Accelerated estimation of long-timescale kinetics by combining weighted ensemble simulation with Markov model “microstates” using non-Markovian theory

Jeremy Copperman¹ and Daniel M. Zuckerman¹,*

¹Department of Biomedical Engineering, Oregon Health and Science University, Portland OR
*zuckermd@ohsu.edu

ABSTRACT

The weighted ensemble (WE) simulation strategy can provide unbiased sampling of non-equilibrium processes, such as molecular folding or binding. Once converged to steady state, exact kinetics can be extracted from any discrete clustering of the configuration space based on a history-dependent trajectory labeling process using any lag time. However, the convergence of WE to steady state may require unaffordably long simulations in complex systems. Here we show that by clustering molecular configurations into many (thousands of) microstates using methods developed in the Markov State Modeling (MSM) community, unbiased kinetics can be obtained from WE data using history-augmented Markov State Models (haMSMs) before steady-state convergence of the WE simulation itself. Because arbitrarily small lag times can be used within the exact haMSM formulation, accurate kinetics can be obtained with significantly less trajectory data than traditional MSMs, while bypassing the often prohibitive convergence requirements of the non-equilibrium weighted ensemble. We validate the method by comparison to recently obtained unbiased estimations of atomistic protein folding rates using WE molecular dynamics.

Introduction

In this study, we address a straightforward technical question: Can steady-state (SS) kinetic information (i.e., the rate constant) be extracted from unbiased transient data obtained via weighted ensemble (WE) simulation? The data examined below will show that this is indeed the case, and that rate-constant estimates are most reliable when extracted using a non-Markov analysis¹–³ of a very fine discretization of configuration space – typically much finer than was used to run the original WE simulations themselves. WE simulations typically use a discretization of configuration space into bins, each of which may contain several trajectories running in parallel⁴, ⁵ – thus limiting the number of bins which can be used to tens or hundreds in practical cases.

We are considering the rate constant \( k_{AB} = \frac{1}{MFPT(A \rightarrow B)} \), from one macrostate (A) to another (B), where A and B could be two non-overlapping conformational states, folded and unfolded states of a protein, or unbound and bound states of a complex. The MFPT is the mean first-passage time for the process.

It has been well established that \( k_{AB} \) can be obtained from WE simulations which have reached SS based on the Hill relation¹, ², ⁶,

\[
k_{AB} = \text{Flux}(A \rightarrow B; SS)
\]

which uses the probability flux into state B – i.e., the probability arriving per unit time. However, complex systems inevitably will require long relaxation times to reach SS, and during the transient relaxation period “direct” WE estimates obtained from the probability flux will typically underestimate the true steady-state flux: see Fig. 1, 2, and 3. Our concern is to extract accurate estimates of the SS flux based on transient WE data – i.e., before steady state has been reached.

In prior work, we showed that a history-dependent non-Markov analysis of WE bins could be used to estimate the steady-state rate based on stationary solution of an appropriate transition matrix¹. Here, we term the non-Markov formulation a “history-augmented Markov state model” (haMSM) to emphasis both its relation to, and difference from, standard MSMs. In a haMSM, one constructs a separate transition matrix for each direction (A-to-B or B-to-A) based on the subset of trajectories which were most recently in macrostate A for the A-to-B direction (or B, for B-to-A). The history is equivalent to the directionality – i.e., the labeling of which macrostate was visited more recently. Once the history has been used to select the trajectory subset, rate estimation proceeds much as it would in a standard MSM calculation. However, haMSMs provide unbiased estimates of the rates (inverse MFPT), whereas standard MSMs are biased for finite-sized bins³. We also emphasize
that we are not computing “implied timescales” based on eigenvalues but the MFPT corresponding to a well-defined physical process.

Here we show that using finer (smaller) bins in a haMSM gives better performance for rate estimation from WE data, and we employ the clustering/binning processes which have been extensively developed in the MSM community. The motivation for using smaller bins is that larger bins are likely to possess internal free energy barriers leading to slower internal relaxation, which in turn will bias the transition probabilities (transition matrix elements) of the haMSM and ultimately the macroscopic transition rate estimate. Procedurally, we use fine “microstates” generated by analyzing uni-directional (A-to-B) data with the pyEMMMA software. These microstates are used to generate a haMSM, whose stationary solution provides the desired rate constant $k_{AB}$. Our primary focus is showing that finer/smaller bins yield more accurate rate estimates, especially when compared to the relatively large bins used to run WE simulations. We note that practical WE simulations are limited in the number of bins which can be used because computing cost scales linearly with the number of bins.

Theory and Procedures

We wish to describe the kinetics of macrostate transitions in a molecular system; we define a source state (A) and a sink state (B), which are arbitrary non-overlapping regions of phase space, and we wish to efficiently calculate the mean first-passage time (MFPT) from A to B based on the Hill relation (1). For the $A \rightarrow B$ transition we only need the $\alpha$ subset of trajectories which were most recently in A; those most recently in B are denoted $\beta$. To this end we employ weighted ensemble simulation for the $\alpha$ subset: trajectories initiated at A which subsequently arrive at the absorbing sink at B are regenerated at A. WE provides an unbiased representation of the $\alpha$ reactive trajectory ensemble.

Here wish to post-analyze the molecular dynamics in a discretized configurational space using a transition matrix $T$. The matrix encodes (conditional) transition probabilities $T_{ij}$ among bins or “microstates” $i$ and $j$ and can bypass the need to sample from the correct steady-state distributions, so long as each microstate internally has the correct steady-state distribution. Markov state models (MSMs) have been used to stitch together many independent simulations to approximate long-timescale processes but do not distinguish the $\alpha$ and $\beta$ trajectory subsets.

The haMSM is a transition matrix formulation containing history labels, namely $\alpha$ or $\beta$, so that transition matrix elements are calculated solely from the corresponding trajectory subsets. Compared to a standard MSM, a haMSM expands the transition matrix formulation with $N \times N$ states into a $2N \times 2N$ labeled rate matrix. Here, we are concerned solely with the $\alpha$ (last in A) ensemble of the source/sink system, which limits our attention to the $N \times N$ transition (sub)matrix

$$T_{ij}^\alpha = P\{X_{t+\tau} = j|X_t = i\}$$

with $\tau$ the lag time of the transition matrix.

The discretized version of the Hill relation (1) then becomes

$$k_{AB} = \text{Flux}(A \rightarrow B; SS) = \sum_{i \in B, j \in B} p_i^{SS} T_{ij}^\alpha \quad \text{(ha MSM)}$$

with $p_i^{SS}$ the steady-state probability of microstate $i$ based on steady-state solution of the transition matrix $T^\alpha$. This non-equilibrium $\alpha$ steady-state breaks detailed balance because of the net flux into the B state originating in the A state. The haMSM formulation (3) yields the correct MFPT independent of the lag-time used to construct the transition matrix. This is a powerful distinction from the traditional MSM because it means that all transitions collected in the trajectory ensemble can be used to train the haMSM. In the atomistic protein folding example we use a 10ps lag time, which should be contrasted with the $\sim$100ns lag times needed for accurate MSMs of molecular systems. In practice, training the haMSM could require significantly less trajectory data than for a standard MSM.

Because the haMSM transition matrix is built from the conditional transition probabilities between states, the steady-state distribution need only be reached within the defined bins, not between them. As noted above, we expect faster relaxation for smaller bins – that is, for haMSMs with more microstates.

Clustering

History augmented Markov State Models (haMSMs) are constructed directly from the WE simulation trajectories using two distinct approaches for comparison purposes. First, haMSMs are built using WE bins as the “microstates”; these were rectilinear bins along the RMSD progress coordinate - see Ref. 14 for details. Secondly, we employ “MSM-style” microstates, which is the new development in this work. Specifically, we apply unsupervised clustering methods to extract a set of microstates representing the configuration space visited by the WE trajectories. For each haMSM based on clustering, the latest-occurring 100,000 structures from the training window are used as input to the clustering. The training is chosen in different ways to address different questions (see below), and the windows used are always indicated in the results section. A new clustering
was performed for each training window considered, so that only the information available inside the training window is ever used for the analysis. K-means clustering with a minimum RMSD metric based on all protein atom Cartesian coordinates (rotationally and translationally minimized) was performed using the pyEMMA software package, requiring ~24 hours of computation over 12 CPUs for each clustering calculation. Given a set number of desired microstates and an initial condition, a k-means clustering is deterministic, but is random given a change in the desired number of microstates or initial bin. To explore the variability of the clustering process, we therefore use sets of similar numbers of microstates – e.g., 9999, 10000, and 10001.

Dimensionality reduction methods appropriate for non-equilibrium ensembles have been recently developed, and will be explored in future work.

**haMSM construction**

Construction of haMSMs proceeds in a highly similar manner to constructing a standard MSM, except via WE weights instead of simple transition counts. All weights from WE simulation are tracked and available for this analysis. Microstates which are not fully connected (e.g., transitions in but not out) were removed from the analysis. To build the transition matrix, the transitioning weight from state $i$ to state $j$, $w_{ij}$, at each iteration was averaged to construct the haMSM transition matrix using $T_{ij} = \frac{1}{\langle w_i \rangle} \sum_{j} w_{ij}$. Here all transitions are used – that is, transitions only observed a single time were included in the analysis. The lag time $\tau$ of the transition matrix was the same as the WE integration time, or 10.0 ps in this case. Extracting the structural transition information at each step requires a large amount of file input/output, and can be slow depending on file access capability/concurrency, but is not computationally demanding.

In using the transition matrix to extract non-equilibrium steady-state kinetics via Eq. (3), we must apply suitable boundary conditions. The source/sink boundary conditions were enforced by employing the exact target state (sink, or macrostate B) and basis state (source, A) definitions used in the WE simulation when constructing the haMSM, with the transition matrix row for the target state enforcing all probability transfers directly back to the basis state. The steady-state flux and mean first-passage time were estimated from the steady-state of the haMSM transition matrix via (3).

**Results: Application to WE data for atomistic protein folding**

Our goal is to validate the use of the haMSM approach to extract steady-state kinetics from transient trajectory data in complex systems. To that end, we apply the haMSM analysis to WE simulations of atomistic protein folding, which are described in detail in Ref. 14.

Our validation has two stages. (i) When the haMSM is trained with a full set of data which approaches steady-state, then the predicted MFPT should be independent of the clustering. (ii) When the haMSM is trained solely on transient data, we seek haMSMs which reliably predict the MFPT based on the more complete training – i.e., the steady-state value.

We first trained the haMSM on a set of 10 folding simulations of the NTL9 protein (1D progress coordinate, friction $\gamma = 5 \text{ ps}^{-1}$). These simulations approach (but do not fully reach) the steady state at around $t_{\text{mol}} \sim 10 \text{ ns}$. Training the haMSM out to $t_{\text{mol}} = 12 \text{ ns}$, there is only a weak dependence of the estimated steady-state flux with the chosen clustering, and clusterings with 50–9000 microstates all produce a steady-state flux well within an order of magnitude of each other, shown in Fig. 1.

In contrast, if we train the haMSM only from the transient regime $t_{\text{mol}} < 5 \text{ ns}$, then there is a strong dependence on the estimated steady-state rate with the number of microstates in the model. With 50 microstates (the original WE bins, which sub-divide the RMSD to the folded state) the haMSM estimated rate is $\sim 10^3 \text{s}^{-1}$. As the number of microstates in the model is increased, the estimated steady-state flux increases up to $\sim 10$,000 microstates the estimated flux approaches the steady-state value $\sim 10^4 \text{s}^{-1}$, as shown in Fig. 2. Because the estimated flux with 9000 and 10000 microstates is almost identical, it seems that the flux has “converged” and will cease to increase with additional microstates. Generally, there is a balance between convergence due to observing enough transitions in the simulation trajectory, and convergence due to having small enough microstates, a subject which will require much more thorough exploration in future work.

These two analyses validate the expectation that as the trajectory training ensemble approaches steady-state, the haMSM estimated steady-state kinetics are independent of clustering, while haMSMs with smaller microstates can estimate the steady-state flux when trained in the transient regime.

In principle, with sufficient training data and sufficiently small microstates, it should be possible to extract steady-state estimates from arbitrarily short (small $t_{\text{mol}}$) trajectory data. In practice, the amount of trajectory will limit one’s ability to leverage transient information. For a set of five WE simulations of NTL9 folding, we therefore examine the time-dependence of haMSM training windows on steady-state rate estimation. Specifically, we examine WE simulations simulated to $t_{\text{mol}} = 5 \text{ ns}$ of NTL9 folding with a 2D progress coordinate, following the setup of Ref. 14. The use of a second progress coordinate leads to more complete direct sampling as many more WE walkers occupy the bins spanning the 2D progress coordinate space. We compare results from this test set to the steady-state kinetics characterized from ten separate, fully independent WE simulations (i.e., excluding the five analyzed) out to $t_{\text{mol}} = 12 \text{ ns}$. We find that steady-state flux estimates from haMSMs with
Figure 1. Validation of haMSM rate estimation based on a large training set. Plotted is the average flux into the target state from 10 WE folding simulations for the NTL9 protein performed with a 1D progress coordinate (black), and predicted steady-state flux from haMSMs (colored lines) trained from data approaching steady-state ($t_{mol} > 10$ns). The number of “microstates” (nC) is indicated for each haMSM. All haMSMs predict similar SS flux when a large training window is used.
Figure 2. Validation of haMSN rate estimation from a small training window within the transient. Plotted is the running-average flux into the target state from 10 WE folding simulations for the NTL9 protein performed with a 1D progress coordinate (black), and predicted steady-state flux from haMSMs (colored lines) trained from data in the transient regime far from steady-state \((t_{mol} < 5\text{ns})\). The number of “microstates” (nC) is indicated for each haMSN. haMSMs with \(\sim 10,000\) microstates can estimate the SS flux despite the short training window in the transient regime.
$10^4$ microstates approach the steady-state value training in window $t_{\text{mol}} \lesssim 1\text{ns}$, less than half the molecular time needed to reach the same direct flux value, shown in Fig. 3.

**Figure 3.** Effect of varying training haMSM training window on rate estimation. Flux into the target state from a training set of 5 WE folding simulations for the NTL9 protein performed with a 2D progress coordinate (1 ns-windowed average in black, individual runs in gray), and predicted steady-state flux from 5 independently calculated haMSMs (red symbols) is plotted at the final iteration of the training window used. In every case, the training window starts at the first iteration. The spread in values reflects variation in the clustering process, and not the variation between the 5 runs in the training set. The validation confidence window (blue) is from a Bayesian bootstrap analysis of the steady-state flux from an independent set of 10 WE simulations out to $t_{\text{mol}} = 12\text{ns}$ examined in 14. All haMSMs were constructed using $\sim 10^4$ microstates.

**Conclusions**

By using fine “bins” or “microstates” generated from pyEMMA clustering, we show that WE trajectory data from the transient (pre-steady-state) regime can be used to construct (unbiased) haMSMs which reliably estimate steady-state kinetics. The fine bins can sidestep potentially long relaxation times internal to large WE bins – that will occur if there are internal barriers. If a haMSM is not used, standard or “direct” rate calculation from WE would otherwise require reaching the steady state which may be impractical with many systems of interest. Therefore, the approach developed here could be of considerable practical importance.

**Acknowledgements**

We are very appreciative of helpful discussions with Barmak Mostofian and Ernesto Suarez. This work was supported by NIH Grant R01GM115805.
References

1. Suárez, E. *et al.* Simultaneous computation of dynamical and equilibrium information using a weighted ensemble of trajectories. *J. Chem. Theory Comput.* **10**, 2658–2667 (2014).

2. Suárez, E., Pratt, A. J., Chong, L. T. & Zuckerman, D. M. Estimating first-passage time distributions from weighted ensemble simulations and non-markovian analyses. *Protein Sci.* **25**, 67–78 (2016).

3. Suárez, E., Adelman, J. L. & Zuckerman, D. M. Accurate estimation of protein folding and unfolding times: beyond markov state models. *J. Chem. Theory Comput.* **12**, 3473–3481 (2016).

4. Huber, G. A. & Kim, S. Weighted-ensemble brownian dynamics simulations for protein association reactions. *Biophys. journal* **70**, 97–110 (1996).

5. Zuckerman, D. M. & Chong, L. T. Weighted ensemble simulation: review of methodology, applications, and software. *Annu. review biophysics* **46**, 43–57 (2017).

6. Hill, T. L. *Free energy transduction and biochemical cycle kinetics* (Springer New York, 1989).

7. Chodera, J. D. & Noé, F. Markov state models of biomolecular conformational dynamics. *Curr. Opin. Struct. Biol.* **25**, 135–144 (2014).

8. Scherer, M. K. *et al.* PyEMMA 2: A Software Package for Estimation, Validation, and Analysis of Markov Models. *J. Chem. Theory Comput.* **11**, 5525–5542, DOI: 10.1021/acs.jctc.5b00743 (2015).

9. Bhatt, D., Zhang, B. W. & Zuckerman, D. M. Steady-state simulations using weighted ensemble path sampling. *The J. Chem. Phys.* **133**, 014110 (2010).

10. Aristoff, D. Analysis and optimization of weighted ensemble sampling. *ESAIM: Math. Model. & Numer. Analysis* **52** (2018).

11. Singhal, N., Snow, C. D. & Pande, V. S. Using path sampling to build better markovian state models: predicting the folding rate and mechanism of a tryptophan zipper beta hairpin. *The J. Chem. Phys.* **121**, 415–425 (2004).

12. Noé, F., Horenko, I., Schütte, C. & Smith, J. C. Hierarchical analysis of conformational dynamics in biomolecules: transition networks of metastable states. *The J. Chem. Phys.* **126**, 04B617 (2007).

13. Voelz, V. A., Bowman, G. R., Beauchamp, K. & Pande, V. S. Molecular simulation of ab initio protein folding for a millisecond folder ntl9 (1-39). *J. Am. Chem. Soc.* **132**, 1526–1528 (2010).

14. Adhikari, U., Mostofian, B., Petersen, A. & Zuckerman, D. M. Computational estimation of ms-sec atomistic folding times. *bioRxiv* 427393 (2018).

15. Paul, F., Wu, H., Vossel, M., de Groot, B. L. & Noé, F. Identification of kinetic order parameters for non-equilibrium dynamics. *arXiv preprint arXiv:1811.12551* (2018).