Tutorial:

Ensemble refinement of dynamic protein structures on the example of ESCRT-III CHMP3

Johannes Schöneberg[1*], Max Linke[1], Gerhard Hummer[1]
*[1]: Department of Theoretical Biophysics, Max Planck Institute of Biophysics, Frankfurt am Main, Germany. *: johannes.schoeneberg@biophys.mpg.de*

December 2014

Intrinsically disordered domains in protein structures are at times variable enough to defeat current methodologies in structural biology and therefore evade structural resolution.
An ensemble refinement methodology has been developed and implemented that allows combining indirect experimental structure data (such as SAXS, DEER or FRET) with coarse grained molecular simulations in order to study the protein's conformational space and determine its structure[1].
The approach has been used successfully in various cases[2,3]. This tutorial shows the principles of the approach on the example of the ESCRT-III protein CHMP3 and is built on the successful characterization of CHMP3 with the method[4].

[1] Kim, Y.C.; Hummer, G.; *J Mol Biol.* **2008**, 375(5):1416-33.
[2] Boura, E.; Różycki, B.; Herrick, D.Z.; Chung, H.S.; Vecer, J.; Eaton, W.A.; Cafiso, D.S.; Hummer, G.; Hurley, J.H. *PNAS* **2011**, 108, 9437-9442.
[3] Boura, E.; Różycki, B.; Chung, H.S.; Herrick, D.Z.; Canagarajah, B.; Cafiso, D.S.; Eaton, W.A.; Hummer, G.; Hurley, J.H. *Structure* **2012**, 20, 874–886.
[4] Różycki, B.; Kim, Y.C.; Hummer, G.; *Structure* **2011**, 19(1):109-116

Welcome:

Welcome! This tutorial is designed to give an overview over computer simulation based protein conformation exploration and the subsequent refinement with experimental data. All necessary files and binaries will be handed by the instructors. We are happy to answer any questions and hope you enjoy the tutorial.

Requirements:

This tutorial requires:

- an Unix based operating system
- VMD
  The best way to obtain VMD is from its website:
  www.ks.uiuc.edu/Research/vmd/
- IPython
  The easiest way to get IPython to work is via the Anaconda distribution from Continuum analytics: http://continuum.io/downloads

Installation

To install the tutorial, unpack the 'tutorial_chmp3.zip' into the location you desire. This is it and you are ready to go.

Get an Overview

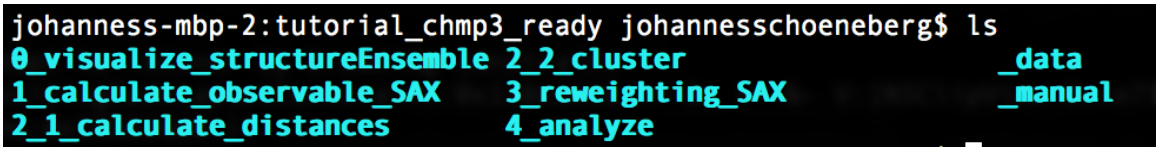
Open a Terminal/Shell and change your directory to the newly unpacked folder.

```
cd ($YOURPATH)/tutorial_chmp3/
```

Now look at the contents of the tutorial folder via typing

```
ls
```

You should see a similar representation as this:



The tutorial is orgaized into 6 distinct steps:

0_visualize_structureEnsemble

In the first step, we are going to have a look at the molecule we are interested in, CHMP3. We will see how it is modeled for the coarse grained simulation and how the results of the coarse grained simulation look like.

1_calculate_observable_SAX

In the second step, we are going to calculate the small angle X-ray scattering (SAXS) profiles from each structure in the simulated ensemble.

2_1_calculate_distances

In this step, we will calculate the DRMS as a distance metric between the structures in the simulated ensemble as a prerequisite for the subsequent clustering.

2_2_cluster

In this step, the simulated structure ensemble is clustered, according to the distances calculated in the step before. We are going to use Quality Treshold (QT) clustering.

3_reweighting_SAX

Based on experimentally derived SAXS profiles, the clustered ensemble will be reweighted by the maximum entropy method. The result is structural information of the protein structures that were likely present when the SAXS experiment was performed, effectively adding atomic resolution information to SAXS experiments that are otherwise limited to shape information.

4_analyze

In this final step, the results of the SAXS reweighting will be visualized and analyzed.

Apart from the distinct tutorial steps, you will find the following additional folders in your tutorial:

_data

This folder contains the input data for the tutorial, consisting of the pdb structure of CHMP3 (`3frt.pdb`), the coarse grained model of CHMP3 as the input structure for the simulations (`chmp3_model.pdb`), the results of the coarse grained structure sampling (`chmp3_ensemble_lowSalt_[0,1,2].pdb`, `chmp3_ensemble_highSalt_0.pdb`) and the experimentally obtained SAXS profiles (`SAXS_LowSalt.txt, SAXS_HighSalt.txt`).

_manual

In this folder, this manual is provided in pdf format.

0_visualize_structureEnsemble

CHMP3

We will first have a look at our molecule of interest, CHMP3:

Assuming that you are currently in the main folder of the tutorial, type

```
cd 0_visualize_structureEnsemble
```

Looking at the contents of the folder via `ls`, should show you a similar view as this:

```
johanness-mbp-2:0_visualize_structureEnsemble johannesschoeneberg$ ls
1_nonAligned    bin             param_tplgy.in
_templates      configure.sh    split.sh
```

First, we now need to compile the vmd scripts via typing

```
./configure.sh
```

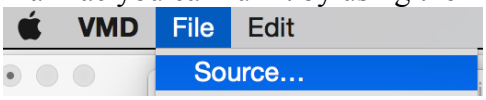
Four new files should now have appeared in the folder. You can check this by typing `ls`:

```
johanness-mbp-2:0_visualize_structureEnsemble johannesschoeneberg$ ./configure.sh
johanness-mbp-2:0_visualize_structureEnsemble johannesschoeneberg$ ls
0_vmd_show_chmp3.tcl              _templates
0_vmd_show_model.tcl              bin
1_nonAligned                      configure.sh
1_vmd_show_nonAligned.tcl         param_tplgy.in
2_vmd_showTogether_nonAligned.tcl split.sh
```

Visualization by VMD

We will now visualize the CHMP3 crystal structure by VMD. There are prepared scripts for visualization that you can use. There are multiple ways to run them in VMD, depending on your operating system and your preferences:

- You can open VMD (by typing `vmd` in the terminal or finding the program icon and opening it by a mouseclick) and then run the script in the VMD shell by typing
  `source $YOURTUTORIALPATH/0_visualize_structureEnsemble/0_vmd_show_chmp3.tcl`

- On a Mac you can run it by using the mouse and the VMD source menue:
  

  and open the `0_vmd_show_chmp3.tcl` script using the appearing file browser.

- On a Linux machine, you can run it by typing
  `vmd -e 0_vmd_show_chmp3.tcl`

Independent of the way you got the vmd script to run, the resulting visualization of the molecule should look similar to Figure 1:

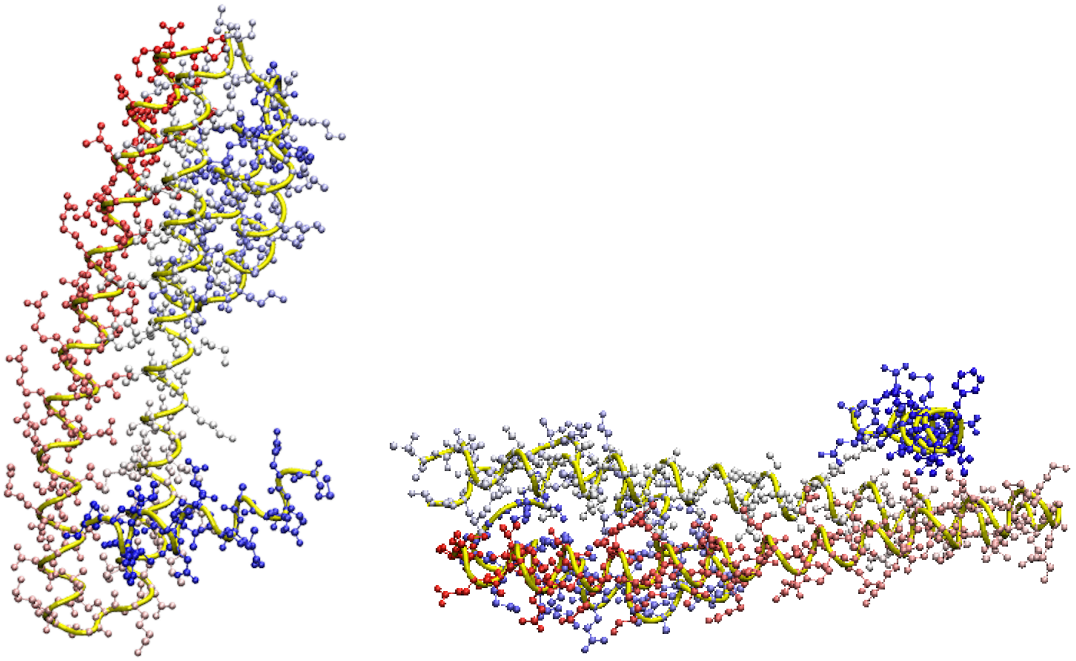


Figure 1. **Crystal structure of CHMP3**. Depicted is the crystal structure of CHMP3 (pdb ID 3FRT.pdb) once in top view (left) and once in side view (right). The backbone of CHMP3 appears in yellow, the individual atoms appear in ball and stick representation, colored by their index in the structure (from red, lowest index, to blue, highest index). There are five helices visible: Helices α_1-α_4 form a large rigid stalk in the center. Helix α_5 (blue) is visible lying on top of the other helices. Note, that α_5 is not connected to the stalk domain by the backbone. Helix α_6 is missing alltogether in the crystal structure. This is a result of the very high intrinsic flexibility of these domains of CHMP3, the reason why they were not resolved by X-ray crystallography.

Modeling the flexible parts of CHMP3 and helix 6

As you can see in Figure 1, important parts of CHMP3 could not be resolved in X-ray crystallography. The software for coarse grained protein complex structure sampling that we developed in our group[5] allows us to model these parts and come up with a model of the full structure of CHMP3. We can not use the software in this tutorial. We can however look at the results of the modeling:

You can visualize the model using the provided vmd script

```
0_visualize_structureEnsemble/0_vmd_show_model.tcl
```

[5] Kim, Y.C.; Hummer, G.; *J Mol Biol.* **2008**, 375(5):1416-33.

just as we used the script for visualizing CHMP3's crystal structure.

You should be seeing an image similar to *Figure 2*.

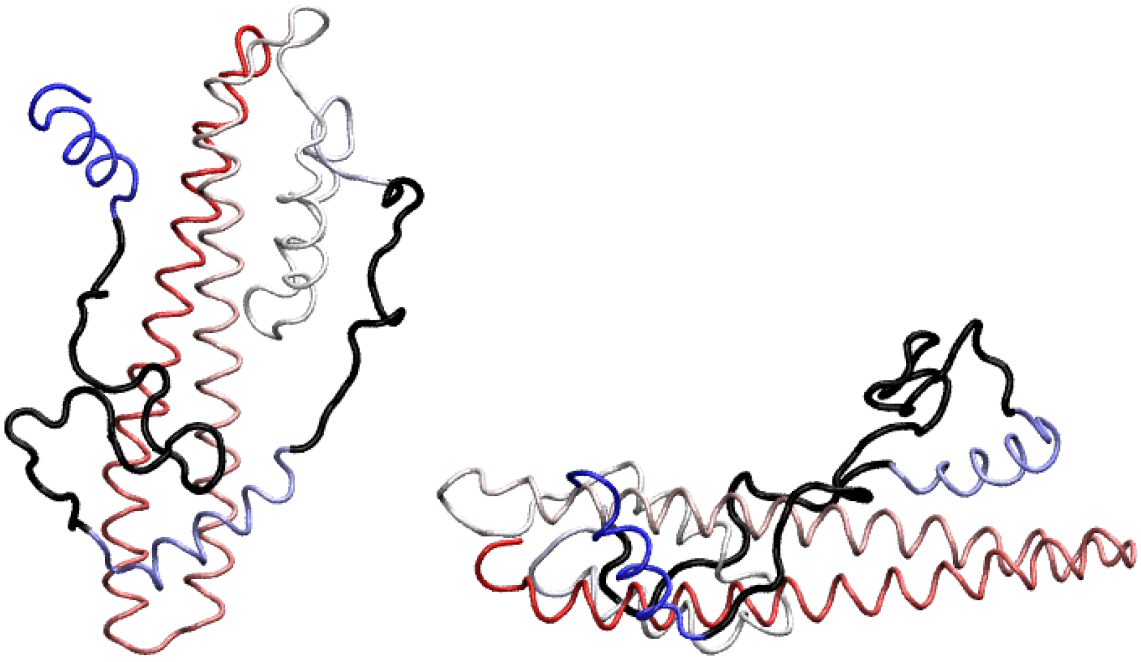


Figure 2. **Full model of CHMP3 including the flexible linkers and helix 6.** Depicted is the crystal structure of CHMP3, including the flexible linkers (black) that connect helix α_5 (light blue) with the stalk (red to white), and that connect helix α_6 (blue) with helix α_5. Both linkers and helix α_6 are absent in the crystal structure (compare Figure 1). Shown is a top view (left) and a side view (right). The coloring corresponds to the index in the protein structure (red, low index, to blue, high index). Consequently, helices α_5 and α_6 appear in light and dark blue.

Sampling the conformational space of the full CHMP3 structure

We intend to investigate the conformational flexibility of the full CHMP3 structure, including the flexible linkers and helix α_6. We are going to do this by a coarse grained molecular simulation of the full structure[6]. The input for these simulations is a model of the full structure that consists of rigid domains and flexible linkers. In Figure 2, these rigid domains appear as colored while the flexible regions appear black.
If you would like to have a look at the definition of this topology, see the file:

```
0_visualize_structureEnsemble/param_tplgy.in
```

We will not be able to run these simulations in the tutorial. However, we provided

[6] Kim, Y.C.; Hummer, G.; *J Mol Biol.* **2008**, 375(5):1416-33.

structure ensembles that are the result of these simulations for this tutorial. They can be found in

```
$YOURTUTORIALPATH/_data/
```

and are named

```
chmp3_ensemble_lowSalt_0.pdb
chmp3_ensemble_lowSalt_1.pdb
chmp3_ensemble_lowSalt_2.pdb
chmp3_ensemble_highSalt_0.pdb
```

To proceed with the tutorial, choose one of the `lowSalt` ensembles and make it ready for use in the tutorial by renaming it to

```
chmp3_ensemble.pdb
```

You can do this in your terminal by typing

```
mv chmp3_ensemble_lowSalt_0.pdb chmp3_ensemble.pdb
```

The tutorial assumes that you chose ensemble `chmp3_ensemble_lowSalt_0.pdb`. If you chose differently, your results might look slightly different than the results depicted in the tutorial, due to the stochastic nature of the sampled structure ensembles and their small sample sizes (1000 structures, usually >> 10,000 are used).

Visualizing the generated structure ensemble

We can now have a look at the ensemble by switching again to the folder

```
$YOURTUTORIALPATH/0_visualize_structureEnsemble
```

The VMD script

`0_visualize_structureEnsemble/1_vmd_show_nonAligned.tcl`

when opened as showed before, will show you all 1000 structures as frames in a trajectory. They will look similar to *Figure 3*.

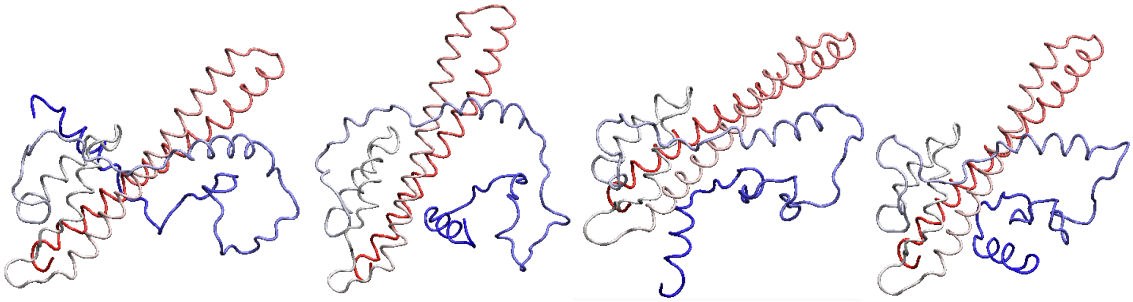


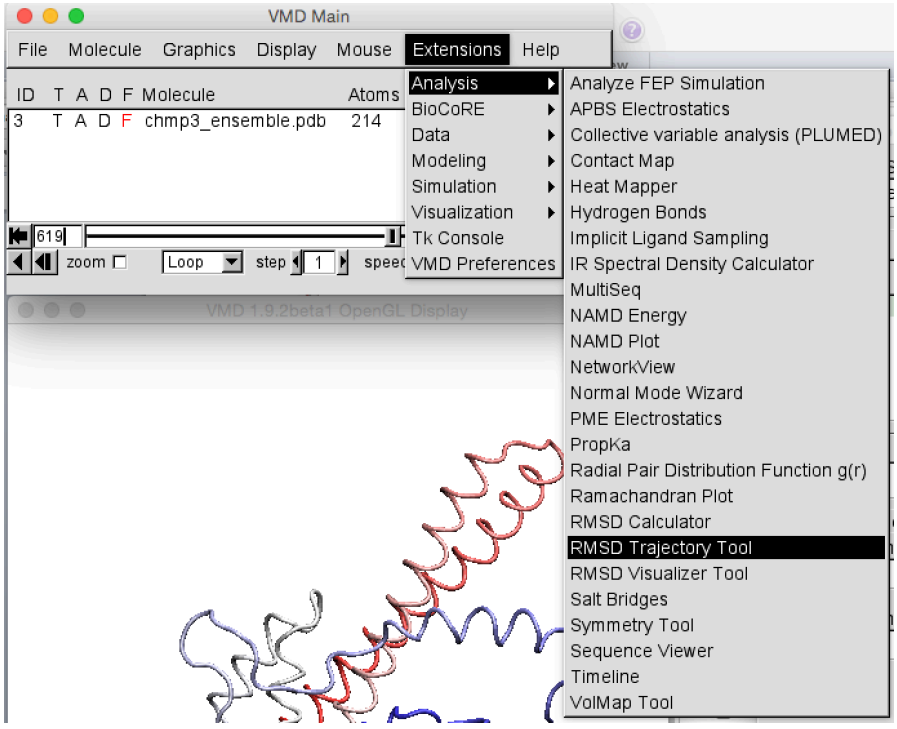
Figure 3. **Structure ensemble of CHMP3 generated by goarse grained simulation**. Depicted are individual structures in the ensemble generate by Monte Carlo conformation sampling as it is described in Kim and Hummer 2008. The backbone is colored by index from red (low index) to blue (high index). The input for the conformation sampling is a model of the full CHMP3 that consists of the stalk and helix 5 as they appear in the crystal structure 3FRT and that consists additionally of helix 6 and two flexible linkers. See Figure 2 for a depiction of the input structure.

Aligning the ensemble structures

In order to get a better view at the actual conformational sampling, i.e. to see the flexibility of the linkers with respect to the rigid domains, we will make use of the 'RMSD Trajectory Tool', a built-in function of VMD:

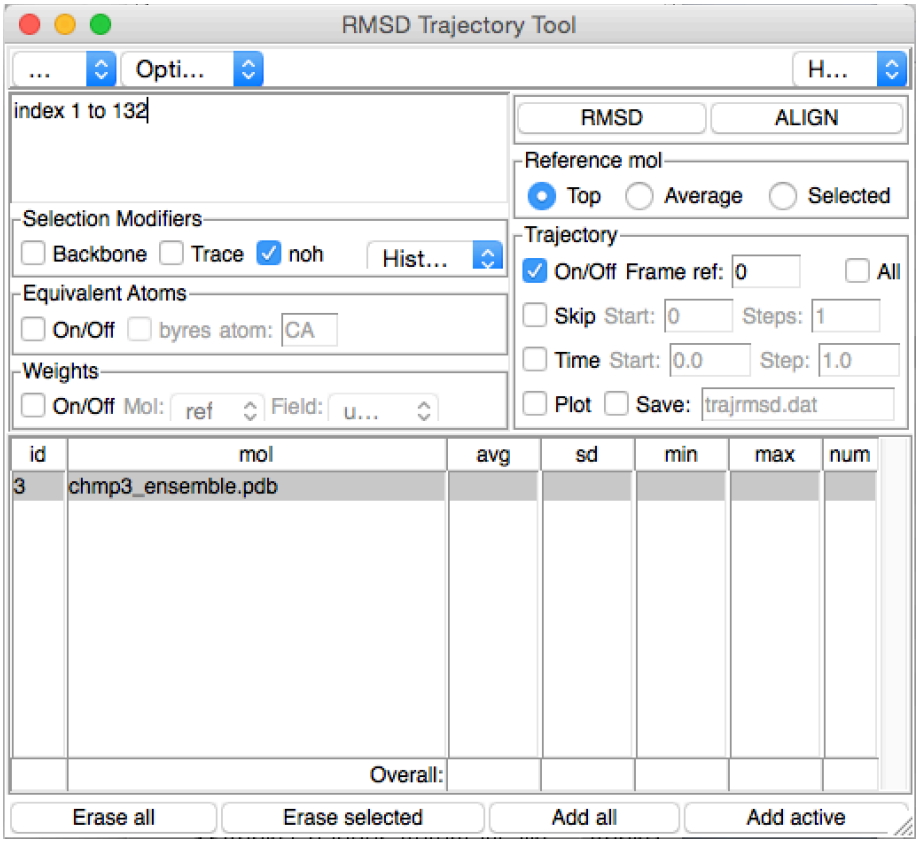
You will find it in the drop down menues of the VMD main panel under

```
Extensions -> Analysis -> RMSD Trajectory Tool
```

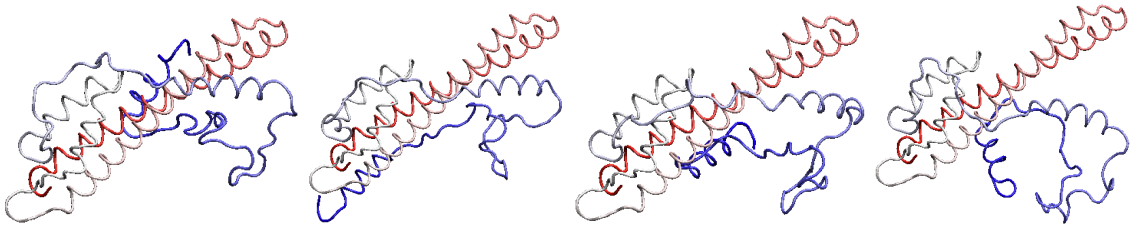


- First click ‘erase all’ to remove all automatically generated input.
- Then write `index 1 to 132` in the small upper left window. This will select only the large rigid stalk domain for alignment of the structure ensemble.
- Now click the ‘Add all’ button in the bottom of the panel. The selection will appear in the list of the panel.
- Finally, click the ‘ALIGN’ button in the upper right corner.
-

Your ‘RMSD Trajectory Tool’ panel should now look like this:



And your ensemble trajectory should now look like this:



This visualization now gives a much better overview over the actual conformational sampling. It is clearly visible that the stalk domain stays rigid, while the linkers are flexible, leading to the multitude of possible conformations of the full CHMP3 structure.

Seeing all structures at the same time:

At times an other visualization option might come in handy: Having a look at all the generated structures together at the same time.

You will achieve this by splitting up the ensemble into individual structures and showing them individually. Scripts are provided for this task. Run the script `split.sh` in the folder

```
$YOURTUTORIALPATH/0_visualize_structureEnsemble
```

by typing

```
./split.sh
```

This script will write the individual structures of the ensemble into the folder

```
0_visualize_structureEnsemble/1_nonAligned
```

You can visualize the first 100 structures of the ensemble at the same time by using the VMD script

```
2_vmd_showTogether_nonAligned.tcl
```

The result should look similar to Figure 4 (left). In using the RMSD Trajectory tool as we used it before, you will get a view as Figure 4 (right).

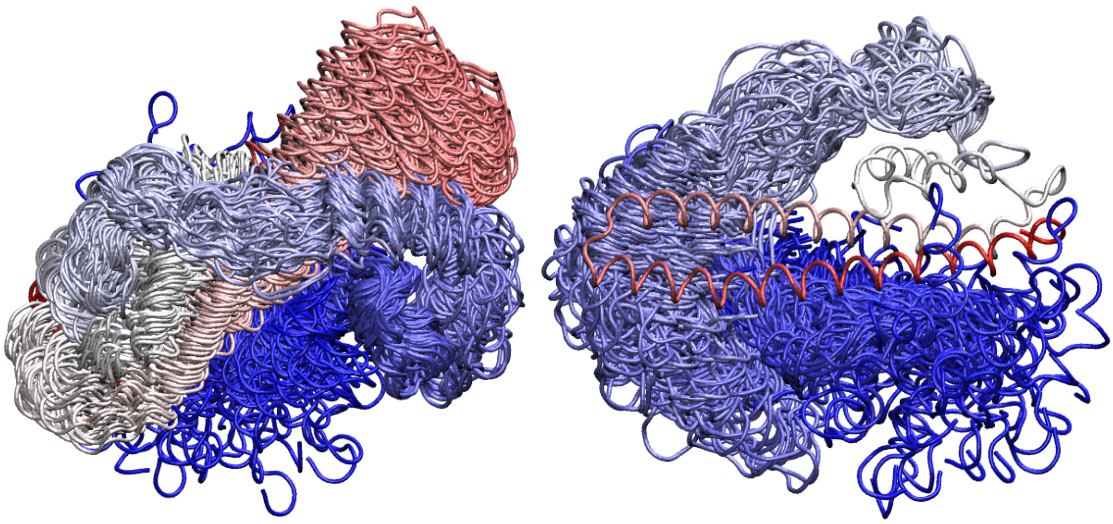


Figure 4. **Aggregated view on the structure ensemble of CHMP3**. The first 100 structures from the structure ensemble are plotted on top of each other. The left hand side shows the non-aligned ensemble, the right hand side shows the aligned ensemble. Colors indicate the index of the structure from red (low) to blue (high). Note the rigid stalk domain within the envelope of multiple conformations of the flexible linkers.

Wrapup

In this section of the tutorial, we met CHMP3 and could observe that its intrinsically flexible parts are poorly or not resolved in the crystal structure (Figure 1). We saw a model of CHMP3 built of rigid domains connected by flexible linkers (Figure 2) that comprises the full protein. This model served as input for generating a structure ensemble by coarse grained Monte Carlo sampling as in Kim and Hummer 2008 (Figure 3). Eventually, we had a look at the structure ensemble and could appreciate the intrinsic conformational flexibility of CHMP3 (Figure 4).

In the next steps of the tutorial, we will learn how this ensemble of structures is combined with experimental data to determine the most prominent conformations of the protein.

1_calculate_observable_SAX

This step of the tutorial builds upon the previous step in which we generated a structure ensemble of the full CHMP3 protein. We will eventually combine our generated structure ensemble with experimentally determined SAX data. In order to do this, we have to compute the expected SAX scattering profiles for each protein in the generated structure ensemble.
Please refer to Putnam et al.[7] For further reading of the computational methods to calculate SAX scattering profiles computationally.

This part of the tutorial is contained in the folder

```
$YOURTUTORIALPATH/1_calculate_observable_SAX
```

Switch to this folder by typing in your terminal (replace `$YOURTUTORIALPATH` by the local path on your machine to where you unpacked the tutorial):

```
cd $YOURTUTORIALPATH/1_calculate_observable_SAX
```

Now, you can compute the SAXS profiles for every structure in the ensembly by running the `run.sh` script

```
./run
```

This will take several minutes to compute. The method relies on spherical harmonics to calculate the SAXS profiles.

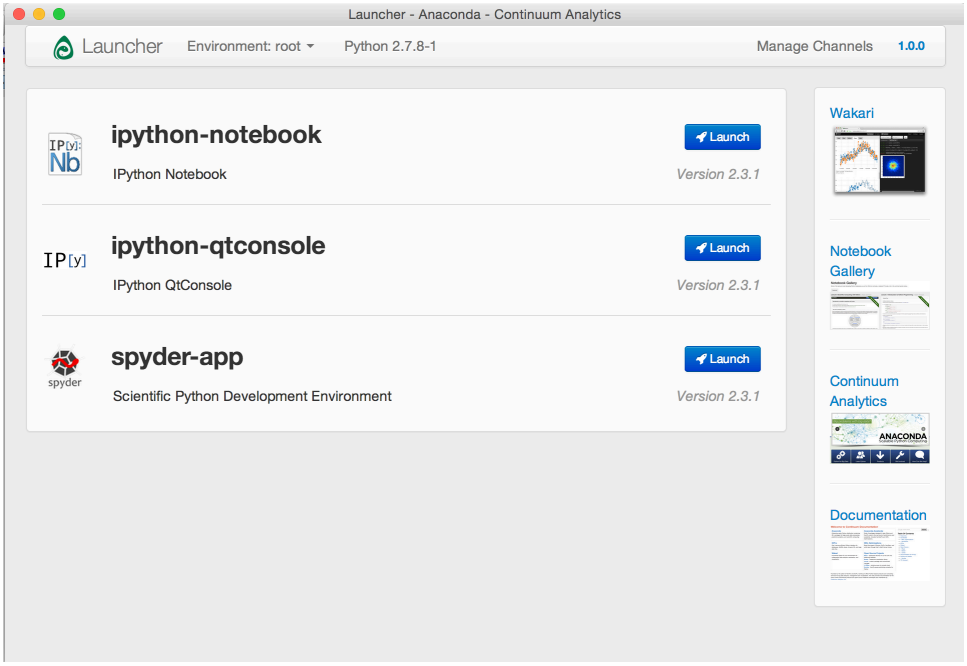
Plot the calcualted SAXS profiles.

Once the computation is finished, you can visualize the result using IPython. We assume in the tutorial that you have the Anaconda IPython distribution installed on your computer.

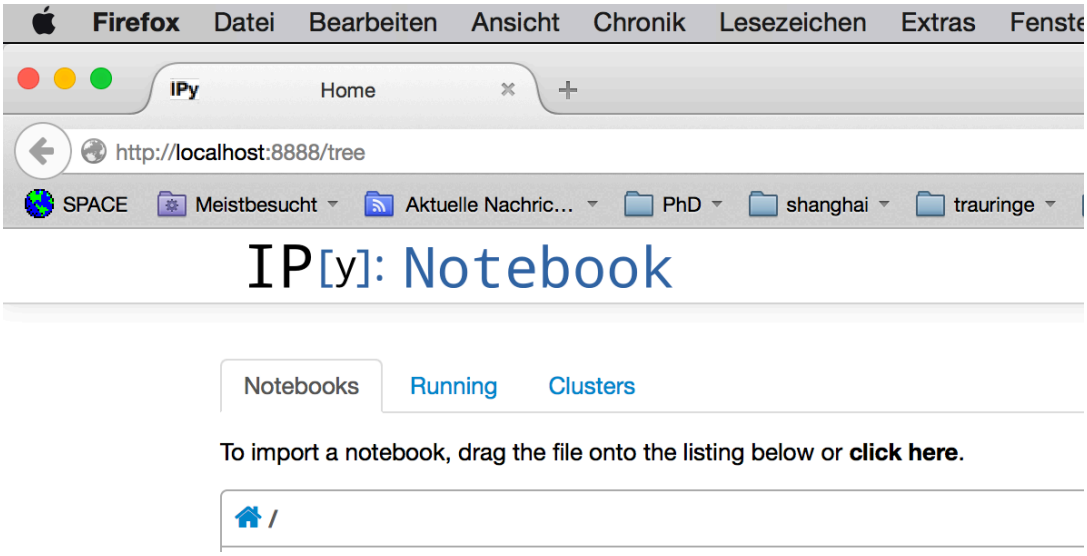
Open IPython in your web browser by using the Anaconda Launcher. This will look similar to the following picture on a Mac.

Click the 'Launch' button for IPython-notebook.

[7] Putnam, D. K.; Lowe Jr., E. W.; Meiler, J.; *Comput. Struct. Biotechnol. J.* **2013**, 8



In the browser, you should see the IPython screen similar to this:



Now browse in the file system to your tutorial folder and eventually into the folder

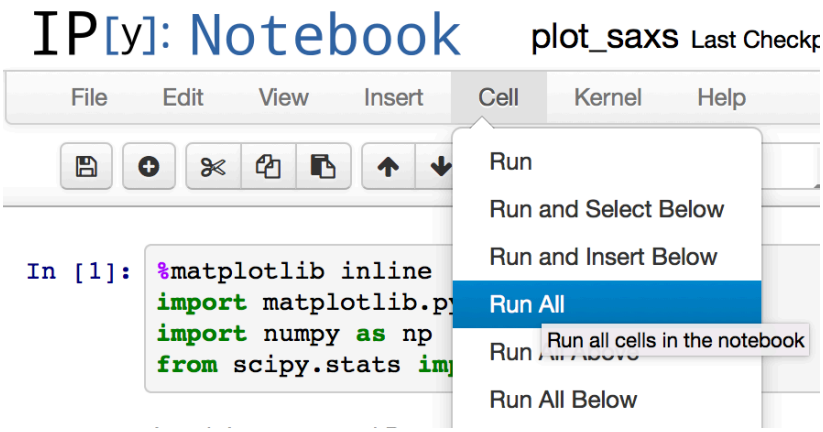
```
$YOURTUTORIALPATH/1_calculate_observable_SAX
```

Once you have done that, a single IPython notebook should appear similar to the following picture:

Click on the notebook to open it.

If you have computed the SAXS profiles as described above, you can run the notebook and plot the SAXS profiles by clicking

```
Cell -> Run All
```



The resulting plot should look similar to Figure 5.

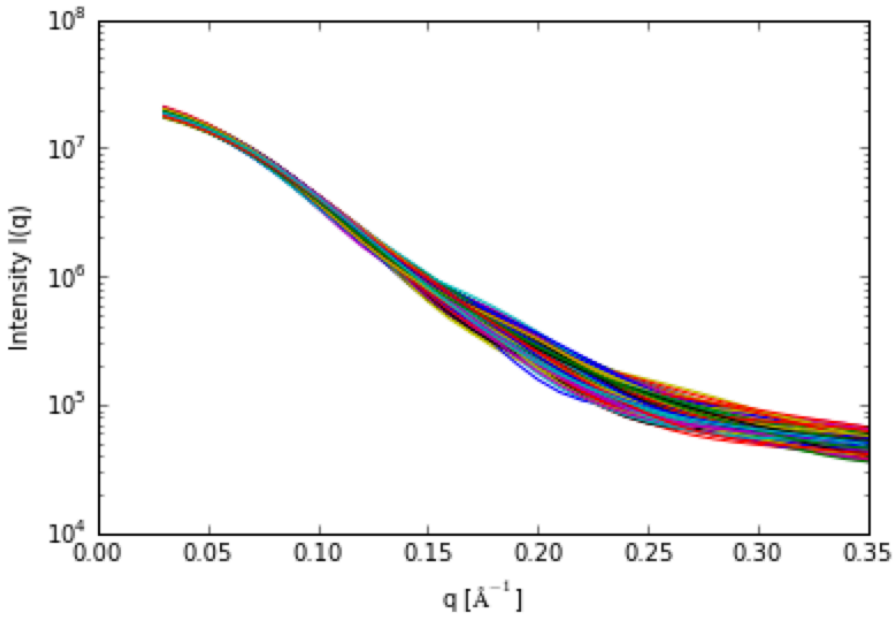


Figure 5. **Calculated SAXS profiles for each structure in the CHMP3 ensemble**. Plotted are the computed intensity profiles for each of the 1000 structures in the CHMP3 structure ensemble. Each color represents an individual structure.

Wrapup

In this section, we calculated the expected SAXS profiles for each structure in the generated CHMP3 structure ensemble. This is a prerequisite step for step 3, the fitting and reweighting according to experimentally determined SAXS data.

2_1_calculate_distances

In this step, we are calculating the distances between different structures of the generated CHMP3 structure ensemble. Step `0_visualize_structureEnsemble` is a prerequisite for this step of the tutorial.

Calculating a distance between two different protein structures is a challenging task. A commonly used method is the root mean squared deviation (RMSD) of all atom positions. This method, despite its advantages has also its drawbacks. In our case, we have a model that consists of rigid domains and flexible linkers in between. We are therefore most interested in the different placement of the rigid domains with respect towards each other. Consequently, we are using the DRMS metric between two different structures g and h:

$$\mathrm{DRMS}(g,h) = \left(\frac{1}{N_2} \sum_{n,m} \left(d_{n,m}^{(g)} - d_{n,m}^{(h)} \right)^2 \right)^{1/2}$$

where $d_{n,m}^{(g)}$ is the Cartesian distance of the α-carbon atoms of residues n and m in two different rigid domains of structure g, and N_2 is the number of residue pairs over which the sum is performed.

All distance calculations are performed in the folder

```
$YOURTUTORIALPATH/2_1_calculate_distances
```

To calculate the distance, run the script `run.sh` by typing in the Terminal

```
./run.sh
```

It will take a while to calculate all distances between the 1000 structures of the ensemble (i.e. a 1000 x 1000 matrix).

Plot the results

To plot the results, we will make use of IPython again.

Launch the IPython notebook viewer as shown above and navigate to the tutorial folder. Here navigate to the calculate distances folder and open the IPython notebook:

```
$YOURTUTORIALPATH/2_1_calculate_distances/plot_distances.ipynb
```

Run the notebook. The resulting distance matrix visualization should look similar to Figure 6.

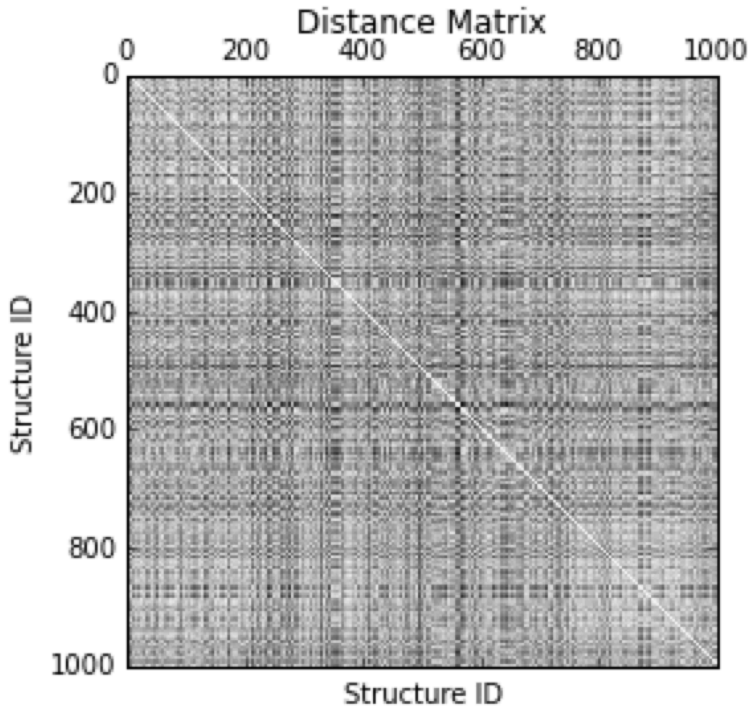


Figure 6. **Calculated DRMS distances for the structure ensemble**. The distance matrix for the DRMS distances between all pairs of structures in the ensemble is depicted. Darker values correspond to larger distances, smaller values to smaller distances. The diagonal shows zero distance between the same structures.

Wrapup

In this section we calculated the mutual distanes between all structures of the ensemble (Figure 6) as a prerequisite for the clustering of structures.

2_2_cluster

In this step, the structure ensemble will be clustered according to the DRMS distance between the structures. We are going to use Quality Treshold (QT) clustering. This clustering method applies a hard threshold (quality control) for clustering. In our case, this theshold will be 0.5 nm.

The structure clustering will be done in the folder

```
$YOURTUTORIALPATH/2_2_cluster
```

Move to this folder and run the clustering by running the `run.sh` script:

```
./run
```

To plot the results, use IPython: Move to the clustering folder and open the IPython notebook

```
$YOURTUTORIALPATH/2_2_cluster/plot_clusters.ipynb
```

Run the notebook and have a look at the results of the clustering. You should get a similar result as depicted in Figure 7.

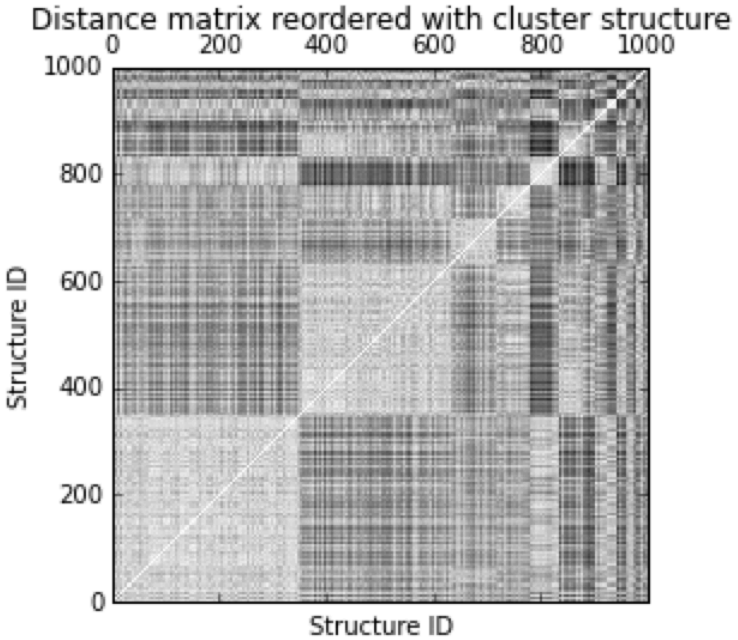


Figure 7. **Clustering of the structures according to DRMS distance**. QT clustering with a threshold of 0.5 nm was applied to the DRMS distances between the structures of the ensemble. 22 Clusters have been found for this ensemble. Note that two large clusters each comprise about one third of the whole ensemble. The other 20 clusters are much less occupied.

3_reweighting_SAX

In this step, we will compare the computed SAXS intensities with the experimentally measured SAXS intensity. Furthermore, we will improve the aggreement of the calculated with the experimental data by performing a reweighting of the computed SAXS intensities. Overfitting will be prevented by using the maximum-entropy method (Jaynes 1957).

Comparison of the Simulated structures with the experimental result without fitting

The SAXS intensity of the simulated structure ensemble can be computed as follows:

$$I_{sim}(q_i) = \sum_{k=1}^{N_c} w_k I_k(q_i)$$

With $I_k(q_i)$ indicating the arithmetic mean of the calculated SAXS intensity of cluster k, w_k the the normalized weight of cluster k and N_c the total number of clusters.

We can compute the discrepancy between the calculated and the experimental SAXS by

$$\chi^2 = N_q^{-1} \sum_{i=1}^{N_q} (cI_{sim}(q_i) - I_{exp}(q_i))^2 / \sigma^2(q_i)$$

with N_q the number of data points of the SAXS curve, $\sigma^2(q_i)$ the experimental error and scale factor c.

There is a IPython notebook in which you can calculate the χ^2:

```
$YOURTUTORIALPATH/3_reweighting_SAX/fitweights.ipynb
```

Despite the name, the notebook is prepared such that it will not perform a fitting initially. If you run it, it will compute the initial χ^2 of the structure ensemble vs the experimental data. It will look similar to the following figure:

```
print 'chi2 initial = {}'.format(calc_chi2(w0, Iclu, Iexp))
print 'G initial = {}'.format(energy(w0, w0, Iclu, Iexp, 1))
print 'have to be equal'
```

```
chi2 initial = 0.104187272231
```

This discrepancy is already very low.
To get a visual impression what this discrepancy looks like, two plots are provided in the end of the notebook. They should look similar to Figure 8.

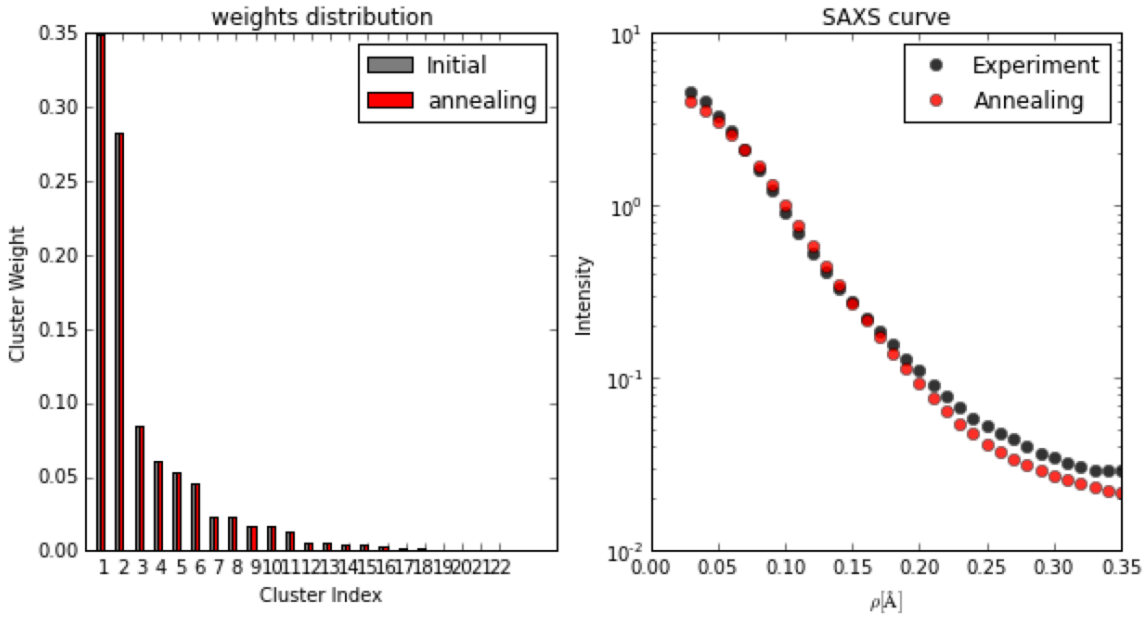


Figure 8. **Cluster weights and comparison of the SAXS intensities without fitting**. The left plot shows the weights w_k of the respective clusters before and after fitting. Since no fitting was performed here, both are equal. The right plot shows the SAXS intensity profiles of the experiment (black) and the simulated ensemble (red). Since no fitting and therefore no reweighting was performed, the red curve corresponds to the unaltered simulated SAXS intensity. Note that the discrepancy is already very low between the simulated structures and the experiment, indicating that the simulation and the subsequent SAXS computation can capture the major dynamics of the intrinsic flexibility of CHMP3 very well.

Application of fitting to improve the aggreement between experiment and simulation.

In the next step, a fitting procedure is applied to enhance the aggreement between experiment and simulation. The weights in the calculation of χ^2 get varied randomly and a new configuration of weights is accepted if the new configuration results in a smaller χ^2 or with probability exp(-delta(χ^2)).

In order to prevent overfitting, the maximum entropy method is used:

Now, it is no longer the χ^2 that is optimized but a 'free energy' function

$$G = \chi^2 - \theta S$$

with overfitting control parameter θ and entropy S:

$$S = -\sum_{k=1}^{N_c} w_k \ln \frac{w_k}{w_k^{(0)}}$$

with w_k the vector of newly found cluster weights that are to be tested and $w_k^{(0)}$ the initial cluster weights. Note here, that if $w_k = w_k^{(0)}$, S = 0.

It has been determined, that the energy function used for structure simulation has an approximate error of 2kT. Consequently we allow a fitting towards weights that correspond in a +- 2kT change. This corresponds to $\theta = 0.02$.

To apply this fitting methodology, we can again use the IPython notebook

```
$YOURTUTORIALPATH/3_reweighting_SAX/fitweights.ipynb
```

This time we have to change the value θ from 1000 to 0.02:

```
args = (w0, Iclu, Iexp, 0.02)
w_final = annealing(energy, w0.copy(), 0.01, step_w, Nrun=(Nclu * 1),
                    verbose=True, args=args)
```

```
999 0.0688768910785
```

The resulting weight distribution and SAXS profiles should look similar to Figure 9.

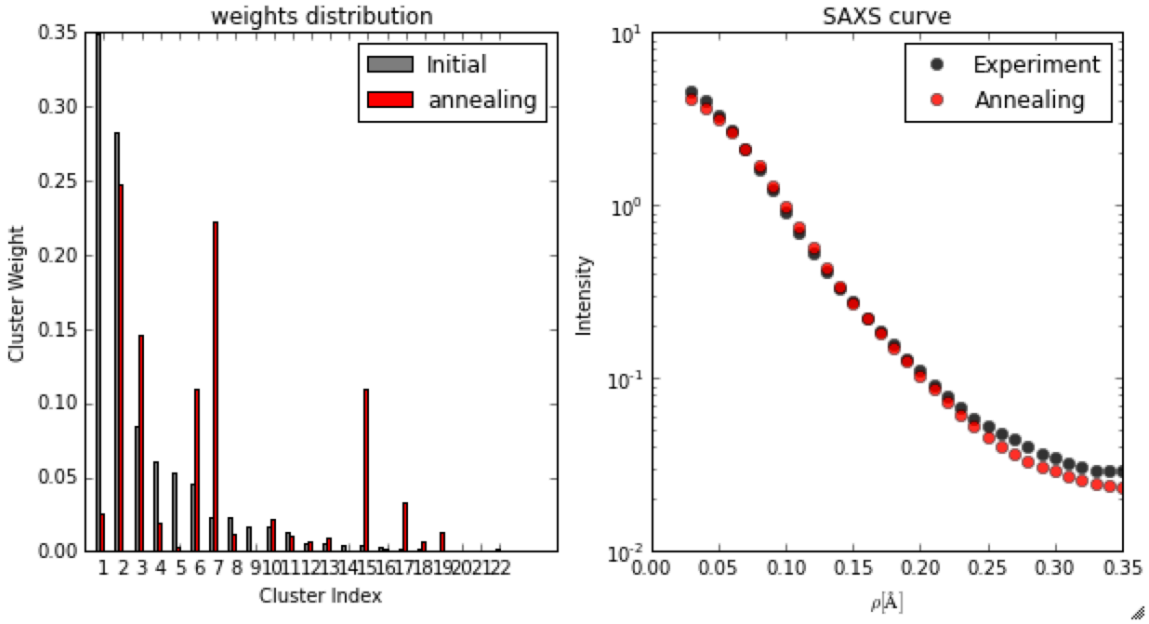


Figure 9. **Cluster weights and comparison of the SAXS intensities using fitting.** By adjusting the weights in the fitting procedure with a treshold of 2kT, the experimental SAXS curve is even better approximated.

4_analyze

In this final step, we will visualize the results of the tutorial: Clustering of the structures allowed us to calculate an expected average SAXS profile for the simulated structures that we could compare to the measured SAXS profile. Figure 8 showed, that two clusters dominate the ensemble and that the calculated average SAXS profiles correspond already very well with the experiment.
Fitting at the maximally allowed threshold shifted the weights an placed clusters 2 and 7 to be the most dominant clusters instead of clusters 1 and 2.
In this step, we will have a look at the structures of the cluster centers and will see what structures most likely dominate the ensemble.

This step of the tutorial is contained in the folder

```
$YOURTUTORIALPATH/4_analyze
```

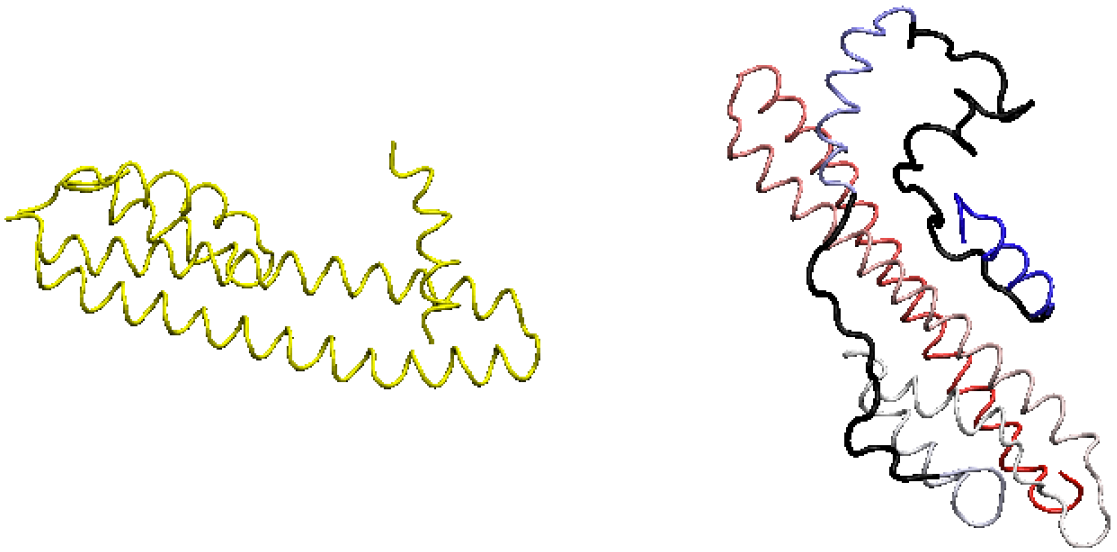
Go to this folder in the Terminal and run both scripts:

```
./configure
./run
```

A new VMD script has been generated by these actions:

```
0_vmd_show_weightedClusterCenters+init.tcl
```

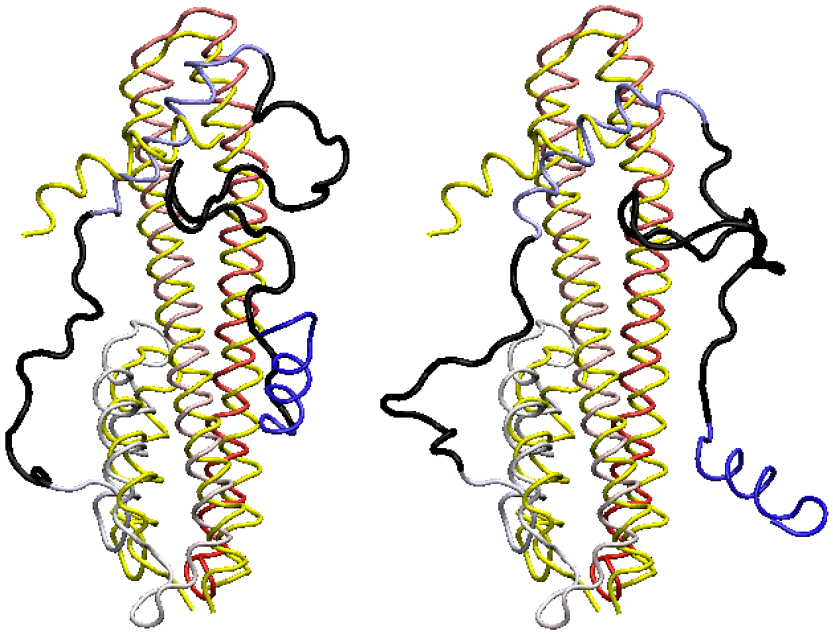
Run it with VMD. You will see a representation similar to the following:



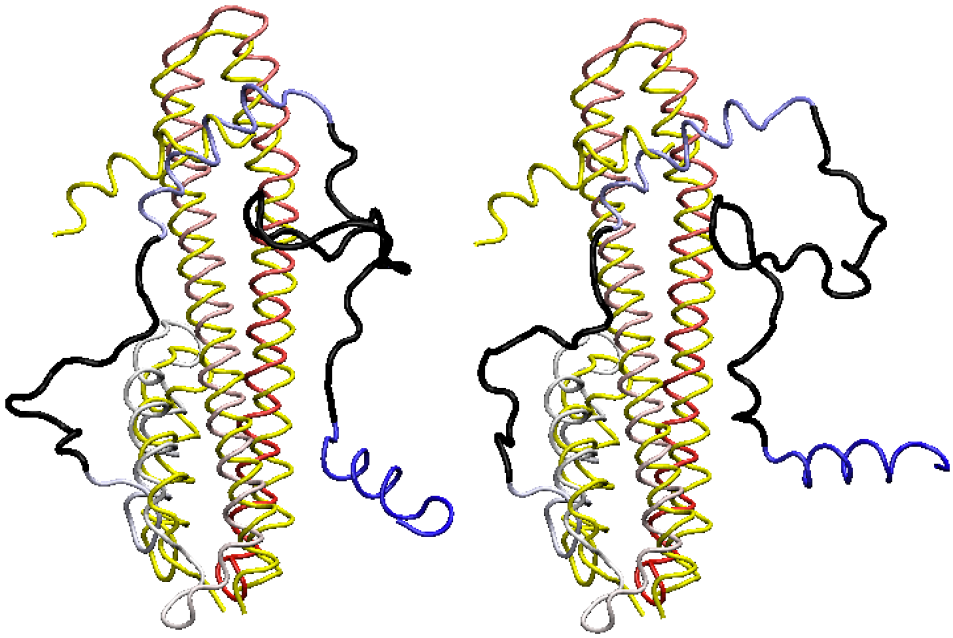
Depicted are the crystal structure in yellow, and the 22 cluster center structures, ordered by decreasing weight.

Align the crystal structure and the cluster center structures by the ‘RMSD Trajectory Tool’

For the unfitted ensemble, the two dominant structures (cluster centers of cluster 1 and 2) look as follows:



For the fitted ensemble, the two dominant structures (cluster centers of cluster 2 and 7) look as follows:



Make sure that you run the `4_analyze/run.sh` after you have performed a new fit.

Wrapup and closing remarks

We are at the end of our tutorial. Although you could not generate the structure ensemble yourself, you were able to calculate SAXS profiles for the ensemble, perform a distance calculation and clustering on them and eventually compare and fit the calculated ensemble structures to experimental data.

We have seen, that CHMP3 most likely assumes closed conformations in the low salt conditions, with helix 6 being attached to the stalk domain.

This combination of computational modeling with experiments allows to bring both methods to a higher level. The computationally generated model is refined by the experiment to result in a method that can give atomistic insights into the structure ensemble of intrinsically disordered proteins.

Further steps from here

If you have enjoyed the tutorial and would like to proceed a little further, the same analysis has also been performed on CHMP3 under very high salt conditions. An ensemble generated under such conditions is provided:

```
$YOURTUTORIALPATH/_data/chmp3_ensemble_highSalt_0.pdb
```

If you are curious how these conditions alter the conformation sampling and the full procedure, re-run the tutorial by using this ensemble instead of one of the low salt ensembles. The only change you have to make is to alter the line

```
exp_datafile = '../_data/SAXS_LowSalt.txt'
```

to

```
exp_datafile = '../_data/SAXS_HighSalt.txt'
```

in

```
$YOURTUTORIALPATH/3_reweighting_SAX/fitweights.ipynb
```

to make sure, that you are comparing to experimental data that were measured at high salt concentrations.

Feedback

If you have any suggestions, found errors or typos, please contact us and let us know about your feedback. We will very much welcome it. Thanks!