A simple method for automated equilibration detection in molecular simulations

John D. Chodera[1, *]

[1] *Computational Biology Program, Sloan Kettering Institute,*
Memorial Sloan Kettering Cancer Center, New York, NY 10065

(Dated: July 3, 2015)

Molecular simulations intended to compute equilibrium properties are often initiated from configurations that are highly atypical of equilibrium samples, a practice which can generate a distinct initial transient in mechanical observables computed from the simulation trajectory. Traditional practice in simulation data analysis recommends this initial portion be discarded to *equilibration*, but no simple, general, and automated procedure for this process exists. Here, we suggest a conceptually simple automated procedure that does not make strict assumptions about the distribution of the observable of interest, in which the equilibration time is chosen to maximize the number of effectively uncorrelated samples in the production timespan used to compute equilibrium averages. We present a simple Python reference implementation of this procedure, and demonstrate its utility on typical molecular simulation data.

Keywords: molecular dynamics (MD); Metropolis-Hastings; Monte Carlo (MC); Markov chain Monte Carlo (MCMC); equilibration; burn-in; timeseries analysis; statistical inefficiency; integrated autocorrelation time

INTRODUCTION

Molecular simulations use Markov chain Monte Carlo (MCMC) techniques [1] to sample configurations x from an equilibrium distribution $\pi(x)$, either exactly (using Monte Carlo methods such as Metropolis-Hastings) or approximately (using molecular dynamics integrators without Metropolization) [2].

Due to the sensitivity of the equilibrium probability density $\pi(x)$ to small perturbations in configuration x and the difficulty of producing sufficiently good guesses of typical equilibrium configurations $x \sim \pi(x)$, these molecular simulations are often started from highly atypical initial conditions. For example, simulations of biopolymers might be initiated from a fully extended conformation unrepresentative of behavior in solution, or a geometry derived from a fit to diffraction data collected from a cryocooled crystal; solvated systems may be prepared by periodically replicating a small solvent box equilibrated under different conditions, yielding atypical densities and solvent structure; liquid mixtures or lipid bilayers may be constructed by using methods that fulfill spatial constraints (e.g. PackMol [3]) but create locally aytpical geometries, requiring long simulation times to relax to typical configurations.

As a result, traditional practice in molecular simulation has recommended some initial portion of the trajectory be discarded to *equilibration* (also called *burn-in*[1] in the MCMC literature [4]). While the process of discarding initial samples is strictly unnecessary for the time-average of quantities of interest to eventually converge to the desired expectations [5], this nevertheless often allows the practitioner to avoid what may be impractically long run times to eliminate the bias in computed properties in finite-length simulations induced by atypical initial starting conditions. It is worth noting that a similar procedure is not a practice universally recommended by statisticians when sampling from posterior distributions in statistical inference [4]; the differences in complexity of probability densities typically encountered in statistics and molecular simulation may explain the difference in historical practice.

As a motivating example, consider the computation of the average density of liquid argon under a given set of reduced temperature and pressure conditions shown in Figure 1. To initiate the simulation, an initial dense liquid geometry at reduced density $\rho^* \equiv \rho\sigma^3 = 0.960$ was prepared and subjected to local energy minimization. The upper panel of Figure 1 depicts the average relaxation behavior of simulations initiated from the same configuration with different random initial velocities and integrator random number seeds (see *Simulation Details*). The average (black line) and 95% confidence interval (shaded grey) of 500 realizations of this process show a characteristic relaxation behavior away from the initial density toward the equilibrium density. The expectation of the running average of the density over many realizations of this procedure (Figure 1, lower panel) significantly deviates from the true expectation (dashed line), leading to significantly biased estimates of the expectation unless simulations are sufficiently long to eliminate this starting point dependent bias—a surprisingly long 30 ns in this case. Note that this bias is present even in the average of many realizations because the *same* atypical starting condition is used for every realization of this simulation process.

To develop an automatic approach to eliminating this bias, we take motivation from the concept of *reverse cumulative averaging* from Yang et al. [6], in which the trajectory statistics over the production region of the trajectory are examined for different choices of the end of the discarded equilibration region to determine the optimal production region to use for computing expectations and other statistical properties. We begin by first formalizing our objectives mathematically.

[*] Corresponding author; john.chodera@choderalab.org

[1] The term *burn-in* comes from the field of electronics, in which a short "burn-in" period is used to ensure that a device is free of faulty components—which often fail quickly—and is operating normally [4].

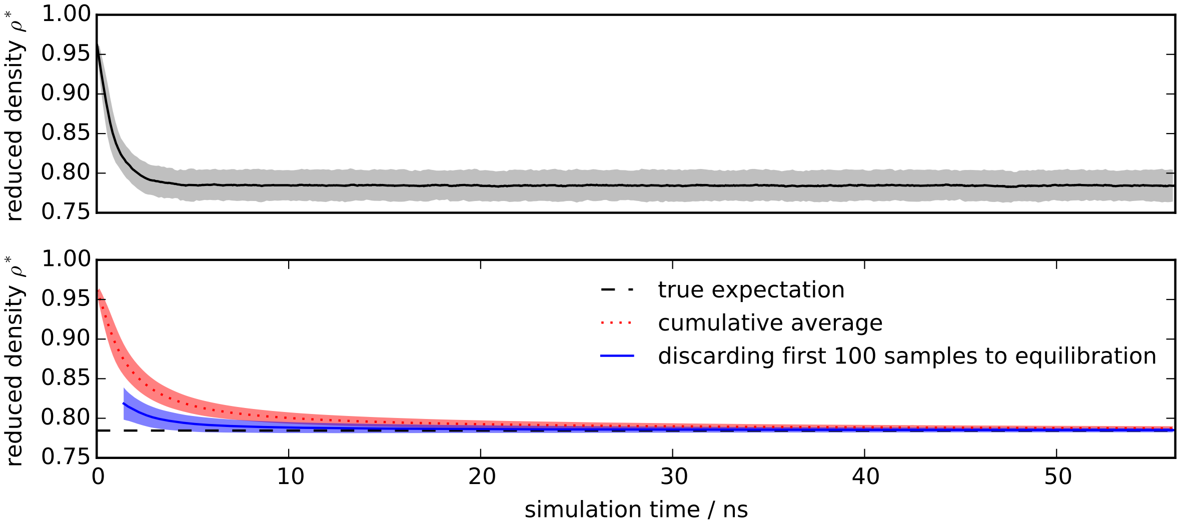

FIG. 1. Illustration of the motivation for discarding data to equilibration. To illustrate the bias in expectations induced by relaxation away from initial conditions, 500 replicates of a simulation of liquid argon were initiated from the same energy-minimized initial configuration constructed with initial reduced density $\rho^* \equiv \rho\sigma^3 = 0.960$ but different random number seeds for stochastic integration. **Top:** The average of the reduced density (black line) over the replicates relaxes to the region of typical equilibrium densities over the first few ns of simulation time. **Bottom:** If the average density is estimated by a cumulative average from the beginning of the simulation (red dotted line), the estimate will be heavily biased by the atypical starting density even beyond 10 ns. Discarding even a small amount of initial data—in this case 500 initial samples (∼1.4 ns, blue solid line)—results in a cumulative average estimate that converges to the true average (black dashed line) much more rapidly. Shaded regions denote 95% confidence intervals.

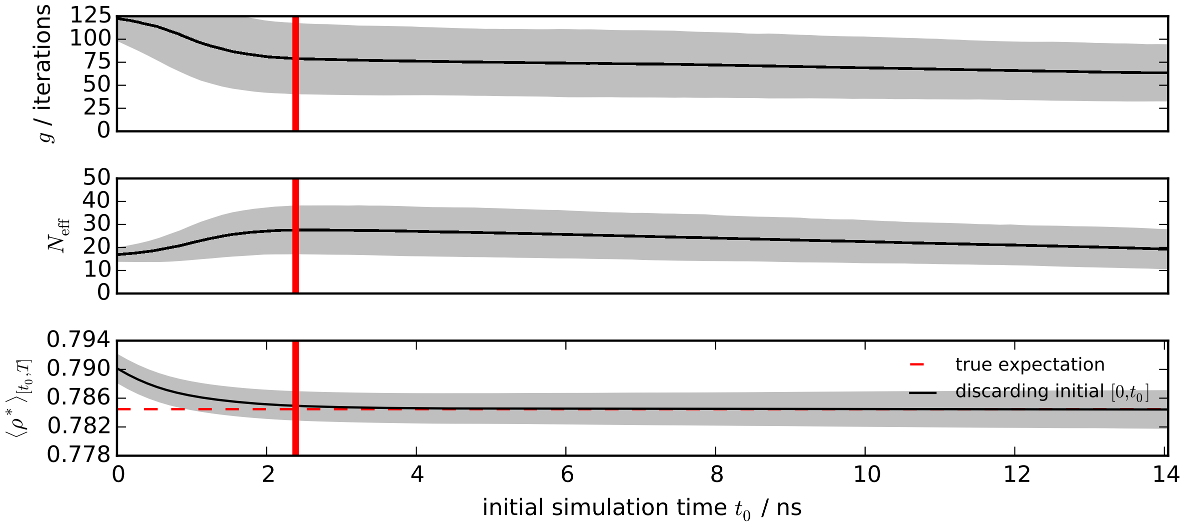

FIG. 2. Statistical inefficiency, number of uncorrelated samples, and bias for different equilibration times. Trajectories of length $T = 2\,000$ iterations (∼28 ns) for the argon system described in Fig. 1 were analyzed as a function of equilibration time choice t_0. Averages over all 500 replicate simulations (all starting from the same initial conditions) are shown as dark lines, with shaded lines showing standard deviation of estimates among replicates. **Top:** The statistical inefficiency g as a function of equilibration time choice t_0 is initially very large, but diminishes rapidly after the system has relaxed to equilibrium. **Middle:** The number of effectively uncorrelated samples $N_{\text{eff}} = (T - t_0 + 1)/g$ shows a maximum at $t_0 \sim 200$ iterations (∼2 ns), suggesting the system has equilibrated by this time. The red vertical line in all plots marks this choice of $t_0 \sim 200$. **Bottom:** The cumulative density average $\langle\rho^*\rangle$ computed over the span $[t_0, T]$ shows that the bias (deviation from the true estimate, shown as red dashed lines) is minimized for choices of $t_0 \geq 200$ iterations ($\sim$ 2 ns). The standard deviation among replicates (shaded region) grows with t_0 because fewer data are included in the estimate. The choice of optimal t_0 that maximizes N_{eff} (red vertical line) strikes a good balance between bias and variance. The true estimate (red dashed lines) is computed from averaging over the range [5 000, 10 000] iterations over all 500 replicates.

STATEMENT OF THE PROBLEM

Consider T successively sampled configurations x_t from a molecular simulation, with $t = 1, \ldots, T$. We presume we are interested in computing the expectation

$$\langle A \rangle \equiv \int dx \, A(x) \, \pi(x) \qquad (1)$$

of a mechanical property $A(x)$. For convenience, we will refer to the timeseries $a_t \equiv A(x_t)$, with $t \in [1, T]$. The estimator $\hat{A} \approx \langle A \rangle$ constructed from the entire dataset is given by

$$\hat{A}_{[1,T]} \equiv \frac{1}{T} \sum_{t=1}^{T} a_t . \qquad (2)$$

While $\lim_{T \to \infty} \hat{A}_{[1,T]} = \langle A \rangle$ for an infinitely long simulation[2], the bias in $\hat{A}_{[1,T]}$ may be significant in a simulation of finite length T.

By discarding samples $t < t_0$ to equilibration, we hope to exclude the initial transient from our sample average, and provide a less biased estimate of $\langle A \rangle$,

$$\hat{A}_{[t_0,T]} \equiv \frac{1}{T - t_0 + 1} \sum_{t=t_0}^{T} a_t . \qquad (3)$$

We can quantify the overall error in an estimator $\hat{A}_{[t_0,T]}$ in a sample average that starts at x_0 and excludes samples where $t < t_0$ by the expected error $\delta^2 \hat{A}_{[t_0,T]}$,

$$\delta^2 \hat{A}_{[t_0,T]} \equiv E_{x_0} \left[\left(\hat{A}_{[t_0,T]} - \langle A \rangle \right)^2 \right] \qquad (4)$$

where $E_{x_0}[\cdot]$ denotes the expectation over independent realizations of the specific simulation process initiated from configuration x_0, but with different velocities and random number seeds.

We can rewrite the expected error $\delta^2 \hat{A}$ by separating it into two components:

$$\delta^2 \hat{A}_{[t_0,T]} = E_{x_0} \left[\left(\hat{A}_{[t_0,T]} - E_{x_0}[\hat{A}_{[t_0,T]}] \right)^2 \right] + \left(E_{x_0}[\hat{A}_{[t_0,T]}] - \langle A \rangle \right)^2 \qquad (5)$$

The first term denotes the variance in the estimator $\hat{A}$,

$$\mathrm{var}_{x_0}(\hat{A}_{[t_0,T]}) \equiv E_{x_0} \left[\hat{A}_{[t_0,T]} - E_{x_0}[\hat{A}_{[t_0,T]}] \right]^2 \qquad (6)$$

while the second term denotes the contribution from the squared bias,

$$\mathrm{bias}^2_{x_0}(\hat{A}_{[t_0,T]}) \equiv \left(E_{x_0}[\hat{A}_{[t_0,T]}] - \langle A \rangle \right)^2 \qquad (7)$$

BIAS-VARIANCE TRADEOFF

With increasing equilibration time t_0, bias is reduced, but the variance—the contribution to error due to random variation from having a finite number of uncorrelated samples—will increase because less data is included in the estimate. This can be seen in the bottom panel of Figure 2, where the shaded region (denoting the 95% confidence interval of the mean, computed from twice the standard deviation among sample estimates) increases in width with increasing equilibration time t_0.

To examine the tradeoff between bias and variance explicitly, Figure 3 plots the bias and variance (here, shown as standard error) contributions against each other as a function of t_0 (denoted by color) as computed from statistics over all 500 replicates. At $t_0 = 0$, the bias is large but variance is minimized. With increasing t_0, bias is eventually eliminated but then variance rapidly grows as fewer uncorrelated samples are included in the estimate. There is a clear optimal choice at $t_0 \sim 150$ iterations that minimizes variance while also effectively eliminating bias.

SELECTING THE EQUILIBRATION TIME

Is there a simple approach to choosing an optimal equilibration time t_0 that provides a significantly improved estimate $\hat{A}_{[t_0,T]}$, even when we do not have access to multiple realizations? At worst, we hope that such a procedure would at least give some improvement over the naive estimate, such that $\delta^2 \hat{A}_{[t_0,T]} < \delta^2 \hat{A}_{[1,T]}$; at best, we hope that we can achieve a reasonable bias-variance tradeoff close to the optimal point identified in Figure 3 that minimizes bias without greatly increasing variance. We remark that, for cases in which the simulation is not long enough to reach equilibrium, no choice of t_0 will eliminate bias completely; the best we can hope for is to minimize this bias.

While automated methods for selecting the equilibration time t_0 have been proposed, these approaches have shortcomings that have greatly limited their use. The reverse cumulative averaging (RCA) method proposed by Yang et al. [6], for example, uses a statistical test for normality to determine the point before which which the observable timeseries deviates from normality when examining the timeseries in reverse. While this concept may be reasonable for experimental data, where measurements often represent the sum of many random variables such that the central limit theorem's guarantee of asymptotic normality ensures the distribution of the observable will be approximately normal, there is no such guarantee that instantaneous measurements of a simulation property of interest will be normally distributed. In fact, many properties will be decidedly *non-normal*. For a biomolecule such as a protein, for example, the radius of gyration, end-to-end distance, and torsion angles sampled during a simulation will all be highly non-normal. Instead, we require a method that makes no assumptions about the nature of the distribution of the property under study.

[2] We note that this equality only holds for simulation schemes that sample from the true equilibrium density $\pi(x)$, such as Metropolis-Hastings Monte Carlo or Metropolized dynamical integration schemes such as hybrid Monte Carlo (HMC). Molecular dynamics simulations utilizing finite timestep integration without Metropolization will produce averages that may deviate from the true expectation $\langle A \rangle$ [2].

AUTOCORRELATION ANALYSIS

The set of successively sampled configurations $\{x_t\}$ and their corresponding observables $\{a_t\}$ compose a correlated timeseries of observations. To estimate the statistical error or uncertainty in a stationary timeseries free of bias, we must be able to quantify the *effective number of uncorrelated samples* present in the dataset. This is usually accomplished through computation of the *statistical inefficiency* g, which quantifies the number of correlated timeseries samples needed to produce a single effectively uncorrelated sample of the observable of interest. While these concepts are well-established for the analysis of both Monte Carlo and molecular dynamics simulations [7–10], we review them here for the sake of clarity.

For a given equilibration time choice t_0, the statistical uncertainty in our estimator $\hat{A}_{[t_0,T]}$ can be written as,

$$\begin{aligned}
\delta^2 \hat{A}_{[t_0,T]} &\equiv E_{x_0}\left[\left(\hat{A}_{[t_0,T]} - \langle \hat{A} \rangle\right)^2\right] \\
&= E_{x_0}\left[\hat{A}^2_{[t_0,T]}\right] - E_{x_0}\left[\hat{A}_{[t_0,T]}\right]^2 \\
&= \frac{1}{T_{t_0}^2} \sum_{t,t'=t_0}^{T} \left\{ E_{x_0}\left[a_t a_{t'}\right] - E_{x_0}\left[a_t\right] E_{x_0}\left[a_{t'}\right] \right\} \\
&= \frac{1}{T_{t_0}^2} \sum_{t=t_0}^{T} \left\{ E_{x_0}\left[x_t^2\right] - E_{x_0}\left[x_t\right]^2 \right\} \qquad (8) \\
&+ \frac{1}{T_{t_0}^2} \sum_{t \neq t'=t_0}^{T} \left\{ E_{x_0}\left[a_t a_{t'}\right] - E_{x_0}\left[a_t\right] E_{x_0}\left[a_{t'}\right] \right\},
\end{aligned}$$

where $T_{t_0} \equiv T - t_0 + 1$, the number of correlated samples in the timeseries $\{a_t\}_{t_0}^{T}$. In the last step, we have split the double-sum into two separate sums—a term capturing the variance in the observations a_t, and a remaining term capturing the correlation between observations.

If t_0 is sufficiently large for the initial bias to be eliminated, the remaining timeseries $\{a_t\}_{t_0}^{T}$ will obey the properties of both *stationarity* and *time-reversibility*, allowing us to write,

$$\begin{aligned}
\delta^2 \hat{A}^{\mathrm{equil}}_{[t_0,T]} &= \frac{1}{T_{t_0}} \left[\langle a_t^2 \rangle - \langle a_t \rangle^2\right] \\
&+ \frac{2}{T_{t_0}} \sum_{n=1}^{T-t_0} \left(\frac{T_{t_0} - n}{T_{t_0}}\right) \left[\langle a_t a_{t+n} \rangle - \langle a_t \rangle \langle a_{t+n} \rangle\right] \\
&\equiv \frac{\sigma_{t_0}^2}{T_{t_0}} (1 + 2\tau_{t_0}) = \frac{\sigma_{t_0}^2}{T_{t_0}/g_{t_0}}, \qquad (9)
\end{aligned}$$

where the variance σ^2, statistical inefficiency g, and integrated autocorrelation time τ (in units of the sampling interval) are given by

$$\sigma^2 \equiv \langle a_t^2 \rangle - \langle a_t \rangle^2, \qquad (10)$$

$$\tau \equiv \sum_{t=1}^{T-1} \left(1 - \frac{t}{T}\right) C_t, \qquad (11)$$

$$g \equiv 1 + 2\tau, \qquad (12)$$

with the discrete-time normalized fluctuation autocorrelation function C_t defined as

$$C_t \equiv \frac{\langle a_n a_{n+t} \rangle - \langle a_n \rangle^2}{\langle a_n^2 \rangle - \langle a_n \rangle^2}. \qquad (13)$$

In practice, it is difficult to estimate C_t for $t \sim T$, due to growth in the statistical error, so common estimators of g make use of several additional properties of C_t to provide useful estimates (see *Practical Computation of Statistical Inefficiencies*).

The t_0 subscript for the variance σ^2, the integrated autocorrelation time τ, and the statistical inefficiency t_0 mean that these quantities are only estimated over the production portion of the timeseries, $\{a_t\}_{t=t_0}^{T}$. Since we assumed that the bias was eliminated by judicious choice of the equilibration time t_0, this estimate of the statistical error will be poor for choices of t_0 that are too small.

THE ESSENTIAL IDEA

Suppose we choose some arbitrary time t_0 and discard all samples $t \in [0, t_0)$ to equilibration, keeping $[t_0, T]$ as the dataset to analyze. How much data remains? We can determine this by computing the statistical inefficiency g_{t_0} for the interval $[t_0, T]$, and computing the effective number of uncorrelated samples $N_{\mathrm{eff}}(t_0) \equiv (T - t_0 + 1)/g_{t_0}$. If we start at $t_0 \equiv T$ and move t_0 to earlier and earlier points in time, we expect that the effective number of uncorrelated samples $N_{\mathrm{eff}}(t_0)$ will continue to grow until we start to include the highly atypical initial data. At that point, the integrated autocorrelation time τ (and hence the statistical inefficiency g) will greatly increase (a phenomenon observed earlier, e.g. Fig. 2 of [6]). As a result, the effective number of samples N_{eff} will start to plummet.

Figure 2 demonstrates this behavior for the liquid argon system described above, using averages of the statistical inefficiency g_{t_0} and $N_{\mathrm{eff}}(t_0)$ computed over 500 independent replicate trajectories. At short t_0, the average statistical inefficiency g (Figure 2, top panel) is large due to the contribution from slow relaxation from atypical initial conditions, while at long t_0 the statistical inefficiency estimate is much shorter and nearly constant of a large span of time origins. As a result, the average effective number of uncorrelated samples N_{eff} (Figure 2, middle panel) has a peak at $t_0 \sim 222$ iterations (Figure 2, vertical red lines). The effect on bias in the estimated average reduced density $\langle \rho^* \rangle$ (Figure 2, bottom panel) is striking—the bias is essentially eliminated for the choice of equilibration time t_0 that maximizes the number of uncorrelated samples N_{eff}.

This suggests an alluringly simple algorithm for identifying the optimal equilibration time—pick the t_0 which maximizes the number of uncorrelated samples N_{eff}. In mathematical terms,

$$\begin{aligned}
t_0^{\mathrm{opt}} &= \operatorname*{argmax}_{t_0} N_{\mathrm{eff}}(t_0) \qquad (14) \\
&= \operatorname*{argmax}_{t_0} \frac{T - t_0 + 1}{g_{t_0}} \qquad (15)
\end{aligned}$$

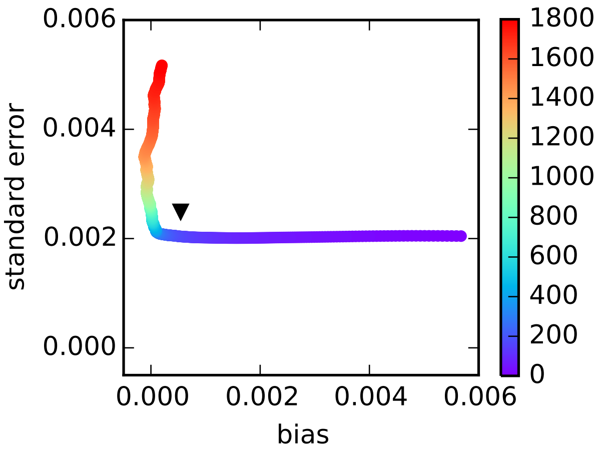

FIG. 3. **Bias-variance tradeoff for fixed equilibration time versus automatic equilibration time selection.** Trajectories of length T = 2 000 iterations (∼28 ns) for the argon system described in Fig. 1 were analyzed as a function of equilibration time choice t_0, with colors denoting the value of t_0 (in iterations) corresponding to each plotted point. Using 500 replicate simulations, the average bias (average deviation from true expectation) and standard deviation (random variation from replicate to replicate) were computed as a function of a prespecified fixed equilibration time t_0, with colors running from $t_0 = 0$ (violet) to $t_0 =$1 1800 iterations (red). As is readily discerned, the bias for small t_0 is initially large, but minimized for larger t_0. By contrast, the standard error (a measure of variance, estimated here by standard deviation among replicates) grows as t_0 grows above a certain critical time (here, ∼200 iterations). If the t_0 that maximizes N_eff is instead chosen *individually* for each trajectory based on that trajectory's estimates of statistical inefficiency $g_{[t_0,T]}$, the resulting bias-variance tradeoff (black triangle) does an excellent job minimizing bias and variance simultaneously, comparable to what is possible for a choice of equilibration time t_0 based on knowledge of the true bias and variance among many replicate estimates.

Bias-variance tradeoff. But how will this strategy work for cases where we do not know the statistical inefficiency g as a function of the equilibration time t_0 precisely? When all that is available is a single simulation, our best estimate of g_{t_0} is derived from that simulation alone over the span $[t_0, T]$—will this affect the quality of our estimate of equilibration time? Empirically, this does not appear to be the case—the black triangle in Figure 3 shows the bias and variance contributions to the error for estimates computed over the 500 replicates where t_0 is individually determined from each simulation using this simple scheme based on selecting t_0 to maximize N_eff for each individual realization. Despite not having knowledge about multiple realizations, this strategy effectively achieves a near-optimal balance between minimizing bias without increasing variance.

Overall RMS error. How well does this strategy perform in terms of decreasing the *overall* error $\delta\hat{A}_{[t_0,T]}$ compared to $\delta\hat{A}_{[1,T]}$? Fiigure 4 compares the expected standard error (denoted $\delta\hat{A}$ in the figure) as a function of a fixed initial equilibration time t_0 (black line with shaded region denoting 95% confidence interval) with the strategy of selecting t_0 to maximize N_eff for each realization (red line with shaded region denoted 95% confidence interval). While the minimum error for the fixed-t_0 strategy (0.002545±0.00009) is achieved in the range of 2–6 ns—a fact that could only be determined from knowledge of multiple realizations—the simple strategy of selecting t_0 using Eq. 14 achieves a minimum error that is only 6% worse (compared to 137% worse should no data have been discarded). While this is certainly statistically significant in being worse than the optimal t_0 given all 500 replicates, it is remarkably good for a simple method employing data from a single simulation.

DISCUSSION

The scheme described here—in which the equilibration time t_0 is computed using Eq. 14 as the choice that maximizes the number of uncorrelated samples in the production region $[t_0, T]$—is both conceptually and computationally straightforward. It provides an approach to determining the optimal amount of initial data to discard to equilibration in order to minimize variance while also minimizing initial bias, and does this without employing statistical tests that require generally unsatisfiable assumptions of normality of the observable of interest. As we have seen, this scheme empirically appears to select a practical compromise between bias and variance even when the statistical inefficiency g is estimated directly from the trajectory using Eq. 12.

A word of caution is necessary. One can certainly envision pathological scenarios where this algorithm for selecting an optimal equilibration time will break down. In cases where the simulation is not long enough to reach equilibrium—let alone collect many uncorrelated samples from it—no choice of equilibration time will bestow upon the experimenter the ability to produce an unbiased estimate of the true expectation. Similarly, in cases where insufficient data is available for the statistical inefficiency to be estimated well, this algorithm is expected to perform poorly. However, in these cases, the data itself should be suspect if the trajectory is not at least an order of magnitude longer than the minimum estimated autocorrelation time.

SIMULATION DETAILS

All molecular dynamics simulations described here were performed with OpenMM 6.2 [11] (available at openmm.org) using the Python API. All scripts used to retrieve the software versions used here, run the simulations, analyze data, and generate plots—along with the simulation data itself and

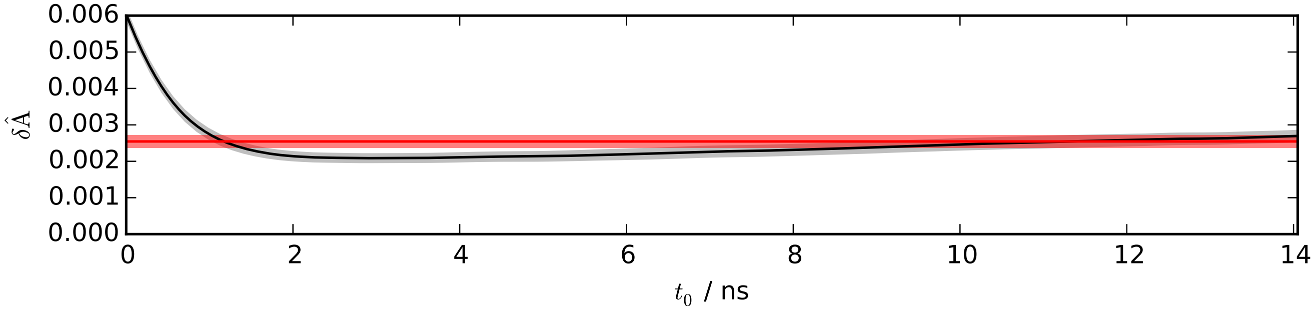

FIG. 4. RMS error for fixed equilibration time versus automatic equilibration time selection. Trajectories of length $T = 2\,000$ iterations (~28 ns) for the argon system described in Fig. 1 were analyzed as a function of fixed equilibration time choice t_0. Using 500 replicate simulations, the root-mean-squared (RMS) error (Eq. 4) was computed (black line) along with 95% confidence interval (gray shading). The RMS error is minimized for fixed equilibration time choices in the range 2–6 ns. If the t_0 that maximizes N_eff is instead chosen *individually* for each trajectory based on that trajectory's estimated statistical inefficiency $g_{[t_0,T]}$ using Eq. 14, the resulting RMS error (red line, 95% confidence interval shown as red shading) is quite close to the minimum RMS error achieved from any particular *fixed* choice of equilibration time t_0, suggesting that this simple automated approach to selecting t_0 achieves close to optimal performance.

scripts for generating figures—are available on GitHub[3].

The LennardJonesFluid model system in the openmmtools package[4] was used with parameters appropriate for liquid argon (σ = 3.4 Å, ϵ = 0.238 kcal/mol), though all results are reported in reduced (universal) units. A cubic switching function was employed, with the potential gently switched to zero over $r \in [\sigma, 3\sigma]$, and a long-range isotropic dispersion correction accounting for this switching behavior used to include neglected contributions. Simulations were performed using a periodic box of $N = 500$ atoms at reduced temperature $T^* \equiv k_B T/\epsilon = 0.850$ and reduced pressure $p^* \equiv p\sigma^3/\epsilon = 1.266$ using a Langevin integrator [12] with timestep $\Delta t = 0.01\tau$ and collision rate $\nu = 1.5\tau^{-1}$, with characteristic oscillation timescale $\tau = \sqrt{mr_0^2/72\epsilon}$ and $r_0 = 2^{1/6}\sigma$ [13]. A molecular scaling Metropolis Monte Carlo barostat with Gaussian simulation volume change proposal moves attempted every 25 timesteps, along with an adaptive algorithm that adjusts the proposal width during the initial part of the simulation [11]. Densities were recorded every 25 timesteps, with each set of 25 timesteps termed an "iteration" of the simulation. The true expectation $\langle\rho^*\rangle$ was estimated from the sample average over all 500 realizations over [5000,10000] iterations.

The automated equilibration detection scheme is also available in the timeseries module of the pymbar package as detectEquilibration(), and can be accessed using the following code:

```
from pymbar.timeseries import detectEquilibration
# determine equilibrated region
[t0, g, Neff_max] = detectEquilibration(A_t)
# discard initial samples to equilibration
A_t = A_t[t0:]
```

PRACTICAL COMPUTATION OF STATISTICAL INEFFICIENCIES

The robust computation of the statistical inefficiency g (defined by Eq. 12) for a finite timeseries a_t, $t = 1, \ldots, T$ deserves some comment. There are, in fact, a variety of schemes for estimating g described in the literature, and their behaviors for finite datasets may differ, leading to different estimates of the equilibration time t_0 using the algorithm of Eq. 14.

The main issue is that a straightforward approach to estimating the statistical inefficiency using Eqs. 11–13 in which the expectations are simply replaced with sample estimates causes the statistical error in the estimated correlation function C_t to grow with t in a manner that allows this error to quickly overwhelm the sum of Eq. 11. As a result, a number of alternative schemes—generally based on controlling the error in the estimated C_t or truncating the sum of Eq. 11 when the error grows too large—have been proposed.

For stationary, irreducible, reversible Markov chains, Geyer observed that a function $\Gamma_k \equiv \gamma_{2k} + \gamma_{2k+1}$ of the unnormalized fluctuation autocorrelation function $\gamma_t \equiv \langle a_i a_{i+t}\rangle - \langle a_i\rangle^2$ has a number of pleasant properties (Theorem 3.1 of [14]): It is strictly positive, strictly decreasing, and strictly convex. Some or all of these properties can be exploited to define a family of estimators called *initial sequence methods* (see Section 3.3 of [14] and Section 1.10.2 of [4]), of which the *initial convex sequence* (ICS) estimator is generally agreed to be optimal, if somewhat more complex to implement.[5]

[3] All Python scripts necessary to reproduce this work—along with data plotted in the published version—are available at: http://github.com/choderalab/automatic-equilibration-detection

[4] available at http://github.com/choderalab/openmmtools

[5] Implementations of these methods are provided with the code dis-

All computations in this manuscript used the fast multiscale method described in Section 5.2 of [10], which we found performed equivalently well to the Geyer estimators (data not shown). This method is related to a multiscale variant of the *initial positive sequence* (IPS) method of Geyer [15], where contributions are accumulated at increasingly longer lag times and the sum of Eq. 11 is truncated when the terms become negative. We have found this method to be both fast and to provide useful estimates of the statistical inefficiency, but it may not perform well for all problems.

ACKNOWLEDGMENTS

We are grateful to William C. Swope (IBM Almaden Research Center) for his illuminating introduction to the use of autocorrelation analysis for the characterization of statistical error, as well as Michael R. Shirts (University of Virginia), David L. Mobley (University of California, Irvine), Michael K. Gilson (University of California, San Diego), Kyle A. Beauchamp (MSKCC), and Robert C. McGibbon (Stanford University) for valuable discussions on this topic, and Joshua L. Adelman (University of Pittsburgh) for helpful feedback and encouragement. We are grateful to Michael K. Gilson (University of California, San Diego) and Wei Yang (Florida State University) for critical feedback on the manuscript itself.

[1] J. S. Liu, *Monte Carlo strategies in scientific computing*, 2nd ed. ed. (Springer-Verlag, New York, 2002).

[2] D. Sivak, J. Chodera, and G. Crooks, Physical Review X **3**, 011007 (2013), bibtex: Sivak:2013:Phys.Rev.X.

[3] L. Martínez, R. Andrade, E. G. Birgin, and J. M. Martínez, J. Chem. Theor. Comput. **30**, 2157 (2009).

[4] S. Brooks, A. Gelman, G. L. Jones, and X.-L. Meng, in *Handbook of Markov chain Monte Carlo*, *Chapman & Hall/CRC Handbooks of Modern Statistical Methods* (CRC Press, ADDRESS, 2011), Chap. Introduction to Markov chain Monte Carlo.

[5] C. Geyer, Burn-in is unnecessary., http://users.stat.umn.edu/~geyer/mcmc/burn.html.

[6] W. Yang, R. Bittetti-Putzer, and M. Karplus, J. Chem. Phys. **120**, 2618 (2004).

[7] H. Müller-Krumbhaar and K. Binder, J. Stat. Phys. **8**, 1 (1973).

[8] W. C. Swope, H. C. Andersen, P. H. Berens, and K. R. Wilson, J. Chem. Phys. **76**, 637 (1982).

[9] W. Janke, in *Quantum Simulations of Complex Many-Body Systems: From Theory to Algorithms*, edited by J. Grotendorst, D. Marx, and A. Murmatsu (John von Neumann Institute for Computing, ADDRESS, 2002), Vol. 10, pp. 423–445.

[10] J. D. Chodera, W. C. Swope, J. W. Pitera, C. Seok, and K. A. Dill, J. Chem. Theor. Comput. **3**, 26 (2007).

[11] P. Eastman, M. Friedrichs, J. D. Chodera, R. Radmer, C. Bruns, J. Ku, K. Beauchamp, T. J. Lane, L.-P. Wang, D. Shukla, T. Tye, M. Houston, T. Stitch, and C. Klein, J. Chem. Theor. Comput. **9**, 461 (2012).

[12] D. A. Sivak, J. D. Chodera, and G. E. Crooks, J. Phys. Chem. B **118**, 6466 (2014).

[13] B. Veytsman and M. Kotelyanskii, Lennard-Jones potential revisited., http://borisv.lk.net/matsc597c-1997/simulations/Lecture5/node3.html.

[14] C. J. Geyer, Stat. Sci. **76**, 473 (1992).

[15] C. J. Geyer and E. A. Thompson, J. Royal Stat. Soc. B **54**, 657 (1992).

tributed with this manuscript.