

Intersection of multiscale MD and NMR

Kevin Sawade

Universität Konstanz

kevin.sawade@uni-konstanz.de

October 1, 2021

Overview

Data Overview

Available data

Methods and Results

Analysis Scheme

Cluster Analysis

General shape of the diUbi proteins

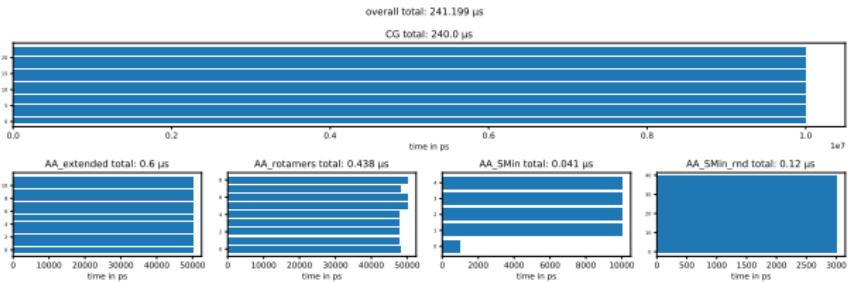
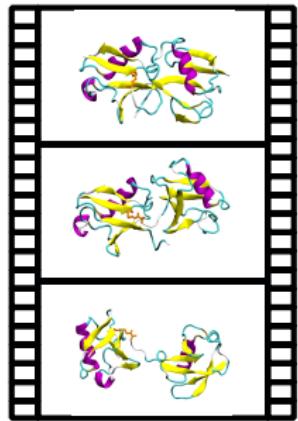
General shape of the diUbi proteins

Second Section

Data available from previous publications

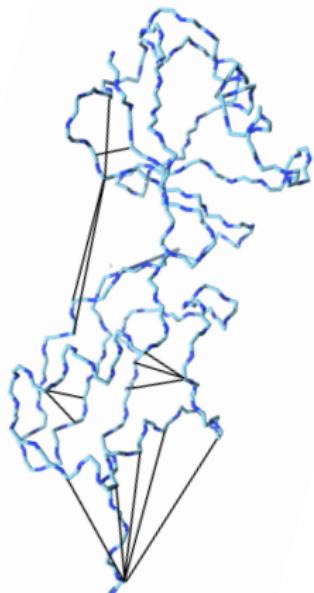
1. All-atom MD simulations using an altered GROMOS54a7 forcefield. Started from extended conformations.
2. Coarse-grained MD simulations using MARTINI v2.2 ff.
3. All-atom MD simulations started from the 4 lowest sketch-map basins of the CG map using BACKWARD.
4. All-atom MD simulations started from 10 random points around the 4 lowest sketch-map basins of the CG map using BACKWARD.
5. All-atom MD simulations started from χ_3 -rotamers of extended structures.

Overview



Analysis Steps

1. Extract high-dimensional CVs from all simulation frames.



$$\text{distances AB} \left\{ \begin{array}{c} x_1 \\ \vdots \\ x_{72} \end{array} \right(\begin{array}{ccc} d_{a_1,b_1} & \cdots & d_{a_1,b_{72}} \\ \vdots & \ddots & \vdots \\ d_{a_{72},b_1} & \cdots & d_{a_{72},b_{72}} \end{array} \right)$$

$$D_{A,B} =$$

$$RWMD_{A,B} = (\min(x_1), \dots \min(x_{72}), \min(y_1), \dots \min(y_{72}))$$

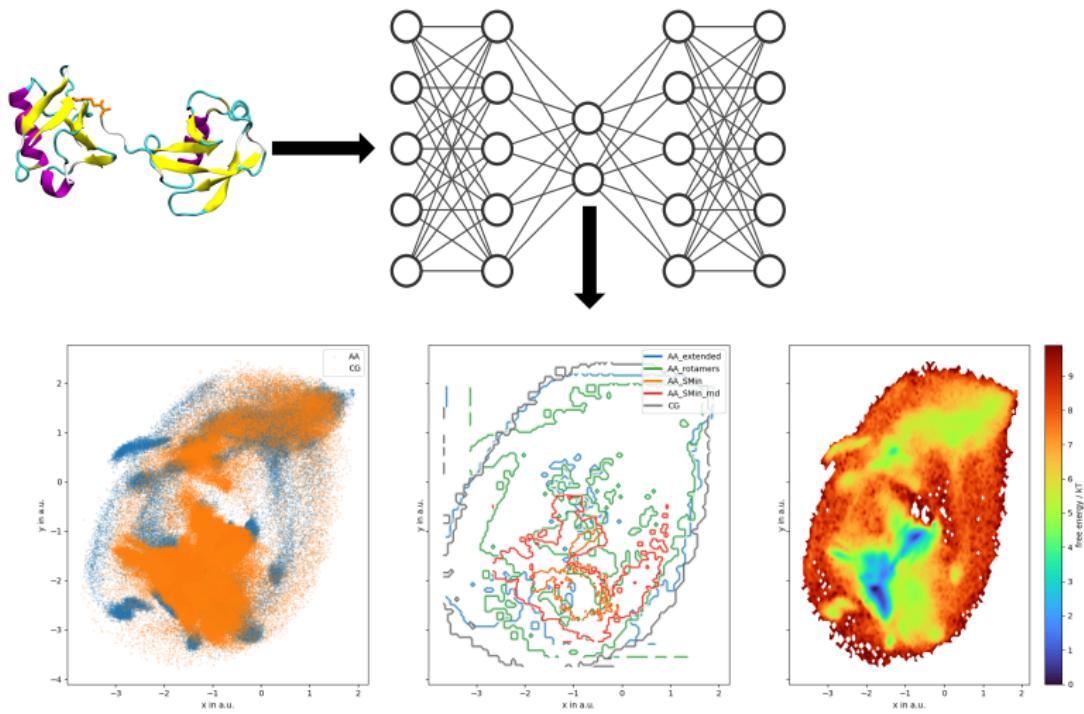
$$\text{distances BA} \left\{ \begin{array}{c} y_1 \\ \vdots \\ y_{72} \end{array} \right(\begin{array}{ccc} d_{b_1,c_1} & \cdots & d_{b_1,c_{72}} \\ \vdots & \ddots & \vdots \\ d_{b_{72},c_1} & \cdots & d_{b_{72},c_{72}} \end{array} \right)$$

$$RWMD_{B,A} = (\min(x_1), \dots \min(x_{72}), \min(y_1), \dots \min(y_{72}))$$

$$RWMD = (RWMD_{A,B} \cup RWMD_{B,A})$$

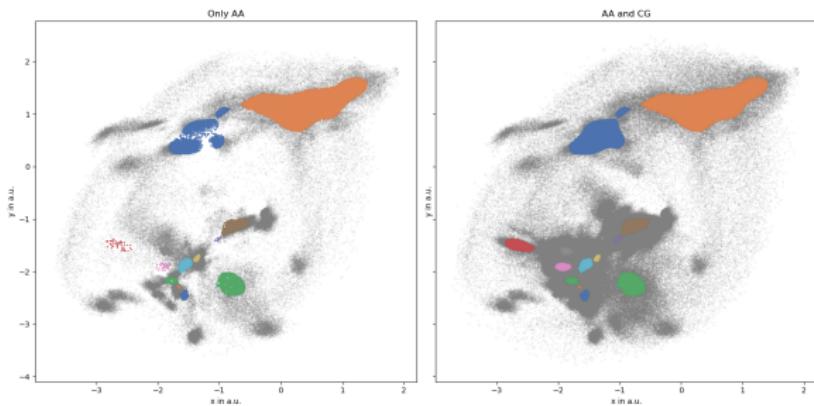
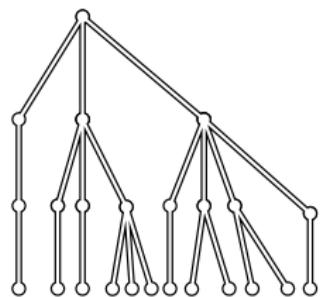
Analysis Steps

2. Run EncoderMap with the high-dimensional CVs as input data.



Analysis Steps

3. Cluster data using HDBSCAN.



Analysis Steps

3. Write a code library, that poses as an interface between current python packages and XPLOR.
(https://github.com/kevinsawade/xplor_functions)
4. Identify possible settings/arguments and define sensible defaults (solution concentration, probe radius, etc.)
5. Parallelize the scoring for faster throughput.
6. Manually parse .psf files to include 15N relaxation data.

Analysis Steps

```
psol:  
call_parameters:  
name:  
type: str  
value: psol  
descr: |
```

This is the name of the potential term assigned to this PSolPot object
It can contain any string and can be used to

restraints:

```
type: file  
value: data/diUbi_k6_sPRE_in.tbl  
descr: |
```

The location of the spre_tbl file that will be passed to XPLOR.

The spre table file needs to be formatted as such:

```
"f'assign (resid {resSeq:<2} and name HN) {sPRE:5.3f} {err:5.3f}"
```

So for example: For Ubiquitin the first three lines of that table look like this:

```
assign (resid 2 and name HN) 5.510 0.711  
assign (resid 3 and name HN) 1.223 1.816  
assign (resid 4 and name HN) 4.381 0.402
```

tauc:

```
type: float
```

```
value: 0.2
```

descr: correlation time

probeR:

```
type: float
```

```
value: 3.5
```

descr: radius of probe molecule

probeC:

```
type: float
```

```
value: 0.24
```

```
descr: probe concentration - units?
```

```
def parallel_xplor(ubq_sites, simdir='/home/andrej/Ricerca/SIMS/2017_1', n_threads='max-2',  
                  df_outdir='/home/kevin/projects/tobias_schneider/values_from_every_frame/from_package',  
                  suffix='_df_no_connect.csv', write_csv=True, fix_isopeptides=True, specific_index=None, parallel=False,  
                  subsample=5, yaml_file='', testing=False, from_tmp=False, max_len=-1, break_after=False, **kwargs):  
    """Runs xplor on many simulations in parallel.
```

This function is somewhat specific and there are some hardcoded directories it. It uses MDTraj and OpenMM to load trajectories from Andrej's sim directory (/home/andrej/Ricerca/SIMS/). These trajectories are provided in a joblib Parallel/delayed construct to 'get_series_from_mdtraj', which results in a list of pandas Series, that are stacked to a long dataframe.

The dataframe is periodically saved (to not loose anything). Check out the function 'xplor.delete_old_csvs' to remove the unwanted intermediate csvs, produced by this function.

Args:

```
ubq_sites (list): A list of ubiquitination sites, that should be recognized.
```

Keyword Args:

```
simdir (str, optional): Path to the sims, that contain the ubq_site substring.  
    Defaults to '/home/andrej/Ricerca/SIMS/2017_1'.  
n_threads (Union[int, str], optional): The number of threads to run.  
    Can be an int, but also 'max' or 'max-2', where 'max' will give  
    make this function use the maximum number of cores. 'max-2' will use  
    all but 2 cores. Defaults to 'max-2'  
df_outdir (str, optional): Where to save the csv files to. Defaults to  
    '/home/kevin/projects/tobias_schneider/values_from_every_frame/from_package'.  
suffix (str, optional): Suffix of the csv files, used to sort different  
    runs. Defaults to '_df_no_connect.csv'.  
write_csv (bool, optional): Whether to write the csv to disk. Defaults  
    to True.  
subsample (int, optional): Whether to subsample trajectories. Give an  
    int and only use every 'subsample'-th frame. Defaults to 5.  
max_len (int, optional): Only go to that maximum length of a trajectory.  
    Defaults to -1, which will use the full length of the trajectories.  
yaml_file (str, optional): Path to a yaml file. If an empty string is provided  
    the defaults.yaml file from xplor/data is loaded. Defaults to ''.  
from_tmp (bool, optional): Changes the executable of the command to  
    work with an ssh interpretor to 134.34.112.158. If set to false,  
    the executable will be taken from xplor/scripts.  
    Defaults to False  
testing (bool, optional): Adds the '-testing' flag to the command.  
    Defaults to False.  
specific_index (Union[int, None], optional): If given, only that Index will  
    be used in the parallel loop. For Debugging. Defaults to None.  
parallel (bool): Whether to do the calculations in parallel. Defaults to False.  
break_after (Union[bool, int], optional): Whether to break the for loop early, stopping the  
    calculation after the specified number of loops. Defaults to False.  
fix_isopeptides (Union[bool, int], optional): Whether to fix isopeptide bonds using the  
    technique developed in 'check_connect'. If set to True all calculations use this  
    technique. Can also take an int, that is larger than 'subsample', in that  
    case, only every 'fix_isopeptides' frame will use this technique, the other  
    frames will use the old, faster protocol. Defaults to 25.  
**kwargs: Arbitrary keyword arguments. Keywords that are not flags  
    of the xplor/scripts/xplor_single_struct_script.py will be discarded.
```

Returns:

```
pd.DataFrame: A pandas Dataframe.
```

..

```
if n_threads == 'max-2':  
    n_threads = multiprocessing.cpu_count() - 2  
elif n_threads == 'max':  
    n_threads = multiprocessing.cpu_count()
```

```
if np.any(fix_isopeptides):
```



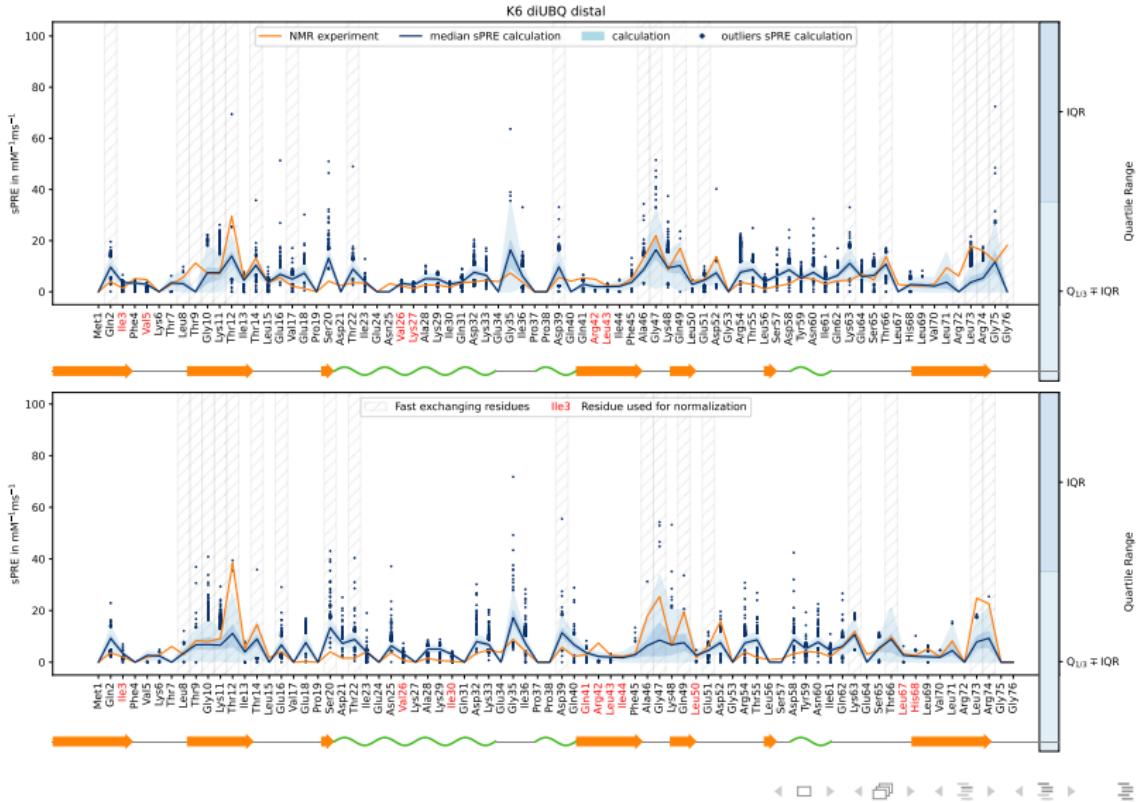
Analysis Steps

8. Normalize sPRE computations.

- ▶ Don't consider the fast-exchanging residues.
- ▶ Consider proximal and distal unit separately.
- ▶ From all simulation frames calculate the variance of the sPRE values for this residue.
- ▶ Take the 10 (not fast exchanging) residues with the smallest variances.
- ▶ Calculate the factor f_i from $f_i = \frac{v_{i,\text{exp}}}{v_{i,\text{sim}}}$ for every of these 10 residues.
- ▶ Calculate the mean of these ten factors as $F = \frac{\sum f_i}{N}$
- ▶ Use F as a factor to normalize the sPRE values of the proximal unit.

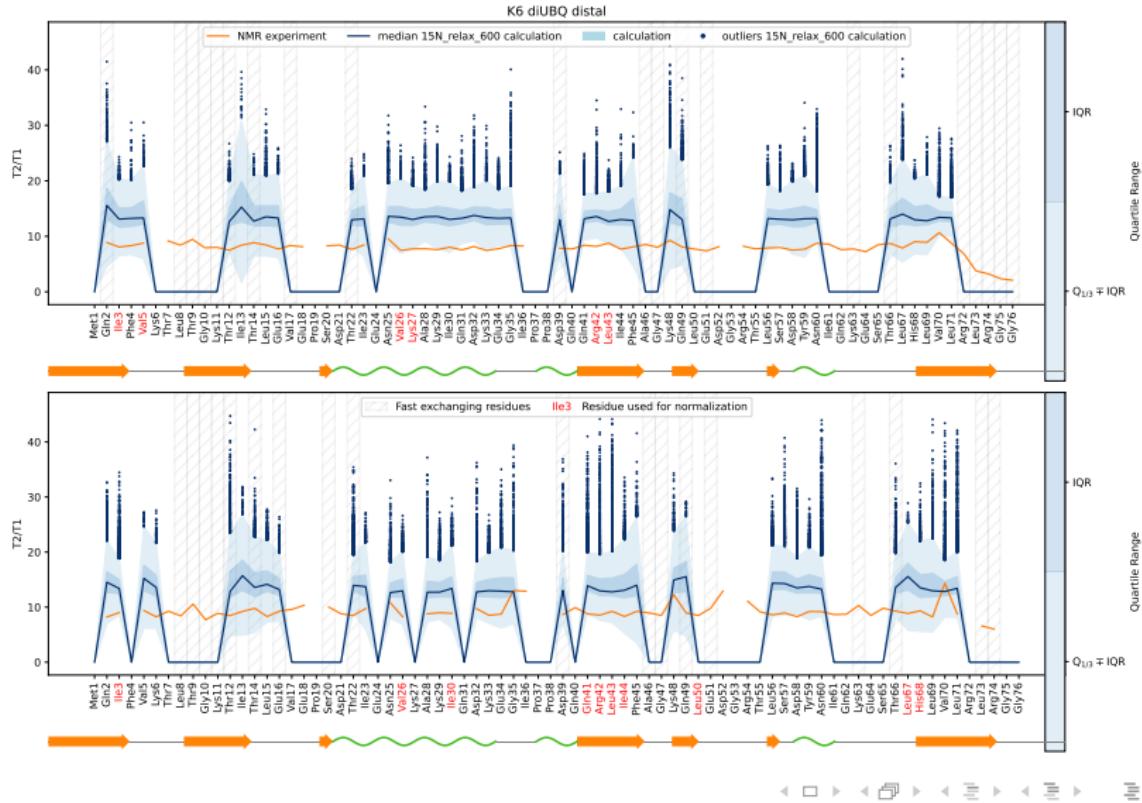
Analysis Steps

9. Calculate sPRE for AA conformations.



Analysis Steps

10. Calculate ^{15}N relaxations for AA conformations.



Cluster coefficients

Solve:

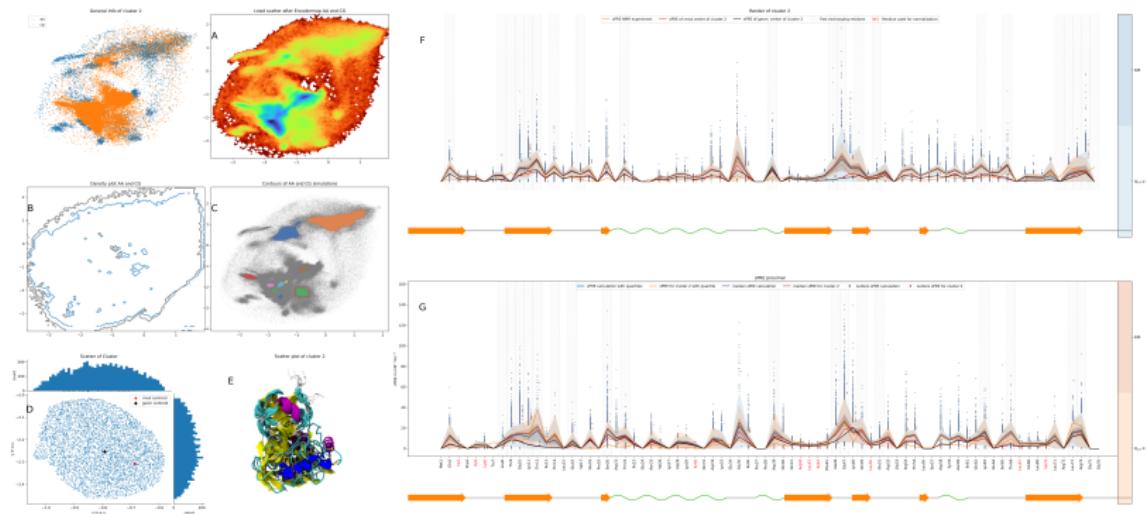
$$\begin{bmatrix} v_{exp, MET1} \\ v_{exp, GLN2} \\ \vdots \\ v_{exp, GLY76} \end{bmatrix} = x_1 \cdot \begin{bmatrix} v_{clu_1, MET1} \\ v_{clu_1, GLN2} \\ \vdots \\ v_{clu_1, GLY76} \end{bmatrix} + x_2 \cdot \begin{bmatrix} v_{clu_2, MET1} \\ v_{clu_2, GLN2} \\ \vdots \\ v_{clu_2, GLY76} \end{bmatrix} + \dots + x_n \cdot \begin{bmatrix} v_{clu_n, MET1} \\ v_{clu_n, GLN2} \\ \vdots \\ v_{clu_n, GLY76} \end{bmatrix}$$

for $(x_1, x_2, \dots, x_n) \in \mathbb{N}$

Single cluster

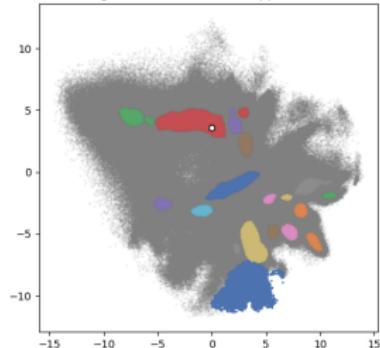
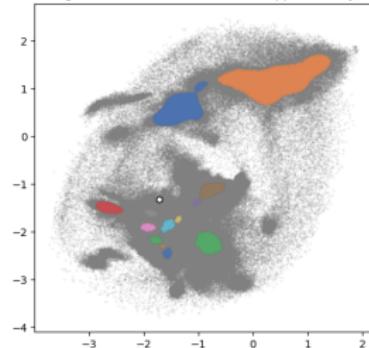
A cluster is extracted from the complete (CG and AA) ensemble but does only contain AA conformations.

- Its contribution to the whole ensemble.
- A coefficient from the linear combination of clusters.

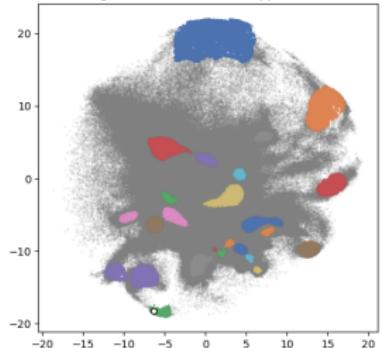


Is the best fitting structure in a cluster?

Best fitting conformation for k6 does not appear in any cluster. Best fitting conformation for k29 appears in cluster 3

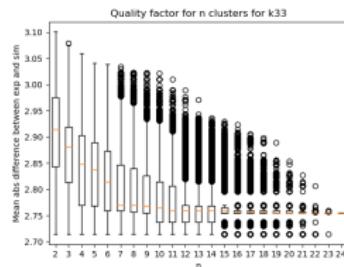
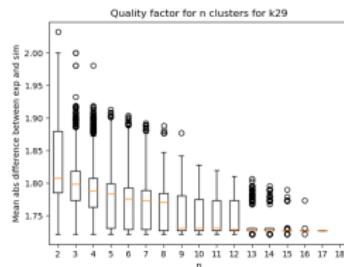
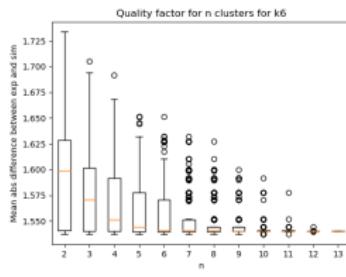


Best fitting conformation for k33 appears in cluster 2

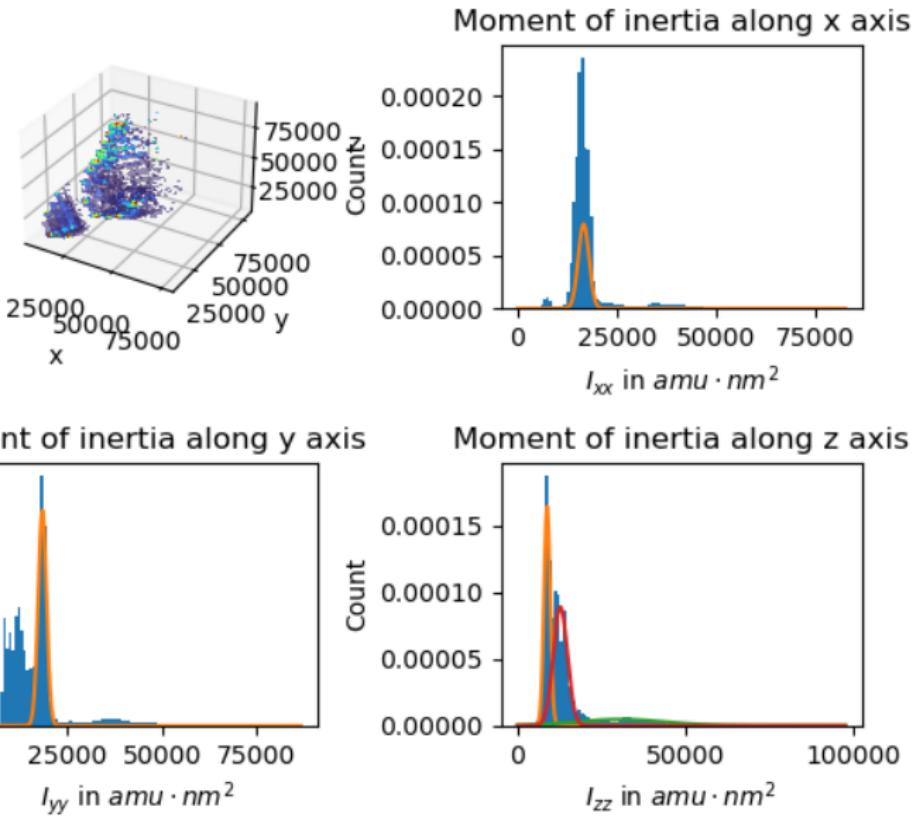


Multiple clusters

Multiple clusters better represent the nature of the sPRE ensemble.
A large coefficient in the linear combination does not necessarily mean that this cluster has similar sPRE values as the experiment.

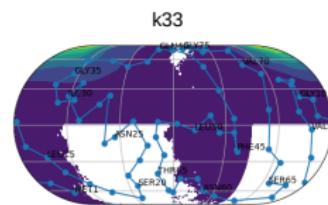
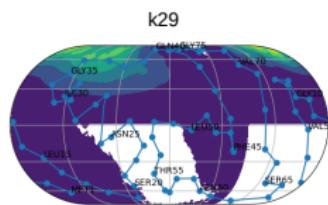
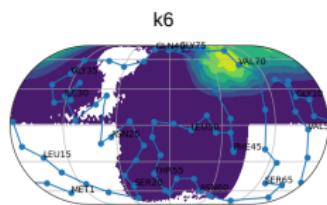


Test Tensors of inertia

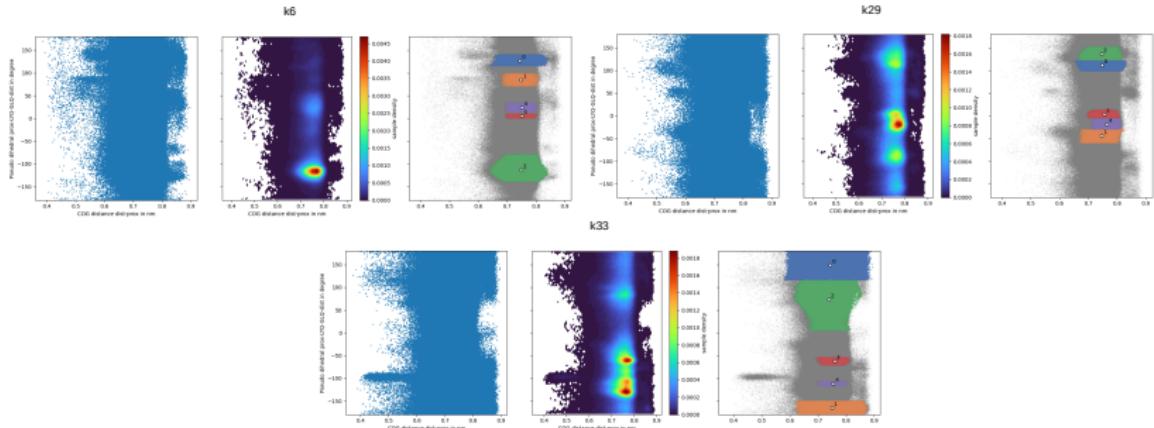


Render of a structures within a 1_{xx} gaussian

Surface coverage might also be a possibility.



Better: pseudo-dihedral and cog-distance



Blocks of Highlighted Text

Block 1

 Lorem ipsum dolor sit amet, consectetur adipiscing elit. Integer lectus nisl, ultricies in feugiat rutrum, porttitor sit amet augue. Aliquam ut tortor mauris. Sed volutpat ante purus, quis accumsan dolor.

Block 2

 Pellentesque sed tellus purus. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos. Vestibulum quis magna at risus dictum tempor eu vitae velit.

Block 3

 Suspendisse tincidunt sagittis gravida. Curabitur condimentum, enim sed venenatis rutrum, ipsum neque consectetur orci, sed blandit justo nisi ac lacus.

Multiple Columns

Heading

1. Statement
2. Explanation
3. Example

Lorem ipsum dolor sit amet,
consectetur adipiscing elit.
Integer lectus nisl, ultricies in
feugiat rutrum, porttitor sit amet
augue. Aliquam ut tortor mauris.
Sed volutpat ante purus, quis
accumsan dolor.

Table

Treatments	Response 1	Response 2
Treatment 1	0.0003262	0.562
Treatment 2	0.0015681	0.910
Treatment 3	0.0009271	0.296

Table: Table caption

Theorem

Theorem (Mass–energy equivalence)

$$E = mc^2$$

Verbatim

Example (Theorem Slide Code)

```
\begin{frame}
\frametitle{Theorem}
\begin{theorem}[Mass--energy equivalence]
$E = mc^2$
\end{theorem}
\end{frame}
```

Figure

Uncomment the code on this slide to include your own image from the same directory as the template .TeX file.

Citation

An example of the \cite command to cite within the presentation:

This statement requires citation [p1].

References

mybib.bib

The End