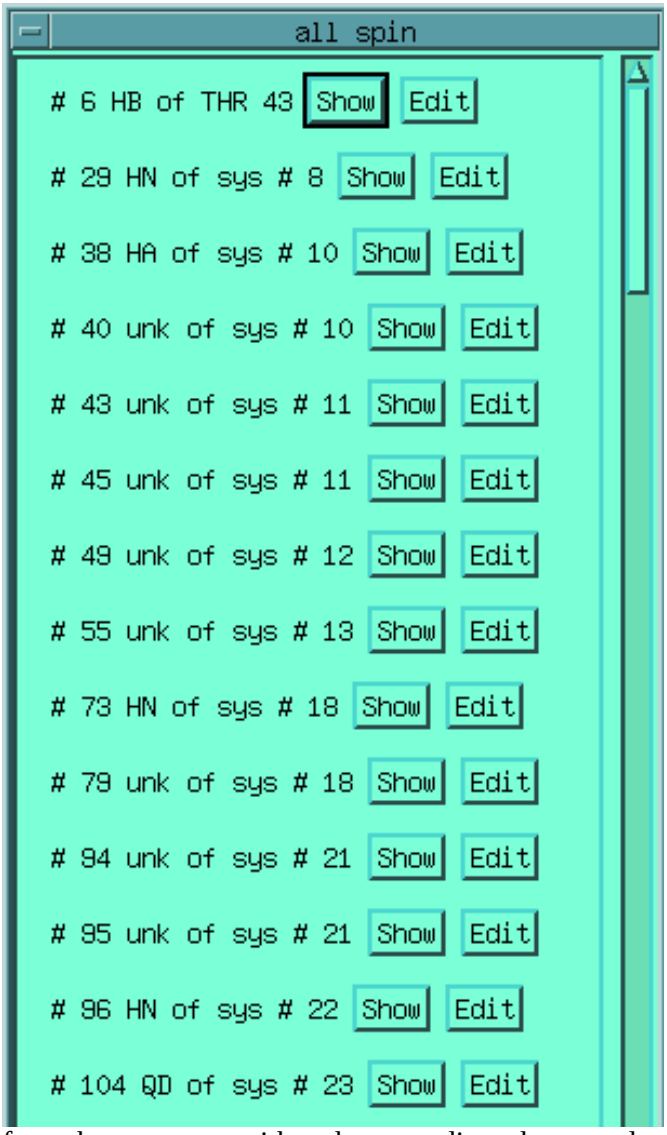
As for peak, you can move or remove the spin, you can also align all the peak in the db to the current chemical shift of this spin.

Create new spin

After selecting this command, the program will ask you along which axis (F1/F2) you want to create a new spin. Then, it will wait until you click on the spectral window, and creates a new entry in the spin data-base. The new spin is then edited.

List all spins

This command produces a clickable list of all the spins in the data-base.



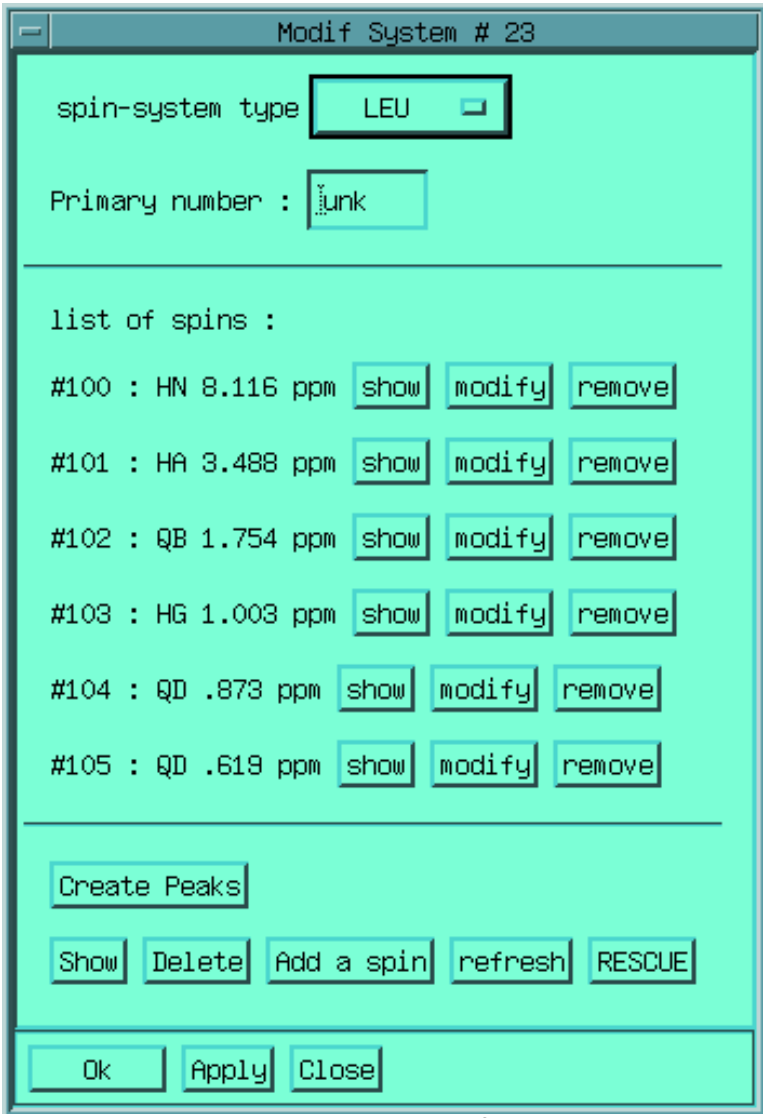
from there, you can either show or edit each currently defined spins.

Find a syst

After selecting this command, the program will wait until you click on a peak on the spectrum, and will high-light the related spin-system, if it exists.

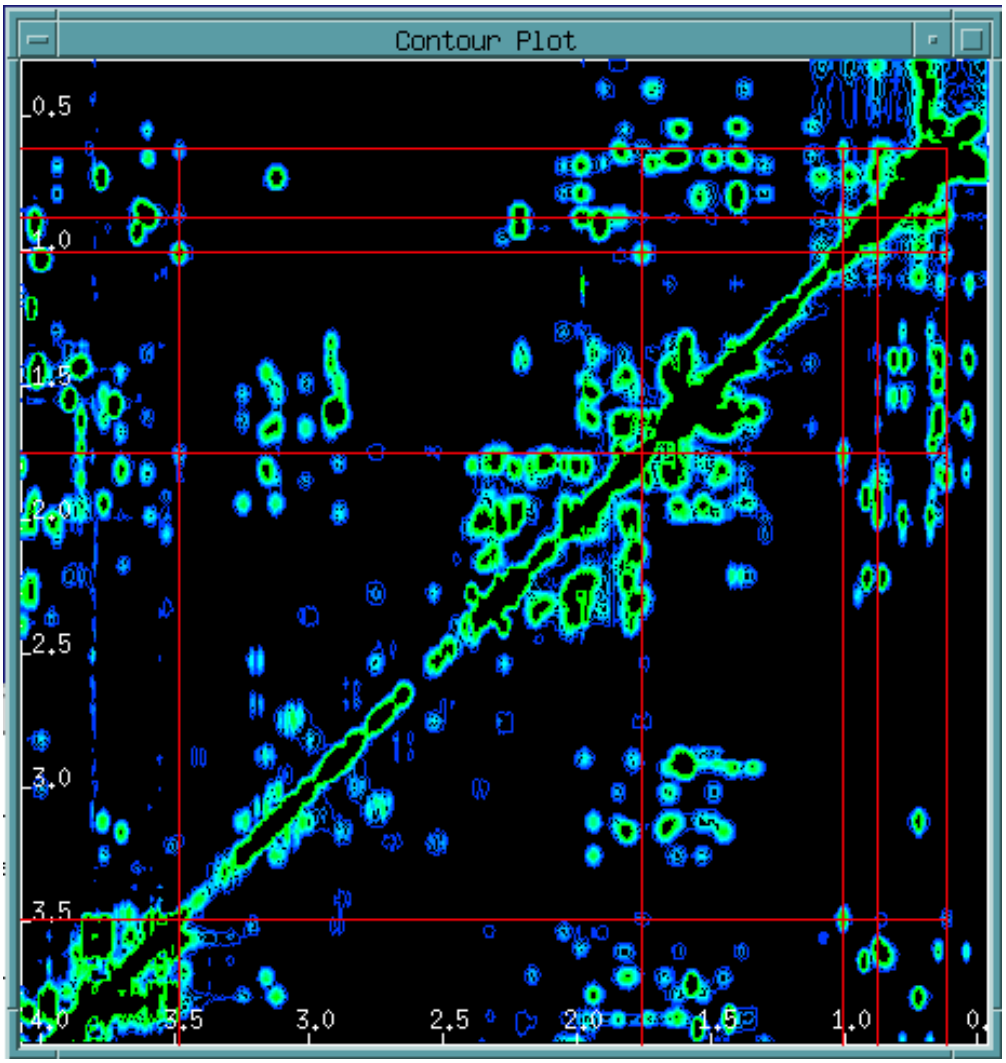
Edit a syst

As the previous command, but an edition box for the selected spin system will show up. You directly see the content of the assignment data-base, and can actually modify it. Related peaks and spins can also be edited.



When the spin system type has been defined by the user, the list of possible spin names is restricted to the list defined in the topology data-base. The `Create Peaks` button, will create peaks at all the peaks in the peak db that could exist at the spin crossing.

The `Show` button will graphically display the spin-system :



The button `refresh` close the opened system formbox, and open a new one containing the last modifications.
The `RESCUE` button will trigger the Rescue computation, which uses a neural network technology for searching assignment possibilities from the set of defined chemical shift. Seehttp://www.infobiosud.univ-montp1.fr/SERVEUR/RESCUE/rescue.html for details. The result in the present case would be:

```
Chemical Shift Input:  8.116:3.488:1.754:1.003:.873:.619
R.E.S.C.U.E. Software Result:
 --First Switch--           --Second Switch--
GROUP    :   SCORE      |    RESIDU    :   SCORE
-----------------------------------------------
IL       :   0.9302     |    I         :   1.0000
A        :   0.0000     |    L         :   0.0000
G        :   0.0000     |
P        :   0.0000     |
T        :   0.0000     |
V        :   0.1361     |
KR       :   0.0000     |
AMX      :   0.0000     |
AMPTX    :   0.0000     |
S        :   0.0000     |
Result       GROUP : IL          RESIDU : I
Reliability  (%)   : 61                  : 45
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
```

Indicating that the spin system is tentatively a Isoleucine or a Leucine, but with a not so high probability, due to the non-zero response for Valine.
Rescue uses a a perl script located in `/usr/local/gifa/com/rescue` . You will need perl v5.0 to use Rescue.
When you close the formbox (with the `Ok` button) , the program check that the topology of the spin system is correct. IUPAC

notation for spin names is enforced.

List all systs

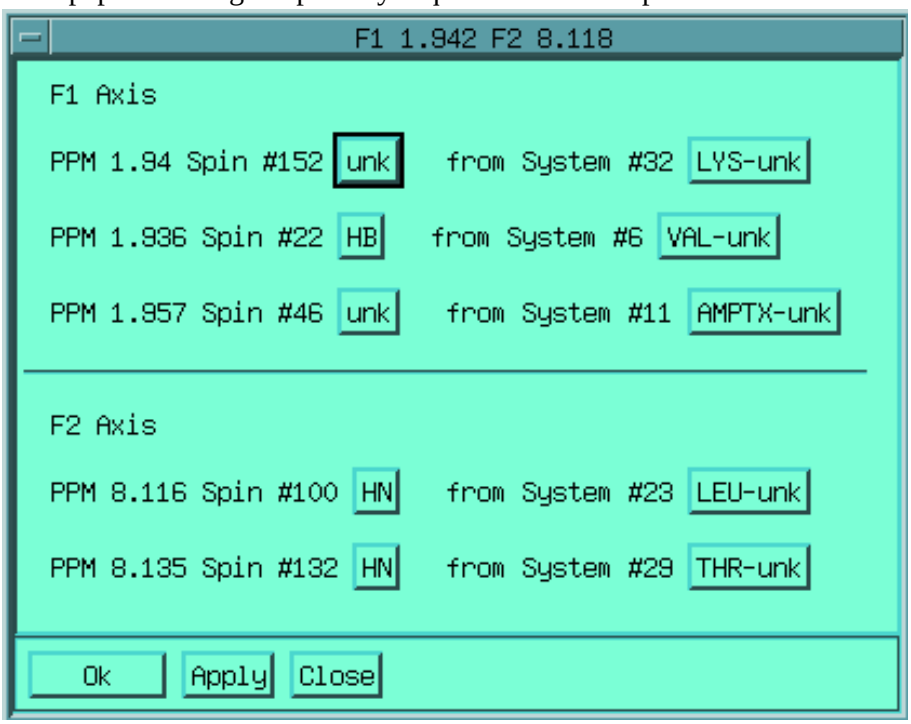
This command produces a clickable list of all the spin-systems in the data-base.

Stat on systs

This command produces statistics of all the spin-systems in the data-base.

Search spins

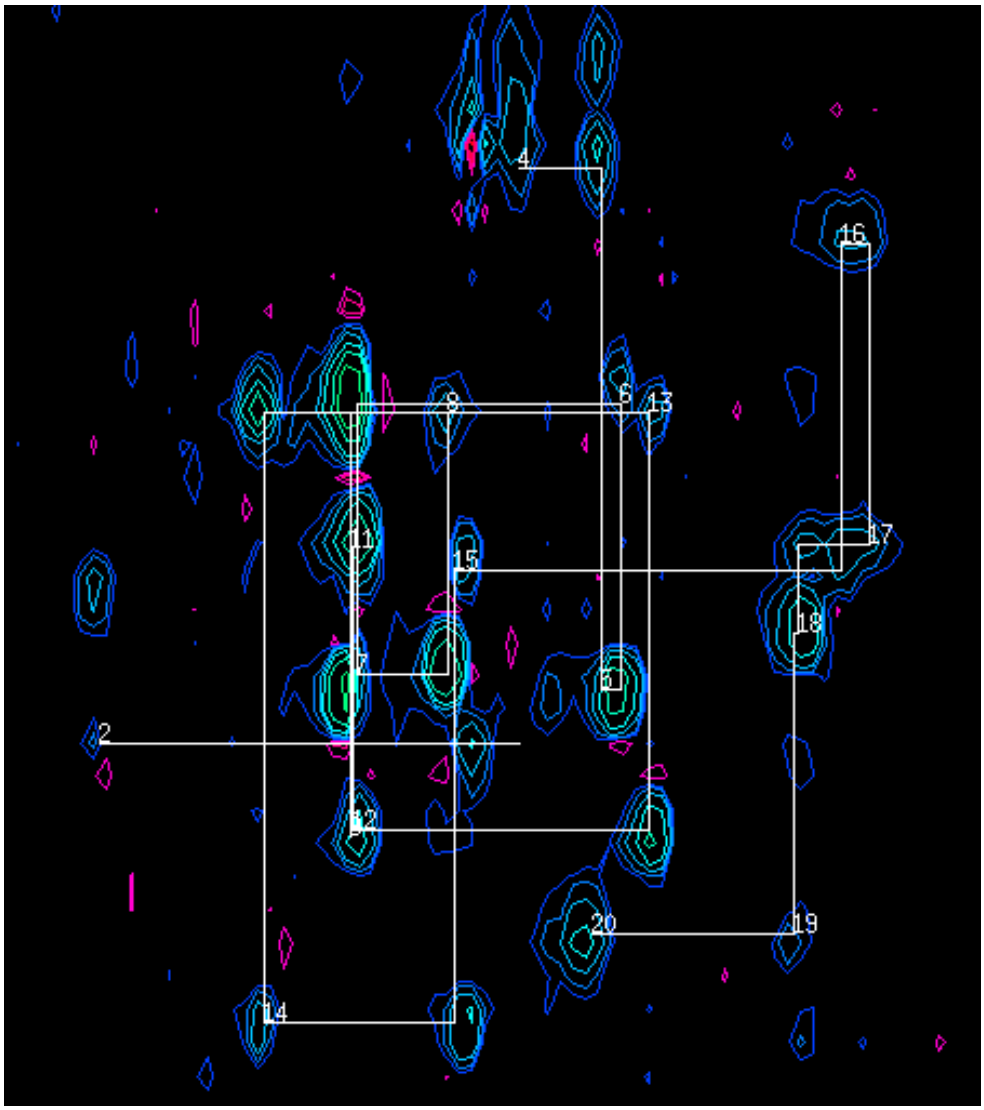
Permits to click on the data-set, and search the spins located within the align distance tolerance, in F1 and F2 axes. The spins names and information are shown in a formbox, which allows to display and edit them individually. Possible sequential match for dipeptides along the primary sequence are also reported.



Draw noesy walk

Draw noesy walk

This command draws the homonuclear NOESY walk in the HN-HN and the HN-HA regions.



Show Primary Seq

This command builds a form box, with one line per residue in the primary sequence of the molecule under study. Each assigned residue is associated to a button showing the corresponding spin-system on screen, and to another button allowing to edit it.

graph tools menu

This menu contains all the basic graphics tool which are used to detect peak alignment and build new spin systems.

Choose base color

Most display command use a contrasting color (see SCOLOR in documentation). This utility permit to define which color will be used.

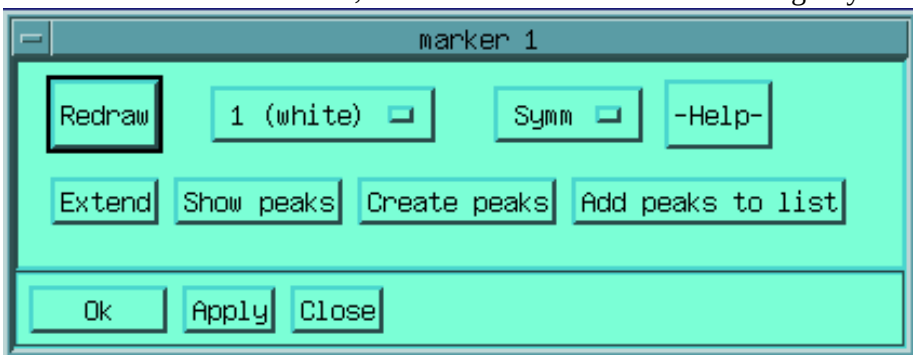
Point

This is equivalent to the standard point macro : you can click on the current spectrum, and the coordinates of the clicked point are printed, and a cross is drawn at the click point location. You exit the point command by clicking on the third button of the

mouse.

Marker

This is the main tool for detecting peak alignment and for building the build list. When activating the command, you are prompted to click on each spectral location that you want to put in the marker. When finished, click on the third button of the mouse. A box is then created, which will remain on screen as long as you do not close it purposely.



From the marker box you can choose to redraw all the horizontal and vertical lines connecting the selected points. You can choose to have diagonal-symmetric locations considered or not, and choose the color. You can extend the current marker, clicking on new locations on the spectrum, this will create a new marker containing the new locations as well as the previous ones. You can show all the peaks in the data-base lying at the intersection of an horizontal and vertical lines, as well as create missing peaks. Finally, you can add all the peaks detected at the intersections to the build list.

Reset build-list

This command empties the build-list, which contains peaks.

Print build-list

This command simply prints the content build-list, by showing the id of the peaks in the build-list..

Show build-list

This will display on the spectral window, the peaks in the actual build-list.

Add peak to list

Permits to add a peak to the build-list.

Promote to spin syst

This is the command that will use the actual build list, and create a new spin-system. First, alignment are detected within the peaks in the build list, and spins are detected for each spectral coordinates. If needed, new spins are created. Then a new spin system is created.
Finally, the tool permitting to edit the spin-system is opened from which you can modify the spin-system parameters and edit each individual spins.

utilities menu

This menu contains various utility commands, which allow to chck the integrity of the current assignment state, its consistency with structural information obtained from a PDB file. These conmands also permit to output assignment information.

Check data bases

Checks all current assignment data-bases for integrity. Because of a bug in the foreach code, this command may have to be applied more than once in case of wrong entries. In particular, this command checks the consistency between the chemical shifts of the spins assigned to a peak and the chemical shifts of this peak. It also checks for internal coherence.

Check topology of spin systs

Check that all spins of each spin system are defined in the current topology database. The topology data base file is located in the /usr/local/gifa/macro/att directory. For each spin system type, it contains the allowed list of spin names, including the individual hydrogen names and generic names for superposed geminal hydrogens. This file can be modified by the user to fit other topologies (see macro programming).

Low-level editor

Creates a formbox which permits direct editing to each spin, syst and peak, according to their index in the data-base.

Find distances

This command permits to click on the spectrum, and high-light the closest peak in the database. Then, it creates a dialog box to look for the distances between hydrogens involved in the selected correlation. The PDB files you can scan are given by the PDB file list.
You can select atoms by their exact names or by a substring of their names (as an example, looking for the 'HB' substring will permit to find all the hydrogens HB1, HB2, HB3,...). If you want to search among all the residues or all the atoms, put the sign '.' into the corresponding field of the dialog box.
This command is based on a perl script calcdst.pl located in /usr/local/gifa/com/att directory.

Peak Listing

Lists all the peak database entries to a file. Here is un example of listing file:

```
# Project : /d1bis/people/terez/ranab
# Experiment : spectra/proc.05
# F1    F2   Spin1  Spin2   Amplitude Peak# (Note)
 0.282  4.855   unk   unk      476142  151
 0.900  1.266 HD-20 QG-20    2597383  443
 0.900  4.223 HD-20 HA-20     682930  442
 0.923  0.912 HG3-7 HG3-7  356122080  430
 0.923  1.267 HG3-7 HG2-7    3268115  369
 0.923  1.531 HG3-7 HG1-7    2074078  336
 0.923  1.886 HG3-7  HB-7    1735731  306
 0.923  4.213  HD-7  HA-7     919385  178
```

Spin Listing

Lists all the spin database entries to a file. Here is un example of listing file:

```
# Project : /d1bis/people/terez/ranab
#PPM   Name  System Spin#  (Note)
1.014    HG  15       63
1.083    HG   8       37
1.255   HG1   7       30 super with ILE 13
1.266   HG1  20       79
1.473    HB  11       48
1.664    HB  21       83 super with HG LEU 5
1.728   HB2  22       88 super with HD1 Lys 19 and 18
1.771    HG  16       67
```

System Listing

Lists all the spin system database entries to a file. Here is un example of listing file:

```
# Project : /d1bis/people/terez/ranab
2 LEU
------ 8.784 HN 10
------ 4.456 HA 10
------ 1.654 QB 10  super with HG Leu 2
------ .946 QD 10
3 GLY
------ 8.441 HN 17
------ 4.017 QA 17
4 GLY
------ 8.399 HN 18
------ 4.051 HA1 18
------ 3.979 HA2 18
5 LEU
------ 8.324 HN 21
------ 4.418 HA 21
------ 1.664 QB 21 super with HG LEU 5
------ .969 QD 21
```

Assignment Listing

Lists all the assignment entries to a file. Here is an example of listing file:

```
# Project : /d1bis/people/terez/ranab
1 F assigned to : 1 Arom-Phe 103 104
------ 7.458 3H 26
------ 7.348 2H 26
1 F assigned to : 1 PHE 101 119
------ 4.344 HA 25
------ 3.261 QB 25
2 L assigned to : 2 LEU 42 43 44 45
------ 8.784 HN 10
------ 4.456 HA 10
------ 1.654 QB 10  super with HG Leu 2
------ .946 QD 10
3 G assigned to : 3 GLY 68 69
------ 8.441 HN 17
------ 4.017 QA 17
```

Plot database

Plots on file the labels of the elements of peak current database, which are located in the current zoom window.

Plot one lable

Plots on file only one peak label, usefull for annotating only a few peaks on the plot.

Recalibrate current Experiment

This command permits to recalibrate a given experiment (change the definition of the 0 ppm point) while maintaining the

position of the peaks. Indeed, peaks are handled in ppm, so recalibrating the experiemnt without taking care of the peaks would end-up with peaks db entries not located anymore on the corresponding spectral location.

Build strip plot from 3D HSQC

This command permits to build a composite 2D strip-plot file from a 3D HSQC experiment, much in the way the related command in the 3D module would do. However here, the strip plot is build from the current assignment databse as would be defined for a 2D HSQC. Strips will then be ordered in the pseudo F2 axis in primary sequence order.
Plots on file only one peak label, usefull for annotating only a few peaks on the plot.

Previous Next Contents

INTEGRATION

Integration menu

This menu allows the calculation of peak intensites, and the output of distance constraint files or peak intensity curve.

Help (short)

Produces a short help of the Integration manu as a 'recipe' to use this menu. The complete assignment help contained in the HTML documentation can be called from this short help.

Single data-set integration tool

Performs integration of all the peaks of the opened dbm assignment table (it is not possible to perform this integration only on the current zoom window, because of a bug in dbm described in CAVEAT).

The calulated volumes are put into the dbm dbm amplitude Different integration methods can be used: 'Max in box' gives the data-set value at the peak maximum (located within the integration box), 'sumrec' and 'amoeba' make use of the commands SUMREC and INTEG respectively.

Write build-up constraints file

From the calibration peak set, the program determines by a least-square method a proportionality coefficient between intensity values and the inverse sixth power of the distance. this coefficient is then used to associate a distance to each element of the peak data-base. Then, using the uncertainty on the distance provided the user, it write a constraint file with distance upper and lower bounds. The file format can be XPLOR or DYANA format.

Write qualitative constraints file

From the calibration peak set, the program determines by a least-square method a proportionality coefficient between intensity values and the inverse sixth power of the distance. this coefficient is then used to associate a distance to each element of the peak data-base. Then, using distance interval and upper and lower bounds given for each interval by the user, it write a constraint file with distance upper and lower bounds. The file format can be XPLOR or DYANA format.

Copy db to a peak file

Copy the current assignment data-base to a peak file that can be read with PKREAD, and create a lookup database giving the relation between the peak index in the assignment data-base and in the peak file. The peak file name is basename.atr and the lookup file names are: basename.hash.dir and basename.hash.pag.

Read peak file

Reads a peak file with the command PKREAD.

PkList

Lists the peak table (see command PKLIST).

ShowPeaks

Displays the peaks located in the current zoom window (see command SHOWPEAKS).

Eval noise

Evaluates the noise on the data-sets using a zoom window given by the user.

Integ

Performs the peak integration according to the noise level and the peak table, by calculating for each peak an amoeba (see command INTEG).

Show Amoeba

Show on the disp2d window the amoeba located in the current zoom window (see command SHOW)..

Modify Amoeba

After selecting this command, the program will ask you to click on a peak, and will create a formbox, which permits to interactively and graphically modify the peak amoeba by selecting one-by-one pixels. Two possibilities are available: "add" adds the selected pixel to the peak amoeba, "erase" removes the selected pixel from the peak amoeba.
If the flag is set-up to 'add' and the pixel is in another peak amoeba, the program ask to user to take a decision.

Read Peaks/Amoeba

Reads a peak file and an amoeba file according to a basename given by the user. The peak file name is: basename.pek, and the amoeba file name is: basename.amb

Save Peaks/Amoeba

Saves the current peak table and the current amoeba to a peak file and to an amoeba file according to a basename given by the user. The peak file name is: basename.pek, and the amoeba file name is: basename.amb. If the peaks/amoeba files already exist with the same basename, the user is asked to remove them.

Multiple integration tool

Permits the integration of peaks along a series of experiments according to an assignment database, an amoeba file and the project list of spectra. For each peak, the integral values are written in an independent file, located in the *processing* directory. Two formats are possible: free ascii format or Tela macro format. The user can select in the formbox the spectra he want to be used in the integration.
To perform the multiple integration of peaks, the amoeba basename file should be the same than the basename of the lookup data-base between the indexes of peaks in assignment data-base and peak file (see the command **Copy db to a peak file** above).

Show integration curve

Permits the display of the curve obtained with the multiple integration utility, by clicking on the corresponding peak.

Peak movie

After selecting this command, the program ask to you to click on a peak, and then creates a formbox, which allows to display successively the selected peak on the list of project spectra. The user can select in the list the spectra to be displayed.

Previous Next Contents

DYNAMIC

Dynamic menu

This menu is optionnal, it can be added from the process menu. It contains a set of tools aimed toward the integration and the processing of a set of 2D containing a relaxation experiment. For instance as would be obtained from a heteronuclear relaxation experiment. It assumes that all the experiment are located is spectra of the current projet, and that one of these experiment has a completely assigned peak db. All commands should be applied from this assigned experiment, as the assignement db will be used.
Some commands (one peak processing; integration) work from the current experiemnt while others (processing (fitting); Monte-Carlo analysis) do their processing from file.
All relative files are stored in the "processing" directory.
The tools used here are all defined in the `/usr/local/gifa/macro/att/dyna` directory. There are all written in macro, and rely on the FITGENE command for the relaxation parameter estimation.

Data-set list

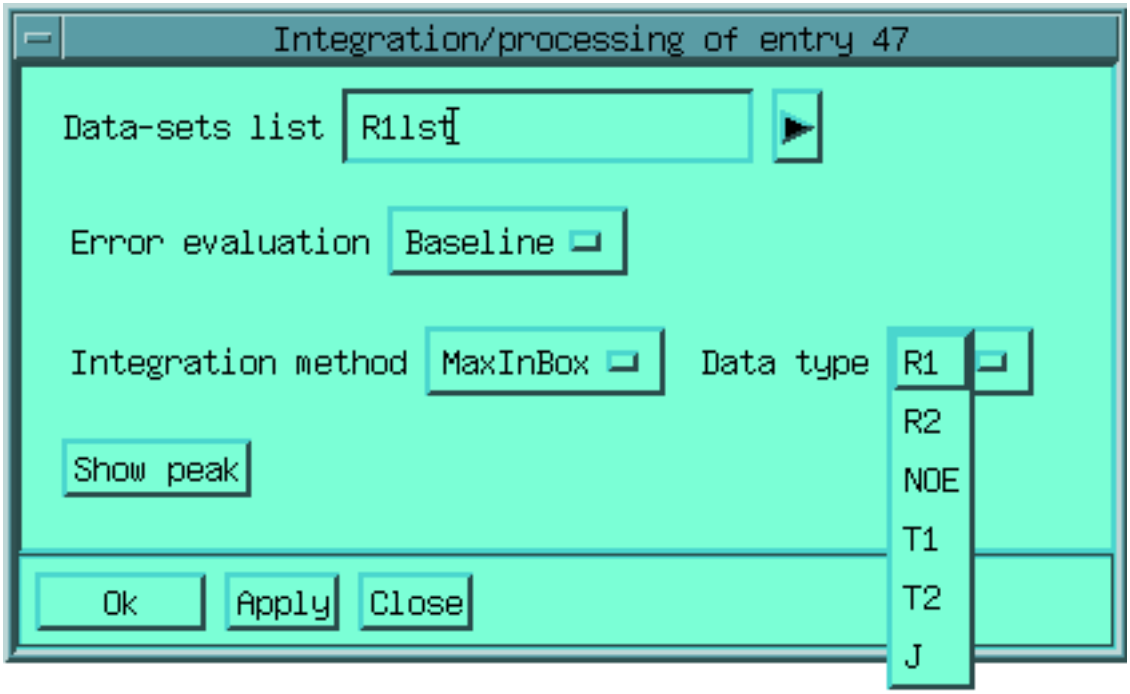
This command permits to define the name of the list which should contain all the entries that will be used for integration and fitting, along with value of the time parameter which was varied. This file should reside in the "processing" directory of the current assignment project.
Here an example of such a file :

```
; list of T1 experiments
0.018 01_T1 01_T1_bis 01_T1_tris
0.054 02_T1
0.102 03_T1
```

```
0.150 04_T1 04_T1_bis 04_T1_tris
0.210 05_T1
0.258 06_T1
0.306 07_T1
0.402 08_T1
0.498 09_T1
0.606 10_T1
0.810 11_T1
1.002 12_T1
1.506 13_T1
2.010 14_T1
2.502 15_T1
3.006 16_T1
```

You can see that : comments are permitted. Each contain first the value of the parameter, and the name of the experiemnt, as it appears in the spectra directory. It is possible to have several experiment for the same delay, in which case the variation of the intensity of a given peak over the different experiments, corresponding to the same parameter, will be used for experimental noise estimate.

Integ/process one peak



This box permits to realize the complete processing : integration and parameter estimate for a given peak in the assignment database. You will have to determine the following parameters :

- the data-set list (as defined above)
- the kind of error evaluation, which can be either Baseline (the baseline noise, estimated on an empty spectral region for each dataset) or Multiexp (estimated from the standard deviation of peaks for the experiemnt which have been realized sevral time for the same time parameter)
- The integration method, which is either MaxinBox (the maximum value in the integration box) of Sumrec (the sum of the point in the integration box). The integration box is defined with the

parameter command in the project menu.

- The data type, which determine the equation which will be fitted :

`R1` is for T_1 relaxation data, fitted in sec^{-1}, equation which is fitted is

$$I_\infty - (I_\infty - I_o)\exp(-R_1 \tau)$$

where τ is the time parameter defined in the dataset list

- R2 is for T2 relaxation data, fitted in sec-1, equation which is fitted is

$$I_o \exp(-R_1 \tau)$$

- NOE is for NOE measurement. In this case, there is no time parameter defined in the dataset list, but the free parameter is replaced by the two keywords : with and without which define respectiveley the experiment performed with saturation and without saturation.
- T1 and T2 are equivalent to R1 and R2 but the following equations are fitted :

$$I_\infty - (I_\infty - I_o)\exp(\frac{-\tau}{T_1}) \quad and \quad I_o \exp(\frac{-\tau}{T_2})$$

- J corresponds to the following equation :

$$I_o \cos(\pi J \tau)\exp(\frac{-\tau}{T_2})$$

which is found in J measurement experiment.

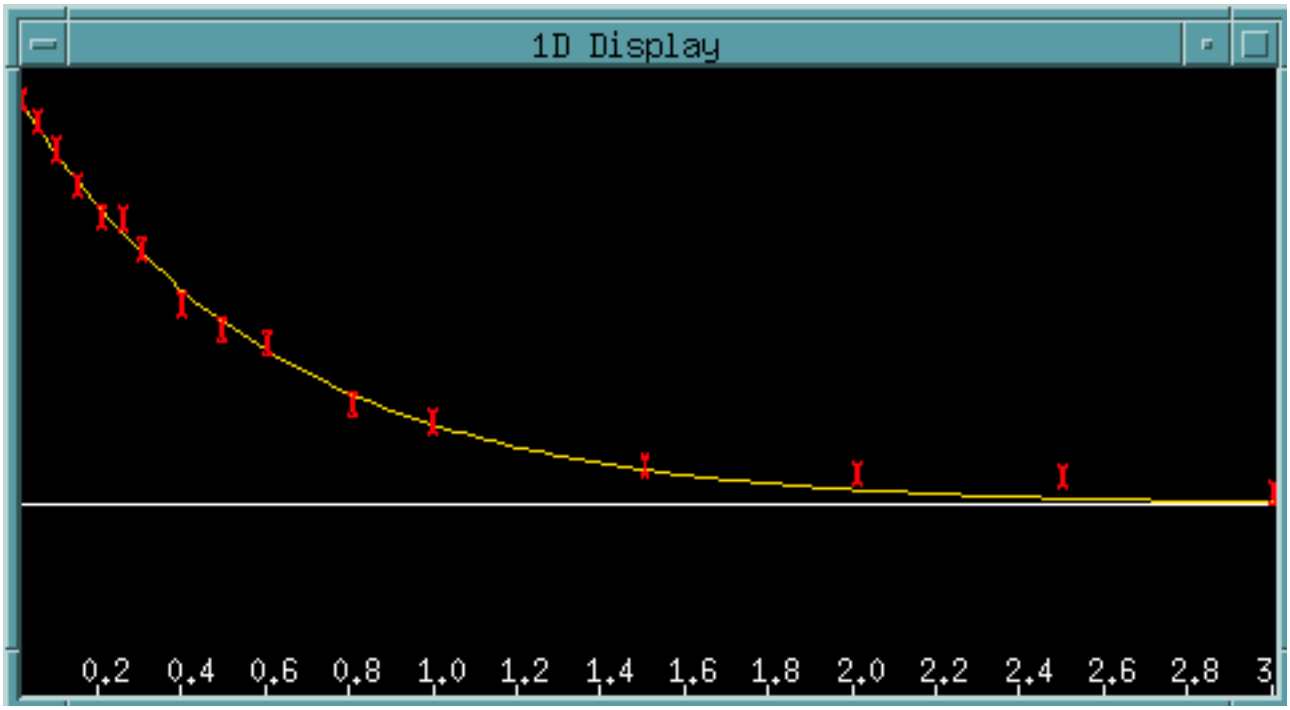
- These equations are defined in the proc1pk.g macro which can be adapted to fit your needs. If you want to use all the tools with the newly defined equation, you will have to adapt also the procMC.g procalllst.g quant1pk.g showfit1pk.g and writproc.g macros as well.

The processing will compute error bars for the fitted parameters, which are determined from the covariance matrix and the size of the intensity error as found from the noise evaluation.
Note that certain parameter combinations may not be meanningfull. For instance, choosing Sumrec integration technique along with the Baseline error estimate will lead to wrong error estimate.

Show last processing

This command simply shows graphically the result of the last fit in the 1D window :

```
FITGENE Final Chi2 :  14.391778
R2 1.63892 +/- .5887275E-01 I0 333448.6 +/- 4905.25
```



Integrate in zoom region

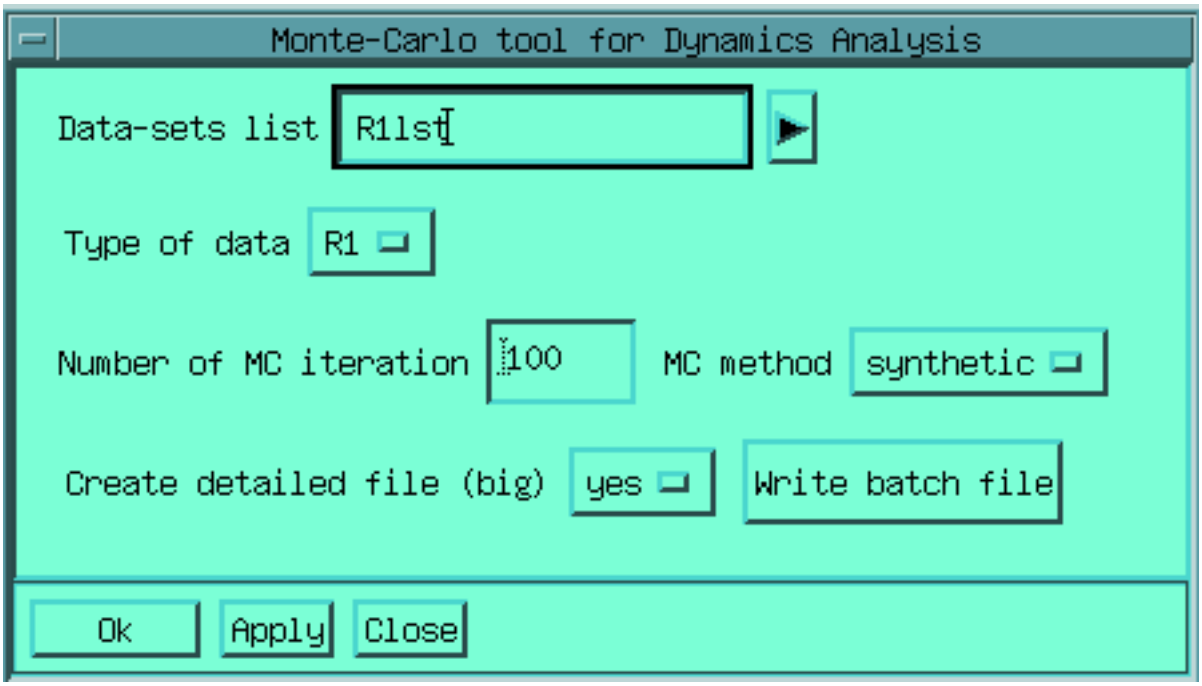
This command performs the first integration step when a complete set of peaks is processed. Start in a zoom window, all assigned peaks within this window will be processed. See `Integ/process one peak` for details on parameter set-. The processing will create a integration file, called as the dataset list file, with `_integ` concatenated at the end. Thus `R1list` becomes `R1lst_integ`. This file is located in the processing directory.

Process from file

This command will take the result of the previous integration command, and will realize a fit for all the peak found in this file. See `Integ/process one peak` for details on parameter set-up. The processing will create a result file, called as the dataset list file, with `_proc` concatenated at the end. Thus `R1list` becomes `R1lst_proc.` This file is located in the processing directory.

Do Monte-Carlo from file

This command takes the result found in the integration file and realize a Monte-Carlo (MC) error estimate on the dataset.



Along with classic parameter you will have to determine the number of MC iterations and the MC method. The MC method can be either 'synthetic' or data. synthetic corresponds to the classical MC run where a first fit is realized on the experimental dataset, then a synthetic dataset is rebuilt from the fitted parameter, and a larg number of simulated experiment, correspomding to synthetic dataset added with noise will be fitted. The statistic on the parameters (mean and standard deviation) are them used as a precise estimate of the precision of the fit. The data method corresponds to a case where the actual experimental dataset are perturbed with additive noise. Results of the MC run will create a result file, called as the dataset list file, with `_MC` concatenated at the end. Thus `R1list` becomes `R1lst_MC.` This file is located in the processing directory.

For MC run you have the option of creating a complete detailed file (called _stat) which permit to verify the chi2 statistic.

As MC runs tend to be very long in the present implementation, you have the possibility of creating a standalone batch file which can then be started in batch.

Previous Next Contents

TYPICAL WORK

assignment

With this set-up, a typical assignment work on a small protein, done in the *Wüthrich* way, consists in:

1. get a set of *nice* processed homonuclear 2D experiments somewhere on the computer system.
2. create a new project, enter the primary sequence, and link all your nice *2D* into the *spectra* directory; then read-in the TOCSY.
3. peak-pick the TOCSY spectrum, at least on the finger print region and copy it over into the assignment data-base, eventually redo so for the NH-side chain regions.
4. use the **marker** tool to note peak alignments, eventually confirm by "showing" also the NOESY, or a TOCSY from other experimental conditions.
5. once a spin system is found, and noted with one or several markers, eventually create the corresponding additional peaks with marker tool, create spins, and put them in the special "build-list";
6. **promote** the spin-list to a spin-system
7. go back to 4) as long as there are spin-systems to assign.
8. load the NOESY in place of the TOCSY. Peak-pick the NOESY spectrum and merge the NOESY peak table with the TOCY peak data-base (**Copy the db from another data-set** and **Merge the peak table into the db** commands)
9. use the NH-NH region and the NH-Ha region to do the sequential assignment, use the marker tool again to create new NOESY peaks in the data-base. Check peaks and spins with the **find peak** and **find spin** tools, use **edit peak** to add peaks in the NOESY data base.
10. once a sequential is found, modify the spin systems accordingly
11. go back to 9 as long as the sequential is not finished
12. assign all remaining NOESY peaks.
13. output the NOESY peaks in the form of a constraint file (see *Integration* menu).

Thirteen steps, not such a big deal after all !
You will notice that the program slows down when the data-base get bigger, this is why it is not recommended to start with a big peak-picking, and then to handle a big data-base throughout the whole process.

Qualitative or quantitative intensity analysis

The intensity analysis (*Integration*) menu implements a set of simple tools dedicated to the analysis of peak intensity according to the assignment database. Here is a simple 'recipe' on how to use this menu: If you want to produce a constraint file used for structure generation:

1. Determine intensities of database peaks using the command **Single data-set integration tool**.
2. Determine a set of calibration peaks, which will be used to define distance estimates on the current peak of the database (see command **Choose the calibration intensities** above). This set of peaks contains the calibration distances chosen for a series of peaks, for which the corresponding intensities are contained in the database.
3. Write the output constraint file, according to the set of calibration peaks. The file can be written in XPLOR or DYANA format, and the distance estimates can be generated in the 'build_up' or 'qualitative' ways. 'Build-up' way means that a precise distance estimate is quantitatively determined from the information contained in the peak calibration set. Then, a general uncertainty can be supposed for all the distances.

If you want to generate files containing intensity variation on a series of data-sets (in the case you want to perform quantitative T1, T2 or nOe analysis).

1. Copy the database to an ascii peak file (same format than those used in pkread/pkwrite commands), and save the lookup table giving the peak index in function of the database index (**Copy db to a pk file)**.
2. Then read this peak file (**PkRead**) and integrate it using the amoeba procedure. Save the amoeba file using the same basename than the peak file.
3. Finally, perform a multiple integration of the series of data-sets according to the saved amoeba file, using the **Multiple Integration Tool**. For each peak in the database, an intensity file is generated and you can check it by using the **Show Integration Curve** command.

Heteronuclear Relaxation analysis

The Dynamic analysis (*Dynamic*) menu implements a set of simple tools dedicated to the analysis of heteronuclear relaxation. Here is a simple 'recipe' on how to use this menu:

1. Get a set of 2D relaxation experiments, in the same acquisition and processing conditions (be carefull with ppm calibration)
2. Fill up the assignment database for all the peak you want to analyse.
3. Set up the dataset list file, which determine which experiments should be used, and what are the experimental parameters
4. Try to process one peak to check if everything is correct.
5. zoom on the region you want to process and start the integration tool.
6. Once the integration is finished, start the parameter estimate.
7. You may finally want to check if every thing is correct by running the Monte-Carlo tool. This

one is currently very slow, so may choose to write a batch file and start it off-graphic with the following command issued in the project directory :

```
gifa < batch_file > log_file &
```

Previous Next Contents

MACRO PROGRAMMING

The complete assignment module is written in nothing but macros. So you could have written it yourself ! At least you can modify it to fit your needs. Here is some help to do so.
All the macros are in */usr/local/gifa/macro/att* , this address is added to the `GifaPath` when entering the module. Static information (list of possible atom names, residue names, etc...) are defined in the *basic_db.g* and *build_static_db.g* macros. You can very simply adapt this one for some new residues. Some static information is stored in dbm data-bases: the "topology" data-base stores the structures of the different residues defined and the names of valid spins. The data-base 3let_1let and 1let_3let stores lookup tables permitting to convert rapidly 1 lettre residue names (C Y K..) to 3 letters names (Cys Typ Lys...). These 3 dat-bases are also stored in /usr/local/gifa/macro/att and can be recreated with the build_static_db.g macro. This can be done if you want to add new types of residues, it may also be needed in the case of a new installation of Gifa, since the dbm format is not fully binary compatible among all UNIX platforms (be carefull, you, Limux users).
The static data-bases are opened when entering the assignment module. Other data-bases are associated to the project associative arrays `att[] spin[]` and `sys[]` for the peak data-base, the spin data-base, and the spin-system data-base respectively. Entries are thus of the kind : `$att[$peak_id]` for instance. The different pieces of information are stored as blank separated fields in the variable. Coding is the following :

```
$att[#att] = f1 f2 amp #spin1 #spin2 type note
$spin[#spin] = delta  name #sys note
$sys[#sys] = index type list_of_spin note
```

where
`#att, #spin #sys` are used here to note the indexes.
`f1` and `f2` are coordinates in ppm;
`amp` is the peak amplitude in arbitrary unit;
`type` codes for the kind of experiment;
`delta` is chemical shift in ppm;
`index` is the number in the primary sequence
`type` is the name of the residue
`note` is a free field, that you can use for whatever function.
Each dbm associative array `att[] spin[]` and `sys[]` contain the special entry LARGEST, which contains the index of the LARGEST id (`#att, #spin, or #sys`) yet assigned. So, when creating a new entry (spin in the example), you are supposed to do something like :

```
set new_id = ($spin["LARGEST"] + 1)
set $spin[$new_id ] = "New entry ..."
set $spin["LARGEST"] = $new_id          ; updated only if no error occured
```

When programming some function that scan the whole data-base, you will probably end-up writing something like :

```
foreach i in att                ; let's scan all the peak as an example
        if ($i s! "LARGEST") then         ; don't forget this one !
                set peak = $att[$i]                  ; this is the complete entry
     ; then parse the entry, this is one way :
                set f1 = (head($peak))  set peak = (tail($peak))
                set f2 = (head($peak))  set peak = (tail($peak))
                set amp = (head($peak)) set peak = (tail($peak))
                                                ; etc...
```

```
        endif
endfor
```

When loaded, the calibration distances defined by the command **Choose the calibration intensities**, are stored in an associative array called calib_dst[].
If you manage to make something useful, you can transmit it to me so that I will make it available to other users.

Previous Next Contents

CAVEAT

Note that this is a preliminary work, quite a few people have been using this tool here in our lab, however, I'm sure there is still a lot of bugs.

- There ARE known bugs, for instance deleting spin-systems might, in certain cases, let the data-base within wrong incoherent informations (however version 1.2 is much more robust than previous ones). Use the `check database` command. In this case, the best way of doing is probably to modify directly the wrong entries. Note that the command Low level editor permits a direct editing to each peak, spin and system, and can be helpful in debugging the data-bases. You can also modify entries from the Gifa command-line level (print $att[xx] set att[xxx] = "---"). In desesperate case, you might want to use perl to access directly the db files.
- There are things that will change. For instance the LARGEST entry, being numerical, interferes with the FIND mechanism. This will change, probably, the LARGEST entry will be prefixed with some non numerical char.
- There are missing tools. To cite the most likely to come first : 'edit build list' ...
- An integration tool based on the line-fitter commands is missing.
- Because of a bug in some dbm namagers (Not in Linux, but most other UNIX implementation), it is not possible to use the foreach syntaxe for modifying entries in the data-bases. So, you cannot use the following:

```
foreach i in att
  if ($i s! "LARGEST") then
     set att[i] = "what you want"
  endif
endfor
```

Rather use the following:

```
for i = 1 to $att["LARGEST"]
  if (exist('att['//$i//']')) then
     set att[i] = "what you want"
  endif
endfor
```

If you wish to help, please contact me !

Assignment on-line Documentation - Contents
