Using correlated motions to determine sufficient sampling times for molecular dynamics

Ryan L. Melvin, Jiajie Xiao, Kenneth S. Berenhaut, Ryan C. Godwin, and Freddie R. Salsbury, Jr.

Phys. Rev. E **98**, 023307 — Published 24 August 2018

DOI: 10.1103/PhysRevE.98.023307

Using correlated motions to determine sufficient sampling times for molecular dynamics

Ryan L. Melvin

Wake Forest University Department of Physics and

Wake Forest University Department of Mathematics and Statistics

Jiajie Xiao

Wake Forest University Department of Physics and

Wake Forest University Department of Computer Science

Kenneth S. Berenhaut

Wake Forest University Department of Mathematics and Statistics

Ryan C. Godwin and Freddie R. Salsbury Jr.*

Wake Forest University Department of Physics

(Dated: August 6, 2018)

Abstract

Here we present a novel time-dependent correlation method that provides insight into how long a system takes to grow into its equal-time (Pearson) correlation. We also show a novel usage of an extant time-lagged correlation method that indicates the time for parts of a system to become decorrelated, relative to equal-time correlation. Given a completed simulation (or set of simulations), these tools estimate (1) how long of a simulation of the same system would be sufficient to observe the same correlated motions, (2) if patterns of observed correlated motions indicate events beyond the timescale of the simulation, and (3) how long of a simulation is needed to observe these longer timescale events. We view this method as a decision-support tool that will aid researchers in determining necessary sampling times. In principle, this tool is extendable to any multi-dimensional time series data with a notion of correlated fluctuations; however, here we limit our discussion to data from Molecular Dynamics simulations.

* salsbufr@wfu.edu

I. INTRODUCTION

What is an appropriate amount of sampling in molecular dynamics (MD) simulations is a recurring, open question (see for example [1–4]). For example in amyloid studies of fibril and oligomer stability, this problem is particularly severe (see for instance [5–9]). Especially when limited by computational resources, a choice often must be made between running few long-timescale simulations to capture rare conformations and slow events or running many short simulations for stronger statistics. Here we present two tools that will aid researchers in this decision.

Extant methods for addressing issues of sufficient sampling in MD tend to modify the method of sampling itself. For example, umbrella sampling techniques modify a system's potential in order to access regions of conformation space that traditional MD might not reach under limited sampling. For a thorough discussion of such techniques, see [10, 11] and the citations therein. Another popular technique – replica exchange Markov chain Monte Carlo sampling (also called parallel tempering) – runs multiple simulations in parallel at various temperatures and exchanges conformations among them [12, 13].

Rather than adjusting the sampling method, we take on the issue of determining how much sampling is sufficient for a given system in the absence of experimental guidance. If experimental data is available and indicates the physiological time for some process, that information provides the best possible guidance for how much simulated time is needed; see for example [3, 14–25]. However, for systems or processes without clear experimental data, researchers are left to make educated guesses at sufficient timescales for their simulations. The two techniques we present here refine such initial conjectures. Given a completed pilot simulation (or set of simulations), the methods presented here determine (1) how short of a simulation of the same system would be sufficient to observe the same correlated motions as seen in the pilot simulation(s), (2) if patterns of correlated motions observed in the pilot simulation(s) indicate events beyond the timescale of the simulation(s), and (3) how long of a simulation is needed to observe these longer timescale events. Also, it is worth mentioning that this method could be implemented to run periodically during a simulation to inform stopping conditions.

II. METHODOLOGY

We first perform this analysis using a well-established time-lagged correlation technique (see for example [26]), which modifies Pearson correlation to ask how is atom j at time α correlated to atom k at time $\alpha + \tau$. By calculating the correlation coefficient of the time-lagged correlation matrix with the equal time (also called Pearson [27, 28]) correlation matrix for various values of τ and various systems (Figure 1), we see roughly exponential decay of correlated motions from the equal time correlations. Mathematically, such a decay pattern is expected as the correlation matrices are exactly the same at $\tau = 0$. We can then fit exponential curves (supplemental Figures 1-3) to these data sets to estimate the number of events (number of exponential terms needed) and the timescale of these events (decay constant for a given exponential term).

This section proceeds with a discussion of time-lagged correlation, exponential fitting and our novel correlation propagator. For those wishing to reproduce our analysis and/ or apply it to their own work, we have made our python code and the time series data for the examples presented in this work available at `https://doi.org/10.6084/m9.figshare.6613481`.

A. Time-lagged correlation

If r_i and r_j are position vectors of two atoms in the sample, then the typical covariance of their motion is calculated using

$$\tilde{C}_{ij} \equiv \sum_{\alpha=1}^{N} \frac{\left(r_i^\alpha - \langle r_i^\alpha \rangle\right) \cdot \left(r_j^\alpha - \langle r_j^\alpha \rangle\right)}{N} \tag{1}$$

Here N indicates the total number of frames and alpha is the index over each frame of the trajectory. We then calculate the correlation by normalizing the covariance. To do so, we divide by the square root of the product of $\tilde{C}_{ii}$ and $\tilde{C}_{jj}$, making the diagonal elements all 1 – that is, an atom always fluctuates exactly with itself. That is, the correlation matrix is calculated by

$$C_{ij} = \frac{\tilde{C}_{ij}}{\sqrt{\tilde{C}_{ii}\tilde{C}_{jj}}} \tag{2}$$

For time-lagged correlation with a lag time of τ, we modify $\tilde{C}_{ij}$ to

$$\tilde{C}_{ij}^{(\tau)} \equiv \sum_{\alpha=1}^{N-\tau} \frac{(r_i^\alpha - \langle r_i^\alpha \rangle) \cdot \left(r_j^\alpha - \langle r_j^{\alpha+\tau} \rangle\right)}{N-\tau} \tag{3}$$

and similarly normalize to make the diagonals 1, as in Equation (2), to arrive at the the time-lagged correlation matrix with a lag time of τ. Intuitively, consider at time α, the movement of x provides information about the future movement of y at time $\alpha + \tau$.

B. Exponential fitting

Having calculated both the typical correlation matrix and time-lagged correlation matrices for a series of values of τ, we calculate the correlation coefficient of the typical correlation matrix with each time-lagged matrix, resulting in a time series of correlation coefficients. We perform exponential fitting on this time series to estimate the number of events and the respective timescales thereof. The intuition here is that the number of overlapping exponential curves is the number of "events." That is, each correlated motion "event" has its own decay constant.

The results of exponential fitting are sensitive to the algorithm and parameters (e.g. initial guess) used. For extensive discussions, see for instance [29, 30] and the citations therein. We describe here the algorithms used for the fits displayed in supplemental Figures 1-6.

For single exponential fits, with $\mathbf{y}$ as the observed time series of correlation coefficients and $\boldsymbol{\tau}$ as the corresponding lag times, we want to fit a model of the form

$$y = A^* e^{B^* \tau}. \tag{4}$$

For this case, we make a transformation in τ and define a new response variable Y, that is linear with respect to $\ln(\tau)$. This transformation allows for fitting with typical least squares regression.

$$Y = A + B \ln(\tau). \tag{5}$$

To perform a least squares linear regression of the form $Y \sim \ln(\tau)$ using the `polyfit` method in the Python NumPy package [31] to obtain estimates for A and B, $\hat{A}$ and $\hat{B}$, and calculate the corresponding fitted values

$$\hat{Y} = \hat{A} + \hat{B} \ln(\tau). \tag{6}$$

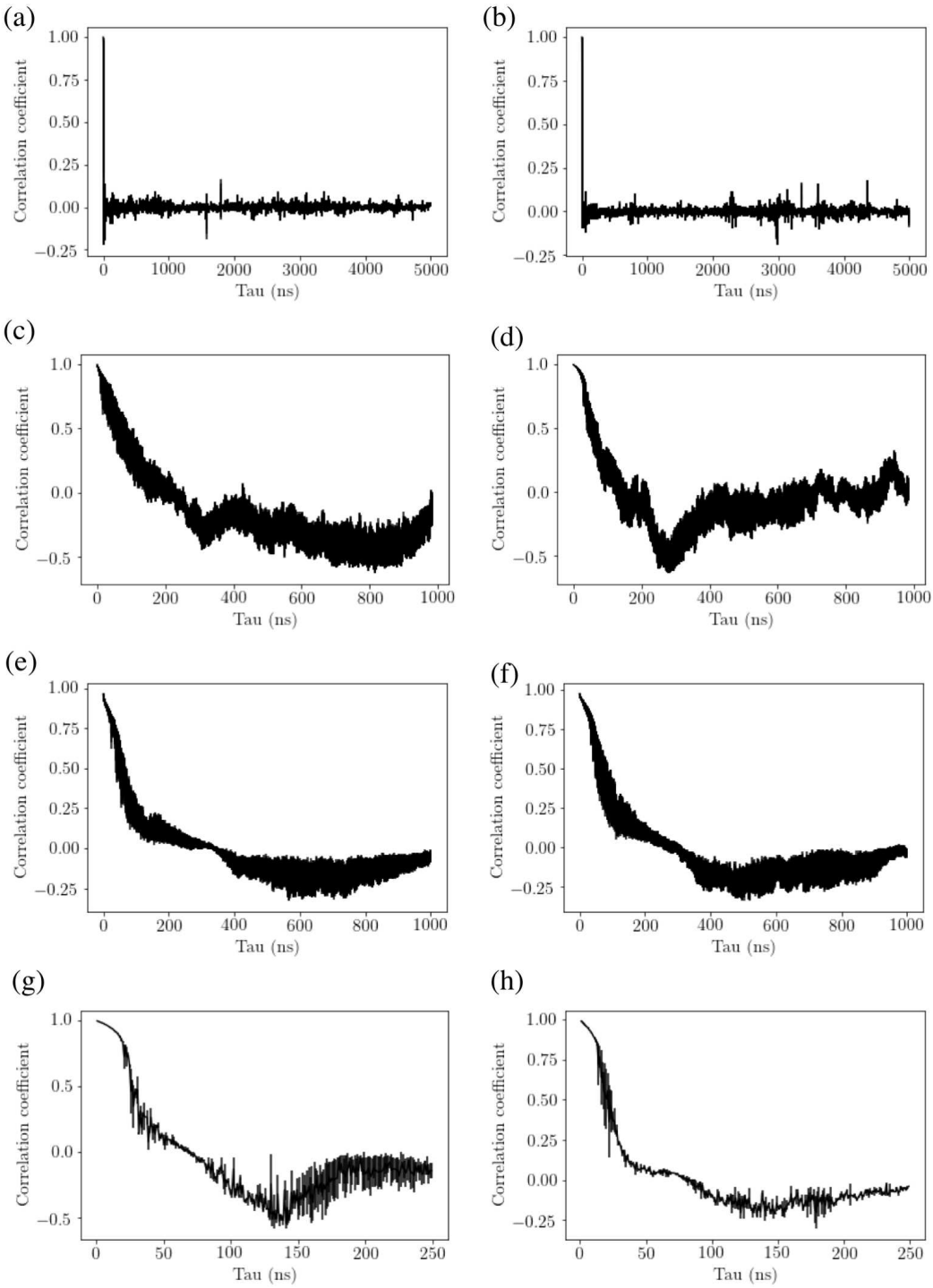

FIG. 1. Correlation coefficients of time-lagged correlation matrix at a series of τ values to the typical equal time correlation matrix are presented for (a-b) therapeutic DNA strand F10 in physiological solvents – (a) potassium-heavy and (b) sodium-heavy. NEMO zinc finger with (c) and without (d) a bound zinc ion, Thrombin in a (e) stabilizing and (f) destabilizing solvent, and MutSα with (g) cisplatinated DNA and (h) carboplatinated DNA. Bars indicate one standard error.

In the case of single exponential fitting, we preferred this transformation-then-fit due to the intuitive, ubiquitous nature of least squares fitting. However, one could certainly use the Levenberg-Marquardt method discussed below for all cases.

For double (or triple) exponential fitting, we want to fit a model of the form

$$y = Ae^{B\tau} + Ce^{D\tau} + \left(Ee^{F\tau}\right). \tag{7}$$

In, this case, we use the `optimize.curve_fit` method in the SciPy Python package [32, 33]. Here we used all default options for this method. The default non-linear least squares regression method is the Levenberg-Marquardt [34, 35] (or damped least squares) algorithm. For a critical discussion of this method and available alternative see [36]. One issue of particular note is that Levenberg-Marquardt, as with many such methods, may report a solution that lies in a local (rather than global) minimum. To partially address this issue, for each such fit, we plot the observed and fitted values to judge the reasonableness of the solution (supplemental Figures 2-3 and 5-6), iteratively adjusting the initial values. We recognize such analysis cannot be performed without bias, and we refer readers to [36] for alternatives. The algorithm we have chosen here is relatively fast for our particular use case [36].

It is worth noting at this point that more than three exponential terms may be appropriate for a given scenario. Furthermore, there is a level of iterative validation involved in this process. Therefore, we view this method as a type of decision-support tool, not a definitive, faultless procedure for determining necessary sampling with absolute certainty.

C. Correlation propagator

We also introduce a new time-lagged correlation measure that we have termed *Correlation Propagator*. Let $\mathbf{r}^\alpha$ be the $M \times 3$ matrix of atom positions (e.g., Cartesian coordinates) at time α for M atoms in a trajectory of N steps. Let τ be a user-specified lag time of interest. Then, define the matrix

$$\mathbf{P}^{\alpha,\tau} \equiv \mathbf{r}^\alpha - \mathbf{r}^{\alpha+\tau}, \tag{8}$$

which is an $M \times 3$ matrix. We then define the correlation propagator to be the time-averaged dot product of this matrix with itself. That is, let i index dimensions (e.g., 1 to 3 for Cartesian coordinates x, y and z) and j and k index atoms in the trajectory (i.e., the

rows of $\mathbf{r}^{\alpha}$) then the correlation propagator is

$$\mathbf{C}_{j,k} \equiv \frac{\sum_{\alpha=1}^{N-\tau} \sum_{i=1}^{3} \mathbf{P}_{j,i}^{\alpha,\tau} \mathbf{P}_{k,i}^{\alpha,\tau}}{N-\tau}.$$

By calculating the correlation coefficient of the correlation propagator matrix with the equal time correlation matrix for various values of τ and various systems (Figure 2), we see roughly exponential convergence to the typical equal time correlation matrix. Intuitively, this convergence makes sense, as we would expect $\mathbf{r}^{\alpha+\tau}$ to go to the average position vector for large values of τ. The development of the correlation propagator is phenomenological, and we hope that it adds to the ongoing discussion of correlated motions and allostery in the field of biophysics [37] and the use of autocorrelation for general time series analysis in other fields.

Using the correlation coefficient between the correlation propagator matrix with the equal time correlation matrix, we can then fit exponential curves – as described above – to these data sets to estimate the number of events (number of exponential terms needed) and the timescale of these events (decay constant for a given exponential term). See supplemental Figures 4-6 and the Python notebooks we have made available at `https://doi.org/10.6084/m9.figshare.6613481` for examples.

III. EXAMPLES

In the case of a relatively unstructured [38] nucleic acid strand in two physiological solvation conditions, we see growth into the equal time correlation on a timescale of nanoseconds (Figure 2a-b and supplemental Figures 4a-b, 5a-b and 6-ab), indicating this would be a system where many short simulations would be appropriate. In the case of the small, structured NEMO zinc finger with a bound zinc ion, we predict that 500 ns simulations are needed, agreeing with previous results for this system [39]. In the case of the zinc-unbound case, which is known to be less stable [40, 41], we see indications of a nanosecond-scale process observable with relatively short simulations (Figure 2c-d and supplemental Figures 4c-d, 5c-d and 6c-d).

A similar story emerges in the case of Thrombin, where the presence of the known stabilizer sodium produces longer timescale process than destabilizing [42] potassium (Figure 2e-f and supplemental Figures 4e-f, 5e-f and 6e-f). In the case of a large protein complex, MutSα, exponential fitting (supplemental Figures 4g, 5g and 6g) predicts one roughly 50 ns

process and one microsecond timescale process for both types of DNA damage investigated – cisptlatinated (Figure 2g) and carboplatinated (Figure 2h and supplemental Figures 4h, 5h and 6h) DNA. The second process is beyond the time simulated in the pilot simulations, indicating much longer simulations are needed for this system. In the eight brief case studies presented here, we see the utility of this technique in predicting how long a simulation is needed to observe the correlated motions observed in the pilot simulations and the timescales of events indicated by exponential fitting.

IV. SIMULATED DATA METHODS

A. General computational methods

All protein simulations were run under the isothermal-isobaric ensemble (NPT) in ACEMD [43]; F10 simulations were run under the canonical ensemble (NVT), the recommended ensemble for ACEMD[43]. Damping via a Langevin thermostat [44] maintained target temperature of 300 K via a damping coefficient of 0.1. A Berendsen pressure piston [45] held all systems at roughly 1.01325 Bar using a relaxation time of 400fs. Hydrogen mass repartitioning allowed for 4fs time steps in our production runs. We applied a 9Å cutoff and 7.5Å switching distance for VdW and electrostatic forces, calculating long-range electrostatics with a smooth particle mesh Ewald (SPME) summation method [46, 47].

These simulations were run on Titan GPUs in Metrocubo workstations produced by Acellera. All systems were solvated in explicit TIP3P water [48], using VMD's [49] "Add Solvation Box" feature. The CHARMM27 forcefield parameters used here are based on interaction energies of small model systems calculated by quantum mechanics computations and direct experiment [50–52].

B. F10

For F10, the CHARMM27 forcefield was supplemented with FdU-specific MD parameters [53, 54], which have been validated in previous studies [54, 55]. F10 was solvated in a cubic water box – 50 angstroms on a side – with the concentration of counter ion (in the form of salt, i.e., NaCl or KCl) set to 150mM using VMD's [49] "Add Solvation Box" and "Add

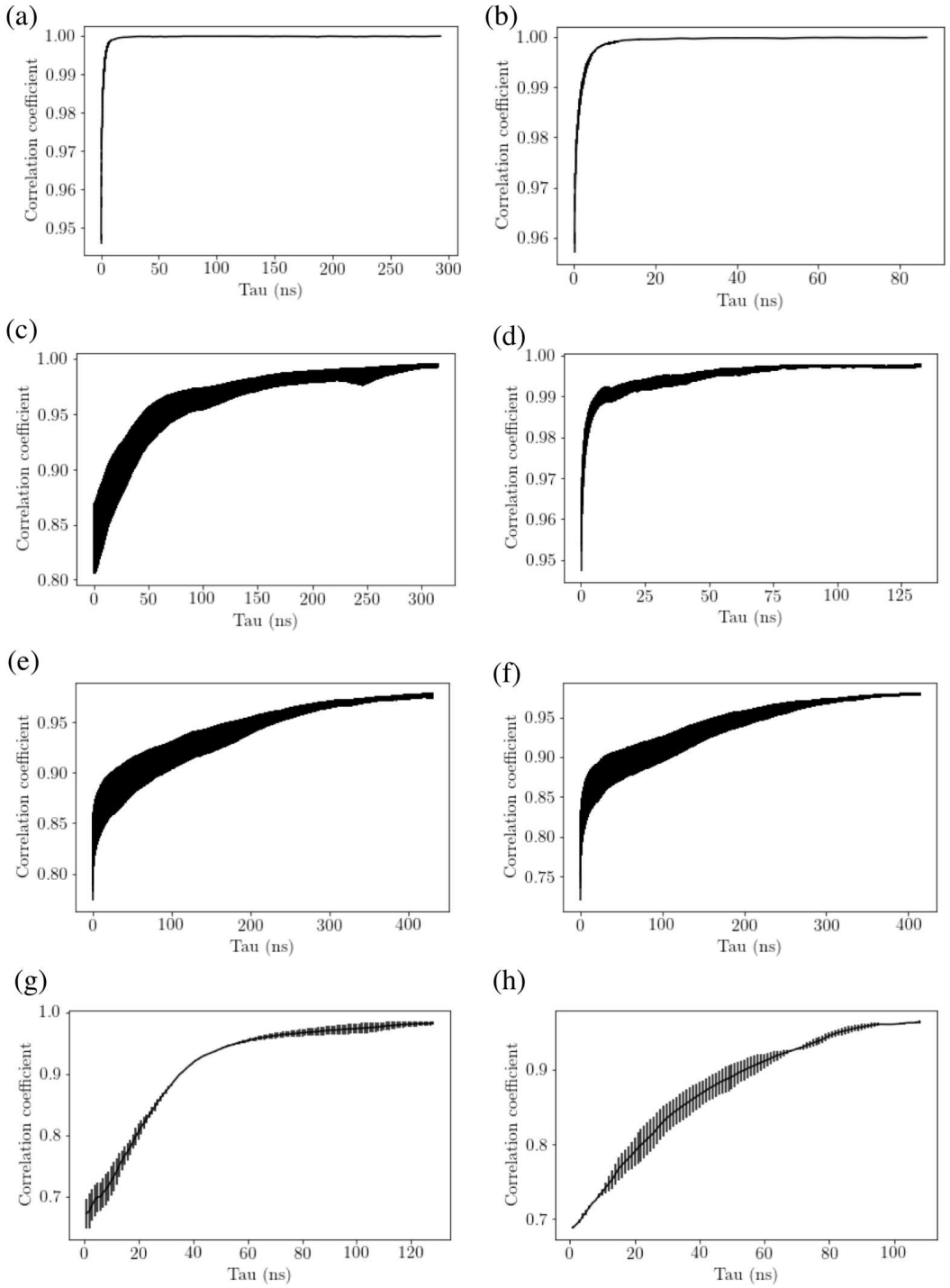

FIG. 2. Correlation coefficients of the correlation propagator at a series of τ values to the typical equal time correlation matrix are presented for (a-b) therapeutic DNA strand F10 in physiological solvents – (a) potassium-heavy and (b) sodium-heavy, NEMO zinc finger with (c) and without (d) a bound zinc ion, Thrombin in a (e) stabilizing and (f) destabilizing solvent, and MutSα with (g) cisplatinated DNA and (h) carboplatinated DNA. Bars are one standard error.

Ions" plugins with salt concentrations set using the "Neutralize and set" option and all other parameters left as default.

C. NEMO zinc finger

Initial coordinates are based on an experimental crystal structure, accessed from the RCSB data bank – PDB ID 2JVX [56]. Before production runs, each fully solvated system underwent conjugate gradient minimization for 5000 time steps. Subsequent equilibration lasted approximately 18ns, as measured by the RMSD. Prior to analysis, this equilibration period was removed from each of each of eight 1μs trajectories used here. All NEMO systems were solvated in 150mM NaCl using the "Add Ions" extension in VMD.

D. Thrombin

Initial coordinates are based on an experimental crystal structure from the RCSB data bank – PBD ID 4DII and 4DIH[57]. Missing residues (18 and 19 out of 295 residues respectively for PDB 4DIH and 4DII) in these PDBs were added via Modeller [58] – a structural template-based atom fill-in tool. Hydrogen atoms were added by VMD's "psfgen" tool using the topology file in CHARMM 27 force field. The system was solvated in the buffer with 125mM KCl and NaCl respectively. Systems underwent 1000 steps of conjugate gradient minimization. The final trajectory analyzed here is a concatenation of 5 simulations, each of 1μs.

E. MutSα

Initial coordinates are based on an experimental crystal structure from the RCSB data bank – PDB ID 208E [59]. For cisplatinated and carboplatinated DNA, we used additional, pre-existing, cisplatin and carboplatin parameters [53, 54, 60, 61]. We fitted cross-linked structures of modified DNA strands into the mismatched binding pocket – as seen in RCSB PDB ID 208E [59]. All systems were solvated in 150mM NaCl using the "Add Ions" plugin in VMD. Each trajectory analyzed here was concatenated from two 250ns all-atom MD production runs. Before production runs, each system underwent 1000 steps of conjugate

gradient minimization and 250ps of thermal equilibration.

V. CONCLUSIONS AND FUTURE WORK

Here we have presented a novel time-dependent correlation method that provides insight into how long a system takes to grow into its equal-time (Pearson) correlation. We have shown a novel usage of an extant time-lagged correlation method that indicates the time for parts of a system to become decorrelated, relative to equal-time correlation. The tools presented here estimate (1) how long of a simulation of the same system would be sufficient to observe the same correlated motions, (2) if patterns of observed correlated motions indicate events beyond the timescale of the simulation, and (3) how long of a simulation is needed to observe these longer timescale events. We view this method as a decision-support tool that will aid researchers in determining necessary sampling times.

A logical extension of this work may be to use the information from this decision support tool to define stopping conditions for a "smart" simulation scheme. That is, one might iteratively check and restart a pilot simulation, as described above, until no additional (de)correlation events are observed and the simulation has achieved the predicted timescale of the observed events events.

A. Figure note

For all figures, data are truncated at the maximum (Figure 2) or minimum (Figure 1) y-axis value.

ACKNOWLEDGMENTS

The authors wish to acknowledge the support of the Wake Forest Baptist Comprehensive Cancer Center Crystallography & Computational Biosciences Shared Resource, supported by the National Cancer Institute's Cancer Center Support Grant award number P30CA012197. The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Cancer Institute. This work was partially supported by National Institutes of Health grant T32-GM095440, supporting RLM and RCG. JX acknowledges

a fellowship from the Center for Molecular Signaling at Wake Forest University. Some computations were performed on the Wake Forest University DEAC Cluster, a centrally managed resource with support provided in part by the University.

---

[1] R. Galindo-Murillo, D. R. Roe, and T. E. Cheatham, Biochimica et biophysica acta **1850**, 1041 (2015).

[2] J. B. Clarage, T. Romo, B. K. Andrews, B. M. Pettitt, and G. N. Phillips, Proceedings of the National Academy of Sciences **92**, 3288 (1995).

[3] M. Karplus and J. A. McCammon, Nature structural biology **9**, 646 (2002).

[4] H. Flyvbjerg, in *Advances in Computer Simulation* (Springer, 1998) pp. 88–103.

[5] Y. Sun, W. Xi, and G. Wei, The Journal of Physical Chemistry B **119**, 2786 (2015).

[6] W. M. Berhanu and U. H. E. Hansmann, in *Advances in protein chemistry and structural biology*, Vol. 96 (Elsevier, 2014) pp. 113–141.

[7] W. M. Berhanu and A. E. Masunov, Biopolymers **95**, 573 (2011).

[8] W. M. Berhanu and U. H. E. Hansmann, Proteins: Structure, Function, and Bioinformatics **81**, 1542 (2013).

[9] X. Zhou, W. Xi, Y. Luo, S. Cao, and G. Wei, The Journal of Physical Chemistry B **118**, 6733 (2014).

[10] H. S. Hansen and P. H. Hünenberger, Journal of Computational Chemistry **31**, 1 (2010).

[11] C. D. Christ, A. E. Mark, and W. F. van Gunsteren, Journal of Computational Chemistry , NA (2009).

[12] Y. Sugita and Y. Okamoto, Chemical Physics Letters **314**, 141 (1999).

[13] U. H. Hansmann, Chemical Physics Letters **281**, 140 (1997).

[14] L. Sborgi, A. Verma, S. Piana, K. Lindorff-Larsen, M. Cerminara, C. M. Santiveri, D. E. Shaw, E. de Alba, and V. Muñoz, Journal of the American Chemical Society **137**, 6506 (2015).

[15] D. W. Borhani and D. E. Shaw, Journal of computer-aided molecular design **26**, 15 (2012).

[16] D. E. Shaw, R. O. Dror, J. K. Salmon, J. P. Grossman, K. M. Mackenzie, J. A. Bank, C. Young, M. M. Deneroff, B. Batson, K. J. Bowers, E. Chow, M. P. Eastwood, D. J. Ierardi, J. L. Klepeis, J. S. Kuskin, R. H. Larson, K. Lindorff-Larsen, P. Maragakis, M. A. Moraes, S. Piana, Y. Shan, and B. Towles, in *Proceedings of the Conference on High Performance*

Computing Networking, Storage and Analysis, SC '09 (ACM, New York, NY, USA, 2009) pp. 65:1–65:11.

[17] T. J. Lane, D. Shukla, K. a. Beauchamp, and V. S. Pande, Current Opinion in Structural Biology **23**, 58 (2013).

[18] F. Noé, Biophysical Journal **108**, 228 (2015).

[19] J. A. McCammon, Reports on Progress in Physics **47**, 1 (1984).

[20] J. L. Klepeis, K. Lindorff-Larsen, R. O. Dror, and D. E. Shaw, Current Opinion in Structural Biology **19**, 120 (2009).

[21] K. Henzler-Wildman and D. Kern, Nature **450**, 964 (2007).

[22] G. Ciccotti, M. Ferrario, and C. Schuette, *Molecular Dynamics Simulation*.

[23] D. Kern and E. R. Zuiderweg, Current Opinion in Structural Biology **13**, 748 (2003).

[24] W. C. Swope, J. W. Pitera, and F. Suits, The Journal of Physical Chemistry B **108**, 6571 (2004).

[25] A. van der Vaart, Theoretical Chemistry Accounts **116**, 183 (2006).

[26] B. Podobnik, D. Wang, D. Horvatic, I. Grosse, and H. E. Stanley, EPL (Europhysics Letters) **90**, 68001 (2010).

[27] S. M. Stigler, Statistical Science **4**, 73 (1989).

[28] K. Pearson, Proceedings of the Royal Society of London (1854-1905) **58**, 240 (1895).

[29] D. Halmer, G. von Basum, P. Hering, and M. Mürtz, Review of scientific instruments **75**, 2187 (2004).

[30] L. G. Ixaru and G. V. Berghe, *Exponential fitting*, Vol. 568 (Springer, 2004).

[31] T. E. Oliphant, *A guide to NumPy*, Vol. 1 (Trelgol Publishing USA, 2006).

[32] T. E. Oliphant, Computing in Science & Engineering **9** (2007).

[33] K. J. Millman and M. Aivazis, Computing in Science & Engineering **13**, 9 (2011).

[34] K. Levenberg, Quarterly of applied mathematics **2**, 164 (1944).

[35] D. W. Marquardt, Journal of the society for Industrial and Applied Mathematics **11**, 431 (1963).

[36] K. Madsen, H. B. Nielsen, and O. Tingleff, (1999).

[37] K. Gunasekaran, B. Ma, and R. Nussinov, Proteins **57**, 433 (2004).

[38] R. L. Melvin, W. H. Gmeiner, and F. R. Salsbury, The Journal of Physical Chemistry B **121**, 7803 (2017).

[39] R. Godwin, W. Gmeiner, F. R. Salsbury, F. R. Salsbury Jr, and F. R. Salsbury, Journal of Biomolecular Structure and Dynamics **34**, 125 (2016).

[40] R. C. Godwin, R. L. Melvin, W. H. Gmeiner, and F. R. Salsbury, Biochemistry **56**, 623 (2017).

[41] R. C. Godwin, W. H. Gmeiner, and F. R. Salsbury, Journal of Biomolecular Structure and Dynamics **0**, 1 (2017).

[42] J. Xiao, R. L. Melvin, and F. R. Salsbury, Phys. Chem. Chem. Phys. **19**, 24522 (2017).

[43] M. J. Harvey, G. Giupponi, G. D. Fabritiis, and G. De Fabritiis, Journal of Chemical Theory and Computation **5**, 1632 (2009).

[44] S. E. Feller, Y. Zhang, R. W. Pastor, and B. R. Brooks, The Journal of Chemical Physics **103**, 4613 (1995).

[45] H. J. C. Berendsen, J. P. M. Postma, W. F. van Gunsteren, A. DiNola, and J. R. Haak, The Journal of Chemical Physics **81**, 3684 (1984).

[46] T. Darden, D. York, and L. Pedersen, The Journal of Chemical Physics **98**, 10089 (1993).

[47] M. J. Harvey and G. De Fabritiis, Journal of Chemical Theory and Computation **5**, 2371 (2009).

[48] W. L. Jorgensen, J. Chandrasekhar, J. D. Madura, R. W. Impey, and M. L. Klein, The Journal of Chemical Physics **79**, 926 (1983).

[49] W. Humphrey, A. Dalke, and K. Schulten, Journal of molecular graphics **14**, 33 (1996).

[50] N. Foloppe, A. D. MacKerell, Jr., A. D. J. MacKerell, and A. D. MacKerell, Jr., Journal of Computational Chemistry **21**, 86 (2000).

[51] A. D. MacKerell and N. K. Banavali, Journal of Computational Chemistry **21**, 105120 (2000).

[52] A. D. MacKerell, N. Banavali, and N. Foloppe, Biopolymers **56**, 257 (2000).

[53] S. Ghosh, F. R. Salsbury Jr, D. a. Horita, and W. H. Gmeiner, Nucleic Acids Research **39**, 4490 (2011).

[54] W. H. Gmeiner, F. R. Salsbury Jr, C. M. Olsen, and L. a. Marky, Journal of nucleic acids **2011**, 631372 (2011).

[55] S. Ghosh, F. R. Salsbury Jr, D. a. Horita, and W. H. Gmeiner, Journal of Biomolecular Structure and Dynamics **31**, 1301 (2013).

[56] F. Cordier, E. Vinolo, M. Véron, M. Delepierre, and F. Agou, Journal of molecular biology **377**, 1419 (2008).

[57] I. Russo Krauss, A. Merlino, A. Randazzo, E. Novellino, L. Mazzarella, and F. Sica, Nucleic Acids Research **40**, 8119 (2012).

[58] A. Šali, T. L. Blundell, A. Sali, and T. L. Blundell, Journal of molecular biology **234**, 779 (1993).

[59] J. J. Warren, T. J. Pohlhaus, A. Changela, R. R. Iyer, P. L. Modrich, and L. S. Beese, Molecular Cell **26**, 579 (2007).

[60] E. D. Scheeff, J. M. Briggs, and S. B. Howell, Molecular pharmacology **56**, 633 (1999).

[61] L. Negureanu and F. R. Salsbury Jr, Journal of Molecular Modeling **19**, 4969 (2013).