



Communication

Self-Guided Molecular Simulation to Enhance Concerted Motion

Xiongwu Wu * and Bernard R. Brooks

Laboratory of Computational Biology, National Heart, Lung and Blood Institute (NHLBI), National Institutes of Health (NIH), Bethesda, MD 20892, USA

* Correspondence: wuxw@nhlbi.nih.gov

Abstract: Self-guided (SG) molecular simulation methods, namely self-guided molecular dynamics (SGMD) and self-guided Langevin dynamics (SGLD), enhance conformational search by promoting low-frequency motion. A simple local time averaging scheme is used to extract low-frequency properties with little overhead in computing costs. For molecular processes to form ordered structures like ligand binding and protein folding, it is believed that concerted motions play crucial roles. To enhance the concerted motion in molecular systems, we propose a spatial averaging scheme to extract the concerted motion of a local region. Applying guiding forces based on spatial averaging, self-guided molecular simulations can enhance concerted motion and reach ordered structures more efficiently. Through simulations of amyloid fibril peptides, we demonstrated that the spatial averaging in self-guided Langevin dynamics results in accelerated β -sheet formation.

Keywords: molecular simulation; molecular dynamics; Langevin dynamics; algorithm; conformational search; conformational sampling

1. Introduction

Self-guided molecular dynamics (SGMD) and self-guided Langevin dynamics (SGLD), namely self-guided (SG) molecular simulation methods, are developed to enhance conformational search and sampling. Many applications can be found in the review article [1]. Most enhanced conformational sampling methods rely on certain biased potentials to help overcome energy barriers, such as metadynamics [2], accelerated molecular dynamics [3], and self-adapted accelerated molecular dynamics [4]. SG methods do not rely on a priori energy barrier information to enhance sampling. Instead, they achieve an enhanced conformational search through promoting low-frequency motion. Low-frequency motion is a well-known limiting factor for conformational searching and sampling. Low-frequency momentum is related to diffusional motion, which is essential for diffusion-controlled conformational change, while low-frequency force is related to a smoothed energy surface, which is responsible for energy-barrier-controlled conformational change.

SG methods are unique in their ability to extract low-frequency properties through a simple local averaging scheme, which does not incur a significant overhead in computing costs. This local averaging scheme produces running time average properties. Guiding forces based on these local time averages would promote low-frequency motion. There are many types of motion in molecular systems, such as thermal motion, diffusion, binding, folding, etc. Most motion is random in nature, whereas some is concerted and leads to more ordered structures. Concerted motions are believed to be essential for structure changes in molecular systems, such as crystallization, ligand binding, protein folding, etc. To identify the concerted motion, we propose a spatial averaging scheme in addition to the local time



Academic Editors: Paulino Gómez-Puertas and James K. Bashkin

Received: 24 January 2025

Revised: 18 February 2025

Accepted: 22 February 2025

Published: 25 February 2025

Citation: Wu, X.; Brooks, B.R.

Self-Guided Molecular Simulation to Enhance Concerted Motion. *Int. J. Mol. Sci.* **2025**, *26*, 1969. <https://doi.org/10.3390/ijms26051969>

Copyright: © 2025 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license

(<https://creativecommons.org/licenses/by/4.0/>).

averaging. Guiding forces based on the concerted motion can be used to enhance such motion in SG simulations.

The concept and implementation of the spatial averaging scheme are described in the method section. A demonstration simulation of β -sheet formation is presented as an application of the method. The spatial averaging presented here has been implemented in current versions of CHARMM [5] and Amber [6], which are molecular simulation and modeling packages widely used in computational studies.

2. Results and Discussion

To demonstrate the effect of spatial averaging in SGLD simulations, we chose a system of amyloid fibrils to simulate the formation of β -sheets. Aggregation of amyloid fibrils has been identified as a leading cause of Alzheimer's disease. Extensive experimental and computational studies have been carried out to study the mechanism of amyloid fibril aggregation. Computer simulation is a valuable approach for studying this mechanism at the atomic level. However, due to the large conformational space and rugged energy landscape, molecular simulation of amyloid fibril is often trapped into local minimum conformations. Therefore, an enhanced conformational search technique is needed to efficiently simulate the aggregation process.

To illustrate the effects of spatial averaging, we performed LD, SGLD, and SGLD with spatial averaging. The amyloid-driving segments of the tetrameric thyroxine transport protein transthyretin (TTR) [7] were chosen for their simplicity. Figure 1 shows the structure of the amyloid-driving segments of TTR, $^{119}\text{TAVVTN}^{124}$.

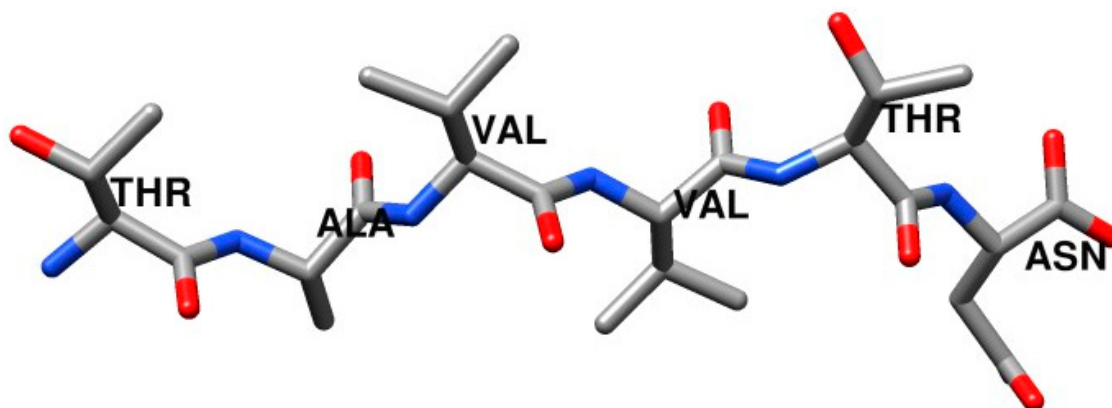


Figure 1. The amyloid-driving segments of the tetrameric thyroxine transport protein transthyretin (TTR), $^{119}\text{TAVVTN}^{124}$. Only heavy atoms are shown. C, O, and N atoms are colored as grey, red, and blue, respectively.

Instead of using a periodic boundary condition, we construct an EMAP box to limit the simulation molecules from flying away. An EMAP box is a map object that has densities at the boundary and which interacts with simulation objects with a map potential [8]:

$$E_{\text{map}} = -c_{\text{map}} \sum_a^N m_a \hat{\rho}(x_a, y_a, z_a) \quad (1)$$

$\hat{\rho}(x_a, y_a, z_a)$ is the density at position (x_a, y_a, z_a) . The restraint constant, C_{map} , sets the strength of the map-restraint. The units of atomic mass, m_a and c_{map} are g/mol and kcal/g, respectively. The EMAP boundary is generated in the simulation, and no EMAP file is needed. However, existing boundary EMAP files can be used if users want. In this work, the boundary map is a $20 \times 20 \times 20 \text{ \AA}^3$ cube with a grid space of 4 \AA . A negative restraint constant, $c_{\text{map}} = -1.0 \text{ g/mol}$, is used to repel atoms from approaching the boundary.

The simulations are performed with a friction constant of $\gamma = 1/\text{ps}$. For SGLD simulations, the guiding factor is set to $\lambda = 0.2$, and the local average time is $t_L = 0.2/\text{ps}$. For the spatial averaging, we chose $d_L = 5\text{\AA}$. The AMBER force field ff19sb [9] is used for energy calculation. The generalized Born solvation model is used to calculate the solvation energy [10,11]. AMBER simulation program PMEMD [6] is used to perform the simulation. The SGLD method and its spatial averaging scheme, as well as the EMAP functions, have been implemented in current versions of CHARMM [5] and Amber [6] and are available in either Sander or PMEMD.

Figure 2 shows the initial conformation for the simulations, where 40 TAVVTN peptides are randomly placed in the EMAP boundary box. Starting from this initial conformation, three simulations, one with Langevin dynamics (LD), one with SGLD, and one with SGLD and spatial averaging, are carried out.

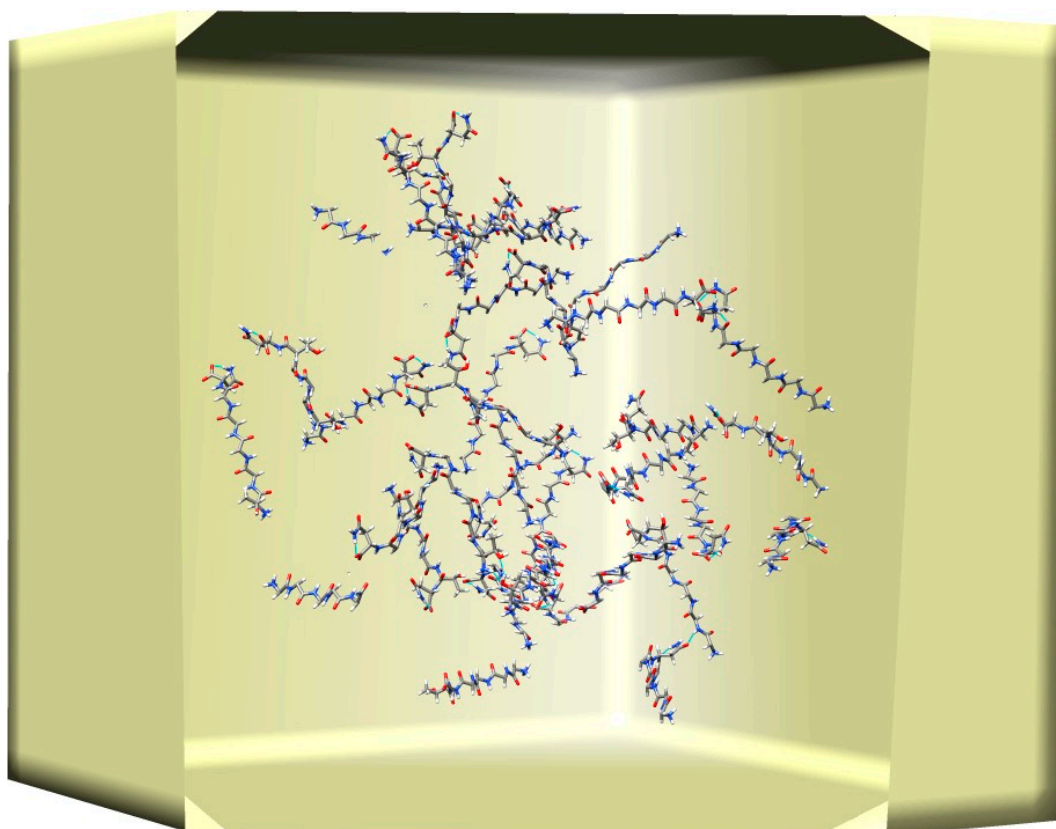


Figure 2. The initial conformation of the simulations, where 40 TAVVTN peptides are randomly placed in a $20 \times 20 \times 20 \text{\AA}^3$ cubic EMAP boundary box. Only heavy atoms are shown. C, O, and N atoms are colored as grey, red, and blue, respectively.

Figure 3 shows the potential energy profiles along the three trajectories. As can be seen, potential energies decrease along rugged curves. The SGLD curve shows frequent energy jumps, indicating its guiding forces can lead to conformational changes with significant energy increases. With spatial averaging, SGLD has a faster energy decrease at the beginning than LD. In 5000 ps, all three simulations reach similar energy levels.

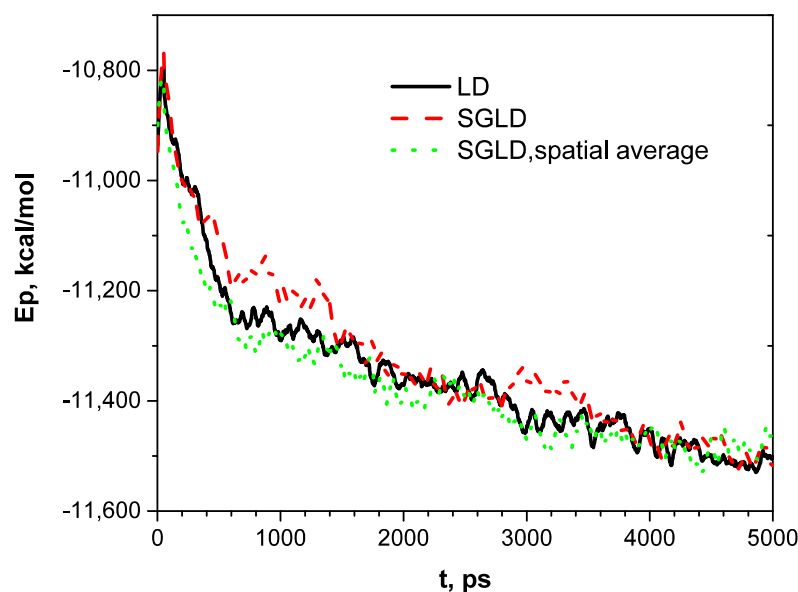


Figure 3. Potential energies along the LD, SGLD, and SGLD/spatial averaging simulation trajectories.

Figure 4 shows the conformational changes, as measured by β -strand fractions. The directory of secondary structure of proteins (DSSP) [12] is employed for the analysis. Starting from a fraction of 0, the LD simulation reaches a fraction of 0.18 in the first 400 ps, then slowly grows to 0.22 and fluctuates around this level. The SGLD simulation reaches 0.2 after 600 ps, a little bit slower than the LD simulation, but grows to 0.3 after 2200 ps and fluctuates around this value after that. With spatial averaging, SGLD quickly reaches 0.3 in 600 ps and fluctuates around this level afterward. These simulation results show that the LD simulation quickly descends to a local minimum conformation with a β -strand fraction of 0.18 and is trapped there for a long time with slow conformational changes. The SGLD simulation can quickly overcome energy barriers and reach states with high β -strand fractions. With spatial averaging, the SGLD quickly reaches states with high β -strand fractions, faster than the SGLD simulation without spatial averaging. In other words, the spatial averaging promoted the formation of the β -sheet conformation. From the fluctuations in Figure 4, one can see that SGLD and SGLD/spatial average can result in large structure changes that would help a system overcome energy barriers. SG simulations will not focus on any specific structure but will move faster from one structure to another structure.

Figure 5 illustrates a typical β -sheet conformation reached in the SGLD/spatial averaging simulation. Clearly, we can see several β -sheets formed by β -strands of the peptides. These β -sheets formation processes provide us opportunities to analyze the atomic details of amyloid fibril aggregation, which is beyond the scope of this paper and will be discussed in future studies.

The acceleration in β -sheet formation demonstrates enhanced concerted motion due to spatial averaging. Concerted motion is different in structured regions and unstructured regions. Motion in intrinsically disordered proteins will be enhanced by regular time local averaging but may not be enhanced by spatial averaging because, in unstructured regions, concerted motion may be weak.

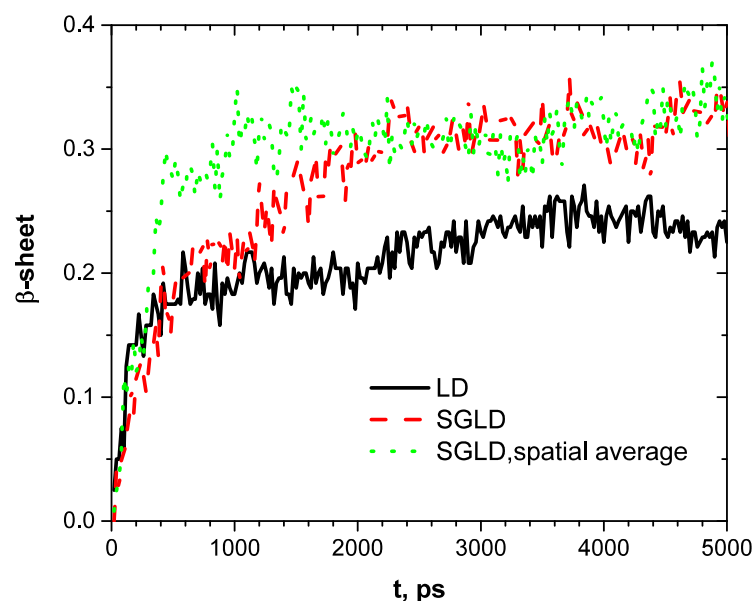


Figure 4. β -sheet fractions along the LD, SGLD, and SGLD/spatial averaging simulation trajectories.

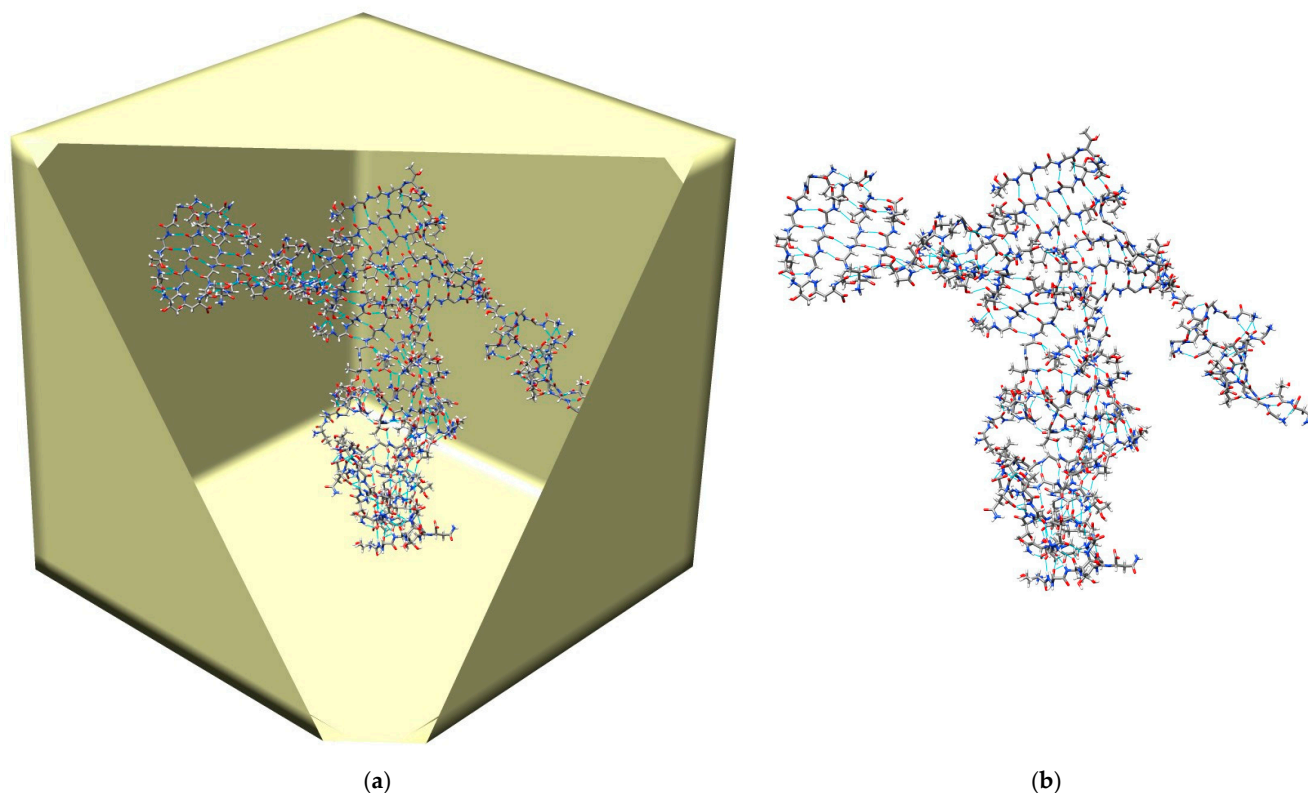


Figure 5. A typical β -sheet conformation formed in the SGLD/spatial averaging simulation. (a) The β -sheet inside the EMAP box; (b) the β -sheet alone. Only heavy atoms are shown. C, O, and N atoms are colored as grey, red, and blue, respectively.

The EMAP boundary box used in these simulations has significant effects on the simulation results. First, the size of the boundary limits the conformational space of the simulation system. Second, the strength of the boundary potential may control how fast a molecule bounces back when hitting the boundary. Third, the shape of the boundary may limit the geometry of the aggregation. Therefore, the boundary box should be large enough to minimize these effects and small enough to reduce the conformational search space.

The above demonstration simulations show the differences among LD, SGLD, and SGLD/spatial averaging. Thorough analysis and comparison will be conducted in our future publications.

3. Materials and Methods

3.1. The Equation of Self-Guided Motion

The equation of generalized self-guided motion [13] has the following form:

$$\dot{\mathbf{p}} = \mathbf{F} + \mathbf{g} \quad (2)$$

Here, \mathbf{p} is the atom's momentum, and $\dot{\mathbf{p}}$ is its time derivative. \mathbf{F} is the apparent force acting on the atom, including all contributions such as molecular interactions, constraint forces from SHAKE [14] if present, friction forces and random forces in Langevin dynamics, velocity scaling for constant temperature simulations, etc. \mathbf{g} is the guiding force that is based on the local average momentum, $\tilde{\mathbf{p}}$, and the local average forces, $\tilde{\mathbf{F}}$ and $\tilde{\tilde{\mathbf{F}}}$:

$$\mathbf{g} = \lambda \zeta \tilde{\mathbf{p}} + \mu (\tilde{\mathbf{F}} - \tilde{\tilde{\mathbf{F}}}) \quad (3)$$

λ , and force guiding factor, μ . The low-frequency characteristics of the guiding forces provide a way to alter the low-frequency motion of the simulation system. The parameter ζ represents an apparent friction constant of the simulation system, which is estimated during the SG simulation as follows:

$$\zeta = - \frac{\langle (\tilde{\mathbf{F}} - \tilde{\tilde{\mathbf{F}}}) \cdot \tilde{\mathbf{p}} \rangle}{\langle \tilde{\mathbf{p}} \cdot \tilde{\mathbf{p}} \rangle} \quad (4)$$

The tilde “~” on a property represents its local time average, which is calculated during a simulation using the following scheme along trajectories:

$$\tilde{P}(t) = \frac{1}{t_L} \int_{-\infty}^t P e^{-\frac{t-\tau}{t_L}} d\tau \approx \left(1 - \frac{\delta t}{t_L}\right) \tilde{P}(t - \delta t) + \frac{\delta t}{t_L} P(t) \quad (5)$$

Here, P is any time-dependent property, and \tilde{P} is its local time average. Likewise, $\tilde{\tilde{P}}$ is the local time average of \tilde{P} . t_L is the local average time, which defines a frequency threshold, $1/t_L$. Properties with frequencies higher than this threshold are filtered down, while those with lower frequencies are mostly kept by the averaging.

While both guiding forces can influence the low-frequency motion, the momentum guiding force (when $\lambda > 0$) would promote diffusion-controlled conformational search, and the force guiding force (when $\mu < 0$) would promote energy-barrier-controlled conformational search. The two guiding effects would cancel out each other when the guiding factors satisfy the following relation:

$$\lambda_\mu = (1 + \mu)^2 - \frac{1}{1 + \mu} \quad (6)$$

We call λ_μ the balanced momentum guiding factor of μ .

3.2. The Spatial Averaging Scheme

The idea of spatial averaging is to extract common motions shared in a local region. Vibration, shaking, random walk, etc., are not shared motions and will be canceled out in spatial averaging. Complexes, folded structures, crystalline, etc., will have shared motion

by their atoms. The understanding that these shared motions are essential for the formation of these structures leads us to develop the spatial averaging scheme.

We define the spatial averaging by the following formula:

$$\hat{p}_i = m_i \frac{\sum_{j \in \Omega_i} \tilde{p}_j w_{ij}}{\sum_{j \in \Omega_i} m_j w_{ij}} \quad (7)$$

Here, we use \hat{p}_i to represent the spatial averaging of property p_i . w_{ij} is a weighting factor for spatial averaging. Typically, $w_{ij} = 1$. Ω_i defines the spatial range of atom i , which can be chemical structures or a region in space. Several spatial range types are available in the CHARMM [5] and Amber [6] implementations:

Type 1: No spatial averaging. $\Omega_i = i$

Type 2: Averaging over bond-length and bond-angle atoms. Ω_i contains atoms strongly bonded with atom i .

Type 3: Averaging over bond-length, bond-angle, and dihedral angle atoms. Ω_i contains atoms in its local chemical structure.

Type 4: Averaging over all atoms within a local distance, d_L . Ω_i contains all atoms within a certain distance.

For type 4 spatial averaging, we adopted EMAP functions [8,15] implemented in AMBER [6] and CHARMM [5] for calculation. As shown in Figure 6, atomic properties (local average momentums and forces) are distributed over grid points to form a map of objects. The interval between grid points is d_L . Ω_i contains all grid points directly surrounding atom i .

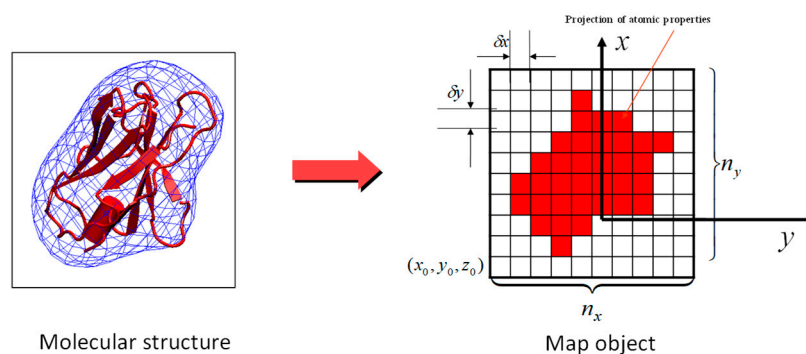


Figure 6. Map representation of local spatial averages. The intervals between grid points are $\delta x = \delta y = \delta z = d_L$. Atom-occupied grids are colored red. Local average momentum and forces of atoms are projected to the grid points directly surrounding them.

Introducing the local spatial averaging only changes how the local average properties are calculated but will not change the equation of the self-guided motion. Therefore, we expect that the simulation can be reweighted using the SG method described previously [13].

3.3. A Leap-Frog Algorithm for SGMD/SGLD with Spatial Averaging

Here, we present a leap-frog algorithm to implement the SGMD/SGLD simulation method with the spatial averaging scheme. This algorithm needs vector arrays to store $\tilde{\mathbf{r}}$, $\tilde{\mathbf{r}}$, and $\tilde{\mathbf{p}}$ and uses these arrays to calculate all other quantities. In addition, an EMAP object is needed for type 4 spatial averaging. The following algorithm has been implemented in current versions of CHARMM [5] (49b1) and Amber [6] (2024). Currently, SGLD implementations support the PMEMD GPU version, but SGLD/spatial average is only implemented in the CPU version. We will implement it in the GPU version only after being widely tested.

(1) At each time step, t , calculate the potential energy, E_p , and forces, \mathbf{f} , which include the random forces in Langevin dynamics.

(2) Calculate the local time average properties.

Local average energies:

$$\tilde{E}_p(t) = \left(1 - \frac{\delta t}{t_L}\right) \tilde{E}_p(t - \delta t) + \frac{\delta t}{t_L} E_p(t) \quad (8)$$

$$\tilde{\tilde{E}}_p(t) = \left(1 - \frac{\delta t}{t_L}\right) \tilde{\tilde{E}}_p(t - \delta t) + \frac{\delta t}{t_L} \tilde{E}_p(t) \quad (9)$$

Local average positions:

$$\tilde{\mathbf{r}}(t) = \left(1 - \frac{\delta t}{t_L}\right) \tilde{\mathbf{r}}(t - \delta t) + \frac{\delta t}{t_L} \mathbf{r}(t) \quad (10)$$

$$\tilde{\tilde{\mathbf{r}}}(t) = \left(1 - \frac{\delta t}{t_L}\right) \tilde{\tilde{\mathbf{r}}}(t - \delta t) + \frac{\delta t}{t_L} \tilde{\mathbf{r}}(t) \quad (11)$$

Other local average properties, $\tilde{\mathbf{p}}$ and $\tilde{\tilde{\mathbf{F}}}$ and $\tilde{\mathbf{F}}$ are calculated from $\tilde{\mathbf{r}}(t)$ and $\tilde{\tilde{\mathbf{r}}}(t)$:

$$\tilde{\mathbf{p}}(t) = \frac{m}{t_L} (\mathbf{r}(t) - \tilde{\mathbf{r}}(t)) \quad (12)$$

$$\mathbf{p}(t) = \tilde{\mathbf{p}}(t - \delta t) + \frac{t_L}{\delta t} (\tilde{\mathbf{p}}(t) - \tilde{\mathbf{p}}(t - \delta t)) \quad (13)$$

$$\tilde{\mathbf{F}}(t) = \frac{1}{t_L} \mathbf{p}(t) - \frac{m}{t_L^2} (\mathbf{r}(t) - \tilde{\mathbf{r}}(t)) \quad (14)$$

$$\tilde{\tilde{\mathbf{F}}}(t) = \frac{m}{t_L^2} (\mathbf{r}(t) - 2\tilde{\mathbf{r}}(t) + \tilde{\tilde{\mathbf{r}}}(t)) \quad (15)$$

Therefore, only the local average positions, $\tilde{\mathbf{r}}(t)$ and $\tilde{\tilde{\mathbf{r}}}(t)$, and $\tilde{\mathbf{p}}(t)$, need to be stored during an SG simulation.

(3) Calculate local spatial averaging properties.

If the spatial averaging type is 1, the spatial averages will be the same as local time averages.

If the spatial averaging type is 2 or 3, the properties of bonded atoms are summed up for averaging according to Equation (7).

If the spatial averaging type is 4, project m , $\tilde{\mathbf{p}}(t)$, and $\Delta\mathbf{F} = \tilde{\tilde{\mathbf{F}}}(t) - \tilde{\mathbf{F}}(t)$ over the grid points around each atom. The grid interval between grid points is set to the spatial averaging distance, d_L .

$$\hat{v}_g = \frac{\sum_{j \in \Omega_i} (1 - |x_{jg}|)(1 - |y_{jg}|)(1 - |z_{jg}|) \tilde{\mathbf{p}}_j}{\sum_{j \in \Omega_i} (1 - |x_{jg}|)(1 - |y_{jg}|)(1 - |z_{jg}|) m_j} \quad (16)$$

$$\hat{a}_g = \frac{\sum_{j \in \Omega_i} (1 - |x_{jg}|)(1 - |y_{jg}|)(1 - |z_{jg}|) \Delta\mathbf{F}_j}{\sum_{j \in \Omega_i} (1 - |x_{jg}|)(1 - |y_{jg}|)(1 - |z_{jg}|) m_j} \quad (17)$$

Here, $x_{jg} = \frac{x_g - x_j}{d_L}$, $y_{jg} = \frac{y_g - y_j}{d_L}$, $z_{jg} = \frac{z_g - z_j}{d_L}$; g represents the index of the grid points directly surrounding the atom.

The spatial averagings for each atom are calculated by interpolating from the grid points around the atom.

$$\hat{\mathbf{p}}_i = m_i \sum_{g \in \Omega_i} (1 - |x_{ig}|)(1 - |y_{ig}|)(1 - |z_{ig}|) \hat{v}_g \quad (18)$$

$$\widehat{\Delta \mathbf{F}}_i = m_i \sum_{g \in \Omega_i} (1 - |x_{ig}|)(1 - |y_{ig}|)(1 - |z_{ig}|) \widehat{\mathbf{a}}_g \quad (19)$$

(4) Calculate apparent friction constants, ζ .

We need ensemble averages of $\widehat{\Delta \mathbf{F}} \cdot \widehat{\mathbf{p}}$ and $\widehat{\mathbf{p}} \cdot \widehat{\mathbf{p}}$ to calculate the apparent friction constant of each atom according to Equation (4). To be able to calculate them on the fly, we replace the ensemble averages by long-time local averages. An average time, t_{avg} , typically 10 times the local average time, t_L , is defined for this purpose. Two scalar arrays, FP and PP, are used to store the averages.

$$FP(t) = \left(1 - \frac{\delta t}{t_{\text{avg}}}\right) FP(t - \delta t) + \frac{\delta t}{t_{\text{avg}}} \widehat{\Delta \mathbf{F}} \cdot \widehat{\mathbf{p}} \quad (20)$$

$$PP(t) = \left(1 - \frac{\delta t}{t_{\text{avg}}}\right) PP(t - \delta t) + \frac{\delta t}{t_{\text{avg}}} \widehat{\mathbf{p}} \cdot \widehat{\mathbf{p}} \quad (21)$$

$$\zeta = -\frac{FP(t)}{PP(t)} \quad (22)$$

The averages and friction constants are calculated for each atom. Alternatively, the apparent friction constants of atoms can be calculated from previous simulations and read in at the beginning of a simulation to avoid their calculation on the fly.

(5) Calculate the guiding forces:

$$\mathbf{g} = \mu \widehat{\Delta \mathbf{F}} + \lambda \zeta \widehat{\mathbf{p}} \quad (23)$$

(6) Calculate the energy conservation scaling factor:

An energy conservation scaling factor, η , is used to cancel energy input due to the guiding force. Previously, a uniform energy conservation factor was used for all atoms. Here, we use atom-specific scaling factors, and their calculations need no information from other atoms, therefore saving inter-processor communication in parallel computing.

$$\eta = \frac{(2 + \gamma \delta t) \mathbf{g} \cdot \mathbf{p}_0}{2 \mathbf{p}_0^2 - \mathbf{g} \cdot \mathbf{p}_0 \delta t} \quad (24)$$

Here,

$$\mathbf{p}_0 = m \mathbf{v} \left(t - \frac{\delta t}{2} \right) + \left(\mathbf{f} + \mathbf{g} - \gamma m \mathbf{v} \left(t - \frac{\delta t}{2} \right) \right) \frac{\delta t}{2} \quad (25)$$

(7) Calculate velocities at $t + \frac{\delta t}{2}$:

$$\mathbf{v} \left(t + \frac{\delta t}{2} \right) = \frac{\left(1 - \frac{(\gamma + \eta) \delta t}{2} \right)}{1 + \frac{(\gamma + \eta) \delta t}{2}} \mathbf{v} \left(t - \frac{\delta t}{2} \right) + \frac{\mathbf{f} + \mathbf{g}}{1 + \frac{(\gamma + \eta) \delta t}{2}} \frac{\delta t}{m} \quad (26)$$

Please note, for SGLD, γ is the friction constant, and \mathbf{f} is the interaction force plus the random force, and for SGMD, $\gamma = 0$, and there is no random force in \mathbf{f} .

(8) Calculate positions at $t + \delta t$:

$$\mathbf{r}(t + \delta t) = \mathbf{r}(t) + \mathbf{v} \left(t + \frac{\delta t}{2} \right) \delta t \quad (27)$$

(9) Go back to step (1) and repeat the above steps for the next time step.

4. Concluding Remarks

The self-guided molecular simulation method was motivated to overcome randomness in conformational search and to accelerate the process to reach global minimum states.

A local averaging scheme is the main characteristic for extracting self-guiding information during a simulation. Previously, the local time averaging scheme has focused on enhancing conformational search and sampling by promoting low-frequency motion. This work explores the spatial averaging scheme with the goal of enhancing concerted motion in simulation systems to facilitate the formation of ordered structures.

This work demonstrates that spatial averaging promotes the β -sheet formation of amyloid fibril peptides, as compared with regular Langevin dynamics and local time-average-based self-guided Langevin dynamics. An extensive examination of the behavior of spatial averaging will be conducted with more systems to fully understand its effects and optimal parameters.

With this new development, we expect SGMD/SGLD will have more options to study problems such as ligand docking and protein folding, where studying ordered structures is the goal.

Author Contributions: Conceptualization, X.W. and B.R.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Wu, X.; Damjanovic, A.; Brooks, B.R. Efficient and Unbiased Sampling of Biomolecular Systems in the Canonical Ensemble: A Review of Self-Guided Langevin Dynamics. *Adv. Chem. Phys.* **2012**, *150*, 255–326. [[PubMed](#)]
2. Christen, M.; Van Gunsteren, W.F. On searching in, sampling of, and dynamically moving through conformational space of biomolecular systems: A review. *J. Comput. Chem.* **2008**, *29*, 157–166. [[CrossRef](#)] [[PubMed](#)]
3. Hamelberg, D.; Mongan, J.; McCammon, J.A. Accelerated molecular dynamics: A promising and efficient simulation method for biomolecules. *J. Chem. Phys.* **2004**, *120*, 11919–11929. [[CrossRef](#)] [[PubMed](#)]
4. Gao, N.; Yang, L.; Gao, F.; Kurtz, R.J.; West, D.; Zhang, S. Long-time atomistic dynamics through a new self-adaptive accelerated molecular dynamics method. *J. Phys. Condens. Matter* **2017**, *29*, 145201. [[CrossRef](#)]
5. Brooks, B.R.; Brooks, C.L., 3rd; Mackerell, A.D., Jr.; Nilsson, L.; Petrella, R.J.; Roux, B.; Won, Y.; Archontis, G.; Bartels, C.; Boresch, S.; et al. CHARMM: The biomolecular simulation program. *J. Comput. Chem.* **2009**, *30*, 1545–1614. [[CrossRef](#)] [[PubMed](#)]
6. Case, D.A.; Betz, R.M.; Botello-Smith, W.; Cerutti, D.S.; Cheatham, T.E., III; Darden, T.A.; Duke, R.E.; Giese, T.J.; Gohlke, H.; Goetz, A.W.; et al. AMBER 2024; University of California: San Francisco, CA, USA, 2024.
7. Saelices, L.; Johnson, L.M.; Liang, W.Y.; Sawaya, M.R.; Cascio, D.; Ruchala, P.; Whitelegge, J.; Jiang, L.; Riek, R.; Eisenberg, D.S. Uncovering the Mechanism of Aggregation of Human Transthyretin*. *J. Biol. Chem.* **2015**, *290*, 28932–28943. [[CrossRef](#)]
8. Wu, X.; Subramaniam, S.; Case, D.A.; Wu, K.W.; Brooks, B.R. Targeted conformational search with map-restrained self-guided Langevin dynamics: Application to flexible fitting into electron microscopic density maps. *J. Struct. Biol.* **2013**, *183*, 429–440. [[CrossRef](#)] [[PubMed](#)]
9. Tian, C.; Kasavajhala, K.; Belfon, K.A.A.; Raguet, L.; Huang, H.; Miguez, A.N.; Bickel, J.; Wang, Y.; Pincay, J.; Wu, Q.; et al. ff19SB: Amino-Acid-Specific Protein Backbone Parameters Trained against Quantum Mechanics Energy Surfaces in Solution. *J. Chem. Theory Comput.* **2020**, *16*, 528–552. [[CrossRef](#)] [[PubMed](#)]
10. Tsui, V.; Case, D.A. Theory and applications of the generalized Born solvation model in macromolecular simulations. *Biopolymers* **2000**, *56*, 275–291. [[CrossRef](#)] [[PubMed](#)]
11. Schaefer, M.; Froemmel, C. A precise analytical method for calculating the electrostatic energy of macromolecules in aqueous solution. *J. Mol. Biol.* **1990**, *216*, 1045–1066. [[CrossRef](#)]
12. Kabsch, W.; Sander, C. Dictionary of protein secondary structure: Pattern recognition of hydrogen-bonded and geometrical features. *Biopolymers* **1983**, *22*, 2577–2637. [[CrossRef](#)] [[PubMed](#)]
13. Wu, X.; Brooks, B.R. Reformulation of the self-guided molecular simulation method. *J. Chem. Phys.* **2020**, *153*, 094112. [[CrossRef](#)] [[PubMed](#)]

14. Ryckaert, J.P.; Ciccotti, G.; Berendsen, H.J.C. Numerical-Integration of Cartesian Equations of Motion of a System with Constraints—Molecular-Dynamics of N-Alkanes. *J. Comput. Phys.* **1977**, *23*, 327–341. [[CrossRef](#)]
15. Wright, J.D.; Sargsyan, K.; Wu, X.; Brooks, B.R.; Lim, C. Protein–Protein Docking Using EMAP in CHARMM and Support Vector Machine: Application to Ab/Ag Complexes. *J. Chem. Theory Comput.* **2013**, *9*, 4186–4194. [[CrossRef](#)] [[PubMed](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.