On-line Optimization of Hamiltonian Replica Exchange Simulations

Justin L. MacCallum[1, *], Mir Ishruna Muniyat[1, †], and Kari Gaalswyk[1, †]

[1]University of Calgary, Calgary, Canada
[*]Corresponding author: justin.maccallum@ucalgary.ca
[†]Both authors contributed equally.

December 2, 2017

Abstract

Replica exchange is a widely used sampling strategy in molecular simulation. While a variety of methods exist for optimizing temperature replica exchange, less is known about how to optimize more general Hamiltonian replica exchange simulations. We present an algorithm for the on-line optimization of both temperature and Hamiltonian replica exchange simulations that draws on techniques from the optimization of deep neural networks in machine learning. We optimize a heuristic-based objective function capturing the efficiency of replica exchange. Our approach is general, and has several desirable properties, including: (1) it makes few assumptions about the system of interest; (2) optimization occurs on-line wihout the requirement of pre-simulation; and (3) it readily generalizes to systems where there are multiple control parameters per replica. We explore some general properties of the algorithm on a simple harmonic oscillator system, and demonstrate its effectiveness on a more complex data-guided protein folding simulation.

Introduction

Replica exchange simulations [1–6] are a widely used sampling technique across a range of disciplines, ranging from molecular simulation [7–12] to Bayesian statistics [13, 14]. The relative ease of implementation of replica exchange has lead to its widespread adoption in the molecular simulation community, with implementations available in many major simulation packages [15–20].

Temperature replica exchange, also known as parallel tempering, samples from a series of flattened or tempered distributions, corresponding to a series of increasing temperatures. In biomolecular systems, N replicas are simulated across a "ladder" of temperatures, often scaled geometrically from say 300 to 500 K. The simulation proceeds by alternating between normal molecular dynamics or Markov-chain Monte Carlo moves, and exchange moves that attempt to exchange configurations between neighboring replicas. Typically, one is interested in the distribution of configurations of the system at the lowest temperatures, while the higher temperatures allow the system to escape local minima, thus potentially enhancing sampling [6]. One limitation of temperature replica exchange is that efficiency gains are not guaranteed if the barriers to sampling are not dependent on temperature, e.g. entropic barriers [21, 22]. Another limitation is that the exchange probability depends on the total potential energy of the system. For large systems, this means that many closely-spaced replicas are needed, which may be computationally prohibitive [8, 21, 23].

Hamiltonian replica exchange is a more general formulation that uses arbitrary perturbations to the Hamiltonian rather than temperature as the basis for exchange [17, 23–25]. These perturbations can be more targeted, making them potentially more efficient than simply modifying the temperature. One motivating example from work in our lab is integrative structural biology [26], where experimental data—potentially sparse, ambiguous, and unreliable—is used to guide folding [27–29] or binding [30–32] through data-driven restraints. In general, it is possible to combine variations in temperature and multiple different perturbations of the Hamiltonian in a single Hamiltonian replica exchange simulation, where the parameters that determine the Hamiltonian and temperature for each replica are called the *control parameters*.

The efficiency of replica exchange is critically dependent on the values of the control parameters. If the exchange probability between neighboring replicas is too low, it creates a "bottleneck" to replica diffusion that can reduce sampling efficiency. In our view, the difficulty of finding efficient values for the control parameters has been a major impediment to the more widespread adoption of Hamiltonian replica exchange simulations.

The aim of this paper is to develop an online optimization strategy for general Hamiltonian replica exchange. There is extensive literature on the optimization of temperature replica exchange [23,33–36], but far less is known about how to optimize the control parameters for Hamiltonian replica exchange [25,37–39]. While there are several algorithms to optimize control parameters, these often suffer from several drawbacks: (1) they often make assumptions, e.g. that the system has constant heat capacity which may not be true; (2) they often require pre-simulation to gather statistics across the range of control parameters; and (3) most importantly, they are designed for a single control parameter, usually temperature, and do not readily generalize the case of multiple control parameters. Overcoming these limitations is critical to achieving efficient,0 flexible, and general Hamiltonian replica exchange schemes.

In this work, we present an algorithm that avoids these drawbacks. Our approach is to: (1) define an objective function that captures what we mean by efficient replica exchange sampling; (2) compute the derivatives of this objective function with respect to the control parameters so that it can be optimized using gradient-based methods; and (3) perform on-line optimization of the control parameters during the simulation using techniques drawn from machine learning. We examine several properties of the algorithm in a simple harmonic oscillator system, and demonstrate the utility of the algorithm on a data-guided protein folding problem similar to those encountered in integrative structural biology.

Theory and Methods

Notation and basic theory of replica exchange

This work considers one-dimensional replica exchange simulations, where a series of replicas are arranged in a "ladder" so that each interior replica has two neighbors. To avoid ambiguity, we use the terminology of *replicas* and *walkers*. Each replica is associated with a Hamiltonian parameterized by a set of control parameters, e.g. temperatures, force constants, etc. A walker is a particular configuration of the system that moves between the replicas through a series of exchange moves, as described below.

The replicas are indexed by $i = 1 \ldots N$, each with a Hamiltonian, $H_i(\boldsymbol{x}, \boldsymbol{\lambda})$, parameterized by the vector of control parameters, $\boldsymbol{\lambda}$. We focus on cases where each control parameter affects only a single replica, and where each replica has the same number and type of control parameters, although neither restriction is required. For a system of N replicas, each with an associated temperature and force constant, $\boldsymbol{\lambda}$ would consist of N temperatures and N force constants, for a total of $2N$ parameters. The control parameters for the top and bottom replica are held fixed.

In general, the Hamiltonians, partition functions, acceptance probabilities, and their averages depend on $\boldsymbol{\lambda}$. For notational simplicity, we suppress the $\boldsymbol{\lambda}$-dependence unless it is necessary for clarity. Throughout, we use the reduced Hamiltonian, $h_i(\boldsymbol{x}) = (RT_i)^{-1} H_i(\boldsymbol{x}, \boldsymbol{\lambda})$, where T_i is the temperature of state i (possibly a control parameter), and R is the gas constant.

A cycle of replica exchange consists of updating the configuration of each walker by performing a series of molecular dynamics or Markov chain Monte Carlo steps, followed by a series of exchange moves that attempt to swap walkers between randomly selected adjacent replicas. In this work, we consider only exchanges between neighboring replicas, although other strategies are possible [40].

The acceptance probability to swap walker $\boldsymbol{x}_i$ at replica i with $\boldsymbol{x}_{i+1}$ at $i+1$ is

$$A_{i,i+1}(\boldsymbol{x}_i, \boldsymbol{x}_{i+1}) = \min\left[1, \exp(\Delta h_{i,i+1})\right], \tag{1}$$

where

$$\Delta h_{i,i+1} = -h_i(\boldsymbol{x}_{i+1}) - h_{i+1}(\boldsymbol{x}_i) + h_i(\boldsymbol{x}_i) + h_{i+1}(\boldsymbol{x}_{i+1}). \tag{2}$$

The average acceptance probability between replicas i and $i+1$ is given by an average over the ensembles of both replicas

$$\langle\!\langle A_{i,i+1} \rangle\!\rangle = \iint A_{i,i+1}(\boldsymbol{x}_i, \boldsymbol{x}_{i+1}) p_i(\boldsymbol{x}_i) p_{i+1}(\boldsymbol{x}_{i+1})\, d\boldsymbol{x}_i\, d\boldsymbol{x}_{i+1}, \tag{3}$$

with probabilities $p_j(\boldsymbol{x}) = Z_j^{-1} e^{-h_j(\boldsymbol{x})}$ and partition functions $Z_j = \int e^{-h_j(\boldsymbol{x})}\, d\boldsymbol{x}$. We use double angle brackets, $\langle\!\langle \cdot \rangle\!\rangle$, to indicate averages that depend on neighboring replicas, and single brackets, $\langle \cdot \rangle$, to indicate averages that depend on only a single replica. By construction, the average acceptance probabilities are symmetric, so that $\langle\!\langle A_{i,i+1} \rangle\!\rangle = \langle\!\langle A_{i+1,i} \rangle\!\rangle$.

Throughout, we make the simplifying assumption that the Monte Carlo or molecular dynamics updates between exchanges produce uncorrelated samples from each replica. This implies that various quantities, including the instantaneous acceptance rates, are independent of the history of the system. In practice, this assumption is violated for many systems of interest—usually replica exchange is used because sampling is slow under the conditions of interest, producing highly correlated samples. Thus, our algorithm does not produce truly optimal solutions. Nevertheless, this is a common assumption in the literature [33, 36, 37, 41], and our algorithm appears to be quite useful in practice, as demonstrated below.

Replica exchange can be optimized using a heuristic

It is surprisingly difficult to quantify exactly what is meant by "optimal sampling" in the context of enhanced sampling algorithms.

Before diving into this question, we note that for replica exchange, there are two broad cases. In the first case, the data obtained at every replica is useful to us, and simulations would be needed at each temperature anyway. Provided that the computational cost of exchange steps is small relative to that of the configuration updates, replica exchange will almost always improve efficiency over independent simulations. In the second case, which is our focus here, we are primarily interested in the results at one replica. In order to gain efficiency, the computational cost associated with the additional replicas must be offset by an increase in sampling efficiency at the replica of interest. Reweighting methods [42, 43], which combine the results from multiple replicas to estimate quantities at a single replica of interest, lie between these two extremes.

Returning to the question of optimal sampling, there are two relevant relaxation or correlation times that govern sampling efficiency: (1) how rapidly the sampled distribution decays towards equilibrium from an arbitrary starting state, which is related to the "burn-in" time at the start of a simulation; and (2) how rapidly the sampling algorithm generates statistically independent samples of some function of interest, $f(\boldsymbol{x})$, which governs the uncertainty in estimates of $\langle f(\boldsymbol{x}) \rangle$ at equilibrium. Both quantities are directly related to the eigendecomposition of the dynamical operator governing the evolution of the system [44]. Recently, algorithms have been developed for identifying slowly relaxing modes and their associated timescales [44, 45], particularly in the context of Markov state models. However, these algorithms are not practical for the optimization of replica exchange simulations because: (1) they require large amounts of simulation data to accurately identify slowly relaxing modes of the system, and (2) it is not straightforward to link changes of the control parameters to perturbations of the eigendecomposition of system dynamics.

Instead, most work aimed at optimizing replica exchange simulations focuses on the heuristic of minimizing the round-trip time [33, 46]—defined as the average time for a walker to transition from the bottom replica to the top, and then back to the bottom again. We adopt the same approach here. Our goal is to sample statistically uncorrelated configurations at the bottom replica. We assume that at the top replica—with the highest temperature, weakest restraints, etc—the walker configuration rapidly decorrelates, effectively "forgetting" its history. Provided the combination of simulation time between exchanges and conditions at the top replica are sufficient to fully decorrelate the walker, each time a walker makes a round-trip, it is guaranteed to produce an independent, uncorrelated sample. The number of round-trips thus sets a lower bound on the number of statistically independent samples produced. Algorithms that optimize round trip time are inherently pessimistic, in that they assume that independent samples of the quantities of interest can only be obtained by completely decorrelating the configuration of the system and, that a round-trip is the only way to achieve said decorrelation.

The harmonic oscillator is a simple model system

In order to better understand on-line optimization of replica exchange, we initially focus on a classical one-dimensional harmonic oscillator, where the reduced Hamiltonian for replica i is given by

$$h_i = (RT_i)^{-1} k x^2, \tag{4}$$

where we fix $k = 1$ kJ mol^{-1} nm^{-2}, and T_i is a control parameter. For this system, we generate in-

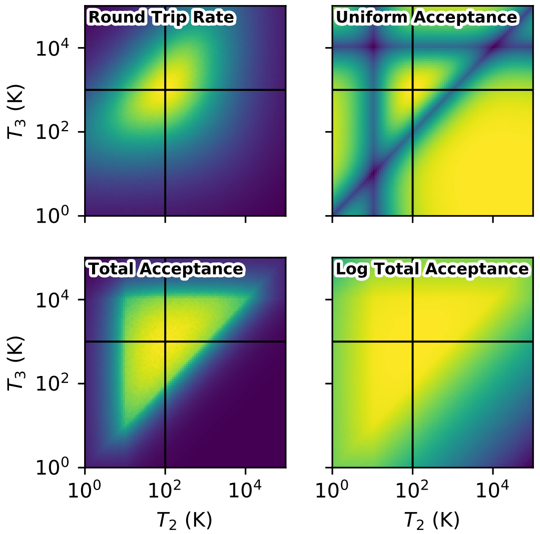

Figure 1: Optimization landscapes for various objective functions as a function of intermediate temperatures T_2 and T_3 with T_1 = 10 K and T_4 = 10,000 K held fixed. Yellow colors indicate more favorable values of the objective function. The system is a classical harmonic oscillator, as described in the text. The black lines indicate the expected optimal solution.

dependent, uncorrelated samples between replica exchange steps.

We consider temperature replica exchange with 4 walkers. The first and last temperatures are fixed at $T_1 = 10$ K and $T_4 = 10{,}000$ K. As a classical harmonic oscillator has constant heat capacity, we expect that an optimal set of temperatures will be geometrically distributed with $T_2 = 100$ K and $T_3 = 1000$ K.

Different objective functions lead to different optimization landscapes

Using the simple harmonic oscillator system described above, we examined four possible objective functions at different combinations of T_2 and T_3 using numerical simulation (Figure 1).

Round trip rate. We first examined the direct use of round-trip rate as an optimization heuristic. There are several approaches to calculating the round-trip rate, but most require either iterative schemes [33] or direct observation of replica flux [46–50]. In either case, it is not immediately clear how to derive an expression for the derivative of the round-trip rate with respect to arbitrary control parameters. Instead, we construct an $N \times N$ transition matrix $\boldsymbol{P}$ for a random walk through replica space. A walker at replica i can jump to $i-1$ with probability $\boldsymbol{P}_{i,i-1} = 0.5 \langle\!\langle A_{i,i-1} \rangle\!\rangle$, jump to $i+1$ with $\boldsymbol{P}_{i,i+1} = 0.5 \langle\!\langle A_{i,i+1} \rangle\!\rangle$, and remain at i with $\boldsymbol{P}_{i,i} = 1 - \boldsymbol{P}_{i,i-1} - \boldsymbol{P}_{i,i+1}$. The mean first passage time matrix $\boldsymbol{M}$ was obtained by the method of Kemeny and Snell [51]

$$\boldsymbol{M} = N\left(\boldsymbol{I} - \boldsymbol{Z} + \boldsymbol{E}\,\mathrm{diag}(\boldsymbol{Z})\right) \tag{5}$$

where $\boldsymbol{I}$ is the $N \times N$ identity matrix, $\boldsymbol{E}$ is the $N \times N$ matrix of all ones, and $\boldsymbol{Z}$ is

$$\boldsymbol{Z} = \left(\boldsymbol{I} - \boldsymbol{P} + N^{-1}\boldsymbol{E}\right)^{-1}.$$

These equations are simplified from those in Kemeny and Snell by the fact that the stationary distribution over replicas is uniform by construction. The objective function is the round-trip rate

$$f_{\mathrm{RT}} = \frac{1}{\boldsymbol{M}_{1,N} + \boldsymbol{M}_{N,1}}, \tag{6}$$

which has an optimum at $(T_2 = 100, T_3 = 1000)$, as expected (Figure 1).

While this objective function directly captures our intent, we observed that it can be unstable when the inverted matrix in the calculation of $\boldsymbol{Z}$ is singular or nearly so. In particular, for reliable inversion we found that several thousand iterations of replica exchange were required in the calculation of $\boldsymbol{P}$, which reduces the rate at which adaptation can occur. Although it may be possible to improve the numerical robustness, we did not pursue this objective function further.

Uniform acceptance. We next considered an objective based on the heuristic that exchange rates should be uniform

$$f_{\mathrm{U}} = -\sum_{i=1}^{N-1} \left(\langle\!\langle A_{i,i+1} \rangle\!\rangle - \bar{A}\right)^2, \tag{7}$$

where $\bar{A}$ is the average acceptance rate across all pairs of adjacent replicas.

There is a local maximum of this objective function at the expected position (Figure 1). However, there are other local maxima that occur in regions of parameter space where all acceptance probabilities are near zero. For some initial control parameters, f_{U} could converge to a good solution, but for other initial control parameters it could converge to a terrible solution. We thus abandoned this objective function.

Total acceptance. Following Shenfeld and coworkers [52], we next considered the total acceptance as an objective function

$$f_{\mathrm{AT}} = \prod_{i=1}^{N-1} \langle\!\langle A_{i,i+1} \rangle\!\rangle^2 . \tag{8}$$

Rather than directly measuring the round-trip rate, this objective function captures the probability of the shortest possible round-trip, where the walker hops from replica 1 to N and back in $2N-2$ consecutive steps.

This objective function has an optimum in the expected location (Figure 1). However, when the total acceptance is near zero, the gradient of the objective function is minuscule (deep purple region of figure), which is not ideal for the gradient-based optimization strategy we pursue here.

Log total acceptance. The poor scaling behavior of the gradients with the overall magnitude of the objective function can be overcome by using the logarithm of Eq. 8 as the objective function

$$f_{\ln \mathrm{AT}} = \sum_{i=1}^{N-1} \ln \langle\!\langle A_{i,i+1} \rangle\!\rangle , \tag{9}$$

where we have dropped an irrelevant factor of 2.

The optimum of this objective is the same as for Eq. 8, as the logarithm is a concave function. There is a clear gradient in the regions of low total acceptance. However, the objective function is now relatively flat around the optimum. Nevertheless, there is still some gradient in this region, and, as described below, the use of adaptive step sizes and learning rate decay allows for successful optimization.

We use Eq. 9 as the objective function for the remainder of this study.

Gradient of the objective function

Our approach is to use gradient-based optimization methods developed for machine learning, which require the gradient of the objective function with respect the the control parameters

$$\nabla f = \sum_{j=1}^{k} \frac{\partial f}{\partial \lambda_j} \widehat{\boldsymbol{\lambda}}_j, \tag{10}$$

where $\widehat{\boldsymbol{\lambda}}_j$ are the basis vectors in the k-dimensional space of control parameters.

For the loss function of Eq. 9, the gradient is

$$\nabla f = \sum_{i=1}^{N-1} \sum_{j=1}^{k} \langle\!\langle A_{i,i+1} \rangle\!\rangle^{-1} \frac{\partial}{\partial \lambda_j} \langle\!\langle A_{i,i+1} \rangle\!\rangle \widehat{\boldsymbol{\lambda}}_j . \tag{11}$$

The derivatives are

$$\begin{aligned} \frac{\partial}{\partial \lambda_j} \langle\!\langle A_{i,i+1} \rangle\!\rangle = \left\langle\!\!\left\langle \frac{\partial}{\partial \lambda_j} A_{i,i+1} \right\rangle\!\!\right\rangle & \\ - \operatorname{Cov}\left(A_{i,i+1}, \frac{\partial}{\partial \lambda_j} h_i \right) & \\ - \operatorname{Cov}\left(A_{i,i+1}, \frac{\partial}{\partial \lambda_j} h_{i+1} \right) &, \end{aligned} \tag{12}$$

where

$$\begin{aligned} &\left\langle\!\!\left\langle \frac{\partial}{\partial \lambda_j} A_{i,i+1} \right\rangle\!\!\right\rangle = \\ &\quad \left\langle\!\!\left\langle \left(\frac{\partial}{\partial \lambda_j} \Delta h_{i,i+1} \right) \exp\left(\Delta h_{i,i+1} \right) \theta(-\Delta h_{i,i+1}) \right\rangle\!\!\right\rangle , \end{aligned}$$

$$\operatorname{Cov}(A, \frac{\partial}{\partial \lambda_j} h) = \frac{\eta}{\eta - 1} \left(\left\langle\!\!\left\langle A \frac{\partial}{\partial \lambda_j} h \right\rangle\!\!\right\rangle - \langle\!\langle A \rangle\!\rangle \langle \frac{\partial}{\partial \lambda_j} h \rangle \right),$$

$\theta(\cdot)$ is the Heaviside step function, and $\Delta h_{i,i+1}$ is defined in Eq. 2. The covariances are corrected for bias at small sample sizes, where η is the number of replica exchange steps included in the computed averages.

The factor of $\langle\!\langle A_{i,i+1} \rangle\!\rangle^{-1}$ in Eq. 11 leads to a singularity when any of the average acceptance probabilities are near zero, which can occur if the initial guess for the control parameters is particularly bad. To mitigate this, we add a small positive constant $\epsilon_1 = 10^{-9}$ to all occurrences of $A_{i,i+1}$ and $\exp(\Delta h_{i,i+1})$, leading to the following modified expression for the gradient

$$\begin{aligned} \nabla f' = \sum_{i=1}^{N-1} \sum_{j=1}^{k} \Bigg(& \frac{1}{\langle\!\langle A_{i,i+1} \rangle\!\rangle + \epsilon_1} \frac{\partial}{\partial \lambda_j} \langle\!\langle A_{i,i+1} \rangle\!\rangle + \\ & \frac{\epsilon_1}{\langle\!\langle A_{i,i+1} \rangle\!\rangle + \epsilon_1} \left\langle\!\!\left\langle \frac{\partial}{\partial \lambda} A_{i,i+1} \right\rangle\!\!\right\rangle \Bigg) \hat{\boldsymbol{\lambda}}_j . \end{aligned} \tag{13}$$

This expression provides an alternative gradient when the acceptance probability is extremely small, making the algorithm more robust.

The objective function is minimized using stochastic optimization

To optimize Eq. 9, we use the Adam algorithm [53], a stochastic optimization method commonly used to train deep neural networks in machine learning. The parameters are updated according to

$$\begin{aligned}
\boldsymbol{m}_t &\leftarrow \beta_1 \boldsymbol{m}_{t-1} + (1-\beta_1)\nabla \boldsymbol{f}'(\boldsymbol{\lambda}_{t-1}) \\
\boldsymbol{\nu}_t &\leftarrow \beta_2 \boldsymbol{\nu}_{t-1} + (1-\beta_2)(\nabla \boldsymbol{f}'(\boldsymbol{\lambda}_{t-1}))^2 \\
\widehat{\boldsymbol{m}}_t &\leftarrow \frac{\boldsymbol{m}_t}{1-\beta_1^t} \\
\widehat{\boldsymbol{\nu}}_t &\leftarrow \frac{\boldsymbol{\nu}_t}{1-\beta_2^t} \\
\boldsymbol{\lambda}_t &\leftarrow \boldsymbol{\lambda}_{t-1} + \boldsymbol{\alpha}_t \frac{\widehat{\boldsymbol{m}}_t}{\sqrt{\widehat{\boldsymbol{\nu}}_t + \epsilon_2}}, \qquad (14)
\end{aligned}$$

where $\boldsymbol{m}_0 = \boldsymbol{\nu}_0 = 0$, t is updated after each adaptation step by $t \leftarrow t+1$, $\boldsymbol{\alpha}_t$ is the learning rate (discussed below), $\boldsymbol{\lambda}_0$ is the initial parameter guess. We set the constants $\epsilon_2 = 10^{-9}$ and $\beta_1 = \beta_2 = 0.9$, as we found that the parameters suggested for machine learning applications, $(\beta_1 = 0.9, \beta_2 = 0.999)$, did not adjust quickly enough to the noise and sudden changes in gradient encountered in our application.

Schedules of learning rate and averaging time allow for efficient optimization

To allow the system to settle into an optimum in the presence of noisy estimates of the gradient, the learning rates are reduced throughout the run by

$$\boldsymbol{\alpha}_t \leftarrow \frac{\boldsymbol{\alpha}_0}{1+\gamma_1 t}, \qquad (15)$$

where γ_1 is the learning rate decay parameter. The initial learning rate vector, $\boldsymbol{\alpha}_0$, is system-dependent. In this work, we set the initial learning rates for all parameters of a given type to be the same, e.g. the temperature parameters for all replicas share the same learning rate. The learning rates for the top and bottom replicas are set to zero.

To compute the gradient, the ensemble averages in Eqs. 11 and 12 are evaluated over a number of replica exchange steps, η, called the averaging time. When η is small, the gradients are noisy, which prevents the parameters from settling into an optimum. Conversely, when η is large, the gradients are less noisy, but this comes at the cost of fewer adaptation steps per unit of simulation time. In this work, we increase the averaging time throughout the simulation by

$$\eta_t \leftarrow \eta_0 \gamma_2^t, \qquad (16)$$

where η_0 is the initial averaging time and γ_2 is the averaging time growth.

After a change in parameters, the system will temporarily be out of equilibrium, causing the average gradient to deviate from the correct value. To mitigate this, we adapt the parameters every $2\eta_t$ steps, where the first η_t steps are discarded and the remainder are used to compute the gradient.

Together, the parameters for the optimization algorithm, $\boldsymbol{\lambda}_0$, $\boldsymbol{\alpha}_0$, η_0, β_1, β_2, γ_1, and γ_2 are referred to as *hyper-parameters*.

Several criteria can be used to terminate optimization

There are several possible criteria to decide when to stop optimization. (1) Optimization may not be terminated at all, with constant updates to parameters throughout the simulation. This may be acceptable in some cases due to the decreasing frequency and magnitude of parameter updates owing to the averaging time growth and learning rate decay. However, for critical applications, even small changes to the parameters and associated non-stationarity of the distributions may be unacceptable. (2) Optimization may be stopped after a fixed number of steps. (3) Optimization may be stopped after the change in parameters between successive updates drops below a particular threshold. (4) Optimization may be stopped once all average exchange rates lie between 7% and 82%, as it is known that under these conditions round-trip rates for optimal and near-optimal solutions can differ by a factor of 2 at most [33]. There are diminishing returns in further optimization compared to increasing the amount of sampling with fixed, if slightly sub-optimal, parameters. In this work, we use option 1, as we are interested in the behavior of the algorithm, rather than the actual results of the simulation. For general use, we recommend either option 3 or 4.

Data-guided protein folding is a more complex model system

For a more complex model system, we examined the data-guided folding of Protein G. This test is designed to mimic the challenges encountered in integrative structural biology applications [26], where the task is to use data that may be sparse, ambiguous, or unreliable [27–29, 54, 55] to guide protein folding.

We focus on the data-guided folding of Protein G using native-centric backbone dihedral and C_α–C_α restraints. This is not meant to be particularly realistic—all of the restraints are derived

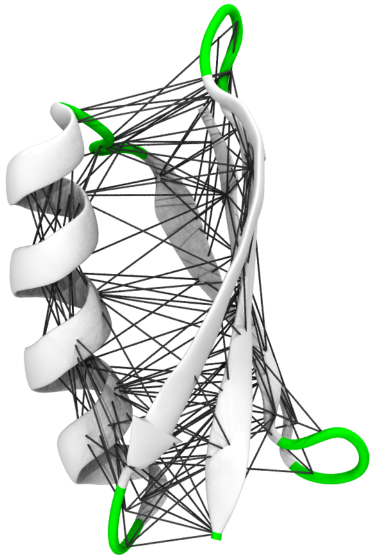

Figure 2: Restraints used during data-guided protein folding simulations. The secondary structures in the white regions were restrained to the correct backbone ϕ/ψ angles, whereas green regions were unrestrained. The black lines indicate C$_\alpha$–C$_\alpha$ distances that were restrained to the experimentally observed values. No restraints were applied to any of the side chain atoms.

from a previously published NMR structure [56], they are accurate, and they are not particularly sparse (Figure 2). Nevertheless, we have found that, even for such simple systems, it is difficult to obtain a good set of parameters for efficient Hamiltonian exchange.

The system was modeled using the ff14SB [57] force field with the OBC implicit solvent model [58] and a grid-based correction to the backbone ϕ/ψ potential to better reproduce secondary structure propensities [59]. Simulations were carried out using the OpenMM library [60, 61], version 7.1. Backbone dihedral angles within secondary structures were weakly restrained to their native values with a force constant of 20 kJ mol^{-1} rad^{-1}. The strength of these restraints was held constant across replicas. Restraints were added for all pairs of C$_\alpha$ with native distances below 10 Å with a force constant that varied across replicas, $k_i \in [e^{-10}, 250]$ kJ mol^{-1} nm^{-2}. The temperature was also varied across replicas, $T_i \in [300, 450]$ K. We used T_i and $\ln k_i$ as control parameters.

We examined two different sets of initial parameters, which we refer to as the *simultaneous* and *separate* cases. All simulations used 16 replicas. For the simultaneous case, T_i and $\ln k_i$ are varied together linearly across the entire range of replicas. For the separate case, T_i varied linearly from 300 to 450 K across replicas 1–8, while $\ln k_i$ varied linearly from -10 to $\ln 250$ kJ mol^{-1} nm^{-2} across replicas 9–16. In all cases, we used an initial guess of hyper-parameters motivated by physical intuition, with $\boldsymbol{\alpha}_0^T = 4$, $\boldsymbol{\alpha}_0^{\ln k} = 0.4$, $\eta_0 = 16$, $\gamma_1 = 1/100$, and $\gamma_2 = 2^{1/100}$.

Results

Learning rate and averaging time affect optimization

Figure 3 shows the influence of $\boldsymbol{\alpha}$ and η on optimization for the simple harmonic oscillator system. In this experiment, the learning rate and averaging time are held constant in each simulation with $\gamma_1 = 0$ and $\gamma_2 = 1$.

The initial transient decay from the initial parameters is more rapid at higher learning rates. However, there is also more noise and the long-term fluctuations are larger. Low learning rates have more stable long-term behavior, but the initial transient response is much slower.

Similar observations can be made for the averaging time. When the averaging time is short, the results are noisy and the system does not settle into a well-defined optimum. The presence of noise in the gradients is akin to temperature in simulated annealing, and it has been suggested that adding noise to the gradients can be helpful for escaping local minima in machine learning [62, 63]. When the averaging time is long, the long-term fluctuations are smaller, but it takes substantially longer to reach the optimal value.

It is evident that results systematically deviate from the optimum when η is small (top row of Figure 3; bottom row of Figure 4). This occurs because the estimator of the gradients in Eq. 11 is biased for small η, but appears to be asymptotically unbiased.

Figure 4 shows the influence of the learning rate decay (γ_1) and averaging time growth (γ_2) on optimization for the simple harmonic oscillator system. Extreme values of either parameter result in large fluctuations or slow decay towards the optimum. However, a variety of combinations result in reasonable behaviour.

Overall, the ideal hyper-parameters are system specific. In this work, we set these hyper-parameters guiding by biophysical intuition. In

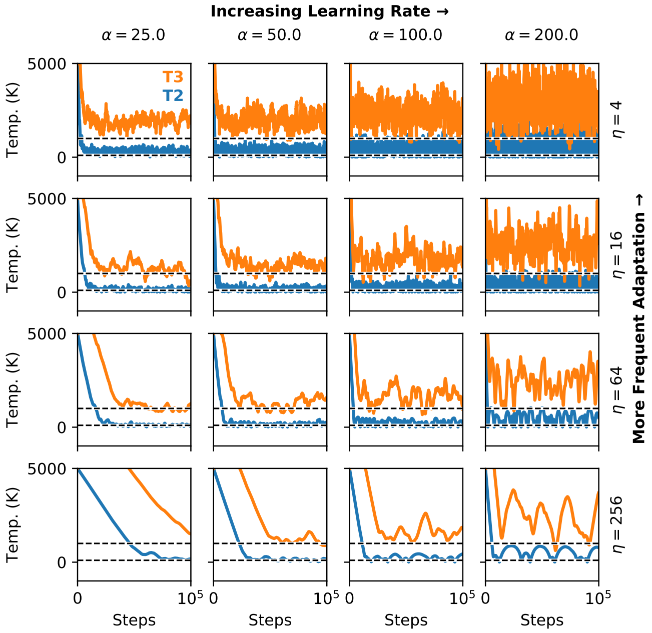

Figure 3: Effect of learning rate (α) and averaging time (η) on the optimization of temperature replica exchange for a simple harmonic oscillator. T_2 and T_3 are initially set to 5000K, while $T_1 = 10$ and $T_4 = 10,000$ K are fixed. The learning rate and averaging time are held constant with $\gamma_1 = 0$ and $\gamma_2 = 1$. The dashed lines indicate the expected solution.

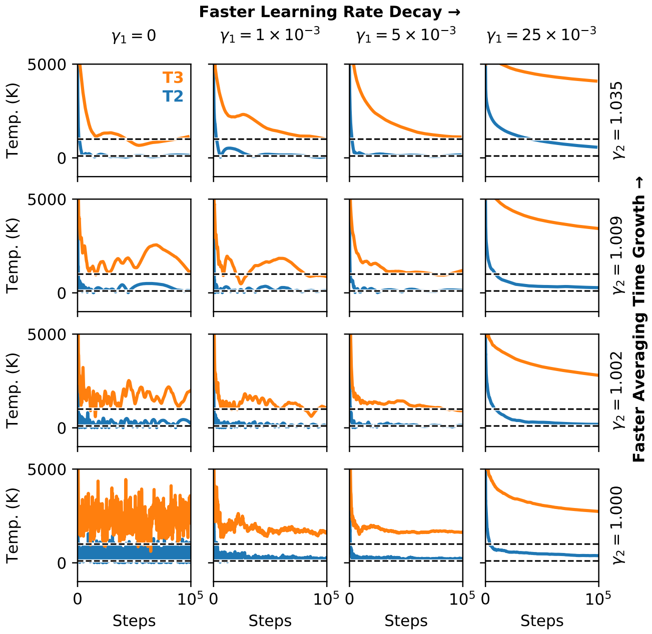

Figure 4: Effect of learning rate decay (γ_1) and averaging time grown (γ_2) on the optimization of temperature replica exchange for a simple harmonic oscillator. T_2 and T_3 are initially set to 5000K, while $T_1 = 10$ and $T_4 = 10,000$ K are fixed. The initial learning rate and averaging time are set to $\boldsymbol{\alpha}_0 = 100$ and $\eta_0 = 4$, respectively. The dashed lines indicate the expected solution.

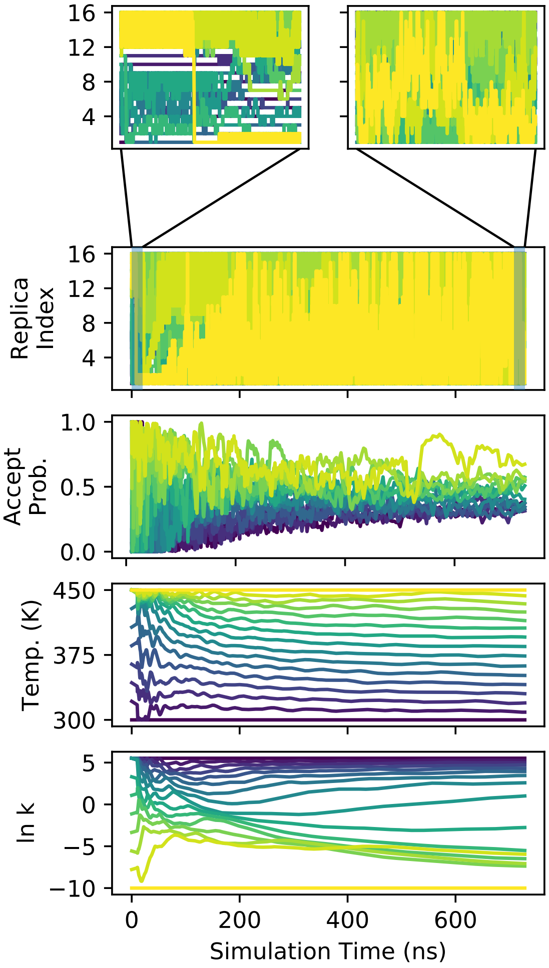

Figure 5: Visualization of replica exchange adaptation for data-guided protein folding starting from the *separate* initial conditions. Colors indicate replica index, from 1 (purple) to 16 (yellow). The top panels show detailed time traces over the first and last 10 ns of the simulation.

our experience, it is straightforward to find values that work sufficiently well. It is also possible to use a grid search or randomized hyper-parameter optimization. We expect that hyper-parameters should generally be transferable between similar systems.

On-line adaptation of replica exchange can optimize data-guided protein folding simulations

Our algorithm was able to successfully optimize replica exchange data-guided protein folding simulations. For the *separate* initial parameters, there were large regions of the replica exchange ladder where the initial exchange rate was poor (Figure 5, upper-left). During the course of the simulation, exchange rates became more uniform and overall diffusivity of replicas was improved dramatically (Figure 5, upper right). We observed similar behavior for the *simultaneous* initial parameters (not shown). Although the *separate* initial parameters varied the temperature only over replicas 1–8, the optimized parameters vary the temperature over the full 16 replicas with the temperatures "spreading out" over the course of the optimization (Figure 6). Similarly, although the changes to $\ln k$ were initially confined to the top 8 replicas, after optimization, the changes are distributed more evenly across replicas.

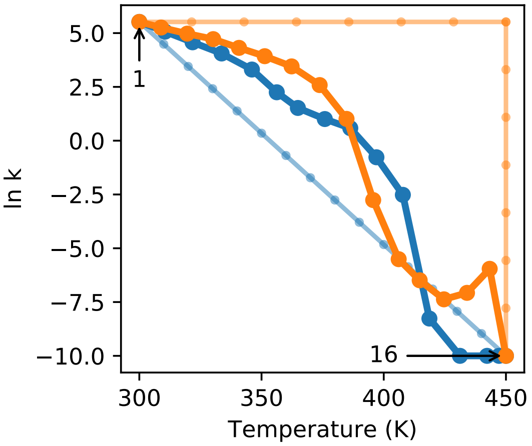

Figure 6: Visualization of initial (faded colors) and final (dark colors) adapted parameters for data-guided protein folding simulations for the *separate* (orange) and *simultaneous* (blue) initial parameters. Numbers indicate the first (1) and last (16) replicas.

The parameters, particularly $\ln k$, are still slowly drifting after 700 ns (Figure 5), which may indicate that the optimal solution has not yet been reached. These results are from a single guess for the hyper-parameters of the algorithm, guided primarily by our physical intuition, and it is possible that a more extensive hyper-parameter search could lead to more rapid optimization. However, as noted previously, acceptance rates can only improve by at most a factor of 2 once all acceptance probabilities fall between 7& and 82% [33]. In practice, the most efficient strategy would be to terminate optimization once this occurs, which in this case would be after ~ 200 ns, resulting in 200

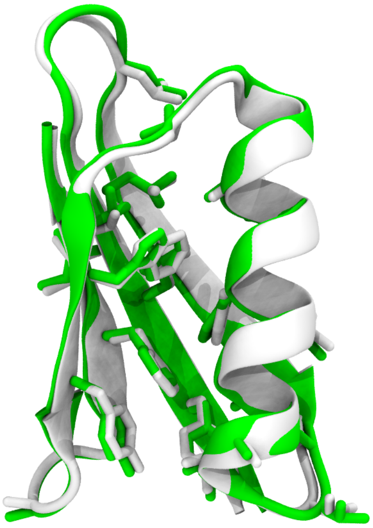

Figure 7: Final frame of data-guided protein folding simulation (green) compared to the reference structure (white; PDB identifier 3gb1). The core side chains are in near-perfect superposition.

ns of optimization and 500 ns of production simulation.

Figure 6 compares the optimized solutions starting from either the *separate* or *simultaneous* initial parameters. Although the initial parameters differ dramatically, the final optimized solutions are quite similar. In both cases, the optimized temperature change occurs over the full range of replicas. The behavior of $\ln k$ is slightly more complex, with two different regimes evident. For the *separate* initial parameters, there is a "spike" in $\ln k$ over the last few replicas, however this occurs when the force constant is very small, $\ln k = -5 \Rightarrow k = 7 \times 10^{-3}$, and thus likely has little influence on the simulation.

As would be expected from the native-centric restraints used to guide folding, the structures generated by data-guided folding are highly accurate (Figure 7). For both the *separate* and *simultaneous* initial conditions, the backbone root mean square deviation is below 1 Å for nearly all structures sampled after the first ~ 200 ns (not shown). More impressive is the near-perfect superposition of side chain conformations, which were completely unrestrained. Through the simulation, walkers are unfolded by round-trips to high temperatures. The side chains are able to correctly repack during the data-guided refolding of the backbone. This is a demonstration of the power of physics-based integrative structural biology, where the use of data to guide certain features of the structure leads to accurate predictions, even for features that are unrestrained by data.

Discussion and Conclusions

We have presented a general framework for the online optimization of Hamiltonian replica exchange simulations. Our algorithm is able to successfully optimize replica exchange for a restrained protein system, reminiscent of those encountered in integrative structural biology.

Our algorithm has three unique properties compared to existing approaches [23, 25, 33–39]. First, it is an on-line algorithm. Given a suitable criterion to terminate optimization, it is possible to both optimize replica exchange and collect equilibrium statistics from a single simulation. Second, our algorithm straightforwardly handles arbitrary modifications of the Hamiltonian, extending beyond temperature replica exchange. This should help to enable new Hamiltonian replica exchange sampling schemes, which have great promise, but are often difficult to tune in practice. Third, our algorithm readily handles multiple control parameters per replica. This is a major advantage, as it allows for multiple perturbations of the Hamiltonian to be combined. In the case of integrative structural biology, we might combine temperature replica exchange with multiple types of data-driven restraints. The optimal set of control parameters is likely system- and data-dependent and difficult to predict, making such simulations difficult without a strategy for optimization.

Our algorithm does have some potential shortcomings. First, optimization requires the specification of several hyper-parameters, and inappropriate choices of these parameters may lead to poor optimization. Nevertheless, we have found it straightforward to find reasonably well-performing hyper-parameters through simple trial and error.

A second major shortcoming is that our algorithm assumes that the sampling between replica exchange steps leads to uncorrelated configurations. In practice, this is almost never the case, as replica exchange is primarily used when relaxation is slow and sampling is difficult—leading to highly correlated configurations. This means that, although our algorithm is able to generate reasonable control parameter values starting from an arbitrary initial guess, it may be possible to identify parameters that produce even better sampling. Indeed, there are several papers— based on variations of a single algorithm—that demonstrate this [34, 35, 46–50, 64]. However, these methods face several challenges of their own. (1) These algorithms inherently depend on global properties of the system, where each walker must visit each replica multiple times before optimization can proceed. In cases where mixing is slow, this can make reliable optimization prohibitively expensive. (2) Anecdotally, we have found that these algorithms can be difficult to converge, as the optimal solution depends on a delicate balance between overall acceptance rate and more closely spacing replicas in regions of parameter space where relaxation is slow. (3) These approaches do not address the important case of multiple control parameters per replica, which is potentially a major limitation for Hamiltonian replica exchange. We believe that the ideas presented in our work can be extended to account for slow relaxation, and work along these lines is underway in our laboratory.

Hamiltonian replica exchange is a powerful sampling strategy that has yet to reach its full potential, in part due to the difficulty of finding appropriate paths through the space of control parameters. The algorithm presented here provides a robust way to optimize these control parameters, which leads to more efficient simulations. Our approach should help enable the development of new Hamiltonian replica exchange schemes.

Acknowledgements

This work is supported by funding from the Natural Sciences and Engineering Research Council of Canada and the Canada Foundation for Innovation, and by computational resources from Compute Canada. JLM is a Tier 2 Canada Research Chair. JLM thanks David Sivak and Steven Large for fruitful discussions.

References

[1] Robert H Swendsen and Jian-Sheng Wang. Replica monte carlo simulation of spin-glasses. *Physical Review Letters*, 57(21):2607, 1986.

[2] E Marinari and G Parisi. Simulated tempering: A New Monte Carlo Scheme. *EPL*, 19(6):451–458, 1992.

[3] Charles J Geyer and Elizabeth A Thompson. Annealing markov chain monte carlo with applications to ancestral inference. *Journal of the American Statistical Association*, 90(431):909–920, 1995.

[4] Koji Hukushima and Koji Nemoto. Exchange monte carlo method and application to spin glass simulations. *Journal of the Physical Society of Japan*, 65(6):1604–1608, 1996.

[5] Yuji Sugita and Yuko Okamoto. Replica-exchange molecular dynamics method for protein folding. *Chemical Physics Letters*, 314(1):141–151, 1999.

[6] David J Earl and Michael W Deem. Parallel tempering: Theory, applications, and new perspectives. *Physical Chemistry Chemical Physics*, 7(23):3910–3916, 2005.

[7] Dietmar Paschek and Angel E García. Reversible temperature and pressure denaturation of a protein fragment: a replica exchange molecular dynamics simulation study. *Physical Review Letters*, 93(23):238105, 2004.

[8] Nitin Rathore, Manan Chopra, and Juan J de Pablo. Optimal allocation of replicas in parallel tempering simulations. *The Journal of Chemical Physics*, 122(2):024111, 2005.

[9] Daniel R Roe, Viktor Hornak, and Carlos Simmerling. Folding cooperativity in a three-stranded β-sheet model. *Journal of Molecular Biology*, 352(2):370–381, 2005.

[10] Angel E García and José N Onuchic. Folding a protein in a computer: an atomic description of the folding/unfolding of protein a. *Proceedings of the National Academy of Sciences*, 100(24):13898–13903, 2003.

[11] Ruhong Zhou, Bruce J Berne, and Robert Germain. The free energy landscape for β hairpin folding in explicit water. *Proceedings of the National Academy of Sciences*, 98(26):14931–14936, 2001.

[12] Karissa Y Sanbonmatsu and Angel E García. Structure of met-enkephalin in explicit aqueous solution using replica exchange molecular dynamics. *Proteins: Structure, Function, and Bioinformatics*, 46(2):225–234, 2002.

[13] WD Vousden, Will M Farr, and Ilya Mandel. Dynamic temperature selection for parallel tempering in markov chain monte carlo simulations. *Monthly Notices of the Royal Astronomical Society*, 455(2):1919–1937, 2015.

[14] Michael Habeck, Michael Nilges, and Wolfgang Rieping. Replica-exchange monte carlo scheme for bayesian data analysis. *Physical Review Letters*, 94(1):018105, 2005.

[15] James C. Phillips, Rosemary Braun, Wei Wang, James Gumbart, Emad Tajkhorshid, Elizabeth Villa, Christophe Chipot, Robert D. Skeel, Laxmikant Kalé, and Klaus Schulten. Scalable molecular dynamics with NAMD. *Journal of Computational Chemistry*, 26(16):1781–1802, 2005.

[16] Mark James Abraham, Teemu Murtola, Roland Schulz, Szilárd Páll, Jeremy C. Smith, Berk Hess, and Erik Lindahl. GROMACS: High performance molecular simulations through multi-level parallelism from laptops to supercomputers. *SoftwareX*, 1–2:19–25, 2015.

[17] Giovanni Bussi. Hamiltonian replica exchange in GROMACS: a flexible implementation. *Molecular Physics*, 112(3):379–384, 2014.

[18] David A. Case, Thomas E. Cheatham, Tom Darden, Holger Gohlke, Ray Luo, Kenneth M. Merz, Alexey Onufriev, Carlos Simmerling, Bing Wang, and Robert J. Woods. The amber biomolecular simulation programs. *Journal of Computational Chemistry*, 26(16):1668–1688, 2005.

[19] Steve Plimpton. Fast parallel algorithms for short-range molecular dynamics. *Journal of Computational Physics*, 117(1):1–19, 1995.

[20] B.R. Brooks, C.L. Brooks, A.D. MacKerell, L. Nilsson, R.J. Petrella, B. Roux, Y. Won, G. Archontis, C. Bartels, S. Boresch, A. Caflisch, L. Caves, Q. Cui, A.R. Dinner, M. Feig, S. Fischer, J. Gao, M. Hodoscek, W. Im, K. Kuczera, T. Lazaridis, J. Ma, V. Ovchinnikov, E. Paci, R.W. Pastor, C.B. Post, J.Z. Pu, M. Schaefer, B. Tidor, R. M. Venable, H. L. Woodcock, X. Wu, W. Yang, D.M. York, and M. Karplus. CHARMM: The biomolecular simulation program. *Journal of Computational Chemistry*, 30(10):1545–1614, 2009.

[21] Jon Machta. Strengths and weaknesses of parallel tempering. *Physical Review E*, 80(5):056706, 2009.

[22] Hugh Nymeyer. How efficient is replica exchange molecular dynamics? an analytic approach. *Journal of Chemical Theory and Computation*, 4(4):626–636, 2008.

[23] Pu Liu, Byungchan Kim, Richard A Friesner, and BJ Berne. Replica exchange with solute tempering: A method for sampling biological systems in explicit water. *Proceedings of the National Academy of Sciences of the United States of America*, 102(39):13749–13754, 2005.

[24] Hiroaki Fukunishi, Osamu Watanabe, and Shoji Takada. On the hamiltonian replica exchange method for efficient sampling of biomolecular systems: Application to protein structure prediction. *The Journal of Chemical Physics*, 116(20):9058–9067, 2002.

[25] Jozef Hritz and Chris Oostenbrink. Hamiltonian replica exchange molecular dynamics using soft-core interactions. *The Journal of Chemical Physics*, 128(14):144121, 2008.

[26] Andrew B. Ward, Andrej Sali, and Ian A. Wilson. Integrative structural biology. *Science*, 339(6122):913–915, 2013.

[27] Justin L. MacCallum, Alberto Perez, and Ken A. Dill. Determining protein structures by combining semireliable data with atomistic physical models by bayesian inference. *Proceedings of the National Academy of Sciences*, 112(22):6985–6990, 2015.

[28] Alberto Perez, Justin L. MacCallum, and Ken A. Dill. Accelerating molecular simulations of proteins using bayesian inference on weak information. *Proceedings of the National Academy of Sciences*, 112(38):11846–11851, 2015.

[29] Alberto Perez, Joseph A. Morrone, Emiliano Brini, Justin L. MacCallum, and Ken A. Dill. Blind protein structure prediction using accelerated free-energy simulations. *Science Advances*, 2(11):e1601274, 2016.

[30] Joseph A. Morrone, Alberto Perez, Qiaolin Deng, Sookhee N. Ha, M. Katharine Holloway, Tomi K. Sawyer, Bradley S. Sherborne, Frank K. Brown, and Ken A. Dill. Molecular simulations identify binding poses and approximate affinities of stapled α-helical peptides to MDM2 and MDMX. *Journal of Chemical Theory and Computation*, 13(2):863–869, 2017.

[31] Joseph A. Morrone, Alberto Perez, Justin MacCallum, and Ken A. Dill. Computed binding of peptides to proteins with MELD-accelerated molecular dynamics. *Journal of Chemical Theory and Computation*, 13(2):870–876, 2017.

[32] Biswaranjan Mohanty, Martin L Williams, Bradley C Doak, Mansha Vazirani, Olga Ilyichova, Geqing Wang, Wolfgang Bermel, Jamie S Simpson, David K Chalmers, Glenn F King, et al. Determination of ligand binding modes in weak protein–ligand complexes using sparse nmr data. *Journal of Biomolecular NMR*, 66(3):195–208, 2016.

[33] Cristian Predescu, Mihaela Predescu, and Cristian V. Ciobanu. On the efficiency of exchange in parallel tempering monte carlo simulations. *The Journal of Physical Chemistry B*, 109(9):4189–4196, 2005.

[34] Walter Nadler and Ulrich HE Hansmann. Optimized explicit-solvent replica exchange molecular dynamics from scratch. *The Journal of Physical Chemistry B*, 112(34):10386–10387, 2008.

[35] Walter Nadler, Jan H Meinke, and Ulrich HE Hansmann. Folding proteins by first-passage-times-optimized replica exchange. *Physical Review E*, 78(6):061905, 2008.

[36] Meher K Prakash, Alessandro Barducci, and Michele Parrinello. Replica Temperatures for Uniform Exchange and Efficient Roundtrip Times in Explicit Solvent Parallel Tempering Simulations. *Journal of Chemical Theory and Computation*, 7(7):2025–2027, 2011.

[37] Danial Sabri Dashti and Adrian E Roitberg. Optimization of umbrella sampling replica exchange molecular dynamics by replica positioning. *Journal of Chemical Theory and Computation*, 9(11):4692–4699, 2013.

[38] Jozef Hritz and Chris Oostenbrink. Optimization of replica exchange molecular dynamics by fast mimicking. *The Journal of Chemical Physics*, 127(20):204104, 2007.

[39] Roman Affentranger, Ivano Tavernelli, and Ernesto E Di Iorio. A novel hamiltonian replica exchange md protocol to enhance protein conformational space sampling. *Journal of Chemical Theory and Computation*, 2(2):217–228, 2006.

[40] John D. Chodera and Michael R. Shirts. Replica exchange and expanded ensemble simulations as gibbs sampling: Simple improvements for enhanced mixing. *The Journal of Chemical Physics*, 135(19):194110, 2011.

[41] Weihong Zhang and Jianhan Chen. Efficiency of Adaptive Temperature-Based Replica Exchange for Sampling Large-Scale Protein Conformational Transitions. *Journal of Chemical Theory and Computation*, 9(6):2849–2856, 2013.

[42] Ayori Mitsutake, Yuji Sugita, and Yuko Okamoto. Replica-exchange multicanonical and multicanonical replica-exchange monte carlo simulations of peptides. i. formulation and benchmark test. *The Journal of Chemical Physics*, 118(14):6664–6675, 2003.

[43] Michael R. Shirts and John D. Chodera. Statistically optimal analysis of samples from multiple equilibrium states. *The Journal of Chemical Physics*, 129(12):124105, 2008.

[44] Guillermo Pérez-Hernández, Fabian Paul, Toni Giorgino, Gianni De Fabritiis, and Frank Noé. Identification of slow molecular order parameters for markov model construction. *The Journal of Chemical Physics*, 139(1):015102, 2013.

[45] Christian R. Schwantes and Vijay S. Pande. Improvements in markov state model construction reveal many non-native interactions in the folding of NTL9. *Journal of Chemical Theory and Computation*, 9(4):2000–2009, 2013.

[46] Simon Trebst, David A. Huse, and Matthias Troyer. Optimizing the ensemble for equilibration in broad-histogram monte carlo simulations. *Physical Review E*, 70(4):046701, 2004.

[47] Simon Trebst, Matthias Troyer, and Ulrich H. E. Hansmann. Optimized parallel tempering simulations of proteins. *The Journal of Chemical Physics*, 124(17):174903, 2006.

[48] Helmut G. Katzgraber, Simon Trebst, David A. Huse, and Matthias Troyer. Feedback-optimized parallel tempering monte carlo. *Journal of Statistical Mechanics: Theory and Experiment*, 2006(3):P03018, 2006.

[49] Walter Nadler and Ulrich H. E. Hansmann. On dynamics and optimal number of replicas in parallel tempering simulations. *Physical Review E*, 76(6):065701, 2007.

[50] Dominik Sidler, Michael Cristòfol-Clough, and Sereina Riniker. Efficient round-trip time optimization for replica-exchange enveloping distribution sampling (RE-EDS). *Journal of Chemical Theory and Computation*, 13(6):3020–3030, 2017.

[51] John G. Kemeny and J. Laurie Snell. *Finite Markov Chains*. Springer-Verlag, New York, 1976.

[52] Daniel K. Shenfeld, Huafeng Xu, Michael P. Eastwood, Ron O. Dror, and David E. Shaw. Minimizing thermodynamic length to select intermediate states for free-energy calculations and replica-exchange simulations. *Physical Review E*, 80(4):046705, 2009.

[53] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[54] Mark Berjanskii, David Arndt, Yongjie Liang, and David S. Wishart. A robust algorithm for optimizing protein structures with NMR chemical shifts. *Journal of Biomolecular NMR*, 63(3):255–264, 2015.

[55] Hajime Tamaki, Ayako Egawa, Kouki Kido, Tomoshi Kameda, Masakatsu Kamiya, Takashi Kikukawa, Tomoyasu Aizawa, Toshimichi Fujiwara, and Makoto Demura. Structure determination of uniformly 13c, 15n labeled protein using qualitative distance restraints from MAS solid-state 13c-NMR observed paramagnetic relaxation enhancement. *Journal of Biomolecular NMR*, 64(1):87–101, 2016.

[56] John Kuszewski, Angela M. Gronenborn, and G. Marius Clore. Improving the packing and accuracy of NMR structures with a pseudopotential for the radius of gyration. *Journal of the American Chemical Society*, 121(10):2337–2338, 1999.

[57] James A. Maier, Carmenza Martinez, Koushik Kasavajhala, Lauren Wickstrom, Kevin E. Hauser, and Carlos Simmerling. ff14sb: Improving the accuracy of protein side chain and backbone parameters from ff99sb. *Journal of Chemical Theory and Computation*, 11(8):3696–3713, 2015.

[58] Alexey Onufriev, Donald Bashford, and David A. Case. Exploring protein native states and large-scale conformational changes with a modified generalized born model. *Proteins: Structure, Function, and Bioinformatics*, 55(2):383–394, 2004.

[59] Alberto Perez, Justin L. MacCallum, Emiliano Brini, Carlos Simmerling, and Ken A. Dill. A grid-based backbone correction to the ff12sb protein force field for implicit-solvent simulations. *Journal of chemical theory and computation*, 11(10):4770–4779, 2015.

[60] Peter Eastman, Mark S. Friedrichs, John D. Chodera, Randall J. Radmer, Christopher M. Bruns, Joy P. Ku, Kyle A. Beauchamp, Thomas J. Lane, Lee-Ping Wang, Diwakar Shukla, Tony Tye, Mike Houston, Timo Stich, Christoph Klein, Michael R. Shirts, and Vijay S. Pande. OpenMM 4: A reusable, extensible, hardware independent library for high performance molecular simulation. *Journal of Chemical Theory and Computation*, 9(1):461–469, 2013.

[61] Peter Eastman, Jason Swails, John D. Chodera, Robert T. McGibbon, Yutong Zhao, Kyle A. Beauchamp, Lee-Ping Wang, Andrew C. Simmonett, Matthew P. Harrigan, Chaya D. Stern, Rafal P. Wiewiora, Bernard R. Brooks, and Vijay S. Pande. OpenMM 7: Rapid development of high performance algorithms for molecular dynamics. *PLOS Computational Biology*, 13(7):e1005659, 2017.

[62] Maxim Raginsky, Alexander Rakhlin, and Matus Telgarsky. Non-convex learning via stochastic gradient langevin dynamics: a nonasymptotic analysis. *arXiv preprint arXiv:1702.03849*, 2017.

[63] Arvind Neelakantan, Luke Vilnis, Quoc V. Le, Ilya Sutskever, Lukasz Kaiser, Karol Kurach, and James Martens. Adding gradient noise improves learning for very deep networks. *arXiv preprint arXiv:1511.06807*, 2015.

[64] Walter Nadler and Ulrich H. E. Hansmann. Generalized ensemble and tempering simulations: A unified view. *Physical Review E*, 75(2):026109, 2007.