JCTC Journal of Chemical Theory and Computation

Open Access

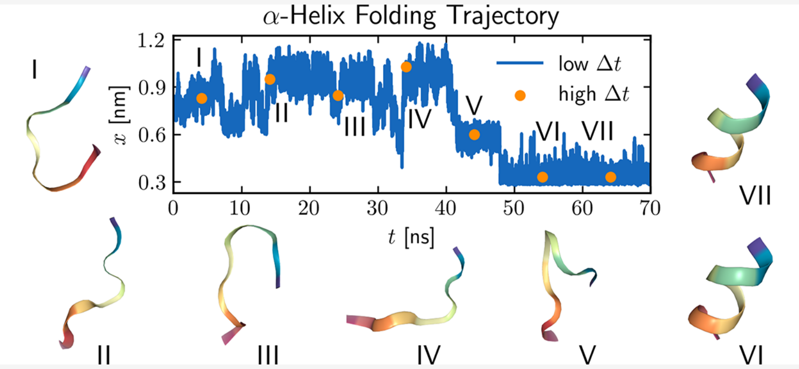
Accurate Memory Kernel Extraction from Discretized Time-Series Data

Lucas Tepper, Benjamin Dalton, and Roland R. Netz*

Cite This: J. Chem. Theory Comput. 2024, 20, 3061−3068

Read Online

ACCESS | Metrics & More | Article Recommendations | Supporting Information



ABSTRACT: Memory effects emerge as a fundamental consequence of dimensionality reduction when low-dimensional observables are used to describe the dynamics of complex many-body systems. In the context of molecular dynamics (MD) data analysis, accounting for memory effects using the framework of the generalized Langevin equation (GLE) has proven efficient, accurate, and insightful, particularly when working with high-resolution time series data. However, in experimental systems, high-resolution data are often unavailable, raising questions about the impact of the data resolution on the estimated GLE parameters. This study demonstrates that direct memory extraction from time series data remains accurate when the discretization time is below the memory time. To obtain memory functions reliably, even when the discretization time exceeds the memory time, we introduce a Gaussian Process Optimization (GPO) scheme. This scheme minimizes the deviation of discretized two-point correlation functions between time series data and GLE simulations and is able to estimate accurate memory kernels as long as the discretization time stays below the longest time scale in the data, typically the barrier crossing time.

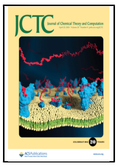
INTRODUCTION

A fundamental challenge in natural sciences involves the creation of a simplified, yet accurate, representation of complex system dynamics using a low-dimensional coordinate. For instance, in spectroscopy, atomic motions are investigated solely through the polarization induced by an electromagnetic field, resulting in spectra.[1] In the case of molecules in fluids, the myriad of interactions with the solvent is often reduced to a one-dimensional diffusion process.[2,3] In numerous studies,[4−8] the folding of a protein is described by a one-dimensional reaction coordinate. These diverse fields all share the common approach of projecting the complete many-body dynamics of 6N atomic positions and momenta onto a few or even a single reaction coordinate. Starting from the deterministic kinetics of a Hamiltonian system, the projection procedure yields a stochastic description based on the generalized Langevin equation (GLE),[9−11] which, in the case of a one-dimensional coordinate $x(t)$ and its corresponding velocity $v(t)$, reads

$$m\frac{\mathrm{d}}{\mathrm{d}t}v(t) = -\frac{\mathrm{d}U[x(t)]}{\mathrm{d}x(t)} - \int_0^t \mathrm{d}s\Gamma(t-s)v(s) + F_R(t) \tag{1}$$

where m is the effective mass of the coordinate x. The potential of mean force $U(x)$ is directly available from the equilibrium probability distribution $\rho(x)$ via $U(x) = -k_B T \ln \rho(x)$, where k_B is the Boltzmann constant, and T is the absolute temperature. Non-Markovian effects arise as a direct consequence of the dimensionality reduction.[12] In the GLE, the memory kernel $\Gamma(t)$ weights the effect of past velocities on the current acceleration. Stochastic effects, represented by the random force $F_R(t)$, are linked to the memory function via the fluctuation−dissipation theorem in equilibrium, $\langle F_R(0)F_R(t)\rangle$

Received: November 23, 2023
Revised: March 25, 2024
Accepted: April 1, 2024
Published: April 11, 2024



ACS Publications

https://doi.org/10.1021/acs.jctc.3c01289

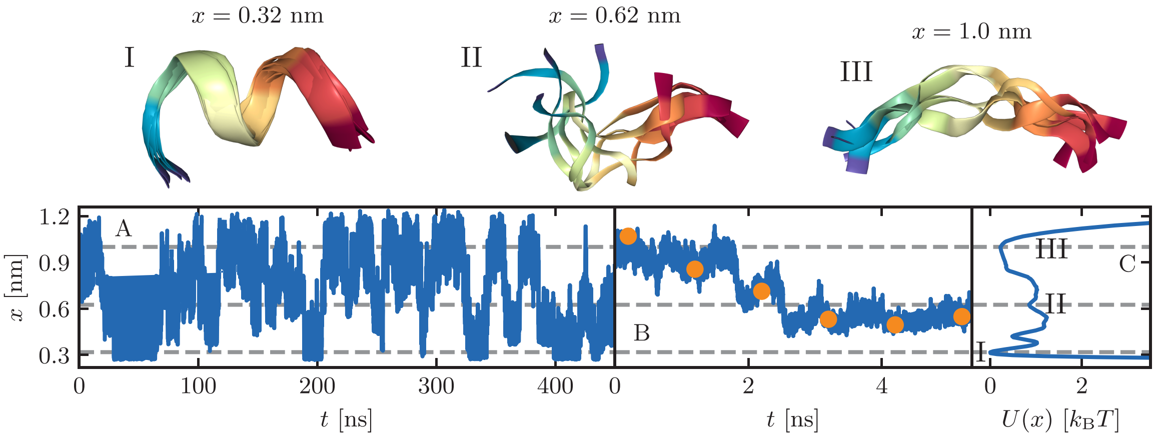


Figure 1. **I-III** Representative snapshots for different values of the mean hydrogen-bond distance reaction coordinate of Ala_9, x, defined in eq 2. **A** Multiple folding and unfolding events occur within a 450 ns trajectory segment. **B** A single folding event. The orange circles indicate the time series discretized at $\Delta t = 1$ ns. **C** The potential landscape $U(x)$ for Ala_9, computed from the trajectory at full resolution. The folded state (**I**) forms a sharp minimum at $x = 0.32$ nm. A local minimum is found at $x = 0.62$ nm (**II**). The unfolded state forms a broad minimum around $x = 1.0$ nm (**III**).

$= k_BT\Gamma(t)$. When the relaxation of the environment governing $\Gamma(t)$ is sufficiently fast, $\Gamma(t)$ approaches a delta kernel, and the Langevin equation emerges from the GLE. Considerable efforts have been dedicated to identifying suitable reaction coordinates to minimize memory effects and enable a Markovian description of protein folding.[4−6,8,12]

In recent works, the memory function $\Gamma(t)$ was extracted from time series data of proteins of biological relevance, allowing the non-Markovian description of a protein's folding kinetics in a nonlinear folding landscape. Memory effects were found to be highly relevant, both in model systems[13] and real proteins.[14,15] Multiple methods exist to extract memory functions from MD data. A much-used method is based on Volterra equations, which are deterministic, integro-differential equations that derive from the GLE and allow for the extraction of the memory kernel from time correlation functions.[16−19] While Volterra equations offer good accuracy when high-quality time-series data are available, it is unclear if they remain efficient when the observations of the system are sampled with long discretization times. A recent research endeavor used an iterative scheme to approximate the memory kernel by adapting a trial kernel with a heuristic update based on the velocity autocorrelation function.[20] Another work parametrized memory kernels by fitting correlation functions to an analytical solution of the GLE.[21] In order to include the short and long time scales of the system dynamics, the fit included both the two-point correlation and its running integral. Both methods share the limitation of not being applicable to a nonlinear potential energy function $U(x)$. A recent paper not suffering from such a limitation used a maximum-likelihood model to estimate the GLE parameters that best fit the given MD data.[22] In a different work on polymer solutions, star polymers were coarse-grained to single beads interacting via a nonlinear $U(x)$. A GLE system was set up to mimic the star polymers' kinetics. The simulation parameters of the GLE system were iteratively changed using Gaussian Process Optimization (GPO) such that the coarse-grained and MD velocity autocorrelations were most similar.[23] The same idea was used to estimate a joint memory kernel over multiple temperatures.[24]

Here, we consider the effects of temporal discretization, motivated by the fact that data are always discretized. For MD simulations, archived data often only contains the atomic positions at time intervals of hundreds of picoseconds to nanoseconds, as in the case of the data from the Anton supercomputer.[25] When considering experimental data, measurement devices limit the time step of the observations, typically at the microsecond scale.[26,27] In a prior publication, discretization effects were examined within the framework of data-driven GLE analysis. The GLE, without a potential, was solved analytically. To deal with discretization effects, the discretized mean-squared displacement and velocity autocorrelation functions were computed, allowing for the direct fitting of the memory kernel.[28] The present work investigates how a GLE with a nonharmonic potential can be parametrized given discretized data by considering a highly nonlinear molecular dynamics test system. The Volterra-based approach is shown to be remarkably resilient to time discretizations. Where the Volterra approach ceases to function, we demonstrate that Gaussian Process Optimization is a suitable method to obtain memory kernels from discrete time series data. In matching correlation functions computed from subsampled data, we present a method to deal with the discretization effects and extend the GLE analysis to nonlinear data at higher discretizations. The choice of correlation functions involves some flexibility, demonstrating the broad applicability of our approach. For a small alanine homopeptide used as a test system, the Volterra method is suitable for discretization times that reach the memory time of about 1 ns. In comparison, the GPO method extends the range to discretization times up to the folding time of 58 ns.

■ RESULTS AND DISCUSSION

We investigate the effect of data discretization starting from a 10-μs-long MD trajectory of alanine nonapeptide (Ala_9) in water, which was established as a sensitive test system for non-Markovian effects in our previous work.[14] As in our original analysis, the formation of the α-helix in Ala_9 is measured by the mean distance between the H-bond acceptor oxygen of residue n and the donor nitrogen of residue $n + 4$

$$x(t) = \frac{1}{3}\sum_{n=2}^{4} \|\vec{r}_n^{\,O}(t) - \vec{r}_{n+4}^{\,N}(t)\| \tag{2}$$

In the α-helical state, x has a value of approximately 0.3 nm, the mean H-bond length between nitrogen and oxygen. The potential of mean force $U(x)$ in Figure 1C displays several metastable states along the folding landscape; Ala$_9$, therefore, is a suitable and nontrivial test system for numerical methods. Figure 1A shows a 450 ns long trajectory. To test how time discretization affects memory extraction, frames of the trajectory are left out to achieve an effective discretization time step Δt. Such a discretized trajectory (orange data points for $\Delta t = 1$ ns) is compared in Figure 1B to the time series at full resolution. The potential $U(x)$ is always estimated from a histogram of the entire data set to separate time discretization from effects arising due to the undersampling of the potential (see section I in the Supporting Information).

Volterra Equations. To extract memory kernels from time-series data, the GLE in eq 1 is multiplied by $v(0)$ and averaged over time. By using the relation $\langle F_R(t)v(0)\rangle = 0$,[9,10] one obtains the Volterra equation[14,19]

$$m\frac{\mathrm{d}}{\mathrm{d}t}C^{vv}(t) = -C^{\nabla Uv}(t) - \int_0^t \mathrm{d}s\Gamma(t-s)C^{vv}(s) \tag{3}$$

where $C^{vv}(t)$ is the velocity autocorrelation function, and $C^{\nabla Uv}(t)$ is the correlation between the gradient of the potential and the velocity. By integrating eq 3 from 0 to t, we derive a Volterra equation involving the running integral over the kernel $G(t) = \int_0^t \mathrm{d}s\Gamma(s)$ and insert $mC^{vv}(0) = C^{\nabla Ux}(0)$[14] to obtain

$$\frac{C^{\nabla Ux}(0)}{C^{vv}(0)}C^{vv}(t) = C^{\nabla Ux}(t) - \int_0^t \mathrm{d}sG(t-s)C^{vv}(s) \tag{4}$$

with $C^{\nabla Ux}(t)$ being the correlation between the gradient of the potential and the position. Computing the memory kernel directly from eq 3 is possible[29,30] but prone to instabilities.[17] Extracting $G(t)$ using eq 4 and computing $\Gamma(t)$ via a numerical derivative improves the numerical stability.[17,31] The discretization and solution of eq 4 are discussed in section II of the Supporting Information. A recent study proposed an alternative technique for extracting memory kernels by Taylor expansion of the convolution integral[32] (we discuss the potential applicability of this Ansatz to our specific problem in section III in the Supporting Information). We fit $\Gamma(t)$ extracted from the full-resolution data at $\Delta t = 1$ fs using least-squares to a multiexponential of the form

$$\Gamma_{\mathrm{fit}}(t) = \sum_{i=1}^{5}\frac{\gamma_i}{\tau_i}\mathrm{e}^{-t/\tau_i} \tag{5}$$

The fitted memory times τ_i and friction coefficients γ_i are presented in Table 1. The fitting involves both $\Gamma(t)$ and $G(t)$, as elaborated in the Methods section, and accurately captures the MD kinetics, similar to our previous work.[14]

In order to estimate the impact of the non-Markovian effects on the kinetics, we turn to a heuristic formula for the mean first-passage time τ_{MFP} of a particle in a double-well potential in the presence of exponentially decaying memory.[13,33,34] Validated by extensive simulations, the heuristic formula accurately described the non-Markovian effects occurring in the folding of various proteins.[15] For a single exponential memory function, the heuristic formula identifies three different regimes by comparing the single memory time τ to the diffusion time scale $\tau_D = \gamma_{\mathrm{tot}}L^2/k_BT$, which is the time it takes for a free Brownian particle to diffuse over a length of L in reaction coordinate space. The first regime is the Markovian limit, where $\tau \ll \tau_D$ and non-Markovian effects are negligible. The second regime is a non-Markovian regime where $\tau_D/100 \lesssim \tau \lesssim 10\tau_D$, in which a speed-up of τ_{MFP} compared to the Markovian description is observed. The third regime occurs when $\tau \gtrsim 10\tau_D$, where τ_{MFP} is slowed down compared to the Markovian description due to non-Markovian memory effects.

Table 1. Fitted Memory Function Parameters for Δt = 1 fs According to Eq 5[a]

i	γ_i [u/ps]	τ_i [ps]
1	$2.2 \cdot 10^3$	0.007
2	$4.4 \cdot 10^4$	18
3	$2.4 \cdot 10^5$	370
4	$6.0 \cdot 10^4$	4100
5	$4.6 \cdot 10^3$	5700
$\gamma_{\mathrm{tot}} = \sum_{i=1}^{5}\gamma_i$	$3.5 \cdot 10^5$	
$\tau_{\mathrm{mem}} = \frac{\int_0^\infty \mathrm{d}s\, s\Gamma(s)}{\int_0^\infty \mathrm{d}s\, \Gamma(s)}$		1000

[a] The fits for Δt > 1 fs are shown in section IV in the Supporting Information.

To compute τ_D, we take $L = 0.22$ nm, the distance between the folded state at $x = 0.32$ nm and the barrier at $x = 0.54$ nm, the total friction $\gamma_{\mathrm{tot}} = \sum_{i=1}^{5}\gamma_i$ and obtain $\tau_D = 6.8$ ns. The τ_i values in Table 1 span times from $\tau_1 = 7$ fs $\ll \tau_D$ up to $\tau_5 = 5.7$ ns $\approx \tau_D$. In a previous work,[15] $\tau_{\mathrm{mem}} = \int_0^\infty \mathrm{d}s\, s\Gamma(s)/\int_0^\infty \mathrm{d}s\, \Gamma(s)$, the first moment of the memory kernel, was proposed as the characteristic time scale for a multiscale memory kernel. For the memory kernel in Table 1, we find $\tau_{\mathrm{mem}} = 1$ ns, correctly predicting the non-Markovian speed-up of τ_{MFP} that a previous study demonstrated for Ala$_9$.[14] In this work, we will establish τ_{mem} as the limit for the discretization time Δt, beyond which the Volterra method ceases to produce accurate results.

In the following, the full-resolution kernel obtained for a time step of $\Delta t = 1$ fs will serve as a reference for results using a higher Δt. Comparing the extracted $G(t)$ with the corresponding fit according to eq 5 (red line) in Figure 2F shows no significant differences in the long time limit. Figure 2C shows oscillations of the extracted $\Gamma(t)$ for $t < 1$ ps, which are discarded by the exponential fit. As we will show later, they do not play a role in the kinetics. For both $\Gamma(t)$ in Figure 2B and $C^{vv}(t)$ in Figure 2A, the oscillations disappear for $\Delta t \geq 0.1$ ps, indicating that they are caused by subpicosecond molecular vibrations. Moreover, the value of $\Gamma(t)$ for $t < 1$ ps is consistently attenuated as Δt increases, mirroring the same trend observed in $C^{vv}(0)$, as illustrated in the inset of Figure 2A. In contrast, $\Gamma(t)$ for $t > 1$ ps in the inset of Figure 2B shows an exponential decay that is well preserved for all $\Delta t < 1$ ns. The running integral $G(t)$ in Figure 2D stays mostly unchanged for discretizations smaller than $\Delta t < 1$ ns. This demonstrates that the Volterra extraction scheme is accurate for discretization times below the mean memory time, i.e. for $\Delta t < \tau_{\mathrm{mem}} = 1$ ns.

The multiexponential kernel in eq 5 allows for the efficient numerical simulation of the GLE by setting up a Langevin equation where the reaction coordinate x is coupled harmonically to one overdamped, auxiliary variable per exponential component[10,35] (see section V in the Supporting Information). Utilizing this simulation technique, Figure 2G compares profiles for the mean first-passage times τ_{MFP} originating from both the folded and unfolded states. For $\Delta t \leq 10$ ps, the τ_{MFP} values obtained from the GLE simulations (colored lines) closely align with those derived from MD simulations (black broken lines), thereby manifesting the precise correspondence between the non-Markovian GLE

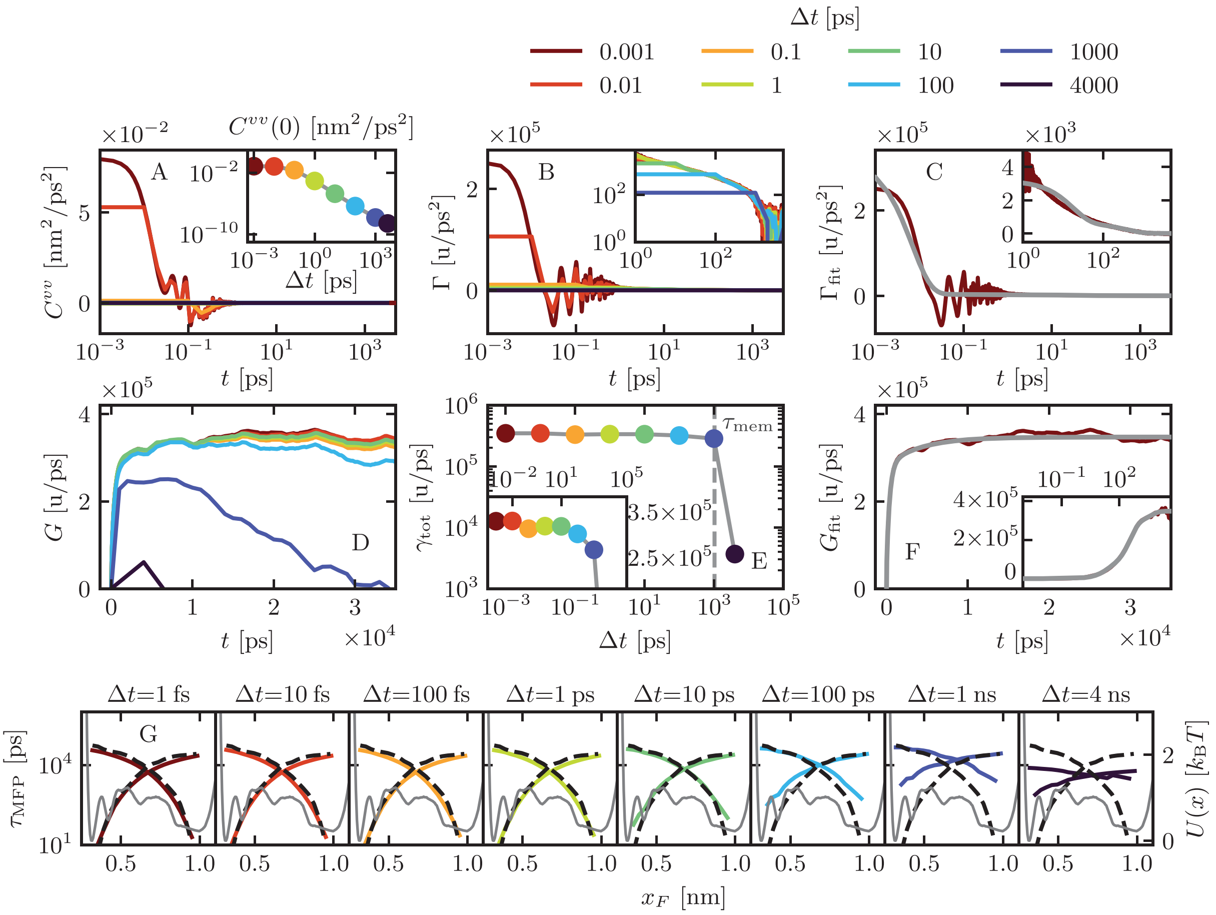


Figure 2. Memory extraction by the inversion of the Volterra eq 4 for different discretization times Δt, using data from MD simulations of Ala_9. **A** Velocity autocorrelation $C^{vv}(t)$. **B** Memory kernel $\Gamma(t)$, from numerical differentiation of $G(t)$. **C** Multiexponential fit of $\Gamma(t)$ computed for $\Delta t = 1$ fs (gray) compared to the corresponding numerical data (dark red). The fitted parameters are shown in Tables 1 and S1. **D** Running integral over the memory kernel $G(t)$. **E** Total friction γ_{tot}, computed from the exponential fits of the kernels. The vertical broken gray line indicates $\tau_{\text{mem}} = \int_0^\infty \text{d}s\, s\Gamma(s)/\int_0^\infty \text{d}s\, \Gamma(s) = 1$ ns. **F** Fit of $G(t)$ (gray) computed at $\Delta t = 1$ fs compared to corresponding numerical data (dark red). **G** Comparison of the mean first-passage times τ_{MFP} computed from the MD data (black broken lines) to τ_{MFP} obtained from GLE simulations using kernels extracted at different Δts (colored lines).

description and the kinetics observed in the MD simulation. In Figure 2E, we present the asymptotic limit $\lim_{t\to\infty} G(t)$, representing the total friction coefficient γ_{tot} of the system, estimated by summing the individual γ_i values obtained from the exponential fits. When $\Delta t \geq 1$ ns, we find that $G(t)$ does not show a plateau value in the long-time limit. Consequently, this leads to a notable discrepancy between the τ_{MFP} profiles presented in Figure 2G and their MD counterparts for $\Delta t \geq 1$ ns. Combining the information provided in Figure 2D, E, and G, it becomes evident that the extracted profile of $G(t)$, the total friction γ_{tot}, and folding times τ_{MFP} all deviate significantly from the MD reference data when the discretization time approaches the memory time τ_{mem}. As a result, we assert that Volterra extraction becomes inadequate when the discretization time exceeds the memory time τ_{mem}. In Section VI of the Supporting Information, we demonstrate that the failure of the Volterra extraction scheme for large Δt is mostly due to discretization effects in the potential gradient-position correlation function $C^{\nabla U x}(t)$.

Gaussian Process Optimization. So far, we have demonstrated that the Volterra equation can be used to extract a consistent memory kernel for a wide range of discretization times up to $\Delta t \approx \tau_{\text{mem}}$. The resulting GLE faithfully captures the underlying kinetics when judged by τ_{MFP} for discretizations below the memory time scale τ_{mem} but fails when exceeding it. Given that the discretization time may exceed the dominant memory time scale in typical experimental settings, an improved method is clearly desirable. In the following, we describe a scheme that is not based on the Volterra equation and allows the extraction of $\Gamma(t)$ for Δt that significantly exceeds τ_{mem}. For this, we use a matching scheme between the discretized time correlation functions of the MD reference system $C^{\text{MD}}(n\Delta t)$ and of the GLE $C^{\text{GLE}}(n\Delta t, \theta)$ via the mean-squared loss

$$\mathcal{L} = \frac{1}{N}\sum_{n=1}^{N} \left(C^{\text{MD}}(n\Delta t) - C^{\text{GLE}}(n\Delta t, \theta)\right)^2 \tag{6}$$

The type of correlation function will be specified later. The loss is evaluated over N samples, where N is determined based on the decay time of the correlation (see Table S3). In an iterative optimization, the friction and memory time parameters in eq 5 that serve as the GLE parameters $\theta = (\gamma_1, \tau_1, ..., \gamma_5, \tau_5)$ are updated, and the GLE is integrated using a simulation time step δt chosen small enough that discretization effects in the GLE simulations are negligible. For the sake of comparability, we maintain a constant mass value of $m = 31.4$ u, derived using the equipartition theorem according to $m = k_BT/\langle v^2\rangle$, from the

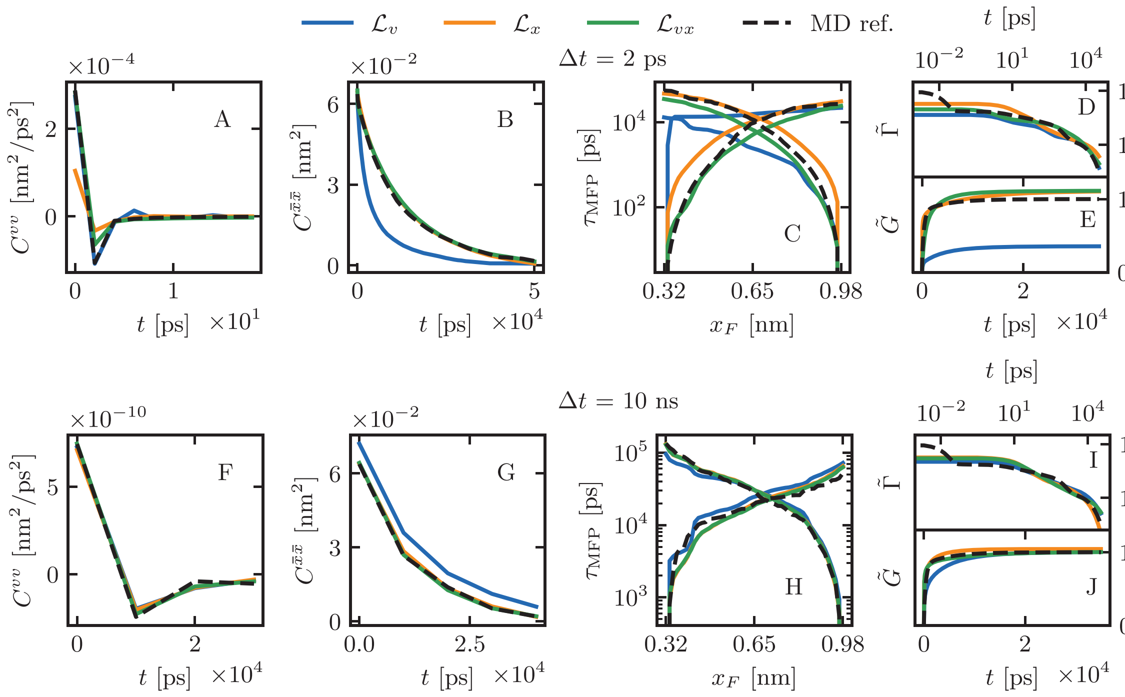


Figure 3. To visualize the Gaussian Process Optimization (GPO), we plot the mean of observables over the 10 best optimization runs. We compare GPO results using the loss $\mathcal{L}_v$ (blue), based on $C^{vv}(t)$, $\mathcal{L}_x$ (orange), based on the autocorrelation of the position $\bar{x}(t) = x(t) - \langle x \rangle$ and $\mathcal{L}_{vx}$, a linear combination of $\mathcal{L}_v$ and $\mathcal{L}_x$ (green). For $\Delta t = 2$ ps, we compare the observables **A** $C^{vv}(t)$, **B** $C^{\overline{xx}}(t)$, **C** τ_{MFP}, **D** $\tilde{\Gamma}(t) = \Gamma(t)/\Gamma(0)$, and **E** $\tilde{G}(t) = G(t)/G(0)$ to the MD reference (black broken line). Equally, for $\Delta t = 10$ ns, we show **F** $C^{vv}(t)$, **G** $C^{\overline{xx}}(t)$, **H** τ_{MFP}, **I** $\tilde{\Gamma}(t)$, and **J** $\tilde{G}(t)$. The kernels in **D**, **E**, **I**, and **J**, parametrized by eq 5, are plotted as time-continuous functions.

MD data. In fact, the precise value of m has no significant influence on the method's outcome since it can be accommodated within the kernel. Furthermore, the system's inertial time $\tau_{\mathrm{m}} = m/\gamma_{\mathrm{tot}} = 0.09$ fs is markedly shorter than all other relevant time scales, leading to an overdamped system in which the mass value is irrelevant. To find the best parameter set, θ, the choice of the optimizer is crucial. The loss $\mathcal{L}$, defined in eq 6, is inherently noisy due to the stochastic integration of the GLE and possesses, in general, many local minima in a high-dimensional space. Faced with such a task, common gradient-based or simplex methods fail.[36,37] Genetic algorithms present a powerful alternative but require many sample evaluations.[38−40] Given the computational cost of a converged GLE simulation, we choose Gaussian Process Optimization (GPO)[41−43] as a method to minimize $\mathcal{L}$. GPO builds a surrogate model of the real loss $\mathcal{L}$ that incorporates noise[44−46] and allows for nonlocal search[47,48] (see section VII in the Supporting Information). As an active learning technique, it guides the sampling of new parameters, improving optimization efficiency.[49−51]

In principle, any correlation function can serve as an optimization target. Figure 2A shows that the velocity autocorrelation function $C^{vv}(t)$ decays to zero after about 1 ps, while Figure 3B,G shows that $C^{\overline{xx}}(t)$, the autocorrelation of the position $\bar{x}(t) = x(t) - \langle x \rangle$, decays much more slowly over about 50 ns. With such a difference in the decay times of the two correlations, we define two losses based on eq 6, $\mathcal{L}_v$, using $C^{vv}(t)$, and $\mathcal{L}_x$, using $C^{\overline{xx}}(t)$, anticipating that the two correlations probe different scales of the dynamics. Furthermore, we define $\mathcal{L}_{vx} = \alpha\mathcal{L}_v + \mathcal{L}_x$, a linear combination of $\mathcal{L}_v$ and $\mathcal{L}_x$, to test if including both correlations in the loss function improves the quality of the GLE parameters. The parameter α is selected for each Δt to achieve a balanced weighting between the two losses and is tabulated in Table S3 in the Supporting Information. For every GP optimization, 300 different θ values are evaluated via 18-μs-long GLE simulations each. The 10 θ samples with the lowest loss form the basis for the following analysis. When optimizing the loss function with a discretization of $\Delta t = 2$ ps, Figure 3A illustrates that $\mathcal{L}_v$ (blue) accurately replicates the MD reference for $C^{vv}(t)$, whereas $\mathcal{L}_x$ (orange) exhibits discrepancies. Conversely, in Figure 3B, $\mathcal{L}_x$ perfectly reproduces $C^{\overline{xx}}(t)$, while $\mathcal{L}_v$ struggles to do so. Remarkably, the combined loss function $\mathcal{L}_{vx}$ (green) successfully aligns with both reference correlations simultaneously. To evaluate the quality of the GLE parameters, Figure 3C provides a comparison of the mean first-passage times τ_{MFP} between GLE results from the GPO solutions and the MD reference. We calculate τ_{MFP} for GPO-based GLE simulations and the MD reference using identical discretizations Δt. Notably, we observe that $\mathcal{L}_v$ fails to align with the MD reference, whereas both $\mathcal{L}_x$ and $\mathcal{L}_{vx}$ exhibit consistency with it. This outcome underscores the insufficiency of $C^{vv}(t)$ in capturing the slow kinetics of barrier crossing. A comparison of τ_{MFP} between $\mathcal{L}_{vx}$ and $\mathcal{L}_x$ reveals a slightly better correspondence to the MD reference for $\mathcal{L}_{vx}$, signifying that the inclusion of $C^{vv}(t)$ improves the optimization. Examining the obtained memory kernels in Figure 3D-E, all loss functions yield kernels that largely conform to the exponential fit of the MD reference but exclude the first memory component with a decay time of approximately $\tau_1 \approx 7$ fs. Both $\mathcal{L}_x$ and $\mathcal{L}_{vx}$ correctly identify the plateau of $G(t)$, while $\mathcal{L}_v$ underestimates it, which we identify as the origin for the failure to correctly predict τ_{MFP}. Next, we evaluated the performance of the GPO

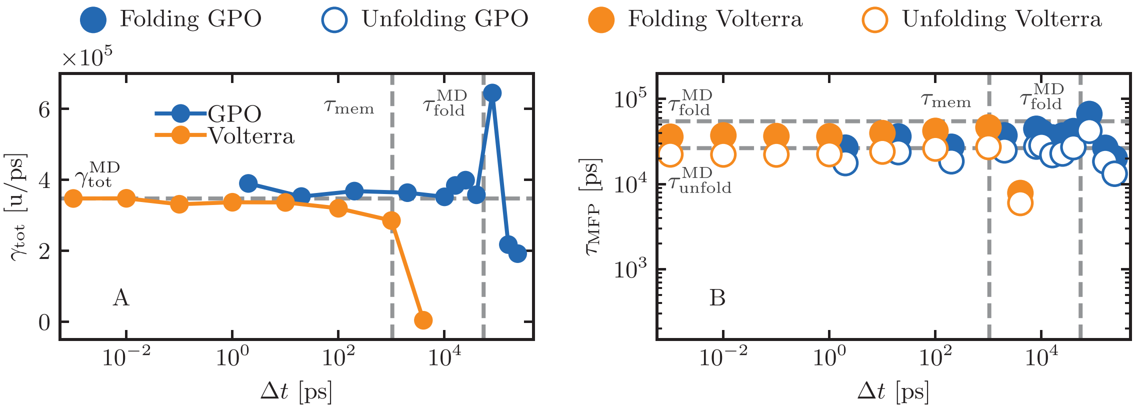


Figure 4. A The total friction $\gamma_{\rm tot} = \sum_{i=1}^{5}\gamma_i$ obtained via the Volterra scheme (orange) is constant for discretizations of $\Delta t < 1$ ns. For Δt higher than the memory time $\tau_{\rm mem} = 1$ ns, it decreases until the extraction fails. Gaussian process optimization (GPO, blue) estimates the correct friction for much higher Δt. The horizontal gray line shows $\gamma_{\rm tot}^{\rm MD}$, the total friction extracted directly from the MD data. **B** The folding and unfolding mean first-passage times from GLE simulations with kernels extracted at different discretizations, given by the mean time it takes the system to first reach from $x = 0.32$ nm to $x = 0.98$ nm (unfolding) and reverse (folding). The MD folding times, $\tau_{\rm fold}^{\rm MD} = 58$ ns and $\tau_{\rm unfold}^{\rm MD} = 26$ ns, are indicated as horizontal gray lines $\tau_{\rm mem} = 1$ ns and $\tau_{\rm fold}^{\rm MD}$ as vertical gray lines. The GPO estimates the correct folding and unfolding times up to $\Delta t \approx \tau_{\rm fold}^{\rm MD}$, significantly higher than the Volterra scheme.

for discretization times exceeding $\tau_{\rm mem}$. In Figure 3F-J, we show the results for $\Delta t = 10$ ns, demonstrating that the GPO approach yields similar results for all differently defined loss functions. The discretized $C^{vv}(t)$, $C^{\overline{xx}}(t)$, and $\tau_{\rm MFP}$ are in perfect agreement with the MD reference. The kernels agree for all but the lowest times. To confirm that the increased discretization used for the $\tau_{\rm MFP}$ computation does not introduce any bias into the results, we perform an additional comparison of $\tau_{\rm MFP}$ computed at the full-time resolution of $\Delta t = 2$ fs (see Figure S4 in the Supporting Information). Figure 4 provides a comparison of the performance of the Volterra and GPO approaches across various discretizations. This comparison focuses on the overall friction, folding, and unfolding mean first-passage times, as these observables are not included in the GPO optimization process. As shown in the previous section, the applicability of the Volterra method is limited to discretizations below memory time $\tau_{\rm mem} = 1$ ns. Extraordinarily, the GPO approach can surpass the boundary set by the memory time and estimates folding times with good accuracy for discretizations up to $\Delta t = 40$ ns. This limit roughly corresponds to the mean time it takes the system to fold, $\tau_{\rm fold}^{\rm MD} = 58$ ns, which is given by the mean first-passage time from the unfolded state at $x = 0.98$ nm to the folded state at $x = 0.32$ nm. For the highest discretization time tested, $\Delta t = 240$ ns, the GP optimization still finds meaningful folding times, while underestimating the total friction.

■ CONCLUSIONS

We investigate the effect that time discretization of the input data has on memory extraction. As a specific example, we consider MD time-series data of the polypeptide Ala$_9$. Computing a memory kernel via the inversion of the Volterra eq 4 requires the velocity autocorrelation and potential gradient-position correlation function. These autocorrelations change significantly as a result of increasing time discretization, and with it a surrogate kernel is obtained that differs from the full-resolution kernel. Our key finding is that given a discretization time lower than the characteristic memory time, the Volterra approach can compute a kernel that reproduces the kinetics of the MD system. Here, we define the characteristic memory time $\tau_{\rm mem}$ via the first moment of the memory kernel, taking into account all decay times of the kernel, and find $\tau_{\rm mem} = 1$ ns for Ala$_9$. By extracting the memory kernel from MD trajectories at different discretizations, we show that the Volterra approach is able to reproduce the kinetics when the discretization time Δt is below $\tau_{\rm mem}$.

To also cover the important regime when $\Delta t > \tau_{\rm mem}$, we introduce a Gaussian Process Optimization (GPO) scheme based on matching discretized time correlation functions of the reference and the GLE system. We test losses based on the velocity and position autocorrelation functions, for which GPO yields memory kernels very similar to the Volterra scheme and is able to reproduce the reaction-coordinate dynamics and the folding times.

We demonstrate the effectiveness of GPO for discretization times up to the folding time of $\tau_{\rm fold}^{\rm MD} = 58$ ns, about 50 times higher than the highest discretization for which the Volterra approach is applicable. As elaborated in previous works,[13−15] memory can affect the kinetics of protein barrier crossing on time scales far exceeding the memory time, up to the longest time scale of the system. Therefore, the presented GPO approach is expected to extend the applicability of non-Markovian analysis to a wide range of discretized systems not suitable for the Volterra method.

In fact, the GPO analysis is not limited to data from MD simulations but can be used whenever encountering highly discrete experimental data. The application to data from single-molecule experiments[52−54] is a promising venue for future research.

■ METHODS

The MD simulation data is taken from our previous publication, see[14] for details. The MD simulation has a simulation time step of $\delta t = 1$ fs, while all GLE simulations use a time step of $\delta t = 2$ fs. In the computation of the hb4 coordinate (eq 2), the distances are computed between the oxygens of Ala2, Ala3, and Ala4 and the nitrogens of Ala6, Ala7, and Ala8, where Ala1 is the alanine residue at the N-terminus of the polypeptide of Ala$_9$.

All analysis code is written in Python[55] or Rust.[56] Table S3 shows the weights α for the loss $\mathcal{L}_{vx}$, which includes $C^{vv}(t)$ and $C^{\overline{xx}}(t)$. The memory kernels are fitted using the differential evolution algorithm implemented in the Python package 'scipy'[57] by minimizing a mean-squared loss, including both the kernel and the running integral over the kernel, $\mathcal{L}_{\text{mem}} = \mathcal{L}_\Gamma + \alpha_{\text{mem}}\mathcal{L}_G$, where $\mathcal{L}_\Gamma$ is the mean-squared loss of the kernel and $\mathcal{L}_G$ is the mean-squared loss of the running integral of the kernel. The resulting kernels and values for α_{mem} are shown in Table S1.

The GPO is performed using the 'GaussianProcessRegressor' implemented in the Python package 'scikit-learn',[58] using 10 optimizer restarts. When computing the loss $\mathcal{L}$, the correlation functions are evaluated over a finite number of sample points, N, always beginning with $t = 0$. The number of sample points N is given in Table S3. To minimize the expected improvement in eq S16 or maximize the standard deviation in eq S17, we use the 'L-BFGS-B' method implemented in 'scipy',[57] starting from 200 random samples drawn uniformly over the space of the parameters θ (see Table S2). When performing the analysis of the GPO on the basis of the 10 best runs, the integrations are repeated with a different seed for the random number generator used in the GLE integration, ensuring that the observables are reproduced by different integration runs with the same GLE parameters θ.

■ ASSOCIATED CONTENT

Supporting Information

The Supporting Information is available free of charge at https://pubs.acs.org/doi/10.1021/acs.jctc.3c01289.

Additional derivations, details for numerical implementations, detailed numerical results, and additional figures (PDF)

■ AUTHOR INFORMATION

Corresponding Author

Roland R. Netz − *Department of Physics, Freie Universität Berlin, 14195 Berlin, Germany;* orcid.org/0000-0003-0147-0162; Email: rnetz@physik.fu-berlin.de

Authors

Lucas Tepper − *Department of Physics, Freie Universität Berlin, 14195 Berlin, Germany;* orcid.org/0000-0001-8403-5275

Benjamin Dalton − *Department of Physics, Freie Universität Berlin, 14195 Berlin, Germany*

Complete contact information is available at:
https://pubs.acs.org/10.1021/acs.jctc.3c01289

Notes

The authors declare no competing financial interest.

■ ACKNOWLEDGMENTS

We gratefully acknowledge support by the Deutsche Forschungsgemeinschaft (DFG) Grant SFB 1449 and the European Research Council (ERC) Advanced Grant NoMaMemo No. 835117.

■ REFERENCES

(1) Quaresima, V.; Ferrari, M. A Mini-Review on Functional Near-Infrared Spectroscopy (fNIRS): Where Do We Stand, and Where Should We Go? *Photonics* **2019**, *6*, 87.

(2) Einstein, A. Über Die von Der Molekularkinetischen Theorie Der Wärme Geforderte Bewegung von in Ruhenden Flüssigkeiten Suspendierten Teilchen. *Ann. Phys.* **1905**, *322*, 549−560.

(3) Brox, Th. A One-Dimensional Diffusion Process in a Wiener Medium. *Ann. Probab.* **1986**, *14*, 1206−1218.

(4) Kitao, A.; Go, N. Investigating Protein Dynamics in Collective Coordinate Space. *Curr. Opin. Struct. Biol.* **1999**, *9*, 164−169.

(5) Best, R. B.; Hummer, G. Coordinate-Dependent Diffusion in Protein Folding. *Proc. Natl. Acad. Sci. U. S. A.* **2010**, *107*, 1088−1093.

(6) Ernst, M.; Wolf, S.; Stock, G. Identification and Validation of Reaction Coordinates Describing Protein Functional Motion: Hierarchical Dynamics of T4 Lysozyme. *J. Chem. Theory Comput.* **2017**, *13*, 5076−5088.

(7) Socci, N. D.; Onuchic, J. N.; Wolynes, P. G. Diffusive Dynamics of the Reaction Coordinate for Protein Folding Funnels. *J. Chem. Phys.* **1996**, *104*, 5860−5868.

(8) Neupane, K.; Manuel, A. P.; Woodside, M. T. Protein Folding Trajectories Can Be Described Quantitatively by One-Dimensional Diffusion over Measured Energy Landscapes. *Nat. Phys.* **2016**, *12*, 700−703.

(9) Mori, H. Transport, Collective Motion, and Brownian Motion. *Prog. Theor. Phys.* **1965**, *33*, 423−455.

(10) Zwanzig, R. Nonlinear Generalized Langevin Equations. *J. Stat. Phys.* **1973**, *9*, 215−220.

(11) Ayaz, C.; Scalfi, L.; Dalton, B. A.; Netz, R. R. Generalized Langevin Equation with a Nonlinear Potential of Mean Force and Nonlinear Memory Friction from a Hybrid Projection Scheme. *Phys. Rev. E* **2022**, *105*, No. 054138.

(12) Plotkin, S. S.; Wolynes, P. G. Non-Markovian Configurational Diffusion and Reaction Coordinates for Protein Folding. *Phys. Rev. Lett.* **1998**, *80*, 5015−5018.

(13) Kappler, J.; Daldrop, J. O.; Brünig, F. N.; Boehle, M. D.; Netz, R. R. Memory-Induced Acceleration and Slowdown of Barrier Crossing. *J. Chem. Phys.* **2018**, *148*, No. 014903.

(14) Ayaz, C.; Tepper, L.; Brünig, F. N.; Kappler, J.; Daldrop, J. O.; Netz, R. R. Non-Markovian Modeling of Protein Folding. *Proc. Natl. Acad. Sci. U. S. A.* **2021**, *118*, e2023856118.

(15) Dalton, B. A.; Ayaz, C.; Kiefer, H.; Klimek, A.; Tepper, L.; Netz, R. R. Fast Protein Folding Is Governed by Memory-Dependent Friction. *Proc. Natl. Acad. Sci. U. S. A.* **2023**, *120*, No. e2220068120.

(16) Berne, B. J.; Harp, G. D. *Advances in Chem. Phys.*; John Wiley & Sons, Ltd.: 2007; pp 63−227.

(17) Lange, O. F.; Grubmüller, H. Collective Langevin Dynamics of Conformational Motions in Proteins. *J. Chem. Phys.* **2006**, *124*, No. 214903.

(18) Deichmann, G.; van der Vegt, N. F. A. Bottom-up Approach to Represent Dynamic Properties in Coarse-Grained Molecular Simulations. *J. Chem. Phys.* **2018**, *149*, No. 244114.

(19) Daldrop, J. O.; Kappler, J.; Brünig, F. N.; Netz, R. R. Butane Dihedral Angle Dynamics in Water Is Dominated by Internal Friction. *Proc. Natl. Acad. Sci. U. S. A.* **2018**, *115*, 5169−5174.

(20) Jung, G.; Hanke, M.; Schmid, F. Iterative Reconstruction of Memory Kernels. *J. Chem. Theory Comput.* **2017**, *13*, 2481−2488.

(21) Daldrop, J. O.; Kowalik, B. G.; Netz, R. R. External Potential Modifies Friction of Molecular Solutes in Water. *Phys. Rev. X* **2017**, *7*, No. 041065.

(22) Vroylandt, H.; Goudenège, L.; Monmarché, P.; Pietrucci, F.; Rotenberg, B. Likelihood-Based Non-Markovian Models from Molecular Dynamics. *Proc. Natl. Acad. Sci. U. S. A.* **2022**, *119*, No. e2117586119.

(23) Wang, S.; Ma, Z.; Pan, W. Data-Driven Coarse-Grained Modeling of Polymers in Solution with Structural and Dynamic Properties Conserved. *Soft Matter* **2020**, *16*, 8330−8344.

(24) Ma, Z.; Wang, S.; Kim, M.; Liu, K.; Chen, C.-L.; Pan, W. Transfer Learning of Memory Kernels for Transferable Coarse-Graining of Polymer Dynamics. *Soft Matter* **2021**, *17*, 5864−5877.

(25) Lindorff-Larsen, K.; Piana, S.; Dror, R. O.; Shaw, D. E. How Fast-Folding Proteins Fold. *Science* **2011**, *334*, 517−520.

(26) Alemany, A.; Rey-Serra, B.; Frutos, S.; Cecconi, C.; Ritort, F. Mechanical Folding and Unfolding of Protein Barnase at the Single-Molecule Level. *Biophys. J.* **2016**, *110*, 63−74.

(27) von Hansen, Y.; Mehlich, A.; Pelz, B.; Rief, M.; Netz, R. R. Auto- and Cross-Power Spectral Analysis of Dual Trap Optical Tweezer Experiments Using Bayesian Inference. *Rev. Sci. Instrum.* **2012**, *83*, No. 095116.

(28) Mitterwallner, B. G.; Schreiber, C.; Daldrop, J. O.; Rädler, J. O.; Netz, R. R. Non-Markovian Data-Driven Modeling of Single-Cell Motility. *Phys. Rev. E* **2020**, *101*, No. 032408.

(29) Gordon, D.; Krishnamurthy, V.; Chung, S.-H. Generalized Langevin Models of Molecular Dynamics Simulations with Applications to Ion Channels. *J. Chem. Phys.* **2009**, *131*, No. 134102.

(30) Shin, H. K.; Kim, C.; Talkner, P.; Lee, E. K. Brownian Motion from Molecular Dynamics. *Chem. Phys.* **2010**, *375*, 316−326.

(31) Kowalik, B.; Daldrop, J. O.; Kappler, J.; Schulz, J. C. F.; Schlaich, A.; Netz, R. R. Memory-Kernel Extraction for Different Molecular Solutes in Solvents of Varying Viscosity in Confinement. *Phys. Rev. E* **2019**, *100*, No. 012126.

(32) Cao, S.; Qiu, Y.; Kalin, M. L.; Huang, X. Integrative Generalized Master Equation: A Method to Study Long-Timescale Biomolecular Dynamics via the Integrals of Memory Kernels. *J. Chem. Phys.* **2023**, *159*, No. 134106.

(33) Kappler, J.; Hinrichsen, V. B.; Netz, R. R. Non-Markovian Barrier Crossing with Two-Time-Scale Memory Is Dominated by the Faster Memory Component. *Eur. Phys. J. E: Soft Matter Biol. Phys.* **2019**, *42*, 119.

(34) Lavacchi, L.; Kappler, J.; Netz, R. R. Barrier Crossing in the Presence of Multi-Exponential Memory Functions with Unequal Friction Amplitudes and Memory Times. *EPL* **2020**, *131*, No. 40004.

(35) Bao, J.-D. Numerical Integration of a Non-Markovian Langevin Equation with a Thermal Band-Passing Noise. *J. Stat. Phys.* **2004**, *114*, 503−513.

(36) Cetin, B.; Burdick, J.; Barhen, J. Global Descent Replaces Gradient Descent to Avoid Local Minima Problem in Learning with Artificial Neural Networks. *IEEE International Conference on Neural Networks*; 1993; Vol. *2*, pp 836−842.

(37) Bandler, J. Optimization Methods for Computer-Aided Design. *IEEE Trans. Microw. Theory Techn.* **1969**, *17*, 533−552.

(38) Zeigler, B. P. *Study of Genetic Direct Search Algorithms for Function Optimization*; 1974.

(39) Fitzpatrick, J. M.; Grefenstette, J. J. Genetic Algorithms in Noisy Environments. *Machine Learning* **1988**, *3*, 101−120.

(40) Buche, D.; Schraudolph, N.; Koumoutsakos, P. Accelerating Evolutionary Algorithms with Gaussian Process Fitness Function Models. *IEEE T. Syst. Man. Cy. C* **2005**, *35*, 183−194.

(41) Williams, C. K. I.; Rasmussen, C. E. *Gaussian Processes for Machine Learning*; MIT Press: 2006.

(42) Gramacy, R. B. *Surrogates: Gaussian Process Modeling, Design and Optimization for the Applied Sciences*; Chapman Hall/CRC: Boca Raton, FL, 2020.

(43) Deringer, V. L.; Bartók, A. P.; Bernstein, N.; Wilkins, D. M.; Ceriotti, M.; Csányi, G. Gaussian Process Regression for Materials and Molecules. *Chem. Rev.* **2021**, *121*, 10073−10141.

(44) Stegle, O.; Fallert, S. V.; MacKay, D. J. C.; Brage, S. Gaussian Process Robust Regression for Noisy Heart Rate Data. *IEEE Trans. Biomed. Eng.* **2008**, *55*, 2143−2151.

(45) Daemi, A.; Alipouri, Y.; Huang, B. Identification of Robust Gaussian Process Regression with Noisy Input Using EM Algorithm. *Chemom. Intell. Lab. Syst.* **2019**, *191*, 1−11.

(46) Lin, M.; Song, X.; Qian, Q.; Li, H.; Sun, L.; Zhu, S.; Jin, R. Robust Gaussian Process Regression for Real-Time High Precision GPS Signal Enhancement. *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*; 2019; pp 2838−2847.

(47) Kaappa, S.; del Río, E. G.; Jacobsen, K. W. Global Optimization of Atomic Structures with Gradient-Enhanced Gaussian Process Regression. *Phys. Rev. B* **2021**, *103*, No. 174114.

(48) Nikolaidis, P.; Chatzis, S. Gaussian Process-Based Bayesian Optimization for Data-Driven Unit Commitment. *Int. J. Electr. Power Energy Syst.* **2021**, *130*, No. 106930.

(49) Zhao, T.; Zheng, Y.; Wu, Z. Improving Computational Efficiency of Machine Learning Modeling of Nonlinear Processes Using Sensitivity Analysis and Active Learning. *Digital Chemical Engineering* **2022**, *3*, No. 100027.

(50) Chang, J.; Kim, J.; Zhang, B.-T.; Pitt, M. A.; Myung, J. I. Data-Driven Experimental Design and Model Development Using Gaussian Process with Active Learning. *Cognit. Psychol.* **2021**, *125*, No. 101360.

(51) Jin, S.-S.; Hong, J.; Choi, H. Gaussian Process-Assisted Active Learning for Autonomous Data Acquisition of Impact Echo. *Autom. Constr.* **2022**, *139*, No. 104269.

(52) Petrosyan, R.; Patra, S.; Rezajooei, N.; Garen, C. R.; Woodside, M. T. Unfolded and Intermediate States of PrP Play a Key Role in the Mechanism of Action of an Antiprion Chaperone. *Proc. Natl. Acad. Sci. U. S. A.* **2021**, *118*, No. e2010213118.

(53) Hinczewski, M.; Gebhardt, J. C. M.; Rief, M.; Thirumalai, D. From Mechanical Folding Trajectories to Intrinsic Energy Landscapes of Biopolymers. *Proc. Natl. Acad. Sci. U. S. A.* **2013**, *110*, 4500−4505.

(54) Neupane, K.; Foster, D. A. N.; Dee, D. R.; Yu, H.; Wang, F.; Woodside, M. T. Direct Observation of Transition Paths during the Folding of Proteins and Nucleic Acids. *Science* **2016**, *352*, 239−242.

(55) Van Rossum, G.; Drake, F. L. *Python 3 Reference Manual*; CreateSpace: Scotts Valley, CA, 2009.

(56) Matsakis, N. D.; Klock II, F. S. The rust language. *ACM SIGAda Ada Letters* **2014**, *34*, 103−104.

(57) Virtanen, P.; et al. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nat. Methods* **2020**, *17*, 261−272.

(58) Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine learning in Python. *Journal of machine learning research* **2011**, *12*, 2825−2830.