JCTC Journal of Chemical Theory and Computation

Article

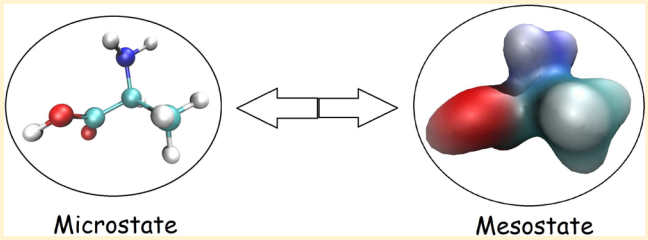
pubs.acs.org/JCTC

Multiscale Factorization Method for Simulating Mesoscopic Systems with Atomic Precision

Andrew Abi Mansour and Peter J. Ortoleva*

Department of Chemistry, Indiana University, Bloomington, Indiana 47405, United States

ABSTRACT: Mesoscopic N-atom systems derive their structural and dynamical properties from processes coupled across multiple scales in space and time. A multiscale method for simulating these systems in the friction dominated regime from the underlying N-atom formulation is presented. The method integrates notions of multiscale analysis, Trotter factorization, and a hypothesis that the momenta conjugate to coarse-grained variables constitute a stationary process on the time scale of coarse-grained dynamics. The method is demonstrated for lactoferrin, nudaurelia capensis omega virus, and human papillomavirus to assess its accuracy.



INTRODUCTION

The objective of the present study is to simulate the behavior of mesoscopic systems based on an all-atom formulation at which the basic physics is presumed known. Traditional molecular dynamics (MD) is ideal for such an approach if the number of atoms and the time scales of interest are limited.[1,2] However, ribosomes, viruses, mitochondria, and nanocapsules for the delivery of therapeutic agents are but a few examples of mesoscopic systems that can provide a challenge for conventional MD. In this paper, we develop a physics-based algorithm that accounts for interactions at the atomic scale and yet makes accurate and rapid simulations for supramillion atom systems over long time scales possible.

Typical coarse-graining (CG) methods include deductive multiscale analysis (DMA),[3,4] inverse Monte Carlo,[5] Boltzmann inversion,[6] elastic network models,[7,8] or other bead-based models.[9−11] DMA methods derived from the N-atom Liouville equation (LE) show great promise in achieving accurate and efficient all-atom simulation.[12−15] The main theme of that work was to construct and exploit the multiscale structure of the N-atom probability density $\rho(\Gamma,t)$ for the positions and momenta of the N atoms (denoted Γ, collectively) as it evolves over time t. Most of the analysis focused on friction dominated, noninertial regime, which is considered here as well. However, in these methods ensembles of all-atom configurations were required for evolving the CG variables. The approach introduced here avoids the need to construct these ensembles by coevolving the all-atom and CG states in a consistent way, and in the spirit of DMA-based methods, it does not make any conjectures on the form of the CG dynamical equations and the associated uncertainty in the form of the equations. A main theme of the present approach is the importance of coevolving the CG and microscopic states. This feature distinguishes our method from others which, for example, require the construction of a potential mean force[16,17] using ensembles of micostates; a challenge for such methods is that the relevant ensembles are not known a priori since they are controlled by the CG state whose evolution is unknown and is in fact the objective of a dynamics simulation. Other multiscale methods, built on the projection operator formalism,[18−20] require the construction of memory kernels. This is typically achieved via a perturbation approach to overcome the complexity of the appearance of the projection operator in the memory kernels. Construction of such kernels is not required in our method.

A first step in the present approach is the introduction of a set of CG variables Φ related to Γ via $\Phi = \tilde{\Phi}(\Gamma)$ for specified function $\tilde{\Phi}(\Gamma)$. When this dependence is well chosen, the CG variables evolve much more slowly than the fluctuations of small subsets of atoms. With these CG variables, the N-atom LE was solved perturbatively in terms of ε,[12,13] the ratio of the characteristic time of the fluctuations of small clusters of atoms, to the characteristic time of CG variable evolution. This is achieved starting with the ansatz that ρ depends on Γ both directly and, via $\tilde{\Phi}$, indirectly. The theory proceeds by constructing $\rho(\Gamma, \Phi;t)$ perturbatively in ε, i.e., by working in the space of functions of $6N + N_{CG}$ variables (where N_{CG} is the number of variables in the set Φ). To advance the multiscale approach, we here introduce Trotter factorization[21−23] into the analysis. Through Trotter factorization, the long-time evolution of the system separates into alternating phases of all-atom simulations and CG variable updating. Efficiency of the method follows from a hypothesis that the momenta conjugate to the CG variables can be represented as a stationary process whose statistics are preserved in an interval of time short to the characteristic time of CG dynamics. The net result is a computational algorithm with some of the character of our earlier MD/OPX method[24,25] but with better control on accuracy, higher efficiency, and more rigorous theoretical basis. Here we develop the algorithm and discuss its implementation as a computational platform, discuss selected results, and make concluding remarks.

Received: May 6, 2013
Published: January 9, 2014

THEORY AND IMPLEMENTATION

Unfolded Dynamical Formulation. The Newtonian description of an N-atom system is provided by the $6N$ atomic positions and momenta, denoted Γ collectively. The phenomena of interest involve overall transformations of an N-atom system. While Γ contains all the information needed to solve the problem in principle, here it is found convenient to also introduce a set of CG variables Φ, that are used to track the large spatial scale, long time degrees of freedom. For example, Φ could describe the overall position, size, shape, and orientation of a nanoparticle. By construction, a change in Φ involves the coherent deformation of the N-atom system, which implies that the rate of change in Φ is expected to be slow.[12,26] This slowness implies the separation of time scales that provides a highly efficient and accurate algorithm for simulating N-atom systems.

With this unfolded description (Γ, Φ), the Newtonian dynamics takes the form

$$\frac{d\Gamma}{dt} = \mathcal{L}\Gamma \tag{1}$$

$$\frac{d\Phi}{dt} = \mathcal{L}\tilde{\Phi}(\Gamma) \tag{2}$$

for unfolded Liouvillian $\mathcal{L} = \mathcal{L}_{micro} + \mathcal{L}_{meso}$, such that

$$\mathcal{L}_{micro} = \sum_{l=1}^{N} \frac{\mathbf{p}_l}{m_l}\cdot\left(\frac{\partial}{\partial \mathbf{r}_l}\right)_{\Phi} + \mathbf{f}_l\cdot\left(\frac{\partial}{\partial \mathbf{p}_l}\right)_{\Phi} \tag{3}$$

$$\mathcal{L}_{meso} = \sum_{k=1}^{N_{CG}} \Pi_k\cdot\left(\frac{\partial}{\partial \Phi_k}\right)_{\Gamma} \tag{4}$$

Here Π_k is the CG velocity associated with the kth CG variable. Equations 1 and 2 have the formal solution

$$(\Gamma(t), \Phi(t)) = S(t)(\Gamma_o, \Phi_o) \tag{5}$$

for initial data indicated by subscript o and evolution operator $S(t) = e^{\mathcal{L}t}$.

TROTTER FACTORIZATION

By taking the unfolded Liouvillian, the time operator now takes the form

$$S(t) = e^{(\mathcal{L}_{micro}+\mathcal{L}_{meso})t} \tag{6}$$

Since $\mathcal{L}_{micro}$ and $\mathcal{L}_{micro}$ do not commute, $S(t)$ cannot be factorized into a product of exponential functions. However, Trotter's theorem[21] (also known as the Lie product formula[22]) can be used to factorize the evolution operator as follows:

$$S(t) = \lim_{M\to\infty}\left[e^{\mathcal{L}_{micro}t/2M}e^{\mathcal{L}_{meso}t/M}e^{\mathcal{L}_{micro}t/2M}\right]^M + O\left(\left(\frac{t}{M}\right)^3\right) \tag{7}$$

By setting t/M to be equal to the discrete time step Δ, the stepwise operator becomes

$$S(\Delta) = \lim_{\Delta\to 0} e^{\mathcal{L}_{micro}\Delta/2}e^{\mathcal{L}_{meso}\Delta}e^{\mathcal{L}_{micro}\Delta/2} + O(\Delta^3) \tag{8}$$

Let the stepwise operators S_{micro} and S_{meso} correspond to $\mathcal{L}_{micro}$ and $\mathcal{L}_{meso}$, respectively. Then $S(n\Delta)$ takes the form

$$S(n\Delta) = \prod_{i=1}^{n} S_{micro}\left(\frac{\Delta}{2}\right)S_{meso}(\Delta)S_{micro}\left(\frac{\Delta}{2}\right) \tag{9}$$

By replacing $S_{micro}(\Delta/2)$ by $S_{micro}(\Delta)S_{micro}(-\Delta/2)$ to the right-hand side, eq 9 becomes

$$S(n\Delta) = S_{micro}\left(\frac{\Delta}{2}\right)\left[\prod_{i=1}^{n} S_{meso}(\Delta)S_{micro}(\Delta)\right]S_{micro}\left(\frac{-\Delta}{2}\right) \tag{10}$$

Since we are interested in the long-time evolution of a mesoscopic system, we can neglect the far left and right end terms, $S_{micro}(\Delta/2)$ and $S_{micro}(-\Delta/2)$, respectively, to a good approximation. Therefore, we can define the stepwise time operator as

$$S(\Delta) = S_{meso}(\Delta)S_{micro}(\Delta) \tag{11}$$

In the next section, we show how this factorization implies a computational algorithm for solving the dynamical equations for Γ and Φ.

IMPLEMENTATION

A key to the efficiency of the mutiscale Trotter factorization (MTF) method is the postulate that the momenta conjugate to the CG variables can be represented by a stationary process over a period of time much shorter than the time scale characteristic of CG evolution. Thus, in a time period significantly shorter than the increment Δ of the stepwise evolution, the system visits a representative ensemble of configurations consistent with the slowly evolving CG state. This enables one to use an MD simulation for the microscopic phase of the stepwise evolution that is much shorter than Δ to integrate the CG state to the next CG time step. For each of a set of time intervals much less than Δ, the friction dominated system experiences the same ensemble of conjugate momentum fluctuations. Thus, if δ is the time for which the conjugate momentum undergoes a representative sample of values (i.e., is described by the stationarity hypothesis), then the computational advantage over conventional MD is expected to be Δ/δ.

The two-phase updating for each time-step Δ was achieved as follows. For the $S_{micro}(\Delta)$ phase, conventional MD was used. This yields a time-series for Γ and hence Π. For all systems simulated here, Π was found to be a stationary process (see Figure 10). Therefore, MD need only be carried out for a fraction of Δ, denoted δ. This and the slowness of the CG variables are the source of computational efficiency of our algorithm. For the S_{meso} phase updating in the friction dominated regime, the Π time series constructed in the microphase is used to advance Φ in time as follows

$$\Phi(t+\Delta) = \Phi(t) + \int_t^{t+\Delta} dt'\Pi(t') \tag{12}$$

Due to stationarity, the integral on the right-hand side reduces to $\Delta/\delta\int_t^{t+\delta}dt'\ \Pi(t')$ (see Figure 10). The expression for Π depends on the choice of CG variables. In this work, we used the space-warping method[27,28] that maps a set of atomic coordinates to a set of CG variables that capture the coherent deformation of a molecular system in space. In the space-warping method, the mathematical relation between the CG variables and the atomic coordinates is

$$\mathbf{r}_i = \sum_{\underline{k}} \mathbf{U}_{\underline{k}i}\mathbf{\Phi}_{\underline{k}} + \sigma_i \tag{13}$$

dx.doi.org/10.1021/ct400615a | J. Chem. Theory Comput. 2014, 10, 518−523

Here $\underline{k}$ is a triplet of indices, i is the atomic index, $\mathbf{r}_i$ is the Cartesian position vector for atom i, and $\mathbf{\Phi}_{\underline{k}}$ is a Cartesian vector for CG variable $\underline{k}$. The basis functions $\mathbf{U}_{\underline{k}}$ are constructed in two stages. In the first stage, they are computed from a product of three Legendre polynomials of order k_1, k_2, and k_3 for the x, y, and z dependence. In the second stage, the basis functions are mass-weighted orthogonalized via QR decomposition.[12,26] For instance, the zeroth order polynomial is $\mathbf{U}_{000}$, the first order polynomial forms a set of three basis functions: $\mathbf{U}_{001}$, $\mathbf{U}_{010}$, $\mathbf{U}_{100}$, and so on. Furthermore, the basis functions depend on a reference configuration $\mathbf{r}^0$ which is updated periodically (once every 10 CG time steps) to control accuracy. The vector $\boldsymbol{s}_i$ represents the atomic-scale corrections to the coherent deformations generated by $\mathbf{\Phi}_{\underline{k}}$. Introducing CG variables this way facilitates the construction of microstates consistent with the CG state.[28] This is achieved by minimizing $\sum_{i=1}^{N} m_i\sigma_i^2$ with respect to $\mathbf{\Phi}_{\underline{k}}$. The result is that the CG variables are generalized centers of mass, specifically

$$\mathbf{\Phi}_{\underline{k}} = \frac{\sum_{i=1}^{N} m_i \mathbf{U}_{\underline{k}i}\mathbf{r}_i}{\sum_{i=1}^{N} m_i \mathbf{U}_{\underline{k}i}^{\,2}} \tag{14}$$

with m_i being the mass of atom i. For the lowest order CG variable, $\mathbf{U}_{000} = 1$, which implies $\mathbf{\Phi}_{000}$ is the center of mass. As the order of the polynomial increases, the CG variables capture more information from the atomic scale, but they vary less slowly with time. Therefore, the space warping CG variables are classified into low-order and high-order variables. The former characterize the larger scale disturbances, while the latter capture short-scale ones.[12,26] Equation 14 implies that $\mathbf{\Pi}_{\underline{k}} = \sum_{i=1}^{N} \mathbf{U}_{\underline{k}i}\mathbf{p}_i / \sum_{i=1}^{N} m_i \mathbf{U}_{\underline{k}i}^{\,2}$, where $\mathbf{p}_i$ is a vector of momenta for the i^{th} atom. With $\mathbf{\Phi}(t + \Delta)$ computed via eq 12, the two-phase Δ update is completed and this cycle is repeated for a finite number of discrete time steps. Details on the necessary energy minimization and equilibriation needed for every CG step was covered in earlier work.[12,24,25] This two-phase coevolution algorithm was implemented using NAMD[1] for the S_{micro} phase within the framework of the DMS software package.[3,12,29] Numerical computations were performed with the aid of LOOS,[30] a lightweight object-oriented structure library for MD simulation analysis.

RESULTS AND DISCUSSION

All simulations were done in vacuum under *NVT* conditions to assess the efficiency and accuracy of the algorithm. The first system used for validation and benchmarking is lactoferrin. This iron binding protein is composed of a distal and two proximal lobes (shown in Figure 1). Two free energy minimizing conformations have been demonstrated experimentally: diferric with closed proximal lobes (PDB code 1LFG) and apo with open ones[31,32] (PDB code 1LFH). Here, we start with an open lactoferrin structure and simulate its closing in vacuum. A total of 10 × 3 CG variables are used (i.e., $k_1 + k_2 + k_3 \leq 2$) to coarse-grain the protein, and the highest-order basis functions are therefore quadratic. This level of coarse-graining captures translation, rotation, contraction, and bending of a macromolecule.[27] The RMSD for lactoferrin is plotted as a function of time in Figure 4; it shows that the protein reaches equilibrium in about 5 ns. This transition leads to a decrease in the radius of gyration of the protein by approximately 0.2 nm as shown in Figure 5.

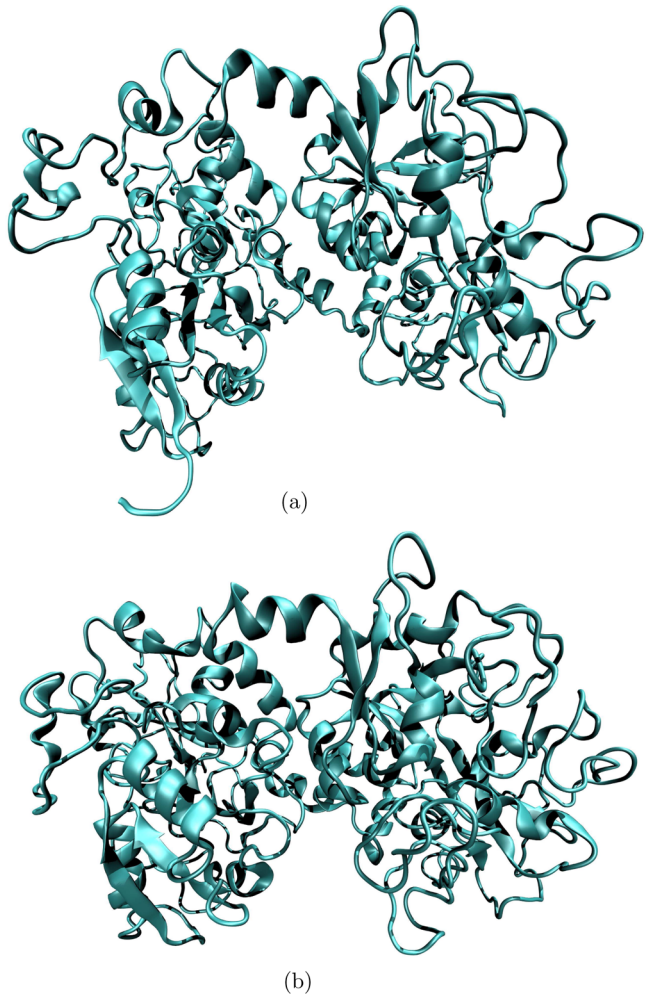


Figure 1. Snapshots of lactoferrin protein in its open state at t = 0 ns (a) and its closed state at t = 19.6 ns (b).

The second system simulated is a triangular structure of the nudaurelia capensis omega virus (Nωv) capsid protein[33] (PDB code 1OHF) containing three protomers (shown in Figure 2).

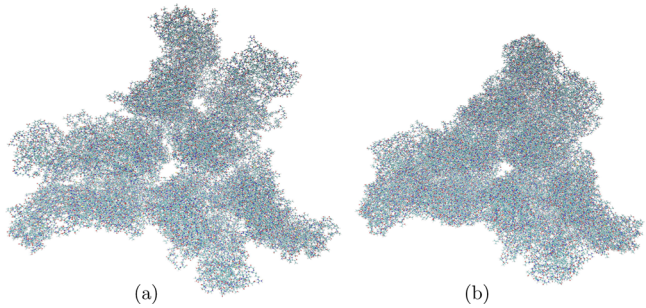


Figure 2. Snapshots of Nωv triangular structure before (initial state at t = 0 ns) (a) and after (shrinking at t = 3.0 ns) (b) contraction due to strong protein−protein interactions.

Starting from a deprotonated state (at low pH), the system was equilibriated using an implicit solvent. A total of 10 × 3 CG variables are used (i.e., $k_1 + k_2 + k_3 \leq 2$) to coarse-grain the structure. This system is characterized by strong protein−protein interactions. As a result, the proteins shrink in vacuum

dx.doi.org/10.1021/ct400615a | J. Chem. Theory Comput. 2014, 10, 518−523

after a short period of equilibriation. The computed radius of gyration of Nωv is shown in Figure 6.

The third system simulated is an assembly of L1 proteins for the human papillomavirus (HPV). This system supports a stable $T = 1$ structure (Figure 3). Here a multiscale simulation is initiated by configuring a set of nine pentamers (in contrast to the twelve pentamers constituting the $T = 1$ structure). Partial structures are of interest as they arise along the self-assembly pathway and their stability could provide a kinetic road block for their complete self-assembly. The space-warping variables were used to coarse-grain the system at the pentamer level, i.e., each of the nine pentamers was replaced by its center of mass. This level of coarse-graining captures the translational movement of the pentamers as they move closer to one another in order to minmize their free energy. The number of hydrogen bonds increases in time, which is captured by the MTF algorithm (shown in Figure 7). For the first and second pentamers, the z-component of their centers of mass and the autocorrelation functions of their y-components are shown in Figures 8 and 9, respectively.

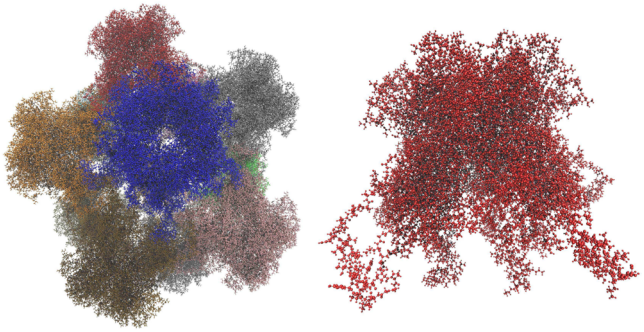


Figure 3. Human papillomavirus (HPV) capsid-like structure of L1 proteins (left) is divided into nine pentameric subsystems (right), each of which has its own set of CG variables (taken here to be the center of mass of each pentamer, i.e., $\mathbf{\Phi}_{000}$).

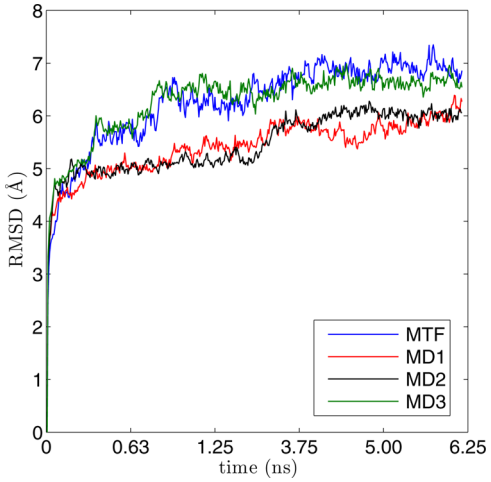


Figure 4. RMSD variation as a function of time for a series of three MD and one MTF runs.

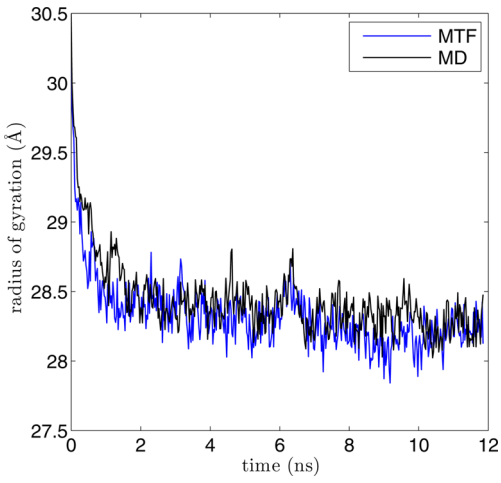


Figure 5. Radius of gyration decreases in time as lactoferrin shrinks.

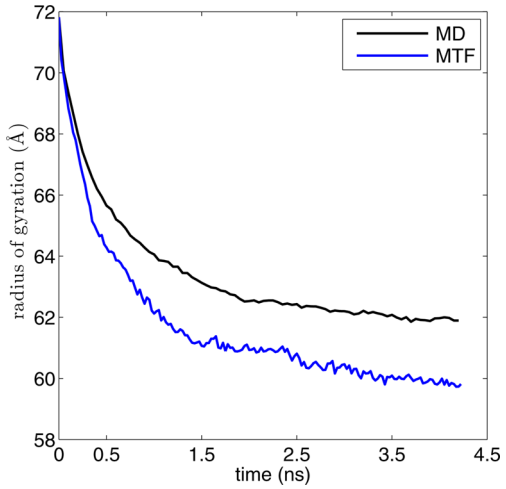


Figure 6. Temporal evolution of the radius of gyration of Nωv computed using MD and MTF.

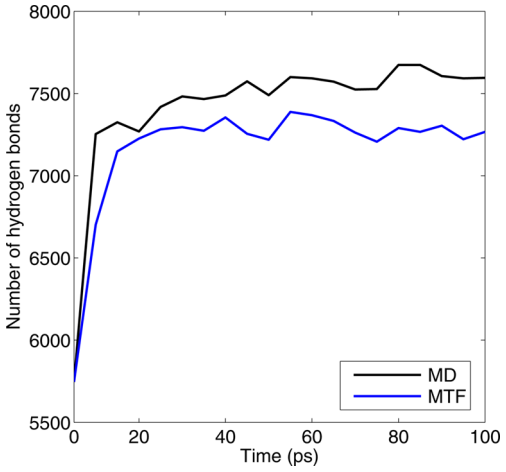


Figure 7. Time variation in the number of hydrogen bonds obtained via MD and MTF for the HPV $T = 1$ assembly of L1 proteins.

On the basis of the convergence of the time integral of Π (shown in Figure 10), the S_{micro} phase was chosen to consist of 5 ps for LFG, 10 ps for Nωv, and 1 ps for HPV. The CG time step, Δ, was taken to be 12.5 ps for LFG, 25 ps for Nωv, and 10 ps for HPV. This yielded a speedup (with respect to MD) of 1.32, 2.15, and 7.15, respectively.

dx.doi.org/10.1021/ct400615a | J. Chem. Theory Comput. 2014, 10, 518−523

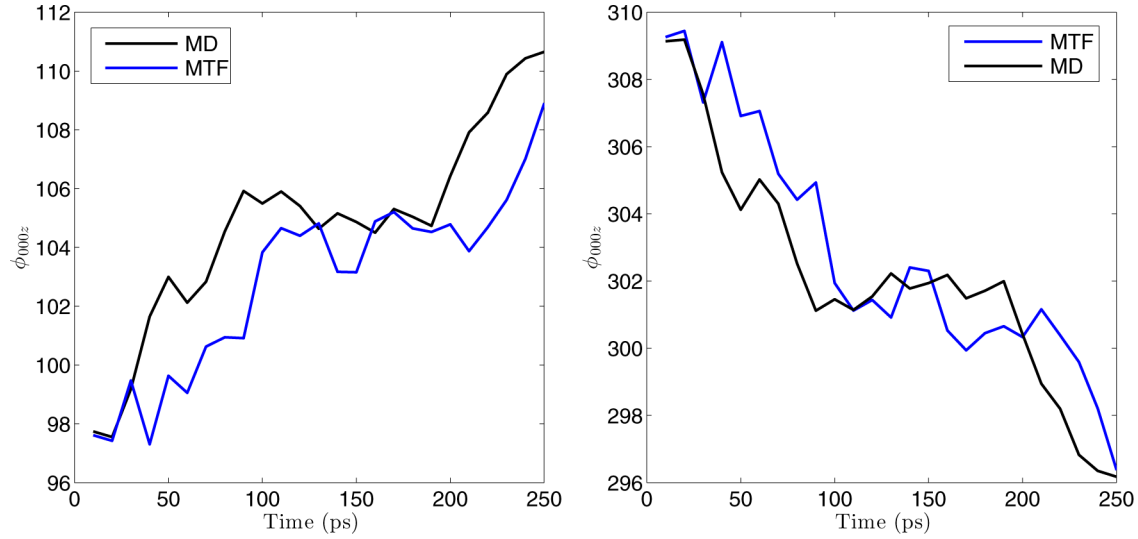


Figure 8. z-Component of the centers of mass for the first (left) and second (right) pentamers increases in time as the nine pentamers in the HPV assembly move closer to one another.

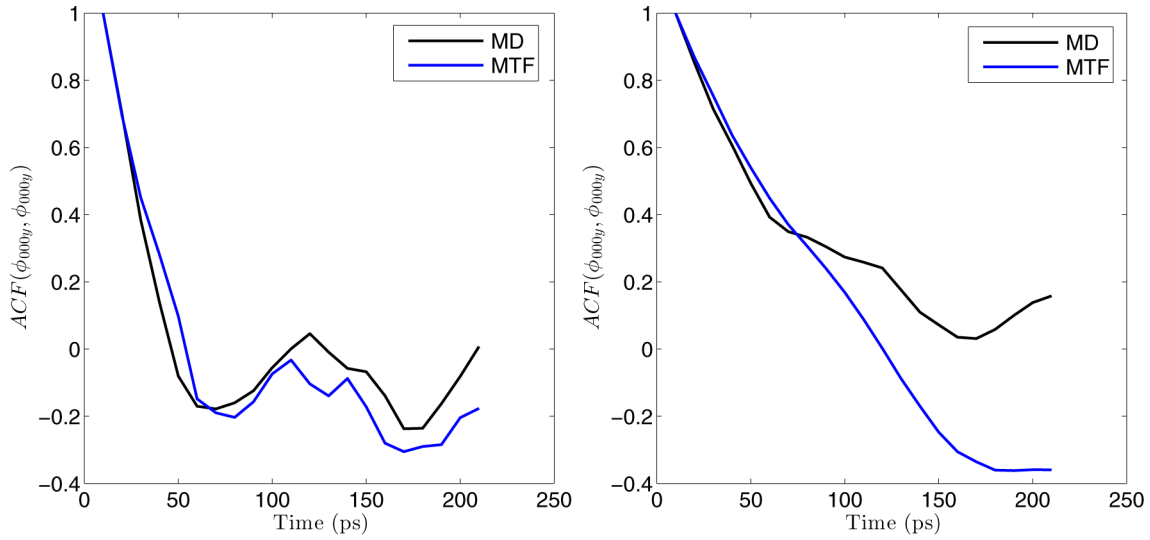


Figure 9. Time decay in the autocorrelation function (ACF) for the y component of the center mass for the first (left) and second (right) pentamer in the HPV assembly of proteins.

CONCLUSIONS

Mesoscopic systems express behaviors stemming from atom−atom interactions across many scales in space and time. Earlier approaches based on Langevin equations for coarse-grained variables showed agreement with MD without comprimising accuracy and captured key atomic scale details.[12,34] However, such an approach requires the construction of diffusion factors, a task that consumes significant computational resources. This is because of the need to use large ensembles and construct correlation functions.

The multiscale factorization method used here introduces the benefits of multiscale theory of the LE. Here we revisit the multiscale method within our the context of MTF which avoids the need for the resource-consuming construction of diffusion factors, and thermal average and random forces. The CG variables for the mesoscopic systems of interest do have a degree of stochastic behavior. In the present formulation, this stochasticity is accounted for via a series of MD steps used in the phase of the multiscale factorization algorithm wherein the state of the system is evolved via $\mathcal{L}_{\text{micro}}$, i.e. at constant value of the CG variables.

The MTF algorithm can be further optimized to produce greater speedup factors. In particular, the results obtained here can be significantly improved with the following: (1) after updating the CGs in the two-phase coevolution Trotter cycle, one must fine-grain, i.e. develop, the atomistic configuration to be used as an input to MD. We have shown that the CPU time to achieve this fine-graining can be dramatically reduced via a constraint method that eliminates bond length and angle strains, (2) information from earlier steps in discrete time coevolution can be used to increase the time step and achieve greater numerical stability; while this was demonstrated for one multiscale algorithm,[29] it can also be applied to MTF, and (3) the time stepping algorithm used in this work is the analogue of the Euler method for differential equations, and greater numerical stability and efficiency could be achieved for a system of stiff differential equations using implicit and semi-implicit schemes.[35]

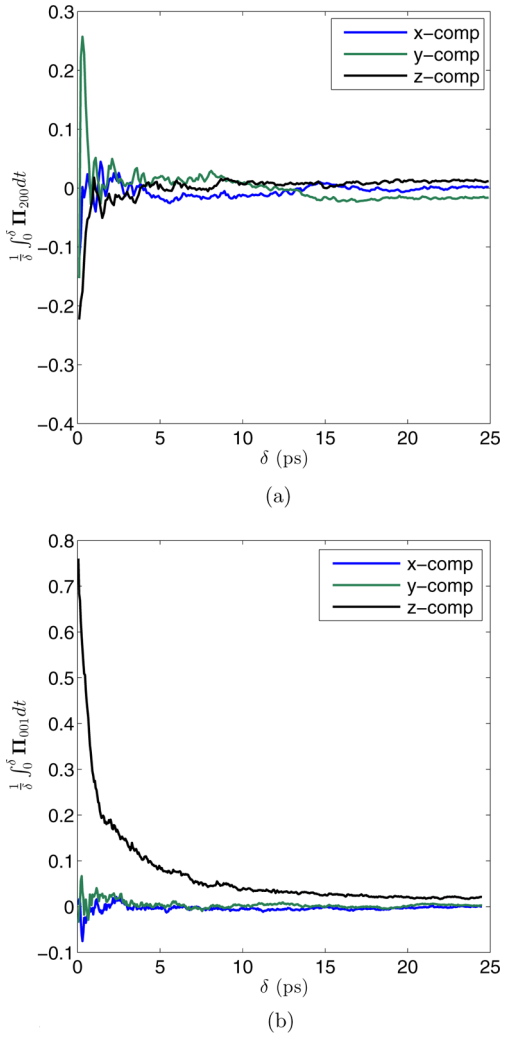


Figure 10. Evidence for the validity of the stationarity hypothesis shown via the convergence of $(1/\delta)\int_0^\delta \Pi(t)\ dt$ as a function of δ for CG variables selected from among those used in simulating the contraction of Nωv. Initially the integral experiences large fluctuations because with small δ, only a relatively few configurations are included in the time average constituting the integral, but as δ increases, the statistics improves, and the integral becomes increasingly flat. (a) Time integral of Π for a high-order CG Φ_{200}. (b) (a) Time integral of Π for a low-order CG Φ_{001}.

AUTHOR INFORMATION

Corresponding Author

*E-mail: ortoleva@indiana.edu.

Notes

The authors declare no competing financial interest.

ACKNOWLEDGMENTS

This project was supported in part by the National Science Foundation (Collaborative Research in Chemistry Program), National Institutes of Health (NIBIB), METAcyt through the Center of Cell and Virus Theory, Indiana University College of Arts and Sciences, and the Indiana University information technology services (UITS) for high performance computing resources.

REFERENCES

(1) Phillips, J. C.; Braun, R.; Wang, W.; Gumbart, J.; Tajkhorshid, E.; Villa, E.; Chipot, C.; Skeel, R. D.; Kale, L.; Schulten, K. *J. Comput. Chem.* **2005**, *26*, 1781−1802.

(2) Spoel, V. D.; Lindahl, E.; Hess, B.; Groenhof, G.; Mark, A. E.; Berendsen, H. J. C. *J. Comput. Chem.* **2005**, *26*, 1701−1718.

(3) Cheluvaraja, S.; Ortoleva, P. J. *J. Chem. Phys.* **2010**, *132*, 75102−75110.

(4) Joshi, H.; Singharoy, A.; Sereda, Y. V.; Cheluvaraja, S. C.; Ortoleva, P. J. *Prog. Bio.-Phys. Mol. Biol.* **2011**, *107*, 200−217.

(5) Murtola, T.; Falck, E.; Karttunen, M.; Vattulainen, I. *J. Chem. Phys.* **2007**, *126*, 75101−75114.

(6) Reith, D.; Putz, M.; Muller-Plathe, F. *J. Comput. Chem.* **2003**, *24*, 1624−1636.

(7) Bahar, I.; Atilgan, R. A.; Erman, B. *Folding Design* **1997**, *2*, 173−181.

(8) Haliloglu, T.; Bahar, I.; Erman, B. *Phys. Rev. Lett.* **1997**, *79*, 3090−3093.

(9) Shiha, A. Y.; Arkhipov, A.; Freddolino, P. L.; Schulten, K. *J. Phys. Chem. B* **2006**, *110*, 3674−3684.

(10) Shiha, A. Y.; Freddolinoa, P. L.; Arkhipova, A.; Schulten, K. *J. Struct. Biol.* **2007**, *157*, 579−592.

(11) Marrink, S. J.; de Vries, A. H.; Mark, A. E. *J. Phys. Chem. B* **2004**, *108*, 750−760.

(12) Singharoy, A.; Cheluvaraja, S.; Ortoleva, P. J. *J. Chem. Phys.* **2011**, *134*, 44104−44120.

(13) Ortoleva, P. J. *J. Phys. Chem. B* **2005**, *109*, 21258−21266.

(14) Pankavich, S.; Shreif, Z.; Ortoleva, P. *J. Phys. A* **2008**, *387*, 4053−4069.

(15) Pankavich, S.; Shreif, Z.; Miao, Y.; Ortoleva, P. J. *J. Chem. Phys.* **2009**, *130*, 194115−194124.

(16) Abrams, J. B.; Tuckerman, M. E. *J. Phys. Chem. B* **2008**, *112*, 1574215757.

(17) Noid, W. G.; Chu, J.-W.; Ayton, G. S.; Krishna, V.; Izvekov, S.; Voth, G. A.; Das, A.; Andersen, H. C. *J. Chem. Phys.* **2008**, *128*, 244114−244124.

(18) Shea, J.-E.; Oppenheim, I. *J. Phys. Chem.* **1996**, *100*, 19035−19042.

(19) Shea, J.-E.; Oppenheim, I. *Phys. A* **1997**, *247*, 417−443.

(20) Shea, J.-E.; Oppenheim, I. *Phys. A* **1998**, *250*, 265−294.

(21) Trotter, H. F. *Proc. Am. Math. Soc.* **1959**, *10*, 545−551.

(22) Hall, B. C. *Lie groups, Lie algebras, and representations: an elementary introduction*; Springer, 2003; Vol. *10* (III); pp 36−37.

(23) Tuckerman, M.; Berne, B. J.; Martyna, G. J. *J. Chem. Phys.* **1992**, *97*, 1990−2001.

(24) Miao, Y.; Ortoleva, P. *J. Comput. Chem.* **2008**, *30*, 423−437.

(25) Miao, Y.; Johnson, J. E.; Ortoleva, P. J. *J. Phys. Chem. B* **2010**, *114*, 11181−11195.

(26) Singharoy, A.; Joshi, H.; Miao, Y.; Ortoleva, P. J. *J. Phys. Chem. B* **2012**, *116*, 8423−8434.

(27) Jaqaman, K.; Ortoleva, P. J. *J. Comput. Chem.* **2002**, *23*, 484−491.

(28) Pankavich, S.; Miao, Y.; Ortoleva, J.; Shreif, Z.; Ortoleva, P. J. *J. Chem. Phys.* **2008**, *128*, 234908−234921.

(29) Singharoy, A.; Joshi, H.; Ortoleva, P. J. *J. Chem. Inf. Model.* **2012**, *52*, 2638−2686.

(30) Romo, T.; Grossfield, A. *31st Annual International Conference of the IEEE EMBS*, Minneapolis, MN, Sep 2−6, 2009; pp 2332−2335.

(31) Norris, G. E.; Anderson, B. F.; Baker, E. N. *Acta Crystallogr. Sect. B* **1991**, *47*, 998−1004.

(32) Kim, M. K.; Jernigan, R. L.; Chirikjian, G. S. *J. Biophys.* **2005**, *89*, 43−55.

(33) Taylor, D. J.; Wang, Q.; Bothner, B.; Natarajan, P.; Finn, M. G.; Johnson, J. E. *J. Chem. Commun.* **2003**, *22*, 2770−2771.

(34) Singharoy, A.; Sereda, Y.; Ortoleva, P. J. *J. Chem. Theory Comput.* **2012**, *8*, 1379−1392.

(35) Iserles, A. *A First Course in the Numerical Analysis of Differential Equations*; Cambridge University Press, 2008; pp 53−63.