# Identification of structure clusters in Molecular Dynamics trajectories from Residue Interaction Networks

Lorenzo Corrado

## 1 Introduction

The protein folding process is the process by which proteins, starting from their polypeptide chain, reach their tertiary structure. Today the folding process is still one of the most complex problems in molecular biology and biochemistry. We can study folding processes using computational simulation, such as Molecular Dynamics (MD). From these time-indexed simulations, we can extract some snapshots that represent the intermediate conformations taken by the protein in the folding process, from their initial to final conformation. Every single snapshot contains the conformational information of the protein at that particular moment, the following information is contained in the PDB files which contain, among other things, the three-dimensional coordinates of each atom involved. The aim of this project is to synthesize the entire trajectory of MD by determining few intermediate conformations assumed by the protein, which are representative of the entire dynamics. While there are several papers in the literature that attempt to summarize MD trajectories from these three-dimensional coordinates, our goal is to summarize starting from the protein contact maps.

## 2 Protein contact maps

A protein contact map is basically a matrix on whose sides there are the positions of the residues in the sequence of the protein: in the cell corresponding to the residues $i$ and $j$ that form a contact, the distance of their $C_\alpha$ is generally entered. The contact matrix contains all the information needed to reconstruct the structure of a protein and can be used to compare the conformations assumed by the protein. In this work, instead of building the contact maps using the distance between the $C_\alpha$ of the residues, we built the contact maps starting from the information provided by the RING software output. This software is able to determine if and which residues form contacts and the type of bond formed. So if the residue $i$ and $j$ form a contact, the corresponding cell of the matrix will have value 1, 0 otherwise. The information related to the RING output is all contained in the EDGES files, which form the starting point of this work. Furthermore, since we are comparing different conformations of the same protein it is not necessary to align the contact maps to make them comparable.

## 3 Algorithm

In this section we will explain the key points of the algorithm, following the order in which the algorithm operates.

### 3.1 Input

The algorithm receives in input essentially two arguments: a PDB file and the EDGES files list. The PDB file contains, among other things, the list of residues in the protein. We assume that this list remains the same along the entire trajectory. The second argument in input is the list of EDGES files, each of them represents a snapshot of the MD and contains, among other things, the list of residues that form a bond and the type of the bond.

### 3.2 Splitting of contact maps

With the list of residues, the algorithm is now able to generate the matrix that represents the contact map on whose sides are present the names of the residues. To make the best use of the information provided by the EDGES files, we choose to create several contact matrices for each snapshot, one for each different type of interaction. For example, if 5 different types of interaction are formed during the trajectory, 5 different contact matrices will be associated with each snapshot. This choice was made because allows us to better highlight the changes in the transition between snapshots. Now, it is also

possible to assign a different importance to each type of interaction by weighting. Furthermore, using a contact matrix for each bond type allow us to use a larger number of distance measures. We decided to use only information about the type of the bond, but not the information about the location of the bond within the protein (MC-SC) as this leads to a significant increase in the computational cost of the algorithm. Lastly, to reduce the noise in the data due to very frequent interactions, only "long" range HBOND were considered, ie bonds between residues with a chain distance greater than 11. The same was applied for VDW bonds, but as this worsened the results of the procedure all VDW were considered.

### 3.3 Distance measure

In order to obtain the distance matrices we must define a criterion, so a distance measure between contact maps. There are many criteria of distance and dissimilarity measures that can be used. The criterion chosen is the *Hamming distance*:

$$d_H = \frac{c_{01} + c_{10}}{n} \tag{1}$$

The Hamming distance is the proportion of disagreeing component of two vectors $u$ and $v$, where $c_{ij}$ is the number of ocurrences where $u[k] = i$ and $v[j] = j$ and $n$ is the length of the two vectors. This measure is often used in the context of information theory and it works quite well with very sparse data in high dimensional space, as in this case. Other distance measures can be used in these type of problems, such as Jaccard distance or the cosine similarity. Indeed, it is important to note that, in high dimensional space, the presence of an attribute is much more important than its absence. Considering the particular nature of this data we shouldn't use distances like euclidean distance because they are not able to capture the similarity in high dimensional space or with very sparse data.

### 3.4 Weighting of interactions and merging

In order to calculate a distance matrix with the formula in (1) we need to keep in mind that it is necessary to vectorize the contact matrices involved in the calculation. Once we calculated the different contact matrices for all snapshots, we can calculate the distance matrix for each type of bond. For example, if we have 5 types of interaction we will get 5 distance matrices, where each distance matrix has dimension $N$x$N$, where $N$ is the number of snasphots. Now, in order to obtain a single distance matrix, we must merge each difference distance matrix with a weight $\omega_i$. The idea behind the weighting is to place greater importance on the distance matrix that represents interactions that are less frequent within the trajectory. So, the weighting criterion that we choose, which is frequently used in the information retrieval, is an adaptation of the *tf-idf* function.

$$\omega_i = (tf)_i \cdot (idf)_i \tag{2}$$

$\forall i = 1, \ldots, n$ different interaction type. Where:

- $(tf)_i = \frac{n_i}{\sum_{i=1}^{n} n_i}$, this term represents the frequency of the $i$-type interaction within the entire trajectory;
- $(idf)_i = \log\left(\frac{\sum_{i=1}^{n} n_i}{n_i}\right)$, this term represents the importance of the $i$-type interaction among all the others, is easy to see that a larger weight is assigned to an infrequent type of bond and vice versa.

In the last formula $n_i$ represents the total number of $i$-type interaction along the MD trajectory. Once the weights are obtained, we can combine the different distance matrices into a single matrix, on which we will then launch the clustering algorithm.

$$D_M = \omega_1 D_1 + \cdots + \omega_i D_i \cdots + \omega_n D_n \tag{3}$$

Where $D_i$ is the distance matrix of the $i$-type interaction.

### 3.5 Hierarchical clustering

Once the distance matrix $D_M$ is obtained, we can run the agglomerative hierarchical clustering algorithm. The parameter that we must define to launch the clustering algorithm is the link method. The link criterion choosen is the *average criterion*. This criterion define the distance between one cluster to another as equal

to the average distance from any other member of one cluster to any other member of the other cluster. This criterion was chosen as it represents a middle ground between the other criteria available, given the complexity of the problem. The optimal number of clusters is automatically calculated by the algorithm and uses the *average silhouette score*, a measure that calculates the degree of similarity between each individual observation and the cluster to which it has been assigned and averages. Of course, the number of clusters that maximize this score has been chosen.

### 3.6 Output

The algorithm return in output several files, the first is the distance matrix $D_M$, on which clustering is based. The second file returned is the dendrogram, which is the graph that represents the partition hierarchy and highlights the groups formed at each stage of the classification, the different clusters are automatically highlighted using a different color for each different cluster. The third file contains a table where for each single snapshot, the cluster labels, whether or not the snapshot is representative for its cluster (in which case the snapshot is indexed by the value 1, 0 otherwise), and the snapshot distance from the representative one. Those distances were taken from the $D_M$ matrix and were normalized in $[0, 1]$. The representative snapshot was chosen as the one that had the maximum silhouette score within its cluster, which is the one that was best classified or, in other words, that is the most "similar" to the other snapshots of its cluster, on average. Finally, is also provided the list of relevant contacts which were formed or destroyed during the trajectory, in the transition from one cluster to another.

## 4 Results

Now it is possible to analyze the starting dataset and the results returned by the algorithm, also comparing our procedure with the clustering based on RMSD.

### 4.1 Dataset

The MD trajectories used to test our algorithm include proteins from 100 to 500 residues, each trajectory has between 100-200 snasphots. The informations about binding types and their frequency is in the table to see if the algorithm returned better results depending on the composition of the links formed within the trajectory.

| Protein | # of residues | # of snapshots | HBOND | IONIC | PICATION | PIPISTACK | SSBOND | VDW |
|---|---|---|---|---|---|---|---|---|
| 6j6y_1_ms_1k0 | 525 | 100 | 25427 | 1392 | 188 | 1123 | 505 | 45534 |
| cdk6_p16ink4a | 482 | 101 | 34552 | 2258 | 107 | 1062 | 0 | 39350 |
| frataxin | 121 | 196 | 15461 | 725 | 1 | 488 | 0 | 17311 |
| p16 | 156 | 101 | 11326 | 233 | 0 | 93 | 0 | 10929 |
| stim1 | 144 | 101 | 12000 | 642 | 16 | 668 | 0 | 10669 |
| vcb | 531 | 101 | 40888 | 1935 | 72 | 2565 | 0 | 45778 |
| vhl | 150 | 101 | 9366 | 285 | 17 | 968 | 0 | 11404 |

**Table 1.** Descriptive characteristics of the proteins involved in the analysis

### 4.2 Clustering based on RMSD

First of all, in order to have a comparison term for our procedure it is necessary to provide the results of clustering based on RMSD. This index measures the degree of overlap between a group of structurally equivalent atoms and is used to compare protein structures.

$$RMSD = \sqrt{\frac{1}{N}\sum_{i=1}^{n} ||\mathbf{v}_i - \mathbf{w}_i||^2} \tag{4}$$

Where $\mathbf{v}$ and $\mathbf{w}$ are vectors containing the 3D-coordinates of the $i^{th}$ atom of the protein $v$ and $w$, and $N$ is the number of equivalent atoms. The distance matrix associated with this clustering was obtained by calculating all-against-all pairwise RMSD between every snapshots and the linkage criterion choosen for the clustering was the *average criterion*.

| Protein | avg RMSD | Clusters | Repr. snapshots | Repr. snapshots RMSD | avg Score |
|---|---|---|---|---|---|
| 6j6y_1_ms_1k0 | 2.717 | 2 | 8, 39 | 6.989 | .471 |
| cdk6_p16ink4a | 8.546 | 8 | 0, 5, 17, 39, 54, 62, 69, 89 | 10.260 | .273 |
| frataxin | 1.131 | 2 | 1, 102 | 1.792 | .247 |
| p16 | 6.271 | 3 | 2, 11, 82 | 9.456 | .238 |
| stim1 | 3.655 | 3 | 0, 15, 69 | 5.299 | .373 |
| vcb | 4.599 | 2 | 1, 51 | 6.795 | .255 |
| vhl | 3.089 | 2 | 43, 68 | 5.420 | .304 |

**Table 2.** Clustering results based on RMSD

As shown in Table 2, in almost all the trajectories 2-3 representative conformations have been found. The only exception is the cdk6_p16ink4a with 8 conformations, but this strictly depends on the chosen linkage criterion. If we try to use a different criterion, like the complete linkage, we will find a different result. In column "Repr. snapshots RMSD", the RMSD between representative snasphots has been calculated. Due to the fact that the average RMSD calculated along the entire trajectory is significantly lower than the RMSD calculated between representative snapshots we can say that this procedure is able to summarize quite well the trajectories. Lastly, if we look at the "avg Score" in Table 2, we see that the average silhouette score is quite high for a clustering problem, hence the observations are consistent with the cluster to which they belong.

## 4.3 Clustering based on contact maps

| Protein | avg RMSD | Clusters | Repr. snapshots | Repr. snapshots RMSD | avg Score |
|---|---|---|---|---|---|
| 6j6y_1_ms_1k0 | 2.717 | 2 | 32, 87 | 4.121 | .050 |
| cdk6_p16ink4a | 8.546 | 2 | 0, 51 | 12.489 | .122 |
| frataxin | 1.131 | 2 | 5, 81 | 1.295 | .093 |
| p16 | 6.271 | 2 | 0, 33 | 8.673 | .200 |
| stim1 | 3.655 | 2 | 0, 52 | 5.648 | .225 |
| vcb | 4.599 | 6 | 3, 15, 28, 47, 61, 96 | 4.817 | .067 |
| vhl | 3.089 | 2 | 22, 94 | 5.113 | .069 |

**Table 3.** Clustering results based on contact maps

As shown in Table 3, in almost all the trajectories 2 representative conformations were identified like before. This indicates a good consistency with the results obtained in RMSD-based clustering. An exception occurs in vcb trajectory, but as before, this is due to the linkage criterion. Instead, if we compare the "Repr. snapshots RMSD" column, it is evident that a lower RMSD value is obtained in almost all the trajectories, but in any case higher than the average, so we may consider the snapshots identified as representative. Lastly, the average silhouette score in quite lower than the previous procedure and this indicates that the procedure does not optimally classify the observations within the clusters.

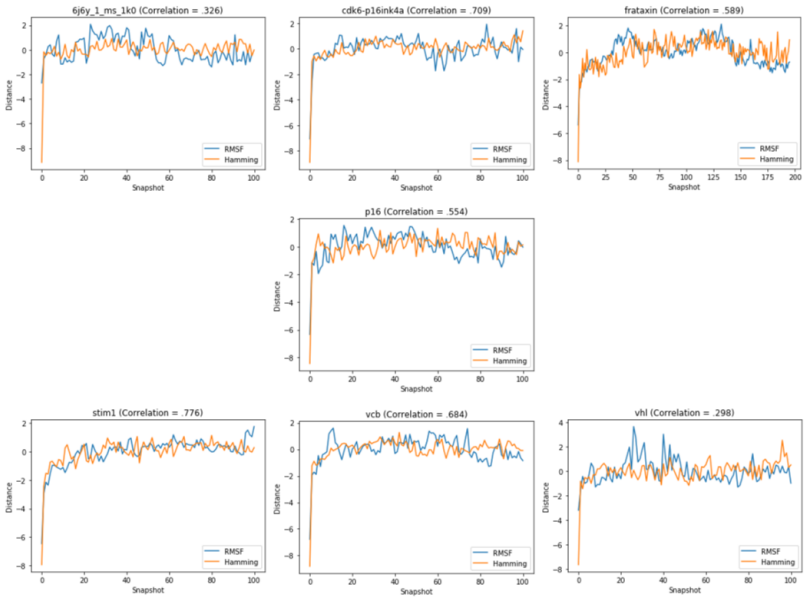


**Figure 1.** Comparison between the RMSF trends and the distance used in our algorithm.

However, it was clear we would have obtained less good results from the RMSD point of view, in a certain sense we have an upper limit due to the fact that we use another distance that cannot give superior results in terms of RMSD compared to a procedure based entirely on RMSD. Now, if we compare the trends present in Figure 1 we see that the correlation between the two procedures is also very significant. One problem with our distance measure is that it often doesn't seem to capture the moments when the protein changes most noticeably, so it gives us a rather flat trend.

# 5 Conclusions

The aim of this project is to synthesize an MD trajectory starting from the contact maps. The trajectory was synthesized by determining some intermediate conformations assumed by the protein representative of the entire dynamics. If we make a comparison between the RMSF trend and the chosen distance measure, as we see in Figure 1, it is evident that some traits of our distance measure between contact maps do not correlate well and therefore are not able to adequately capture the changes occurring within the protein. This may be due to the fact that there are not many new contacts between residuals going from one snapshot to another in these graph strokes. Due to the fact that our algorithm does not use geometric information, the complex of contacts formed in specific regions of the protein can cause a much heavier change in geometric conformation than can be captured by analyzing only the contacts formed and destroyed in the transition between snapshots. One solution could be to assign a different importance to some areas of the contact map where the residues bind to each other and to introduce a method that recognizes bond patterns that increase the distance value. However, given the good consistency of the results returned by the algorithm with the results provided by the clustering based on RMSD, it is possible to state that the procedure is able to satisfactorily capture the representative conformations within the trajectory.

# 6 References


1. Baeza-Yates R., Ribeiro-Neto B. (1999). *Modern information retrieval*, ACM press New York.
2. Deza M.M., Deza E. (2009). *Encyclopedia of distances*, Springer.
3. Ertoz L., Steinbach M., Kumar V. (2002). *A new shared nearest neighbor clustering algorithm and its applications.* In: Workshop on clustering high dimensional data and its applications at 2nd SIAM international conference on data mining (p.105-115).
4. Fraser R., Glasgow J. (2007). *A demonstration of clustering in protein contact maps for alpha helix pairs.* In: International Conference on Adaptive and Natural Computing Algorithms (p.758-766), Springer.
5. Pascarella S., Paiardini A. (2011). *Bioinformatica: dalla sequenza alla struttura delle proteine*, Zanichelli.
6. Piovesan D., Minervini G., Tosatto S. (2016). *The RING 2.0 web server for high quality residue interaction networks.* In: Nucleic acids research (p.W367-W374), Vol.44, No.W1, Oxford University Press.
7. Vassura M., Margara L., Di Lena P., Medri F., Fariselli P., Casadio R. (2008). *Reconstruction of 3D Structures from Protein Contact Maps.* In: IEEE/ACM Transactions on Computational Biology and Bioinformatics (p.357-367), Vol. 5, No.3, IEEE/ACM.