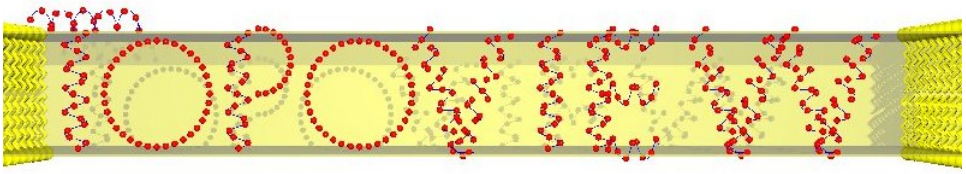
TopoView
Membrane Protein Topology Viewer



Version 0.91

User's Manual

Author: Nadine Schneider

26.02.2008

1 The Application Interface

TopoView is a Membrane Protein topology viewer, mainly developed to display specific structural features of membrane proteins, such as interface helices, "disrupted"[1] transmembrane helices or non contacting consecutive helices. Examples of all these structurally special motifs are given in the next section. This section is an introduction to the usage of the tool "TopoView".

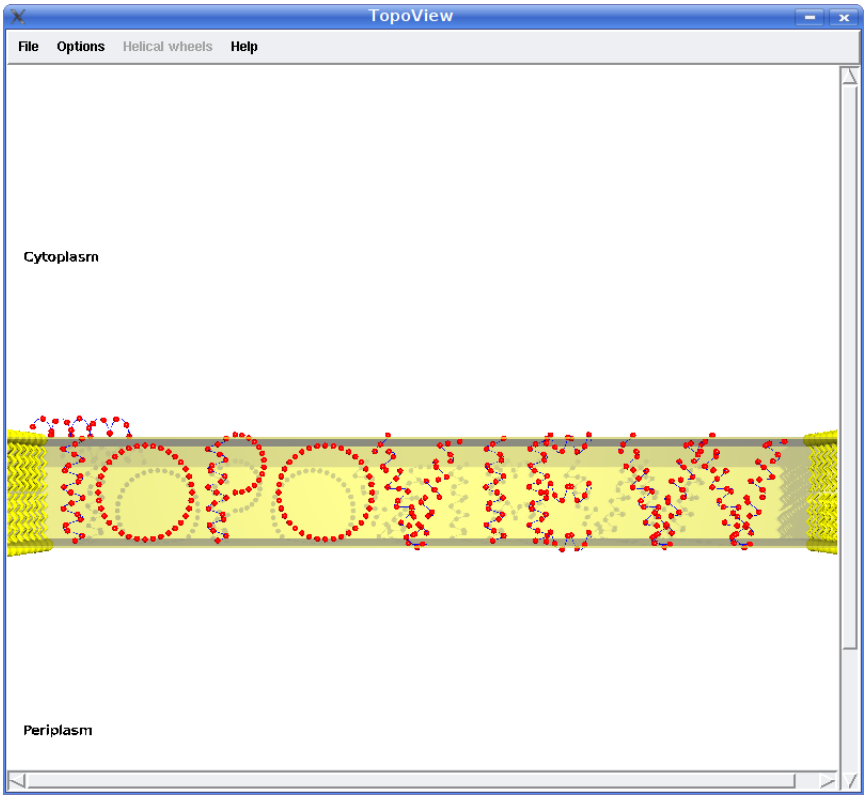


Figure 1: The application window

TopoView consists of two different windows, one for displaying the topology and one for the creation of this topology. Figure 1 shows the main application window. It is divided in two parts, the **menu bar**, where the user can handle the basic procedures and the **representation panel**, where the actual topology images resides.

[1] One long transmembrane helix is "broken" in two short ones, lying above of each other.

1.1 The Menu Bar

The menu bar is always visible and is composed of three main control menus:

- **The File menu**
  - **Create topology**
    Before the topology can be displayed, the user must create it using the "creation window" (figure 2). So by choosing the option "Create topology", the creation window appears.
  - **Load topology**
    Here, the user is able to upload a topology file. An example for such a file can be found in the appendix A.3.
  - **Load TMX prediction**
    Here, the user is able to upload a TMX prediction provided by the TMX server (http://service.bioinformatik.uni-saarland.de/tmx/). An example for such a file can be found in the appendix A.4.
  - **Export image**
    After creation of a topology, the image can be exported with this option as a printer-friendly version (png (portable network graphic), resolution 1000 x 1000 pixels, white background).
  - **Export highquality image**
    This option is for exporting the topology as a highquality image with a resolution of 1800 x 1800 pixels.

- **The Option menu**
  - **Zoom in**
    Here, the image is increased by 100%, so that details can be investigated.
  - **Zoom out**
    By activating this button the general view of the image is shown.
  - **Mark non contact**
    If the user has specified a non contact in the creation window, the non contacting helix pair is not shown until this button is pressed (see "Example" section).

  - **Unmark non contact**
    Here, it is possible to unmark the non contact, so that the original image appears again.

- **The Helical wheels menu**

  - **Property coloring**
    The amino acids of the helical wheels are colored accordingly to their property (small, acidic, polar, ...).

  - **TMX coloring**
    The amino acids of the helical wheels are colored accordingly to their predicted exposedness. Before this option is possible a TMX prediction A.4 must be uploaded in the "File" menu.

  - **Conservation coloring**
    The amino acids of the helical wheels are colored accordingly to their conservation. Before this option is possible a TMX prediction A.4 must be uploaded in the "File" menu.

- **The Help menu**

  - **About**
    The "About" option contains information about the author and the version of the tool.

  - **Help**
    Here, the user gets help through a direct link to the user's manual.

1.2 The Creation Window

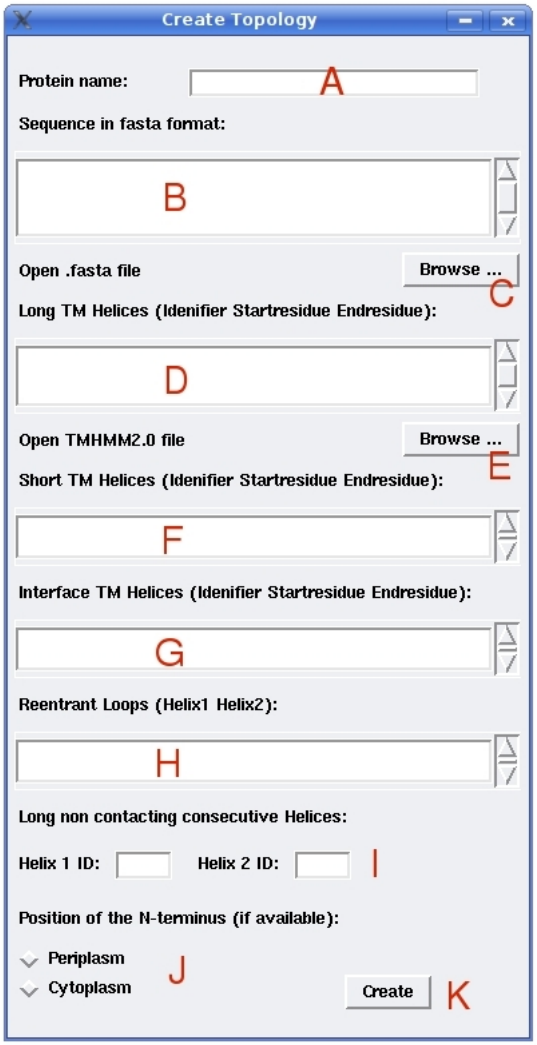


Figure 2: The creation window

A Input box for an abitrary protein name (optional). If a fasta file containing a protein name is loaded up (see C), this protein name is then adopted here.

B Input field for pasting in an amino acid sequence in fasta format (required).

C By pressing this button a File-Dialog window appears and the user is able to upload a fasta file (see appendix A.1 for an example file). This sequence is then displayed in B.

D In this input box the user must define the transmembrane segments (required). This is done by specifying the start- and end-residue of each transmembrane helix. First the helix should get an ID (unique integer), after a space the index of the start-residue should

follow and at last the index of the end-residue (e.g. 1 35 50). For each helix a new line should be used. The user has to take some care in specifying this transmembrane segments: helix IDs must be unique and integers, the index of the start-residue must be smaller than the index of the end-residue and the helix must be at least 15 residues long.

E By pressing this button a File-Dialog window appears and the user is able to upload a output file from the TMHMM 2.0 server[2] (for examples see appendix A.2). The transmembrane segments specified in this file are then displayed in D.

F Input field for specifying short transmembrane helices (optional). In some membrane proteins "broken" helices appear, which are short transmembrane helices lying above of each other and are connected by a very short loop (< 5 residues). Here, the user is able to specify such a kind of structural anomaly. The input is carried out by the same scheme as for the long transmembrane helices in D (unique-Helix-ID start-residue end-residue). Here, the user has to take care again. When using this option, always a pair of short helices must be specified.

G This input box allows the user to define interface helices (same scheme as explained above (D)) (optional). Interface helices often appear in transmembrane proteins and lie nearly parallel to the lipid bilayer. They are mostly only five to ten residues long.

H Here, the user is able to specify reentrant loops between two helices (optional). Reentrant loops are membrane penetrating regions that enter and exit the membrane on the same side. The user has to declare the IDs of the two consecutive transmembrane helices between which the reentrant loop lies (Helix-1-ID Helix-2-ID). The user has to take care that the two helices are already specified before.

I In this part the user can declare two consecutive long transmembrane helices that have no interaction in the folded 3D structure (optional). The IDs of the two helices must be declared in the two input boxes (caution: the IDs must be defined before and the IDs must be consecutive).

J If the location of the N-terminus of the membrane protein is available, the user can specify it here (optional). The default is "cytoplasm", since the N-terminus of most membrane proteins is located there.

K When this button is activated "TopoView" controls whether all inputs are correct and do not break the rules. In the case an input is wrong a message-box opens and informs the user about the mistake and the changes that have to be made. If all inputs are correct "TopoView" starts to calculate the topology image, as soon as the calculation finished the image is displayed in the representation panel.

[2] Transmembrane protein topology server. http://www.cbs.dtu.dk/services/TMHMM/

2 Usage of TopoView

2.1 A simple example

After starting TopoView, the option "Create topology" in the "File" menu has to be chosen. The creation window appears where one can upload a fasta file and a TMHMM 2.0 output file. Figure 3 shows the input in the creation window on the left side the result of this input on the right side.

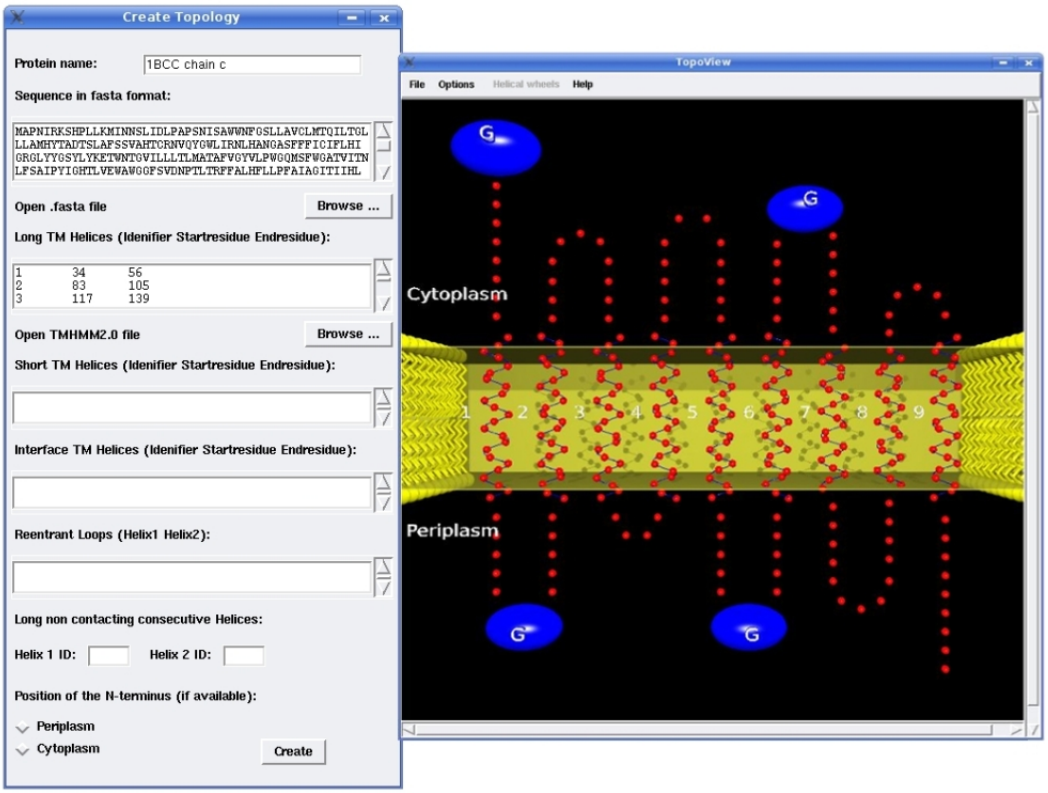


Figure 3: Example input and its result. Loops, that are too long, appear as globular domains (blue spheres).

Then we can use the "Option" menu and the option "Zoom in" to increase our topology image by 100%. The result of this is illustrated in figure 4.
If we are satisfied with our topology image, we can choose the option "Export image (png)" in the "File" menu to save the image as a printer-friendly version. The result of the export can be seen in figure 5.

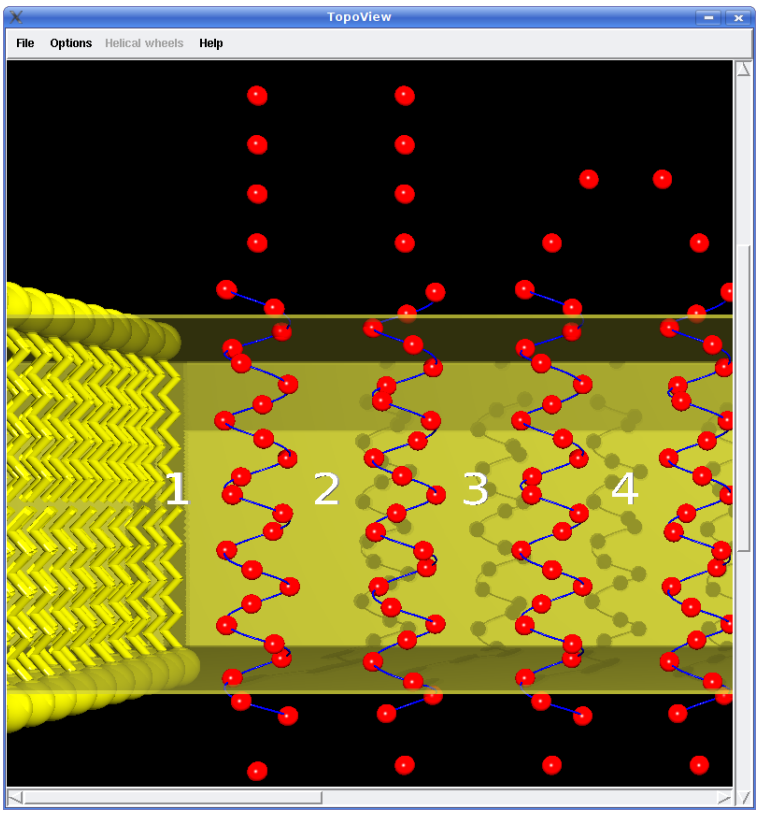


Figure 4: Result of the option 'Zoom in'.

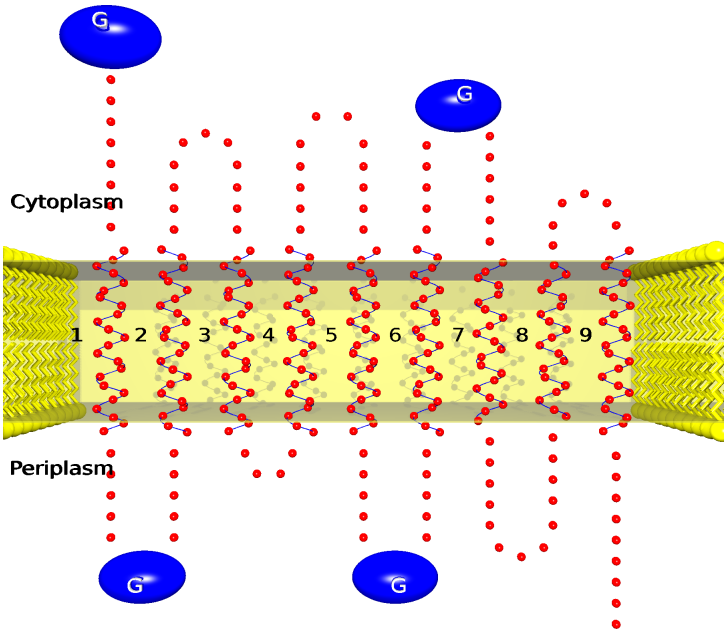


Figure 5: Export of topology image as png with a resolution of 1000 x 1000 pixels.

2.2 Example: Creating short transmembrane helices

In this example we use additionally the option to create two short transmembrane helices. Therefore we define a pair of short transmembrane helices in the corresponding input field of the creation window. Figure 6 shows the input on the left side and its result on the right side.

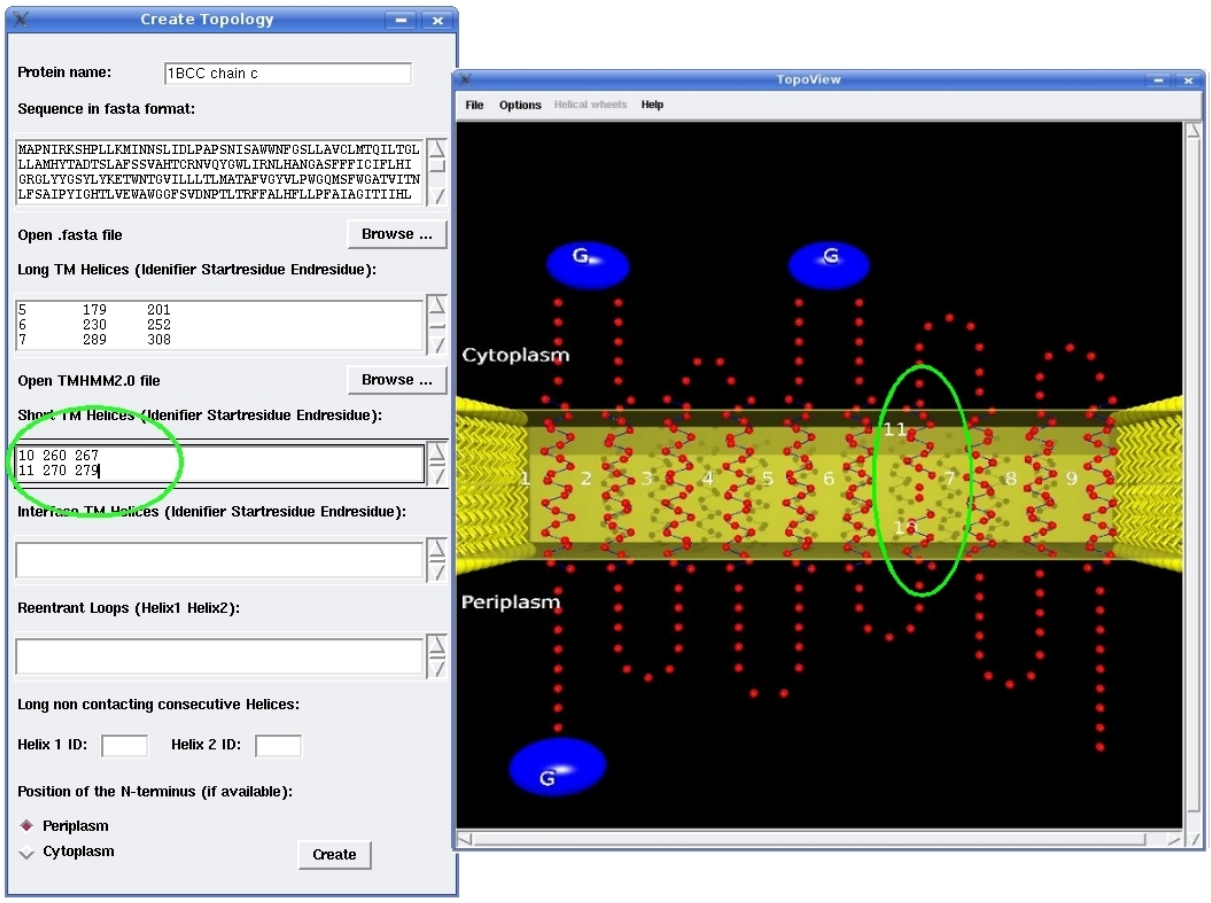


Figure 6: Example: Creating short transmembrane helices (Helix 10 and 11).

It must be pointed out:

- The helix IDs must be unique consecutive integers.
- The index of the start-residue of the helix must be smaller than the index of the end-residue.
- The loop between the helices should not be longer than 5 residues, since this leads to an unrealistic representation.

2.3 Example: Creating interface helices

Here, it is shown how to construct interface helices by the appropriate input box in the creation window. In the figure below the sample input is illustrated on the left and its result on the right.

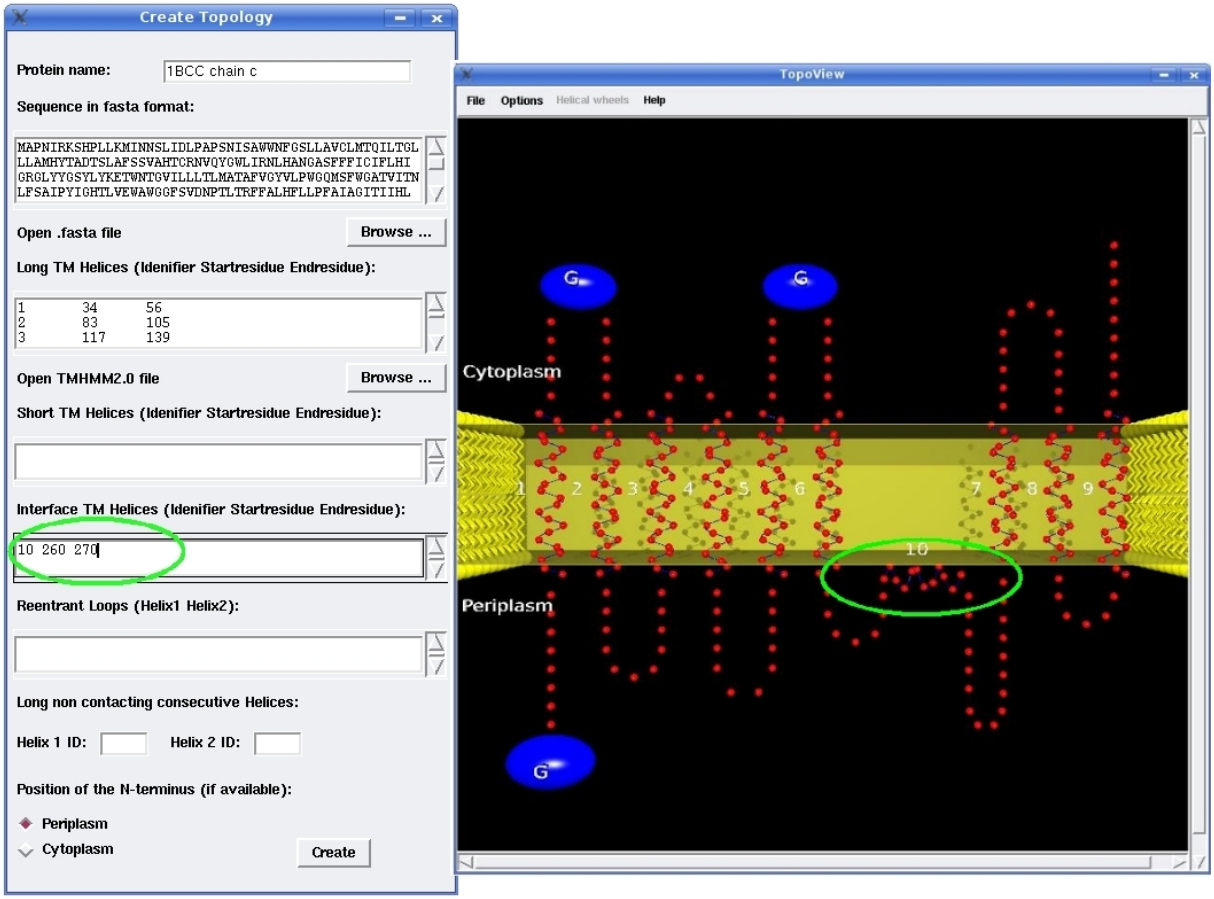


Figure 7: Example: Creating an interface helix (Helix 10).

It must be pointed out:

- The helix ID must be a unique integer.
- The index of the start-residue of the helix must be smaller than the index of the end-residue.

2.4 Example: Creating reentrant loops

In this example we construct a reentrant loop by specifying the two consecutive helices that are neighbours of the reentrant loop. Figure 8 shows the input on the left side and again on the right side the result can be seen.

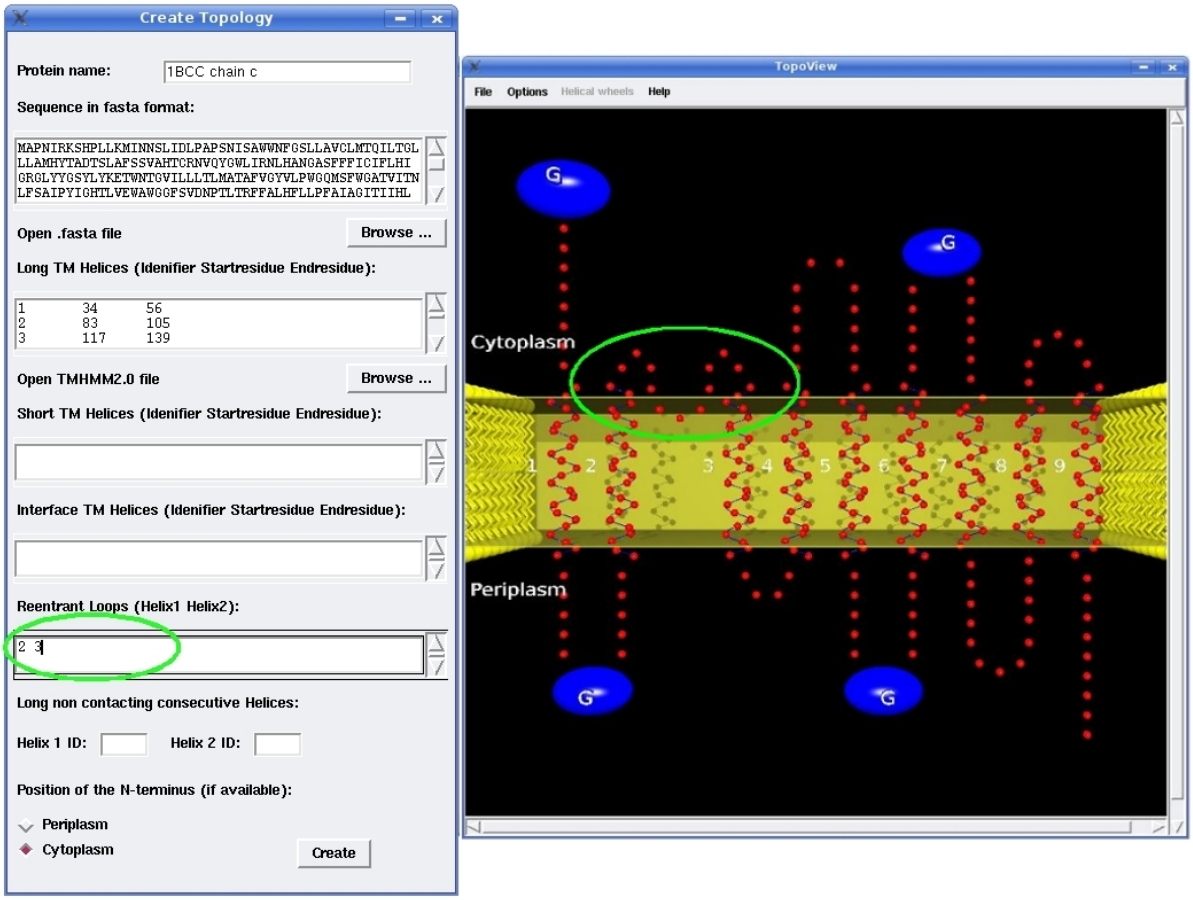


Figure 8: Example: Creating a reentrant loop between helix 2 and 3.

It must be pointed out:

- The helix IDs must be defined before and must be consecutive.
- The length of the reentrant loop should not exceed 20 residues, since this leads to an unrealistic representation. The minimum loop length is 10 residues.
- A reentrant loop between two short transmembrane helices (see above) is not possible.

2.5 Example: Creating consecutive non interacting helices

This example shows how a non contact is assembled. We use the input boxes for the non contact at the bottom of the creation window. Here, we fill in the two IDs of the consecutive helices that should have no interaction. After the topology image has appeared, we use the option "Mark non contact" in the "Option" menu (see figure 9). The result of the sample input is shown in figure 10.

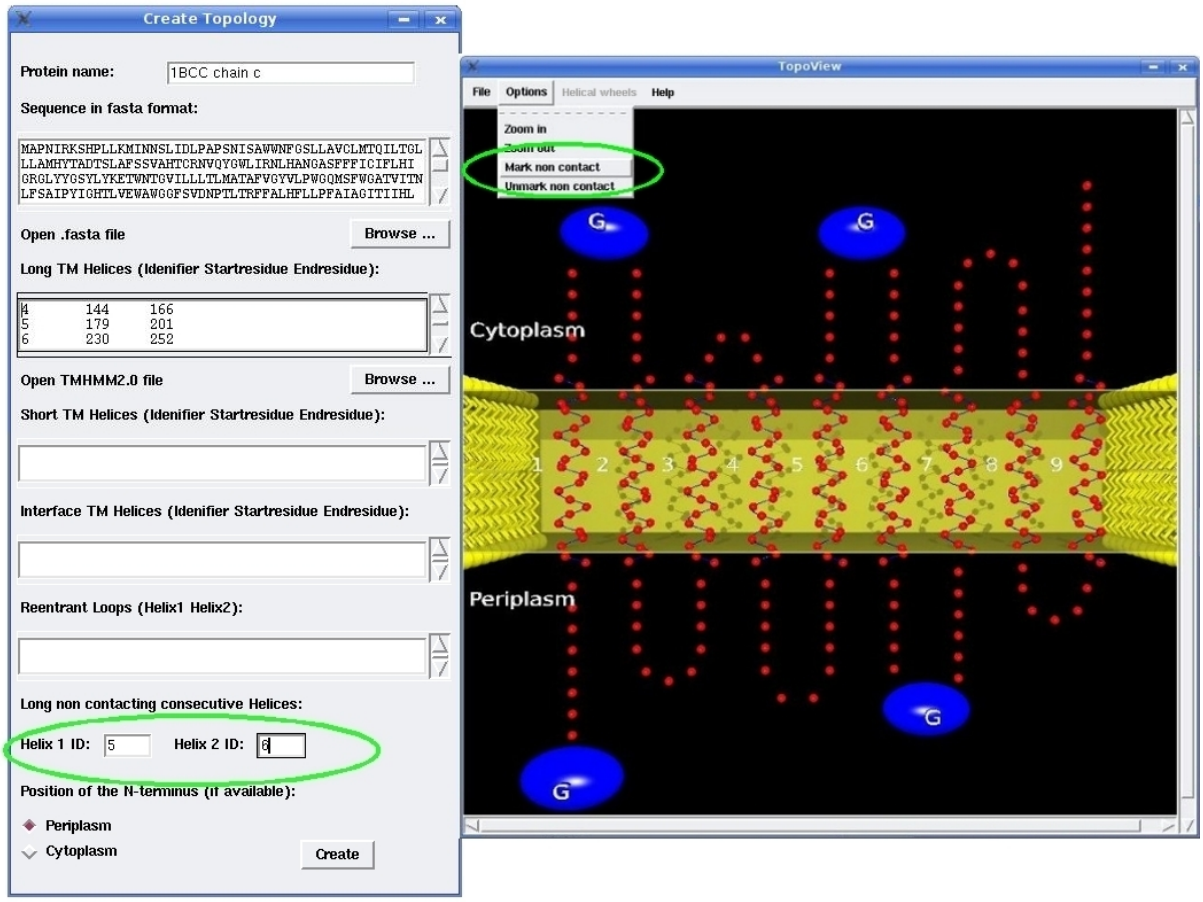


Figure 9: Example: Creating non interacting consecutive helices (helix 5 and 6).

It must be pointed out:

- The helix IDs must be defined before and must be consecutive.
- The loop between the non contact must be at least 10 residues long and the loop of the neighboring helix which is permuted to constitute the non contact must also be at least 10 residues long.
- If a non contact and a reentrant loop are between the same two helices, the helices will

not be permuted but highlighted in green.

- The non contact is only possible between long transmembrane helices. Also at least one neighbor of these helices should be a long transmembrane helix, otherwise it is not possible to shift the helices and they will only be highlighted in green.

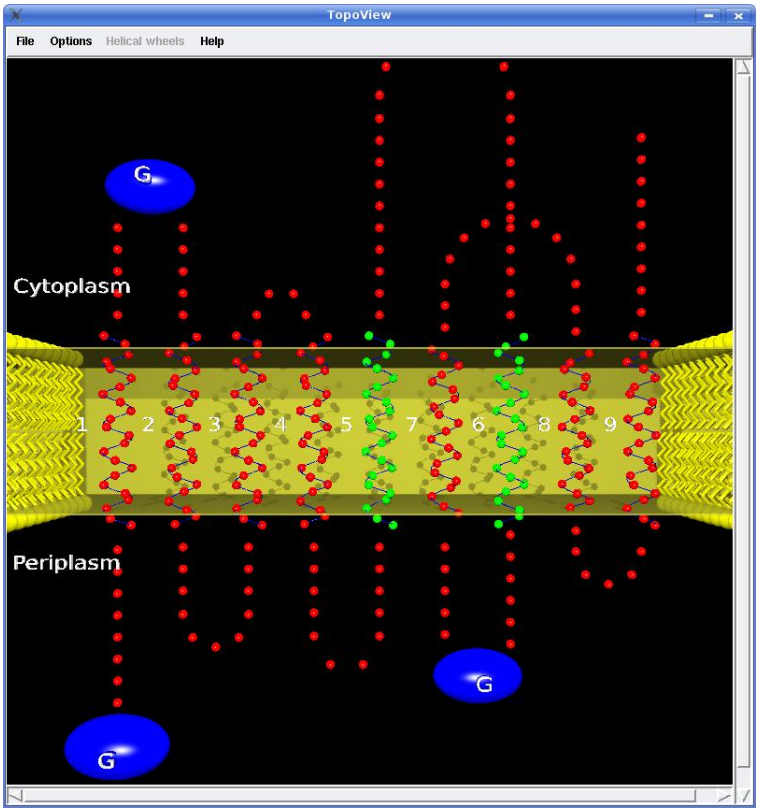


Figure 10: Result of creating non interacting consecutive helices (helix 5 and 6).

2.6 Example: Creating helical wheels

It is also possible to create helical wheels with TopoView. After a topology is created we can get helical wheels of all TM helices in the "Helical wheels" menu. Here it is possible to choose between three different coloring modes of the amino acids (property, exposedness (TMX) and conservation), whereas the last to modes are only available if a TMX predition is uploaded in the "File" menu before. Examples for the three different coloring modes are shown in the figures below.

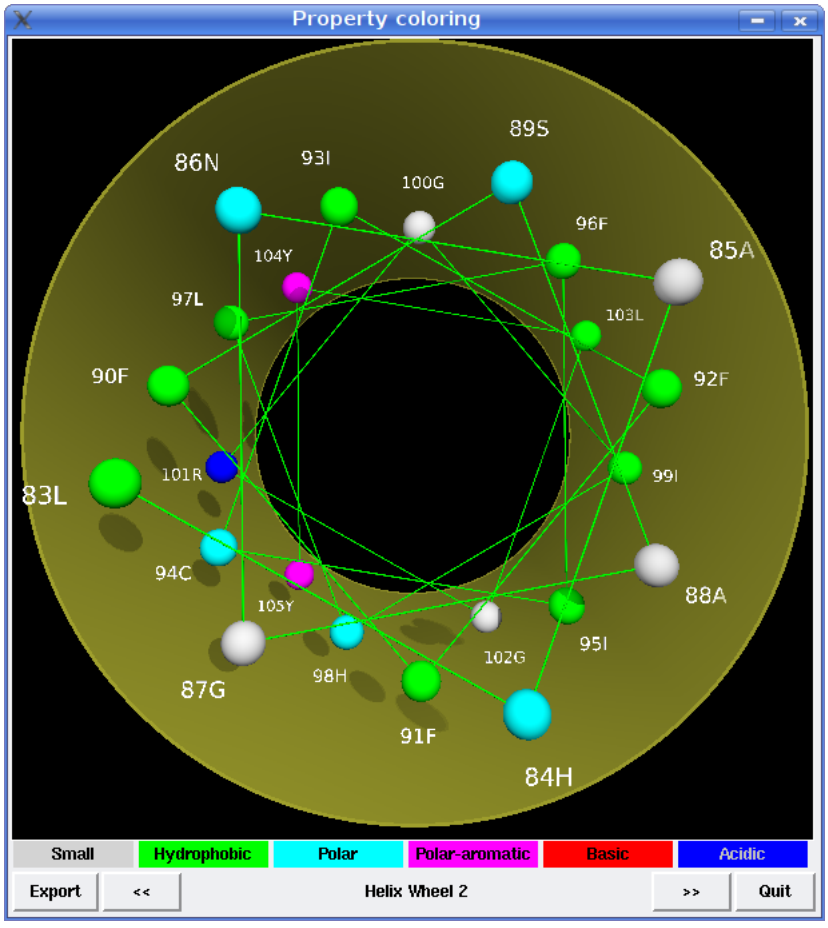


Figure 11: Result of creating a helical wheel with property coloring.

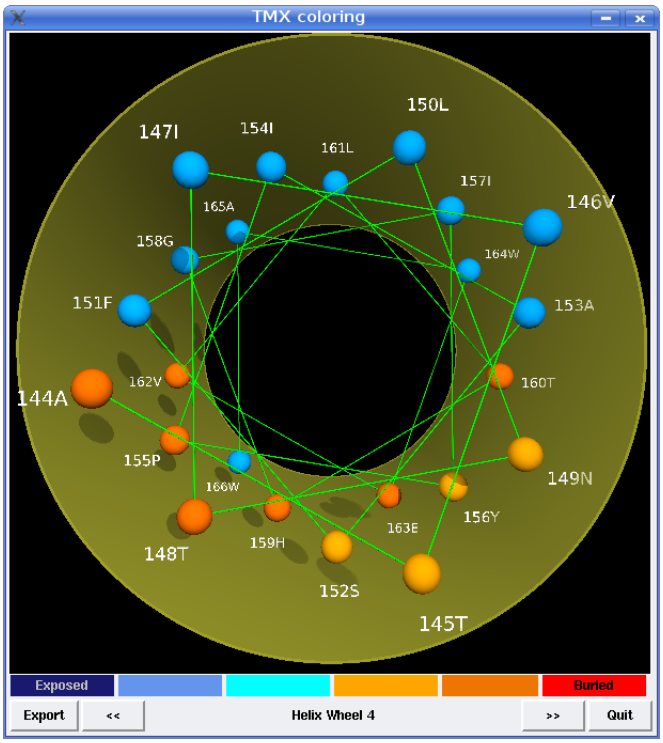


Figure 12: Result of creating a helical wheel with TMX coloring (predicted exposedness).

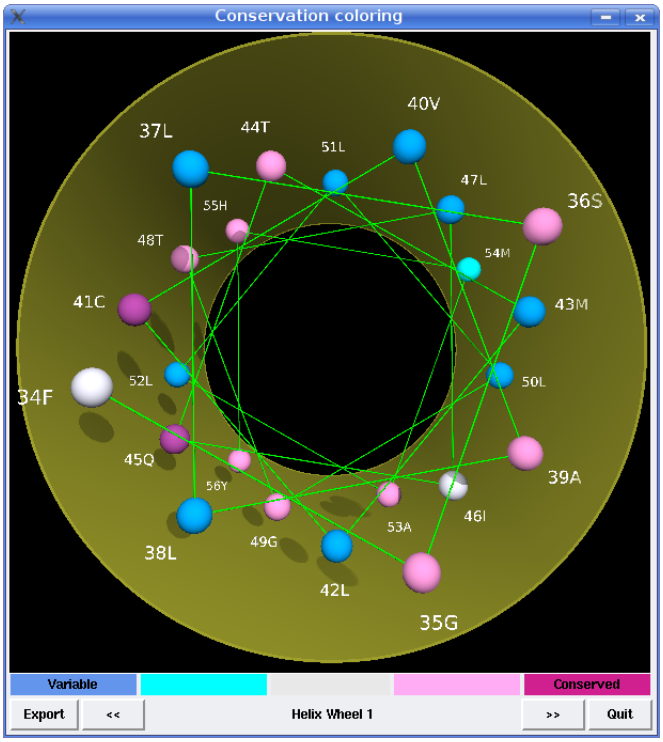


Figure 13: Result of creating a helical wheel with conservation coloring.

A Input File examples

A.1 Fasta file

```
>1BCC chain c
MAPNIRKSHPLLKMINNSLIDLPAPSNISAWWNFGSLLAVCLMTQILTGLLLAMHYTADTSLAFS
SVAHTCRNVQYGWLIRNLHANGASFFFICIFLHIGRGLYYGSYLYKETWNTGVILLLTLMATAFV
GYVLPWGQMSFWGATVIT...
```

Figure 14: Example for a fasta file. The first line, beginning with the greater-than sign contains the protein name. The following lines are reserved for the amino acid sequence in upper letter code.

A.2 TMHMM 2.0 file

```
Sequence      len=380       ExpAA=193.88      First60=22.59       PredHel=9
Topology=i34-56o83-105i117-139o144-166i179-201o230-252i289-308o323-341i348-370o
```

Figure 15: Example for a TMHMM 2.0 output file (one line per protein).

```
# Sequence Length: 380
# Sequence Number of predicted TMHs:  9
# Sequence Exp number of AAs in TMHs: 193.8761
# Sequence Exp number, first 60 AAs:  22.58622
# Sequence Total prob of N-in:        0.73570
# Sequence POSSIBLE N-term signal sequence
Sequence    TMHMM2.0    inside          1    33
Sequence    TMHMM2.0    TMhelix        34    56
Sequence    TMHMM2.0    outside        57    82
Sequence    TMHMM2.0    TMhelix        83   105
Sequence    TMHMM2.0    inside        106   116
Sequence    TMHMM2.0    TMhelix       117   139
Sequence    TMHMM2.0    outside       140   143
Sequence    TMHMM2.0    TMhelix       144   166
.
.
.
```

Figure 16: Example for a TMHMM 2.0 output file (extensive, no graphics).

A.3 Topology file

```
name:1BCC chain c
sequence
>>
MAPNIRKSHPLLKMINNSLIDLPAPSNISAWWNFGSLLAVCLMTQILTG
LLLAMHYTADTSLAFSSVAHTCRNVQYGWLIRNLHANGASFFFICIFLH
IGRGLYYGSYLYKETWNTGVILLLTLMATAFVGYVLPWGQMSFWGATVI
TNLFSAIPYIGHTLVEWAWGGFSVDNPTLTRFFALHFLLPFAIAGITII
HLTFLHESGSNNPLGISSDSDKIPFHPYYSFKDILGLTLMLTPFLTLAL
FSPNLLGDPENFTPANPLVTPPHIKPEWYFLFAYAILRSIPNKLGGVLA
LAASVLILFLIPFLHKSKQRTMTFRPLSQTLFWLLVANLLILTWIGSQP
VEHPFIIIGQMASLSYFTILLILFPTIGTLENKMLNY
>>
N-terminus: cytoplasm
TM      2     33     54
TM      4     77    104
TM      5    111    129
TM      8    173    203
TM      9    224    246
TM     12    320    341
TM     13    346    377
sTM    10    273    283
sTM    11    288    304
IH      1     11     16
IH      3     62     73
IH      6    138    149
IH      7    158    165
RL      8      9
NC      8      9
```

Figure 17: Example for a topology file.

In the first line the name of the protein is written (required). The second and the third line mark the beginning of the amino acid sequence (required). Then the amino acid sequence in upper letter code follows (required). The sign "»" marks the end of the sequence (required). The next line holds the "N-terminus" of the protein, if not known, please write: "N-terminus: noinfo" (required). The lines starting with "TM" hold the borders of transmembrane segments (required). First there must be a unique integer and then the start and the end residue of the transmembrane helix (start residue $<$ end residue). The lines starting with "sTM" represent the short transmembrane helices that are lying above of each other, therefore always a pair is necessary (optional). Here, there must be also a unique integer first and then the start and the end residue of the transmembrane helix. The loop between the two short helices should not exceed 4 residues. Lines with "IH" at the beginning represent the interface helices (optional). Here also the same principle as above holds (unique integer startresidue endresidue). The "RL" represent reentrant loops (optional). Here two helix IDs, specifyied before, are necessary. The loop must have a minimum length of 10 residues. The last structure that is possible is that of non interacting consecutive loops "NC" (optional). Only one non contact is possible and the two helix IDs of the helices that should have no contact are required.

A.4 TMX prediction file

```
Calling predictor for MSA : 1bcc_cclustalw.aln
Frequency profile prepared.
Sequence Submitted: MAPNIRKSHPLLKMINNSLIDLPAPSNISAWWNFG...

Aligned Sequence: MAPNIRKSHPLLKMINNSLIDLPAPSNISAWWNFG...
Starting at 0
Length of Sequence 380
Predictions sorted by AA
typeAmino Acid  Count  Average Confidence Score  Average Conservation Index
TRP             11            0.61                        0.83
PHE             29            0.79                       -0.17
TYR             13            1.13                        0.74
MET             10            1.44                       -0.06
LEU             63            1.54                       -1.13
ILE             33            1.55                       -0.30
VAL             14            0.57                        0.54
ALA             27            0.86                       -0.04
CYS             3             1.11                        1.07
.
.
.
Total Residues Exposed: 193( 50.92)
Total Residues Buried: 186( 49.08)

Summary of Prediction resultsTemp
Protein TempChain TempIndex Residue Conservation Positionalscore Predicted Confidencescore
1bc        A         0         M       0.86            0.10          E          0.10
1bc        A         1         A       0.32            0.10          B          0.10
1bc        A         2         P       0.39            0.04          B          0.04
1bc        A         3         N       0.53            0.10          E          0.10
1bc        A         4         I       0.15            0.20          E          0.20
1bc        A         5         R       0.92            0.03          B          0.03
1bc        A         6         K       0.86            0.06          B          0.06
.
.
.
```

Figure 18: Example for a TMX prediction file (input: Multiple sequence alignment).