RMS2D heatmap edge detection and k-means clustering analysis to determine stable segments of molecular dynamics trajectories

Brigitte Goeler-Slough[2,3], Carol Dalgarno[1], Kristen Scopino[1], Kelly Thayer[2,3,4], Daniel Krizanc[2,3], Michael Weir[1,3]

[1] *Department of Biology, Wesleyan University*
[2] *Department of Mathematics and Computer Science, Wesleyan University*
[3] *College of Integrative Sciences, Wesleyan University*
[4] *Department of Chemistry, Wesleyan University*

Introduction

- Using molecular dynamics (MD) methods, the CAR interaction surface has been identified as an area of the ribosome that we hypothesize regulates translation
- Consists of three residues: two rRNA residues, nucleotides C1274 and A1427 of the yeast 18S rRNA, with corresponding nucleotides C1054 and A1196 in *E. coli* 16S rRNA, and one amino acid, R146 of ribosomal protein Rps3
- The ribosome CAR surface interacts with +1 codon in mRNA
- We hypothesize that sequence dependent interactions regulate translation

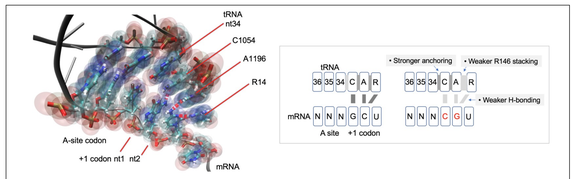

Figure 1. The ribosome CAR surface interacts with the mRNA +1 codon. The CAR surface is positioned close to the mRNA +1 codon allowing for H-bonding between CAR and the codon.

- MD simulations of part of the ribosome can be used to study the behavior of this dynamic system and examine the translation regulation mechanisms
- Determine representative states of this system over time to get an idea how it behaves

Methods

RMS2D Heatmap:

- Pairwise Root Mean Square Deviation (RMSD) calculations are conducted for every possible frame-to-frame comparison of the trajectory
- Forms a matrix of RMSD values which can be plotted to form the heatmap, where the lowest RMSD values are plotted in blue, and the highest RMSD values are plotted in red, with white in between
- The atoms involved in the RMSD calculations were the backbone atoms of the 12 key residues (in the A-site mRNA, +1 codon mRNA, tRNA anticodon, and CAR residues) for a +1 codon of GCU, with 30 experiments of varying initial velocities
- Distinguish the blue squares of the heatmap, as they represent the segments of the trajectory where the RMSD is smallest between frames, meaning the frames are the most similar, or the most stable

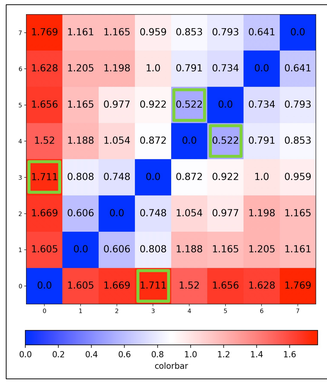

Figure 2. Annotated heatmap with RMSD values. Notice that all the values along the diagonal are 0, as they are the RMSD comparing the frame to itself, so thus the difference between them is 0. The frames highlighted with an RMSD value of 1.711 are comparing frame 0 and frame 3, which are structurally quite different frames. The frames that are highlighted with an RMSD of 0.522 are comparing frames 4 and 5, which are structurally more similar frames.

Methods (continued)

Edge Detection Algorithm

- The edge detection algorithm is a window walking algorithm, with the algorithm starting at the bottom left corner and walking up along the diagonal until the final frame, where the algorithm iterates horizontally to the right to determine where the edge is
- To be considered part of the stable segment (not yet at an edge), 95% of the frames must be below a given RMSD value
- Begins by just checking the first 10 frames from the diagonal, and if the first 10 frames do not pass the test, then the algorithm does not continue for that line. If the initial window of 10 frames passes, then the algorithm iterates one frame at a time to the right, checking after each frame if the threshold of 95% of frames are below the cutoff
- Once the test fails and under 95% of the frames fall below the RMSD cutoff, the algorithm detects an edge and marks that as a black square
- Next step is a sharpening process, where from the right end of the segment, each frame is checked and removed from the stable segment if it is above the RMSD threshold until a value below the RMSD threshold is reached

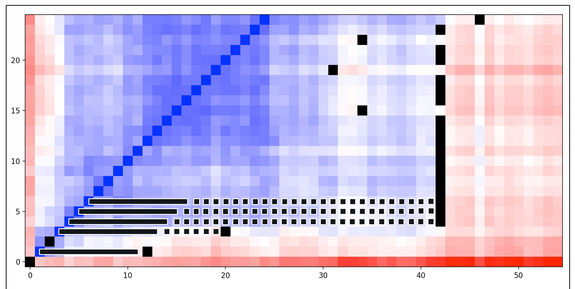

Figure 3. Visual depiction of the window walking edge detection algorithm. The first window of 10 frames does not pass in the first and third line, but passes in every other frame. In the second line, just the initial 10 frames pass, and in the rest more individual frames also continue to fall below the RMSD cutoff.

Centroid Calculation:

- For each stable segment that the edge detection algorithm detects, iterate through each frame in that given segment
- Calculate the average RMSD between the current frame and every other frame in the stable segment by summing the RMSD of that frame with every other possible frame in the segment and then dividing by the total number of frames in the segment
- The smallest overall average RMSD is the representative centroid for that stable segment
- The centroid gives us at which frame we find the representative structure of the system for a stable segment of the trajectory

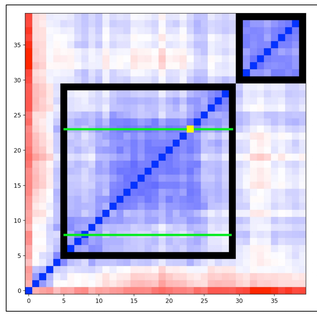

Figure 4. Visual depiction of the centroid calculation. Two frames are highlighted: frame 8 and frame 23. In frame 8, the average RMSD with every other frame is 0.8210, which is the highest/most different, and in frame 23, the average RMSD is 0.5632, which is the lowest/most similar, and therefore frame 23 is the representative centroid for this stable segment.

Methods (continued)

K-Means Clustering:

- K-means clustering splits up frames of a trajectory into k different clusters where frames in the same cluster are the most similar in structure
- The algorithm begins by randomly choosing k random frames to represent the centroid of each cluster
- Then it sorts each other frame of the trajectory into one of the clusters by calculating the RMSD between that frame and each centroid of the cluster and adding that frame to the cluster with the lowest RMSD between the frames
- Centroids are recalculated for each cluster and frames are reassigned to alternative clusters if they are closer to a centroid in another cluster

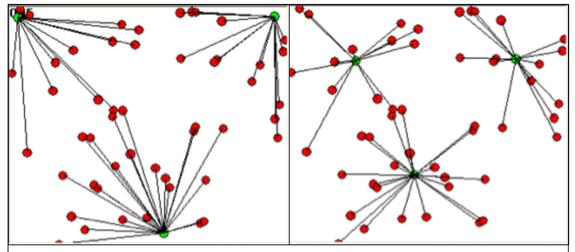

Figure 5. A demonstration of two different centroids for two dimensional k-means. On the left there are poorly chosen centroids where the distances between the points and the centroids are large, and on the right a better choice of centroids where the overall distances between the random points and the centroids tend to be smaller. In our situation we are examining a three-dimensional system of the ribosome, but similar concepts apply with RMSD values instead of simply distance.

Source: Michael Rice

Results

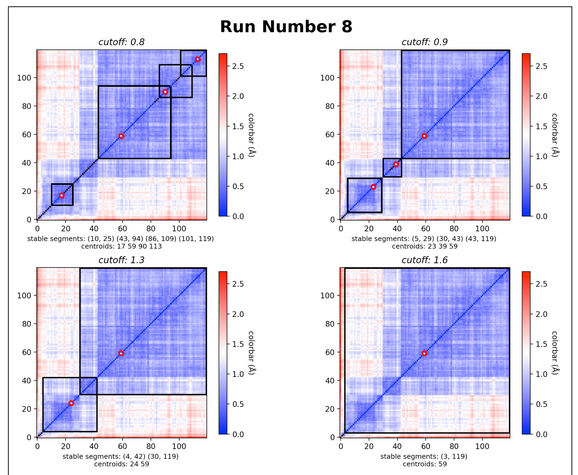

Figure 6. Edge detection algorithm results for experiment 8. Figure 6 shows the algorithm results depicted in black boxes and centroids as white squares (circled in red) along the diagonal. It is demonstrated with 4 different RMSD cutoffs (0.8Å, 0.9Å, 1.3Å, and 1.6Å) with the frames of the stable segments and centroids of each segment listed below.

Results (continued)

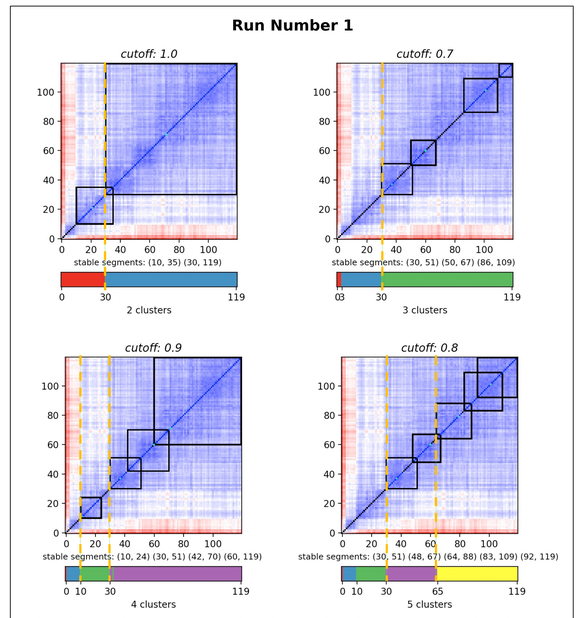

Figure 7. Edge detection algorithm results for experiment 1 compared with k-means clustering results. Results of the edge detection algorithm are shown above with different RMSD cutoffs with the k-means clustering results shown below for results of 2, 3, 4, and 5 clusters/segments. Orange vertical lines demonstrate places where similar boundaries are detected.

Conclusion

- The algorithm is able to detect different stable segments depending on the RMSD cutoff given as input
- The RMS2D heatmap stable segments have some similarities to the stable segments determined by the k-means clustering analysis

Future Directions

- Examine the heterogeneity of different trajectories/experiments starting with different initial velocities of molecules
- Explore the transitions between the five stages of translation translocation using replica exchange
- Increase the running temperature of the MD simulation from 300K to 325K for replica exchange

Acknowledgements

Thank you to Professors Weir, Thayer, and Krizanc for their advice and support in this research. An additional thanks to all the lab members of both the Weir lab and the Thayer lab for their continued help and discussion of my work. Finally, I'd like to thank Henk Meij for maintaining Wesleyan's high-performance computing facility to allow us to use it.