A Self Supervised Solution for Analysis of Molecular Dynamics Simulations In-Situ

Harshita Sahni, Trilce Estrada (University of New Mexico) Hector Carrillo-Cabada (Intuit)
Ekaterina Kots, Michel Cuendet, Harel Weinstein (Weill Cornell Medical Center)
Ewa Deelman (University of Southern California) Silvina Caino-Lores, Michela Taufer (University of Tennessee)

I. Introduction

A Protein is a long chain of amino acid residues [1]. This chain decides the overall structure of the protein. However, this structure is not static; it folds and unfolds to perform the protein's functions. Changes in factors like ph, temperature and composition of a solution can make a protein go through a number of molecular events which induce conformational changes in the protein trajectory. Structural rearrangements, binding events and protein associations are some examples of such changes [2]. Conformation change detection is crucial because it provides information about the biological, self-assembly functions, and molecular mechanisms of the protein [3].

Molecular Dynamics (MD) is used to simulate the folding process of proteins over a period of time [4]. Each time step, which is termed as a Frame, defines the state of the protein. The chain or collection of frames form a trajectory. With modern technology and High Performance Computing (HPC), MD simulations can be task and data parallel. That means, they can be decomposed into multiple independent tasks (i.e., trajectories) with their own data, which can be processed in parallel. Depending on its sampling rate and length, each simulation can produce gigabytes of data [5]. Analysis of MD simulations includes finding specific molecular events and the conformation changes that a protein undergoes. However, traditional analyses rely on a global decomposition of all the trajectories for a specific molecular system, which can be performed only in a centralized way. An example of this pipeline is Time Lagged Component Analysis (TICA) [6] that captures the slowest moving components describing the different molecular events. Contrary to traditional MD data analytics that first generates and saves all the trajectory data to storage and then relies on postsimulation analysis, we analyze data as they are generated, and save to disk only what is really needed for future analysis.

To this end, we propose a lightweight self-supervised machine learning technique to analyse MD simulations In-Situ. That is, we aim to speedup the process of finding molecular events in the protein trajectory at run-time, without having to wait for the entire simulation to finish. This not only allows us to scale the analysis with the simulation, but the findings can be used to dynamically steering the simulation.

II. Method

For a particular protein we use domain knowledge to determine the residues of interest (r). That is, residues that are involved on specific protein functions. Then, for each frame we calculate the Euclidean distance between each pair of residues of interest and build a frame vector x of size $|r|^2$. To build our first model (M_0), we store a fixed window of frames (say w frames) from the running simulation into a matrix X of size $|r|^2 \times w$. We use this collection of frames to perform a Non-Negative Matrix Factorization (NMF) [7]. Given X as the input, NMF finds two other matrices W and H such that $X \approx W \cdot H$, and we calculate the initial set of reconstruction errors $\epsilon_{0,i}$ as MSE($X, W \cdot H$). This reconstruction error serves two purposes: 1) to determine whether the model (e.g., M_0) is able to explain a frame or window of frames, and 2) to identify the specific residues that are involved in a molecular event (see Fig. 1b).

As we are interested in determining the different states of the protein over time, we build a state for every new molecular event that is detected in the trajectory. Where every state S_k comprises of $k+1$ associations with previously built NMF models (M_j for $j = 0$ to k) and a corresponding set of parameters $P(S_k, M_j)$ associated with the "normal" behavior of the protein in this state. The parameters for each model j in a particular state k include the set comprising of running mean and running standard deviation, as well as the critical values for a 80% level of confidence ($p < 0.2$) for a two tailed z-test. To test consecutive windows of frames we evaluate their reconstruction errors with respect to all existing models. For the last state built, we determine if the mean square error (MSE) for each model M_j falls within the critical values in $P(S_k, M_j)$ for all j's. If this is the case, we update the parameters in $P(S_k, M_*)$ by adding the reconstruction errors of the current window to the different distributions. We also traverse the other existing states to decide if the current window can be explained by any of them, but we no longer update their parameters. If none of the existing states can explain the current window, a new state is created.

III. Evaluation

We tested our method on trajectories from five different proteins or protein mutants. [8] They ranged from 162 to 326 number of residues, sampling rates between 100 picoseconds

to 1 microsecond, and lengths varying from 200 nanoseconds to 1 microsecond. We validated our results with respect to TICA and were provided with expected conformation changes determined by domain scientists. From this information, we computed the number of true positives (TP): a change was detected within the expected window, false positives (FP): change was detected when there was no expected change, and true negative (TN): no change was detected in the expected window. Through all the experiments, we were able to correctly identify 42 out of 45 molecular events. We are currently investigating the properties of the three events that we missed. Moreover, we only produced 15 false detections in over 283930 frames. However, most of these false positives are inconclusive, as they align perfectly with TICA predictions.

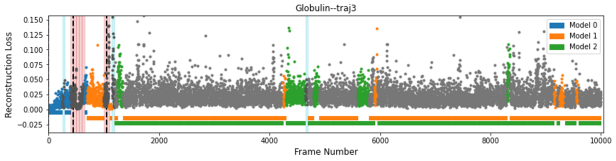
Protein	NT	TR	RI	Frames	TP	FP	TN
Bovine betalactoglobulin	6	162	12	2000-10000	6	2	0
Wild-Type(B cell translocation gene (BTG1) Mutant)	23	129	8	1200-10000	20	5	2
E50K(BTG1 mutant)	18	129	8	1000-14000	11	3	1
R68L(BTG1 mutant)	6	129	8	14000	4	2	0
Opsin	1	326	21	2000	1	3	0

TABLE I: Dataset used for experiments. NT: number of trajectories, TR: total number of residues, RI: residues of interests, Frames: range of frames per trajectory, TP: true positives, FP: false positives, TN: true negatives

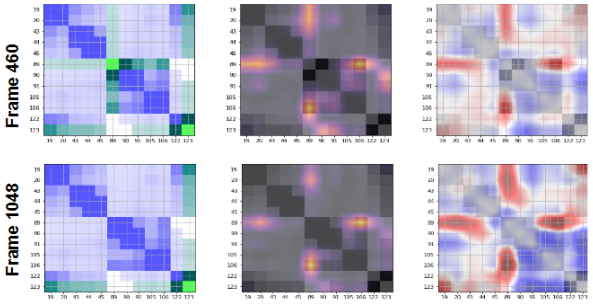
Figure 1 shows evaluation results for Trajectory 3 of the Globulin protein. We show the different states found by our method (a), the sets of residues that contributed more to the molecular event (b), a comparison with two TICA components, and c) performance measurements of our method.

IV. Conclusion

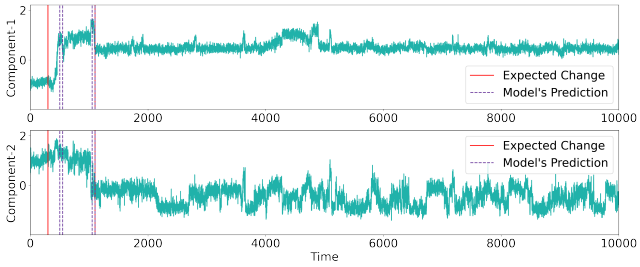
Our method shows a detection behavior that is consistent with TICA. But unlike TICA, the analysis can be performed in the same node as the simulations or run concurrently on a different node; saving time and computational resources. We train an ensemble of light-weight ML models (i.e., in the order of a few kilobytes) that do not require the entire view of the protein to determine the relevant changes. Moreover, our method accurately determines when a protein is not mobile, which saves the scientists a considerable amount of time and effort since a good amount of MD simulations rarely produce new molecular events [10]. In summary: Our algorithm offers a way to monitor molecular events in a protein trajectory In-Situ. With this knowledge, we identify different states of the protein through its trajectory. We are able to explain which residue-pairs contributed the most for a detected conformation change in the protein trajectory. Our method is robust regardless of the type of protein, sampling rate, number of residues, as well as other simulation properties. Our method is efficient (i.e., execution per window takes a few milliseconds) and requires only constant memory (e.g., 50 frames at a time)



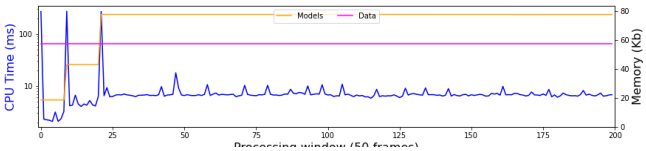
(a) Our method: colored horizontal bars represent states that are able to explain the data (e.g., blue is Model 0 explains the data, orange is Model 1, and green is Model 2). Red vertical bars indicate when a new state was created. Cyan vertical bars are regions of uncertainty that did not create a new state.



(b) Residue interactions that explain the detected molecular event. Top is frame 460 and bottom is frame 1048. Every matrix represents the interaction between each pair of residues. On the left, the GEM encoding. [9] depicts changes in secondary structure. The middle figure highlights the pairs of residue interaction that contributed the most to the detection of the event. Finally, the right matrix shows whether residue pairs moved closer (red) or farther away (blue) than expected.



(c) Plot of two TICA components comparing the expected changes with the changes detected by our method



(d) Memory and CPU utilization of our method

Fig. 1: Evaluation example for Globulin Trajectory 3

References

[1] Alberts B, Johnson A, Lewis J, et al. Molecular Biology of the Cell. 4th edition. New York: Garland Science; 2002. The Shape and Structure of Proteins. Available from: https://www.ncbi.nlm.nih.gov/books/NBK26830/

[2] Tobi, Dror, and Ivet Bahar. "Structural changes involved in protein binding correlate with intrinsic motions of proteins in the unbound state." Proceedings of the National Academy of Sciences 102.52 (2005): 18908-18913.

[3] Koide, Akiko, et al. "Probing protein conformational changes in living

cells by using designer binding proteins: application to the estrogen receptor." Proceedings of the National Academy of Sciences 99.3 (2002): 1253-1258.

[4] Karplus, Martin, and J. Andrew McCammon. "Molecular dynamics simulations of biomolecules." Nature structural biology 9.9 (2002): 646-652.

[5] Wang, Yihang, Joao Marcelo Lamim Ribeiro, and Pratyush Tiwary. "Machine learning approaches for analyzing and enhancing molecular dynamics simulations." Current opinion in structural biology 61 (2020): 139-145.

[6] Pérez-Hernández, Guillermo, et al. "Identification of slow molecular order parameters for Markov model construction." The Journal of chemical physics 139.1 (2013): 07B604_1..

[7] Eggert, Julian, and Edgar Korner. "Sparse coding and NMF." 2004 IEEE International Joint Conference on Neural Networks (IEEE Cat. No. 04CH37541). Vol. 4. IEEE, 2004.

[8] Piana, Stefano, Kresten Lindorff-Larsen, and David E. Shaw. "Protein folding kinetics and thermodynamics from atomistic simulation." Proceedings of the National Academy of Sciences 109.44 (2012): 17845-17850.

[9] Carrillo-Cabada, Hector, et al. "A graphic encoding method for quantitative classification of protein structure and representation of conformational changes." IEEE/ACM transactions on computational biology and bioinformatics 18.4 (2019): 1336-1349.

[10] Botu, Venkatesh, and Rampi Ramprasad. "Adaptive machine learning framework to accelerate ab initio molecular dynamics." International Journal of Quantum Chemistry 115.16 (2015): 1074-1083.