

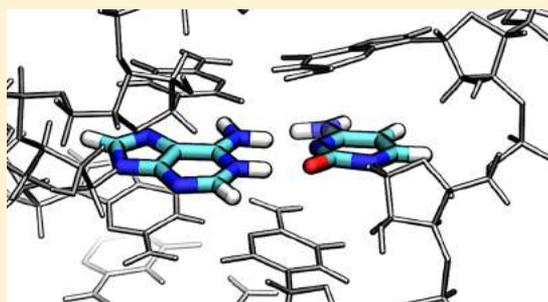
A GPU-Accelerated Parameter Interpolation Thermodynamic Integration Free Energy Method

Timothy J. Giese and Darrin M. York*¹

Laboratory for Biomolecular Simulation Research, Center for Integrative Proteomics Research, and Department of Chemistry and Chemical Biology, Rutgers University, Piscataway, New Jersey 08854-8087, United States

Supporting Information

ABSTRACT: There has been a resurgence of interest in free energy methods motivated by the performance enhancements offered by molecular dynamics (MD) software written for specialized hardware, such as graphics processing units (GPUs). In this work, we exploit the properties of a parameter-interpolated thermodynamic integration (PI-TI) method to connect states by their molecular mechanical (MM) parameter values. This pathway is shown to be better behaved for $\text{Mg}^{2+} \rightarrow \text{Ca}^{2+}$ transformations than traditional linear alchemical pathways (with and without soft-core potentials). The PI-TI method has the practical advantage that no modification of the MD code is required to propagate the dynamics, and unlike with linear alchemical mixing, only one electrostatic evaluation is needed (e.g., single call to particle-mesh Ewald) leading to better performance. In the case of AMBER, this enables all the performance benefits of GPU-acceleration to be realized, in addition to unlocking the full spectrum of features available within the MD software, such as Hamiltonian replica exchange (HREM). The TI derivative evaluation can be accomplished efficiently in a post-processing step by reanalyzing the statistically independent trajectory frames in parallel for high throughput. We also show how one can evaluate the particle mesh Ewald contribution to the TI derivative evaluation without needing to perform two reciprocal space calculations. We apply the PI-TI method with HREM on GPUs in AMBER to predict pK_a values in double stranded RNA molecules and make comparison with experiments. Convergence to under 0.25 units for these systems required 100 ns or more of sampling per window and coupling of windows with HREM. We find that MM charges derived from *ab initio* QM/MM fragment calculations improve the agreement between calculation and experimental results.



1. INTRODUCTION

Free energy calculations are powerful tools for the prediction of a wide range of molecular properties.^{1–4} Alchemical free energy methods are free energy techniques used to determine path-independent properties associated with the relative free energy values between two well-defined thermodynamic states. The path-independent nature of the formalism affords the opportunity to tailor the connection between the states to make the method more amenable to practical computation.^{5,6}

Two of the commonly applied alchemical free energy methods are free energy perturbation (FEP) and thermodynamic integration (TI). The FEP method⁷ can suffer from hysteresis and slow statistical convergence due to inadequate overlap between sampled intermediate states. Advanced analysis techniques such as the Bennett's acceptance ratio⁸ (BAR) or multi-state generalization (MBAR) methods⁹ can help to improve the reliability of the calculated free energy, and they have become essential tools for free energy prediction. The TI method numerically integrates ensemble-averaged derivatives of the potential energy with respect to a perturbing parameter that couples the end-state Hamiltonians.^{10–14} Thermodynamic integration has been demonstrated to be a comparable tool to BAR/MBAR that is also widely used for free energy prediction.¹⁵ The intention of the present manuscript is not to advocate the use of one method over the other

(FEP with BAR/MBAR versus TI). On the contrary, we find it useful to analyze our data using *both* classes of methods to help gauge the reliability of our calculations and derive meaningful error estimates. Ideally, we would like to perform MBAR and TI analysis from simulations performed on Graphics Processing Unit (GPU) accelerated molecular dynamics (MD) software. Furthermore, we would like to improve the extent and consistency of sampling by coupling the intermediate alchemical states via the Hamiltonian replica exchange method (HREM).^{16–19}

Although GPU-accelerated MD has emerged in a wide array of software packages, including NAMD,²⁰ ACEMD,²¹ AMBER,^{22–24} OpenMM,²⁵ GROMACS,^{26,27} and CHARMM,²⁸ the incorporation of free energy methods into the GPU-enabled software has only very recently emerged in a few simulation codes. The result is that many MD software packages do not currently support GPU-accelerated TI, and fewer still have access to the full array of enhanced sampling functionality available to the MD engine. Recent GPU-enabled free energy simulation developments include an implementation of FEP with DESMOND,²⁹ a dual-topology TI method in PMEMD,³⁰ an extensible Python framework for alchemical free energy calculations with YANK,³¹ and the porting

Received: November 20, 2017

Published: January 22, 2018

of constant-pH methods to GPUs.^{28,32} The GROMOS and GROMACS programs can utilize GPUs for those parts of a system not undergoing an alchemical transformation,²⁷ and the OpenMM program can utilize GPUs to perform BAR and MBAR free energy calculations.^{33–36} Other programs, like CHARMM, contain interfaces to OpenMM.²⁸ The incompatibility between CHARMM's free energy module and the other internal software features motivated Boresch and Bruckner to explore alchemical pathways that can be performed as a series of regular MD simulations.³⁷ Arbitrary software implementation limitations are present in AMBER as well; for example, the current version of AMBER's PMEMD program can perform HREM, but not in conjunction with TI.

In the present work, we seek to perform TI with HREM using the GPU-accelerated version of PMEMD. We do this by abandoning AMBER's dual-topology TI framework for performing single-topology transformations. If the two end-states can be described by the same set of molecular mechanical (MM) parameter variables, but different parameter values, then the intermediate alchemical states can be constructed by interpolating the parameter values. The potential energy surface (PES) of a parameter-interpolated intermediate state is fully described by standard MM potential energy functions, so the MD software does not explicitly require a TI implementation to perform the dynamics. Parameter-interpolated TI (PI-TI) thus circumvents arbitrary program incompatibilities that may exist between the MD program's native TI implementation and other software features, such as enhanced sampling methods, so long as those limitations that are not founded upon a theoretical basis. The thermodynamic derivatives required for TI can be implemented in the MD software for on-the-fly evaluation, or they can be evaluated in a post-processing stage that reanalyzes the statistically independent trajectory frames. There is no methodological limitation that prevents PI-TI from being extended to include soft-core potentials for those applications that demand their use, although minor modifications to the force field functional form may then be required.

Parameter-interpolation has previously been used to perform single-topology alchemical transformations. The parameter-interpolated free energy perturbation (PI-FEP) method was pioneered by Jorgensen³⁸ and continues to be used in the BOSS program.³⁹ The parameter-interpolated BAR and MBAR methods (PI-BAR and PI-MBAR) have previously been described³⁶ using the OpenMM program,⁴⁰ and this framework has recently been extended to the Tinker-OpenMM module⁴¹ for applications using the AMOEBA force field.^{34,42–44} Application of the PI-BAR method has been shown to be a practical alternative to using soft-core potentials.³⁷ In addition to these other methods, the concept of PI-TI, specifically, is also very old; Mezei and Beveridge referred to PI-TI as a “nonlinear TI” pathway more than 30 years ago,⁴⁵ and exploratory demonstrations suggested that parameter-interpolation can lead to numerically advantageous TI pathways.^{11,46} Early versions of AMBER could perform PI-TI calculations via the GIBBS program,^{47,48} but the PI-TI approach in AMBER was abandoned in favor of using a dual-topology TI framework⁴⁹ after the introduction of the smooth particle mesh Ewald (PME) method.⁵⁰ It was believed that PME could not be used with PI-TI because PME is not a pairwise decomposable energy.⁵¹ This belief does not appear to have been limited to AMBER development community. Modern versions of GROMOS program continue to only support PI-TI with cutoff or reaction field electrostatics, but not PME electrostatics.^{52,53} The GROMACS program will use PI-TI for some energy terms,³²

but it resorts to a linear-mixing strategy for evaluating the reciprocal-space PME energy.^{26,54,55} Similarly, the CHARMM,²⁸ LAMMPS,⁵⁶ and NAMD²⁰ programs perform two reciprocal-space PME calculations. The evaluation of two reciprocal-space PME calculations has been specifically identified as slowing PMEMD, for example, by 25% relative to non-TI simulations.⁵⁷ The GPU TI implementation in ref 30 reports a 30% performance degradation relative to non-TI simulations. The PI-TI method described in the present work requires only one PME calculation per time step to evaluate the energy, forces, and TI-gradient. That is, we show how one can compute the TI-gradient of the PME reciprocal-space energy using no more information than what is normally required to evaluate the energy and forces in a non-TI simulation.

In this work, we revisit the PI-TI method and implement it with support for PME electrostatics to leverage the performance of existing GPU-accelerated MD software and unlock additional program features, such as HREM.⁵⁸ Out of convenience, we use a modified version of the SANDER MD CPU-code to evaluate the necessary TI derivatives by post-processing trajectories generated with the GPU-enabled version of PMEMD. The PI-TI is well suited for certain types of free energy applications, including point mutations, metal ion solvation and binding, pK_a shift prediction, and QM/MM charge-corrections. In the next sections, we describe the equations necessary to perform the PI-TI method. We compare the potential energy surface of $Mg^{2+} \cdots H_2O$ as the Mg^{2+} is transformed to Ca^{2+} using a standard linear alchemical pathway, soft-core potentials, and PI-TI. We demonstrate that standard TI and PI-TI methods produce identical free energies for the transformation of Mg^{2+} to Ca^{2+} in solution and further show how their “TI gradients” differ. We apply the parameter-interpolated alchemical pathway scheme to predict the pK_a of 3 RNA sequences, establish highly converged simulations with meaningful error estimates, and make comparisons with experiments. In doing so, we compare results between PI-TI, PI-BAR, and PI-MBAR, and examine the convergence of the results as a function of simulation time. We rerun the simulations using several HREM exchange rates to elucidate its advantages. We compare the pK_a results from long MD simulations to a collection of short MD simulations. Finally, we examine how the pK_a results change if the MM charges are replaced with electrostatically fit charges derived from *ab initio* QM/MM calculations.

2. METHODS

2.1. Thermodynamic Integration. The free energy difference between two states $\Delta G = G^{(1)} - G^{(0)}$ can be evaluated using the thermodynamic integration (TI) method:

$$\Delta G = \int_0^1 \left\langle \frac{\partial U(\lambda)}{\partial \lambda} \right\rangle_\lambda d\lambda \quad (1)$$

where $U(\lambda)$ is a λ -dependent potential energy that connects the potential of the initial state $U^{(0)}$ to the final state $U^{(1)}$. Of primary interest in the present work are those transformations where the two states differ only by the values of their MM parameters, \mathbf{P} , in which case: $U^{(0)} = U(\mathbf{P}^{(0)})$ and $U^{(1)} = U(\mathbf{P}^{(1)})$. This type of perturbation can be classified as a “single-topology” transformation; however, the free energy of this transformation can also be evaluated within a “dual-topology” framework. The distinction between these classifications is all atoms within a single-topology framework interact with all other atoms in the system, whereas a dual-topology approach includes two copies of the perturbed atoms—but the two copies do not directly interact with one another.⁴⁶

The dual-topology implementation of a single-topology transformation enforces a restriction that the two copies share a common set of atomic coordinates.⁵⁹ In this case, the values of $U^{(0)}$ and $U^{(1)}$ are the total potential energies when one of the two copies is present and the other is absent.

Linear Alchemical Mixing. Linear mixing of these end-state potential energies are referred to as the “standard TI” (Std.-TI) method:

$$U_{\text{Std.TI}}(\lambda) = U(\mathbf{P}^{(0)}) + \lambda(U(\mathbf{P}^{(1)}) - U(\mathbf{P}^{(0)})) \quad (2)$$

The free energy difference is obtained by integrating the following derivative:

$$\left\langle \frac{\partial U_{\text{Std.TI}}}{\partial \lambda} \right\rangle_{\lambda} = \left\langle U(\mathbf{P}^{(1)}) - U(\mathbf{P}^{(0)}) \right\rangle_{\lambda} \quad (3)$$

Specifically, a series of simulations are performed to compute eq 3 at a discrete set of λ values. Equation 1 is then evaluated from numerical quadrature

$$\Delta G = \sum_{\lambda} w_{\lambda} \left\langle \frac{\partial U(\lambda)}{\partial \lambda} \right\rangle_{\lambda} \quad (4)$$

or by fitting the $\langle \partial U / \partial \lambda \rangle_{\lambda}$ values to a function, such as a cubic spline, that can be integrated analytically.^{60,61} Comparisons between Clenshaw–Curtis, Simpson’s rule, and trapezoidal rule integration formulas can be found in ref 62. The Std.-TI method has the advantage that the thermodynamic derivative is simple, independent of λ , and only requires computation of the endpoint potential energy states. In principle, one could numerically differentiate $U(\lambda)$ to approximate $\partial U / \partial \lambda$; however, the simplicity of fixed-charge force field functional forms produce equally simple analytic derivative expressions that can be implemented in an analysis program without introducing additional numerical error. Alternative free energy methods have been developed that dynamically change the λ values during the course of dynamics, including the slow-growth method,⁴⁷ fast-growth method,^{63–65} and the λ -dynamics method.^{66–71}

Nonlinear Alchemical Mixing and “Soft-Core” TI. For completeness, we note that MD programs often implement a generalization of eq 2, where the endpoint potential energies are weighted by a nonlinear function of λ : $U(\lambda) = (1 - f(\lambda))U^{(0)} + f(\lambda)U^{(1)}$. The generalized form $U(\lambda)$, sometimes called “nonlinear-mixing”,⁷² is used to improve the behavior of $\langle \partial U / \partial \lambda \rangle_{\lambda}$ in certain situations.^{5,73–78} For example, nonlinear-mixing has been used to reduce integration errors resulting from the “endpoint singularity effect”⁷³ encountered when a Lennard-Jones (LJ) particle is removed (“annihilated” or “deleted”) from the system. The soft-core method^{78–81} is an alternative to nonlinear-mixing that replaces the expression for the standard LJ energy with a modified functional form that smoothly decouples (as λ is goes from 0 to 1) the interaction of the atoms undergoing annihilation with the remainder of the system. The pairwise functional form of soft-core TI (SC-TI) LJ-potential used in the present work is

$$U_{\text{SC-LJ},ab}(R_{ab}; \lambda) = \frac{4\epsilon_{ab}(1 - \lambda)}{\left(\frac{\lambda}{2} + 2\left(\frac{R_{ab}}{r_{m,ab}}\right)^6\right)^2} - \frac{4\epsilon_{ab}(1 - \lambda)}{\frac{\lambda}{2} + 2\left(\frac{R_{ab}}{r_{m,ab}}\right)^6} \quad (5)$$

where ϵ_{ab} is the LJ well-depth and $r_{m,ab}$ is the location of the LJ minimum. A region can be “coupled” to the remainder of the system with an analogous expression by replacing λ with $(1 - \lambda)$.

Equation 5 is the form used in AMBER, which was developed in ref 76. The functional form used in other programs, such as GROMACS,⁸² are slightly different, and other expressions for soft-core interactions can be found in the literature.^{72,83} Furthermore, exploratory investigations suggest that the statistical efficiency of atom insertion/deletion processes can be improved by using “linear basis functions”,^{84,85} and other works have avoided the endpoint singularity by using “exponential-six” potentials.⁸⁶ The electrostatic coupling between annihilating/inserting atoms can also be modified with a nonlinear functional form.^{72,78,83} The SC-TI free energy calculations performed in the present work do not use soft-core electrostatics. Instead, the SC-TI simulations are performed using a 3-step procedure: the selected atoms are decharged with Std.-TI, the LJ interactions are changed via eq 5, and the atoms are recharged using Std.-TI.

Parameter Interpolation. Parameter-interpolation methods, such as PI-FEP, PI-BAR, PI-MBAR, and PI-TI, use a standard MM potential energy surface,

$$U_{\text{PI}}(\lambda) = U(\mathbf{P}^{(\lambda)}) \quad (6)$$

whose parameters are mixed from the endpoint values:

$$\mathbf{P}^{(\lambda)} = \mathbf{P}^{(0)} + \lambda(\mathbf{P}^{(1)} - \mathbf{P}^{(0)}) \quad (7)$$

In other words, one writes a parameter file for each value of λ (eq 7) and then performs traditional MD simulations using the potential shown in eq 6. The PI-TI free energy difference is obtained by integrating the following derivative:

$$\left\langle \frac{\partial U_{\text{PI}}}{\partial \lambda} \right\rangle_{\lambda} = \sum_{i=1}^{N_{\text{param}}} \left\langle \frac{\partial U(\mathbf{P}^{(\lambda)})}{\partial P_i^{(\lambda)}} \right\rangle_{\lambda} \frac{\partial P_i^{(\lambda)}}{\partial \lambda} \quad (8)$$

The calculation of $\langle \partial U / \partial P_i^{(\lambda)} \rangle_{\lambda}$ could either be evaluated during the course of dynamics or via reprocessing of the saved trajectory frames. One does not need to record every simulation time step in the trajectory file to achieve reasonable averages because the autocorrelation time of eq 8 is often much longer than a MD time step. Furthermore, the reprocessing of the trajectories statistically independent samples is highly parallelizable because each frame can be examined independently of the others. Our experience has been that the effort spent reprocessing trajectories is trivial in comparison to effort spent performing the MD simulations. The method we describe below for calculating the reciprocal-space PME TI-gradient is computationally advantageous to the Std.-TI method even if the “on-the-fly” calculation of $\partial U / \partial \lambda$ was preferred to post-processing.

In the present work, we perform transformations that scale the atomic charges, q_a

$$q_a^{(\lambda)} = q_a^{(0)} + \lambda(q_a^{(1)} - q_a^{(0)}) \quad (9)$$

and/or the two-body Lennard-Jones well-depths, ϵ_{ab} , and minimum energy distances, $r_{m,ab}$

$$\epsilon_{ab}^{(\lambda)} = \epsilon_{ab}^{(0)} + \lambda(\epsilon_{ab}^{(1)} - \epsilon_{ab}^{(0)}) \quad (10)$$

$$r_{m,ab}^{(\lambda)} = r_{m,ab}^{(0)} + \lambda(r_{m,ab}^{(1)} - r_{m,ab}^{(0)}) \quad (11)$$

The values of the bond and angle force constants, equilibrium values, and the torsion cosine potentials can be similarly interpolated. The implementation of the bonded-energy TI gradient terms is straightforward, and it has been successfully applied many times in the past.^{46–48,87} The TI gradient of the particle mesh Ewald (PME) electrostatic energy, however, is not widely

known to the MD development community. Therefore, TI gradients for the PME electrostatics, as well as the Lennard-Jones nonbonded terms, are discussed separately below.

PI-TI Gradients for PME Electrostatics. Particle mesh Ewald is a method for computing the average electrostatic energy per unit cell of a periodic charge distribution.^{50,88–90} The electrostatic energy is a quadratic function of charge, and for periodic point-charge distributions, it is

$$U_{\text{elec}}(\mathbf{q}^{(\lambda)}) = \frac{1}{2} \sum_a \sum_n \sum_b \frac{q_a^{(\lambda)} q_b^{(\lambda)}}{|\mathbf{R}_{ab} + \mathbf{n}|} \quad (12)$$

where $\mathbf{n} = n_1 \mathbf{a}_1 + n_2 \mathbf{a}_2 + n_3 \mathbf{a}_3$ is a vector to a unit cell; \mathbf{a}_1 , \mathbf{a}_2 , and \mathbf{a}_3 are real-space lattice vectors; the summation over \mathbf{n} considers each unique replicated cell; and the primed-summation excludes the situation where the denominator becomes zero. The electrostatic potential experienced by point charge a is

$$\begin{aligned} p_a^{(\lambda)} &\equiv \frac{\partial U_{\text{elec}}}{\partial q_a^{(\lambda)}} \\ &\equiv \sum_n \sum_b \frac{q_b^{(\lambda)}}{|\mathbf{R}_{ab} + \mathbf{n}|} \end{aligned} \quad (13)$$

where the superscript (λ) emphasizes that the electrostatic potential is caused by the interpolated charges (eq 9). The electrostatic energy can then be concisely written,

$$U_{\text{elec}}(\mathbf{q}^{(\lambda)}) = \frac{1}{2} \sum_a q_a^{(\lambda)} p_a^{(\lambda)} \quad (14)$$

and the PI-TI gradient is

$$\begin{aligned} \frac{\partial U_{\text{elec}}}{\partial \lambda} &= \sum_a \frac{\partial U_{\text{elec}}}{\partial q_a^{(\lambda)}} \frac{\partial q_a^{(\lambda)}}{\partial \lambda} \\ &= \sum_a (q_a^{(1)} - q_a^{(0)}) p_a^{(\lambda)} \end{aligned} \quad (15)$$

Notice that that eq 15 reuses the same electrostatic potential required to compute the energy (eq 14); a second PME calculation is not required. This differs from the Std.-TI method, which performs two PME calculations:

$$U_{\text{elec,Std.TI}}(\lambda) = \frac{1-\lambda}{2} \sum_a q_a^{(0)} p_a^{(0)} + \frac{\lambda}{2} \sum_a q_a^{(1)} p_a^{(1)} \quad (16)$$

Equations 14 and 15 are sufficient instruction to understand how to modify existing PME software implementations. For completeness, eqs 17–25 provide explicit expressions for the PME electrostatic potential.⁸⁹ We drop the superscripts for brevity, and we decompose the potential into three components:

$$p_a = p_{\text{recip},a} + p_{\text{real},a} + p_{\text{charge},a} \quad (17)$$

The first term in eq 17 is the reciprocal-space contribution to the potential,

$$p_{\text{recip},a} = \sum_{\mathbf{t}} \theta(\mathbf{R}_a - \mathbf{R}_t) \frac{4\pi}{V} \sum_{\mathbf{k} \neq 0} e^{i\mathbf{k} \cdot \mathbf{R}_t} \frac{S_{\mathbf{k}} e^{-k^2/4\beta^2}}{k^2 \theta_{\mathbf{k}}} \quad (18)$$

where the unit cell volume and Ewald coefficient are V and β , respectively. The plane waves are indexed by their angular wavenumber,

$$\mathbf{k} = 2\pi(k_1 \mathbf{a}_1^* + k_2 \mathbf{a}_2^* + k_3 \mathbf{a}_3^*) \quad (19)$$

where \mathbf{a}_1^* , \mathbf{a}_2^* , and \mathbf{a}_3^* are the reciprocal space lattice vectors. The PME method performs fast Fourier transforms on a regular grid consisting of N points ($N = N_1 N_2 N_3$) that are indexed by \mathbf{t} and located at \mathbf{R}_t . The B-spline weights used to interpolate from the grid are

$$\theta(\mathbf{r}) = \prod_{d=1}^3 M_n(N_d \mathbf{r}^T \cdot \mathbf{a}_d^* + n/2) \quad (20)$$

and the B-spline weight discrete Fourier transform coefficients are

$$\theta_{\mathbf{k}} = \sum_{\mathbf{t}} e^{-i\mathbf{k} \cdot \mathbf{R}_t} \theta(\mathbf{R}_t) \quad (21)$$

The M_n function appearing in eq 20 is an order- n Cardinal B-spline:

$$M_n(u) = \sum_{k=0}^n \frac{(-1)^k}{(n-1)!} \binom{n}{k} [\max(u-k, 0)]^{n-1} \quad (22)$$

The “structure factors” are discrete Fourier transform coefficients of the B-splined charge density:

$$S_{\mathbf{k}} = \sum_{\mathbf{t}} e^{-i\mathbf{k} \cdot \mathbf{R}_t} \sum_a q_a \theta(\mathbf{R}_t - \mathbf{R}_a) \quad (23)$$

The second term in eq 17 is the real-space contribution to the potential:

$$p_{\text{real},a} = -q_a \frac{2\beta}{\sqrt{\pi}} + \sum_{b \neq a} q_b \frac{\text{erfc}(\beta R_{ab})}{R_{ab}} \quad (24)$$

The third term in eq 17 is the uniform background contribution to the potential for charged systems:

$$p_{\text{charge},a} = -\frac{\pi}{V\beta^2} \sum_b q_b \quad (25)$$

PI-TI Gradients for Lennard-Jones Nonbonded Terms. The Lennard-Jones (LJ) energy used in this work explicitly accounts for all interactions within the nonbond cutoff, R_{cut} , and it includes a tail correction for the long-range attractions beyond the cutoff:

$$U_{\text{LJ}}(\mathbf{A}^{(\lambda)}, \mathbf{B}^{(\lambda)}) = \sum_{\substack{b>a \\ R_{ab}<R_{\text{cut}}}} \frac{A_{ab}^{(\lambda)}}{R_{ab}^{12}} - \frac{B_{ab}^{(\lambda)}}{R_{ab}^6} + U_{\text{tail}}(\mathbf{B}^{(\lambda)}) \quad (26)$$

where

$$A_{ab}^{(\lambda)} = \epsilon_{ab}^{(\lambda)} r_{m,ab}^{(\lambda)12} \quad (27)$$

$$B_{ab}^{(\lambda)} = 2\epsilon_{ab}^{(\lambda)} r_{m,ab}^{(\lambda)6} \quad (28)$$

$$U_{\text{tail}}(\mathbf{B}^{(\lambda)}) = -\sum_{ab} \frac{2\pi B_{ab}^{(\lambda)}}{3VR_{\text{cut}}^3} \quad (29)$$

Chain-rule differentiation leads to the following expression for the LJ TI gradient:

$$\left\langle \frac{\partial U_{\text{LJ}}}{\partial \lambda} \right\rangle_{\lambda} = \sum_{u,v}^{N_{\text{types}}} \left\langle \frac{\partial U_{\text{LJ}}}{\partial A_{uv}} \right\rangle_{\lambda} \frac{\partial A_{uv}^{(\lambda)}}{\partial \lambda} + \left\langle \frac{\partial U_{\text{LJ}}}{\partial B_{uv}} \right\rangle_{\lambda} \frac{\partial B_{uv}^{(\lambda)}}{\partial \lambda} \quad (30)$$

$$\frac{\partial A_{ab}^{(\lambda)}}{\partial \lambda} = r_{m,ab}^{(\lambda)12}(\epsilon_{ab}^{(1)} - \epsilon_{ab}^{(0)}) + 12\epsilon_{ab}^{(\lambda)}r_{m,ab}^{(\lambda)11}(r_{m,ab}^{(1)} - r_{m,ab}^{(0)}) \quad (31)$$

$$\frac{\partial B_{ab}^{(\lambda)}}{\partial \lambda} = 2r_{m,ab}^{(\lambda)6}(\epsilon_{ab}^{(1)} - \epsilon_{ab}^{(0)}) + 12\epsilon_{ab}^{(\lambda)}r_{m,ab}^{(\lambda)5}(r_{m,ab}^{(1)} - r_{m,ab}^{(0)}) \quad (32)$$

Direct interpolation of the A_{ab} and B_{ab} coefficients is also possible, and it results in van der Waals transformations that mimic the Std.-TI pathway. We will show that the interpolation of the ϵ_{ab} and $r_{m,ab}$ parameters can produce a more uniform distribution of alchemical states. Other models for the van der Waals energy exist, including: Born-Mayer functions,^{91–93} 12–6–4 potentials,^{94–96} 14–7 potentials,⁹⁷ models based on atomic overlap,^{98–101} and isotropic periodic summation.^{50,102,103}

2.2. Computational Details. General Simulation Protocol. Unless otherwise explicitly stated, we use the Amber ff14SB force field parameters,¹⁰⁴ the TIP4P/Ew water model,¹⁰⁵ and the Li and Merz TIP4P/Ew 12-6 divalent metal ion parameters.¹⁰⁶ All Std.-TI and SC-TI simulations were performed with the Amber 16 CPU implementation of PMEMD.¹⁰⁷ The PI-TI simulations were performed¹⁰⁸ with the Amber 16 GPU implementation of PMEMD.^{23,107} The dynamics were propagated with a 1 fs time step. Bonds involving hydrogen were constrained with SHAKE.¹⁰⁹ Nonbond interactions were explicitly computed within a 10 Å direct-space cutoff, and the LJ interactions beyond the cutoff were modeled with a long-range tail correction (eq 29). Electrostatic interactions were evaluated with the particle mesh Ewald method using a 1 point/Å³ fast Fourier transform grid spacing, fourth-order cubic B-spline interpolation, and tinfoil boundary conditions.⁵⁰ A charge-canceling uniform background plasma correction was applied to those systems with a net charge.^{89,110} A constant temperature of 298 K was regulated with the Langevin thermostat (5 ps collision frequency).¹¹¹ All production simulations were performed at constant temperature and volume.

The following procedure was used to equilibrate the systems:

1. The density of the $\lambda = 0$ state was equilibrated for 20 ns in the NPT ensemble at 298 K using the Berendsen barostat to maintain a pressure of 1 atm (1.013 bar).
2. The $\lambda = 0$ and $\lambda = 1$ states were then equilibrated in the NVT ensemble for an additional 20 ns, starting from the final coordinates obtained in step 1.
3. Each λ -window was equilibrated in the NVT ensemble for an additional 10 ns. The $\lambda < 0.5$ and $\lambda \geq 0.5$ simulations began from the $\lambda = 0$ and $\lambda = 1$ equilibrations performed in step 2, respectively.

Production statistics for each λ -window were collected after the equilibration described in step 3.

Free Energy Calculation and Error Analysis. The value of $\partial U/\partial \lambda$ was output every 5 ps (Std.-TI and SC-TI). Similarly, the PI-TI simulations wrote the atomic coordinates to a trajectory file every 5 ps for post-processing. The time-series of $\partial U/\partial \lambda$ values were extracted (or computed) for each λ -simulation and stored to a file. The free energy analysis was performed using the ALCHEMICAL-ANALYSIS program.¹¹² The ALCHEMICAL-ANALYSIS program makes use of the PYMBAR library, which implements time-series algorithms¹¹³ and various free energy analysis methods, including the Bennett acceptance ratio^{8,114} (BAR) and multi-state Bennett acceptance ratio⁹ (MBAR) methods. The reported TI free energies use the trapezoidal rule to perform the numerical integration shown in eq 4, except for those labeled PI-TI3, Std.-TI3, or SC-TI3, which perform the integration from a natural cubic spline interpolation

of the $\langle \partial U/\partial \lambda \rangle_\lambda$ data. Details of the TI3 integration procedure and error propagation can be found in the appendix of ref 115.

Our best estimate of the mean $\langle \partial U/\partial \lambda \rangle_\lambda$ is the time-average value:

$$\left\langle \frac{\partial U}{\partial \lambda} \right\rangle_\lambda = \frac{1}{N_t} \sum_{t=1}^{N_t} \frac{\partial U(\lambda; t)}{\partial \lambda} \quad (33)$$

If the time-series was uncorrelated, then the standard error of the mean, $\sigma_{\langle \lambda \rangle}$, could be computed from the standard deviation of the $\partial U/\partial \lambda$ time-series, σ_λ :

$$\sigma_{\langle \lambda \rangle} = \sigma_\lambda / \sqrt{N_t} \quad (34)$$

$$\sigma_\lambda^2 = \frac{1}{N_t} \sum_{t=1}^{N_t} \left(\frac{\partial U(\lambda; t)}{\partial \lambda} \right)^2 - \left(\frac{1}{N_t} \sum_{t=1}^{N_t} \frac{\partial U(\lambda; t)}{\partial \lambda} \right)^2 \quad (35)$$

The standard errors of the $\langle \partial U/\partial \lambda \rangle_\lambda$ averages are propagated through eq 4 to estimate the error of the calculated free energy:

$$\sigma_{\Delta G} = \sqrt{\sum_\lambda w_\lambda^2 \sigma_{\langle \lambda \rangle}^2} \quad (36)$$

If the time-series is correlated, then eq 34 underestimates the true error of the mean. A better estimate of the error is made by pruning the available data with a stride equal to the *statistical inefficiency*, g_λ :

$$g_\lambda = 1 + 2\tau_\lambda/\nu \quad (37)$$

where τ_λ is the autocorrelation time (eq 38), ν is the correlated sampling rate (5 ps/frame in this work), and g_λ is in units of the sampling interval (the number of frames between uncorrelated samples). The true autocorrelation time is unknown, but it can be approximated from the best estimate of the autocorrelation function (eq 39).

$$\tau_\lambda = \nu \sum_{t=1}^{N_t-1} \left(1 - \frac{t}{N_t} \right) C_{\lambda,t} \quad (38)$$

$$C_{\lambda,t} = \frac{1}{\sigma_\lambda^2} \left(\frac{1}{N_t - t} \sum_{n=1}^{N_t-t} \frac{\partial U(\lambda; n)}{\partial \lambda} \frac{\partial U(\lambda; n+t)}{\partial \lambda} - \left\langle \frac{\partial U}{\partial \lambda} \right\rangle_\lambda^2 \right) \quad (39)$$

To account for the correlation within the samples, the error propagation (eq 36) is performed using the standard errors from the pruned data; that is,

$$\sigma_{\langle \lambda \rangle} = \frac{\tilde{\sigma}_\lambda}{\sqrt{N_t/g_\lambda}} \quad (40)$$

$$\tilde{\sigma}_\lambda^2 = \frac{g_\lambda}{N_t} \sum_{t=1}^{N_t/g_\lambda} \left(\frac{\partial U(\lambda; tg_\lambda)}{\partial \lambda} \right)^2 - \left(\frac{g_\lambda}{N_t} \sum_{t=1}^{N_t/g_\lambda} \frac{\partial U(\lambda; tg_\lambda)}{\partial \lambda} \right)^2 \quad (41)$$

In summary, we run the ALCHEMICAL-ANALYSIS program¹¹² twice: the free energy is computed from the correlated data, and the errors are estimated from the analysis of the pruned/uncorrelated samples.

It is useful to monitor the free energy (and its error) as a function of sampling time. For example, by plotting the free energy

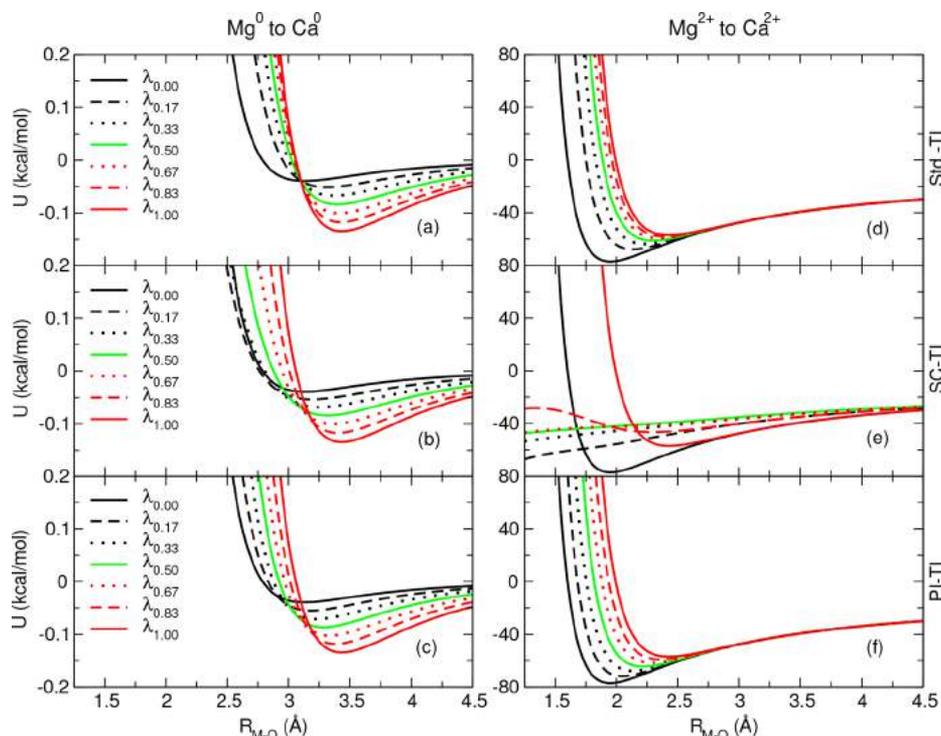


Figure 1. Potential energy surface of metal–H₂O as a function of metal–oxygen distance. Each pane shows the PES for several λ values as the metal is transformed from Mg²⁺ ($\lambda = 0$) to Ca²⁺ ($\lambda = 1$). The left and right columns illustrate Mg⁰ → Ca⁰ and Mg²⁺ → Ca²⁺ transformations, respectively. The rows differ by the choice of TI algorithm. The 1-step SC-TI Mg²⁺ → Ca²⁺ transformation uses both soft-core electrostatic and soft-core LJ potentials.

from the analysis of the first 10, 20, 30 ns, etc. of production statistics; however, we always use our best estimate of the autocorrelation time (from *all* of the available production data) when estimating the errors of petite time-series.

The BAR and MBAR results listed in this manuscript were also evaluated using the `ALCHEMICAL-ANALYSIS` program.¹¹² The analysis required us to recompute the energy $U(\lambda)$ of each frame within the trajectory file generated from the $U(\lambda')$ simulation. That is, a matrix of energies $U_{\lambda,\lambda'}$ is produced at each time step. The BAR and MBAR free energy errors were evaluated from the uncorrelated samples using the statistical inefficiency computed from the correlated $\partial U/\partial \lambda$ values. The labels PI-BAR and PI-MBAR emphasize that the intermediate states sample the parameter-interpolated potential energy surfaces. In other words, the PI-BAR and PI-MBAR results reanalyze the same trajectories that were used to compute the PI-TI free energies.

We sample the necessary information for performing TI, BAR, and MBAR analysis once every 5 ps. Some readers may believe that sampling every frame within the trajectory would lead to better free energy estimates. This belief fails to recognize that the trajectory forms a correlated time-series. Although it is not inherently incorrect to average the TI gradient from all samples within an equilibrium simulation time-series, the correlated samples are not improving the statistical quality of the estimated average.¹¹³ In fact, a careful estimate of an average's standard error would have one prune the data to produce a petite set of statistically independent samples.^{9,112} Excessive sampling of the TI, BAR, and MBAR data, therefore, offers little benefit while also decreasing the performance of the MD software. In this context, it is unfortunate that dual-topology implementations require two reciprocal-space PME evaluations simply to propagate the equations of motion.

2.3. Comparisons between TI Methods. In this section, we examine the transformation: Mg_(aq)²⁺ → Ca_(aq)²⁺ to compare the

PI-TI, Std.-TI, and SC-TI alchemical pathways. This test system is sufficiently simple that we can unambiguously illustrate potential energy surfaces of the alchemical intermediate states to help explain the observed differences between their $\langle \partial U/\partial \lambda \rangle_\lambda$ profiles. We show that the alchemical pathways produce different intermediate alchemical states, but reproduce the same net transformation energy. Furthermore, we examine the sensitivity of the results with respect to the number of intermediate states.

Figure 1 plots the potential energy surface (PES) of each metal ion state coordinated with a single water. The metal ion state is either the Mg²⁺ or Ca²⁺ end-state or an intermediate state mixture between the two. The left panels illustrate the transformation of the LJ potential (the charge of the metal is set to zero), and the right panels illustrate a 1-step transformation between the metal ions. The top, middle, and bottom rows show the Std.-TI, SC-TI, and PI-TI transformations, respectively.

Table 1 compares Std.-TI3, SC-TI3, and PI-TI3 calculations of the free energy difference between a solvated Mg²⁺ ion and a solvated Ca²⁺ ion. The system is a periodic cube filled with 2000 TIP4P/Ew waters and a single metal ion. The real space lattice vectors of the simulation cell are 39.13 Å. The “1-step” transformations change the LJ interactions of the metal; that is, Mg_(aq)²⁺ → Ca_(aq)²⁺. The “3-step” transformations are a sequence of transformations:

1. The charge of the Mg is removed (Mg²⁺ → Mg⁰).
2. The LJ interactions are scaled (Mg⁰ → Ca⁰).
3. The charge of Ca is replaced (Ca⁰ → Ca²⁺).

All stages were performed with 32 λ -windows ($\lambda = i/31$ for $i \in [0,31]$), and 90 ns of production statistics were simulated for each λ -window. Therefore, a 1-step calculation of ΔG was performed using 32 simulations, and a 3-step calculation performed a total of 96 simulations. We do not use soft-core electrostatics,

Table 1. Free Energy Change of Transformation $\text{Mg}^{2+} \rightarrow \text{Ca}^{2+}$ in Solution^a

N_λ	Stage	Std.-TI3		SC-TI3	PI-TI3	PI-TI3
		1-step	3-step	3-step	1-step	3-step
32	Discharge	—	398.90 ± 0.03	(398.90 ± 0.03)	—	398.93 ± 0.03
32	LJ	—	0.28 ± 0.00	0.28 ± 0.01	—	0.28 ± 0.00
32	Recharge	—	-321.55 ± 0.03	(-321.55 ± 0.03)	—	-321.58 ± 0.02
32	Net	77.60 ± 0.02	77.63 ± 0.04	77.63 ± 0.04	77.63 ± 0.02	77.63 ± 0.04
17	Net	77.77 ± 0.02	77.69 ± 0.05	77.70 ± 0.05	77.63 ± 0.02	77.66 ± 0.05
12	Net	78.19 ± 0.03	77.67 ± 0.06	77.66 ± 0.06	77.58 ± 0.03	77.70 ± 0.06
9	Net	78.63 ± 0.03	77.68 ± 0.07	77.68 ± 0.07	77.60 ± 0.03	77.61 ± 0.07
7	Net	80.64 ± 0.04	78.11 ± 0.09	78.11 ± 0.09	77.48 ± 0.04	78.00 ± 0.09

^aValues are in kcal/mol. The \pm values are standard errors. N_λ is the number of λ -states (out of 32) used to perform the analysis.

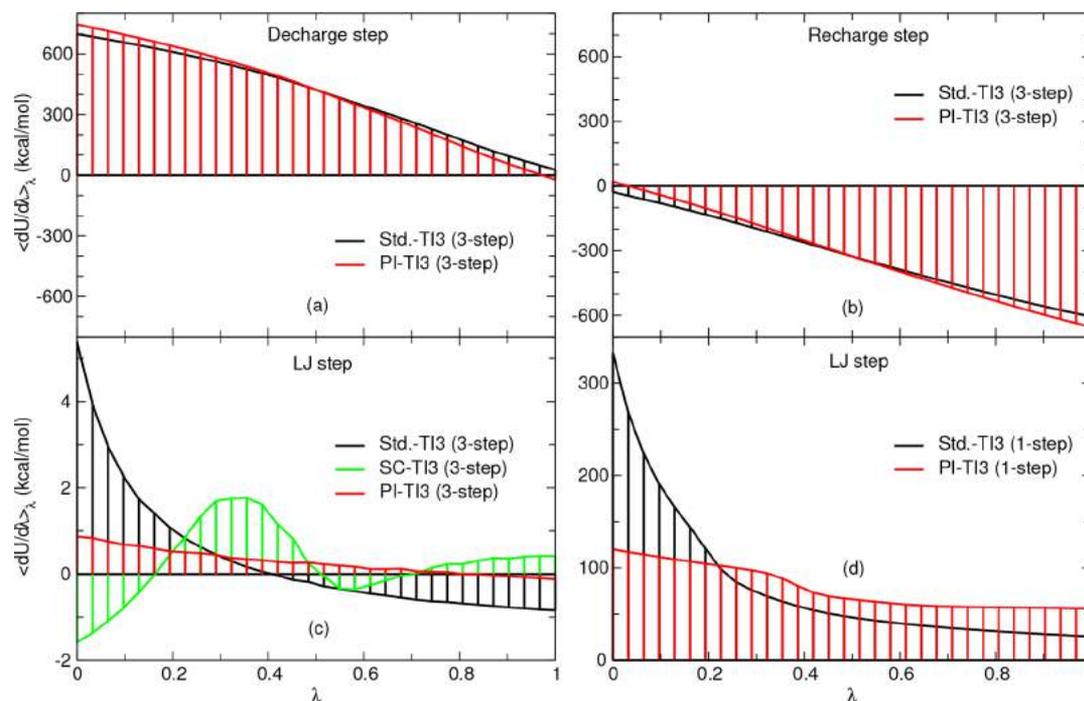


Figure 2. Comparison of $\langle dU/d\lambda \rangle_\lambda$ between PI-TI and SC-TI for the transformation $\text{Mg}^{2+} \rightarrow \text{Ca}^{2+}$.

so the SC-TI3 discharge and recharge stages appearing in Table 1 are taken from the Std.-TI3 calculation. Figure 2 compares the behavior of $\langle \partial U / \partial \lambda \rangle_\lambda$ as a function of λ for each stage appearing in Table 1. We chose to use 32 λ -values to make the plots in Figure 2 look smooth, thus improving the quality of the presentation. This data also serves as a benchmark for comparison in Table 1 when fewer λ -values are used. Figure 3 compares the ΔG values of each stage as a function of production simulation time. The vertical bars are 95% confidence intervals ($1.96 \sigma_{\Delta G}$). For example, the value of ΔG at 40 ns is computed from the first 40 ns of production data, and the $\sigma_{\Delta G}$ value is evaluated from the uncorrelated samples within the first 40 ns of production data, based on the best estimate of the autocorrelation times from the full, 90 ns of production. The $\sigma_{\Delta G}$ value is then scaled by 1.96 to draw the vertical bar in the figure.

The N_λ column in Table 1 lists the number of λ -states used while performing the analysis. The $N_\lambda = 32$ rows integrate $\langle \partial U / \partial \lambda \rangle_\lambda$ from all simulated intermediate states. The $N_\lambda = 17$ row excludes several of the simulations from the analysis; only the $\lambda = 0$ and $\lambda = 1$ endpoints and every-other- λ between them are included in the analysis. The $N_\lambda = 12$ row includes every third simulation between the endpoints, and the $N_\lambda = 9$ and $N_\lambda = 7$ similarly exclude additional interior λ -state simulations.

2.4. Double-Stranded RNA pK_a Benchmark Simulations. Wilcox and Bevilacqua have used ^{31}P NMR to experimentally determine the pK_a of the A \cdot C mismatch base pair within double-stranded RNA (dsRNA) sequences.¹¹⁶ In the present work, we re-examine the three sequences shown in Figure 4. The protonation site is the N1 position of residue A4. The only differences between the three sequences are the base pairs immediately adjacent to A4-C17. The WW, SW, and SS notation refers to relative hydrogen bond strength of the two base pairs immediately flanking A4-C17. The WW sequence places A-U base pairs above and below A4-C17, the SS sequence flanks A4-C17 with G-C base pairs, and the SW sequence substitutes a G-C and A-U base pairs below and above A4-C17, respectively. In other words, the hydrogen bond strength between A-U bases are weak (W) relative to the strong (S) hydrogen bonds between G-C, and this difference causes the pK_a of A4-C17 to shift when the sequence is modified. The geometries of the native and protonated A4-C17 base pairs observed in the dsRNA systems are shown in Figure 5.

To make comparison with the experimental data, we compute the free energy change of the transformation $\text{AH}^+ \rightarrow \text{A}$ of residue A4 within dsRNA, ΔG_{dsRNA} , and of an isolated 5'-adenosine monophosphate (AMP) in solution, ΔG_{AMP} . The acidic proton

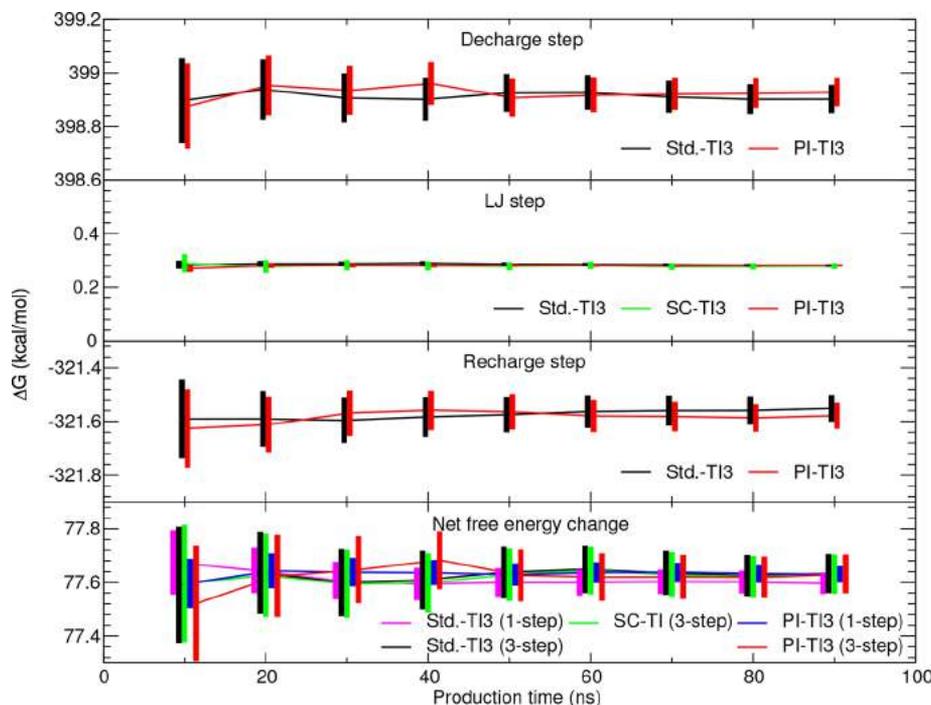


Figure 3. Average ΔG as a function of production time for the transformation $\text{Mg}^{2+} \rightarrow \text{Ca}^{2+}$. The vertical bars are 95% confidence intervals.

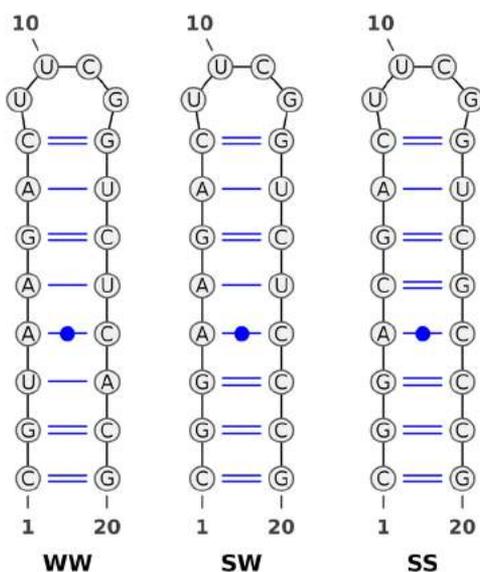


Figure 4. Secondary structures of the three RNA sequences examined in this work. The $\text{p}K_a$ values refer to the N1 position of residue A4.

(the H1 proton) atom-type does not contribute to the LJ energy; only the nucleobase atomic charges differ between the neutral and protonated forms of the adenine. Therefore, each transformation is performed in a single stage that interpolates between the end-state atomic charge vectors of the adenine nucleobase. These charges are provided in the [Supporting Information](#). The calculated $\text{p}K_a$ shift of adenine in dsRNA relative to AMP in solution is

$$\Delta \text{p}K_{a, \text{AMP} \rightarrow \text{dsRNA}} = \frac{\Delta G_{\text{dsRNA}} - \Delta G_{\text{AMP}}}{RT \ln(10)} \quad (42)$$

The absolute $\text{p}K_a$ value in the dsRNA environment is estimated from the experimental $\text{p}K_a$ value of AMP in solution (3.80, ref 117) and the calculated shift:

$$\text{p}K_a = \text{p}K_{a, \text{AMP}}^{\text{Expt.}} + \Delta \text{p}K_{a, \text{AMP} \rightarrow \text{dsRNA}} \quad (43)$$

The $\text{p}K_a$ error values are propagated from the free energy calculations:

$$\sigma_{\text{p}K_a} = \frac{\sqrt{\sigma_{\Delta G_{\text{dsRNA}}}^2 + \sigma_{\Delta G_{\text{AMP}}}^2}}{RT \ln(10)} \quad (44)$$

The $\text{p}K_a$ calculation is repeated for each of the three sequences. The $\text{p}K_a$ shift associated with the mutation of the WW sequence to SS is given by eq 45, and the error is shown in eq 46.

$$\Delta \text{p}K_a(\text{SS}) = \text{p}K_a(\text{SS}) - \text{p}K_a(\text{WW}) \quad (45)$$

$$\sigma_{\Delta \text{p}K_a(\text{SS})} = \sqrt{\sigma_{\text{p}K_a(\text{SS})}^2 + \sigma_{\text{p}K_a(\text{WW})}^2} \quad (46)$$

The expressions for $\Delta \text{p}K_a(\text{SW})$ and $\sigma_{\Delta \text{p}K_a(\text{SW})}$ are analogous.

The dsRNA calculations were performed in a truncated octahedron (58.54 Å lattice vector lengths) containing either the SS, SW, or WW sequence and filled with 4967 waters, 19 Na^+ ions, and 10 Cl^- ions to yield a 0.14 M NaCl approximate bulk ion concentration. Each free energy calculation used 12 λ -windows, and 160 ns of production statistics were produced for each simulation. The free energy calculation was repeated four times: once without using HREM and then using 2.5, 5.0, or 10.0 ps/attempt HREM exchange attempt rates, τ_{HREM} . The $\text{p}K_a$ and $\Delta \text{p}K_a$ values from the long, 160 ns simulations are summarized in Table 2. The \pm values are the errors from eqs 44 and 46. The ΔG_{AMP} values appearing in eqs 42 and 43 are the 0.14 M NaCl and $N_{\text{solv}} = 5579$ results shown in Table 3. The convergence of the PI-TI3 $\text{p}K_a$ values with respect to the amount of production sampling is illustrated in Figure 6. The vertical bars are 95% confidence limits ($1.96 \sigma_{\text{p}K_a}$).

The $\partial U / \partial \lambda$ autocorrelation times, τ_{cor} , observed in the 160 ns simulations of ΔG_{dsRNA} are summarized in Table 4. Each transformation was performed with 12 λ -windows; therefore, there are 12 values of τ_{cor} . Table 4 lists the average and maximum values from the set of 12 autocorrelation times. We ran our simulations

Table 3. Size and Ion Concentration Dependence of 5'-Adenosine Monophosphate N1 Deprotonation Free Energy: $AH^+ \rightarrow A^-$

N_{solv}	L (Å)	0.14 M NaCl				0 M NaCl			
		PI-TI	PI-TI3	PI-BAR	PI-MBAR	PI-TI	PI-TI3	PI-BAR	PI-MBAR
1853	41.8	-77.46	-77.37	-77.45	-77.52	-77.44	-77.35	-77.42	-77.49
2752	47.6	-77.53	-77.45	-77.52	-77.58	-77.43	-77.34	-77.42	-77.49
3651	52.3	-77.53	-77.44	-77.51	-77.58	-77.42	-77.33	-77.40	-77.48
5579	60.3	-77.53	-77.44	-77.52	-77.59	-77.44	-77.36	-77.43	-77.50

^aValues are in kcal/mol. L is the length of the truncated octahedron lattice vectors. N_{solv} is the number of solvent water molecules and ions present in the simulation. The standard error of each TI and MBAR result is ± 0.02 . The standard error of each BAR result is ± 0.01 .

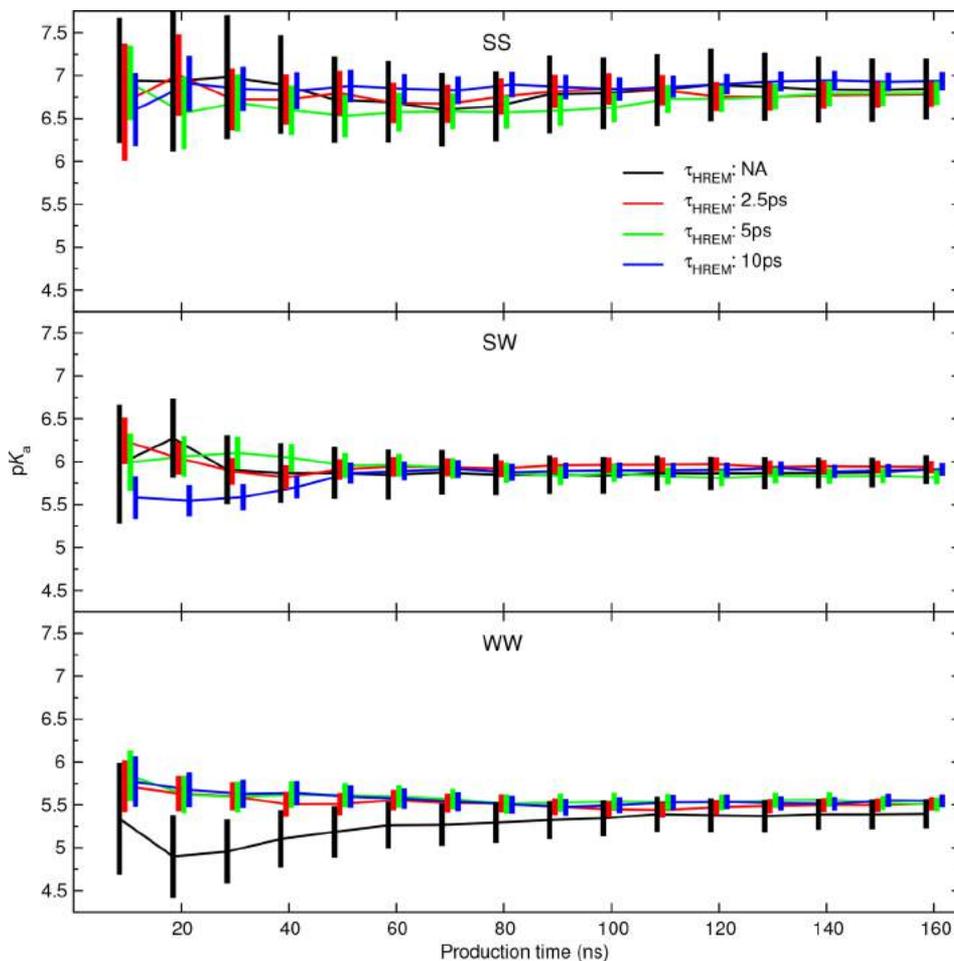


Figure 6. Average pK_a computed from the PI-TI3 method, as a function of production time. The vertical bars are 95% confidence intervals.

Table 4. $\partial U/\partial \lambda$ Autocorrelation Times (ps) from dsRNA Simulations

τ_{HREM}	$\tau_{\text{cor}}(\text{SS})$		$\tau_{\text{cor}}(\text{SW})$		$\tau_{\text{cor}}(\text{WW})$	
	Avg.	Max.	Avg.	Max.	Avg.	Max.
NA	414	2772	199	876	176	732
2.5 ps	85	507	21	104	21	98
5 ps	126	910	39	198	25	104
10 ps	44	239	23	99	22	86

of using the Ewald background plasma correction for charged systems.^{1,118–120} Previous works have suggested that one should perform many, short simulations to obtain an average ΔG from independent estimates rather than from running a few, long simulations.^{121–124} The error estimation described in the Computational Details section accounts for correlation within a simulation, but it does not attempt to estimate the error incurred from finite sampling. We show that the discrepancy

between calculation and experiment is not an error caused by using too few λ -states. Nor does the Ewald background plasma correction significantly affect the ΔG values. Finally, we show that although independent simulations yield a large range of pK_a estimates the resulting error estimate of the average is not significantly altered, and no estimate within the set of independent simulations lie within the experimental uncertainty.

Free Energy Dependence on Simulation Cell Size and Ion-Environment. Table 3 examines the convergence of the mononucleotide reference ΔG_{AMP} with respect to system size, and it compares the sensitivity of the result to NaCl concentration. The simulations were performed in a truncated octahedron filled with a single adenine mononucleotide. The 0 M NaCl simulations do not contain NaCl counterions; therefore, the protonated state has a neutral charge and the native state has a net (1-) charge that is balanced by the Ewald background plasma correction as described in ref 89. N_{solv} is the number of water molecules.

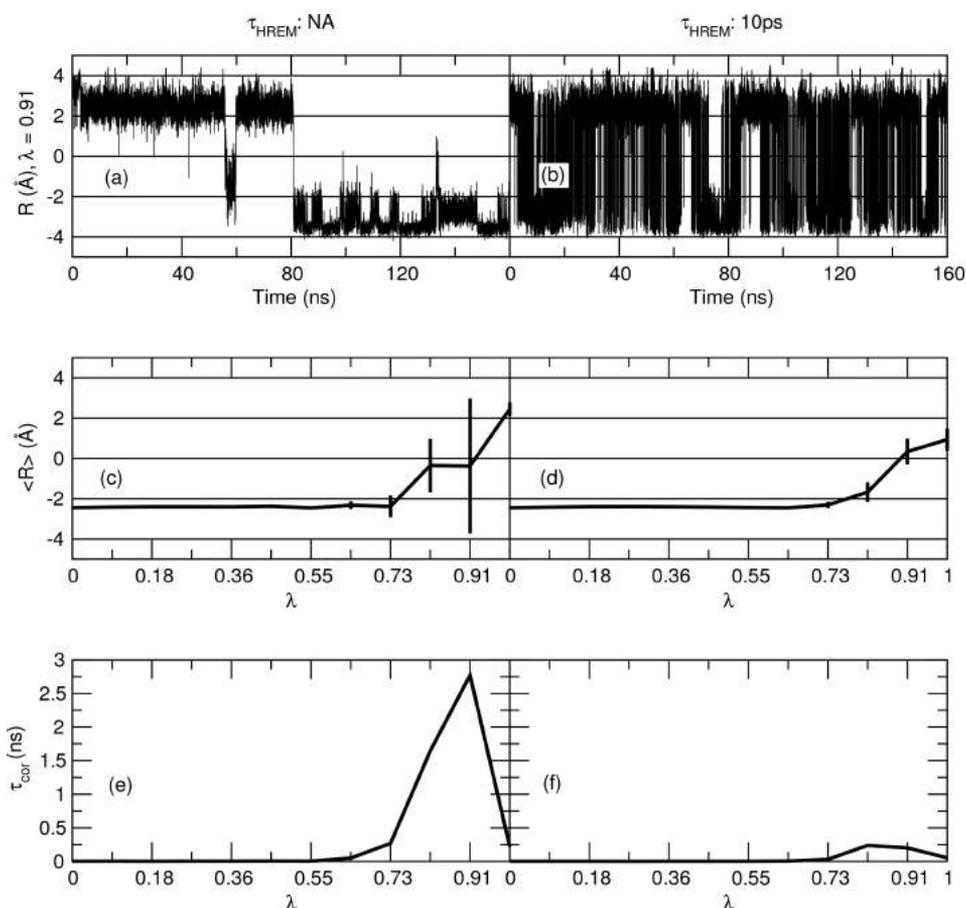


Figure 7. Generalized coordinate is $R = |R_{A:N6} - R_{C:N3}| - |R_{A:N1} - R_{C:N4}|$. The native and protonated geometries shown in Figure 5 are approximately $R = 2 \text{ \AA}$ and $R = -2 \text{ \AA}$, respectively. All plots analyze the SS sequence. The simulations summarized in panels a, c, and e do not use HREM. The simulations in panels b, d, and f use an HREM exchange attempt rate of 10 ps. Panels a and b show how R varies with time in the $\lambda = 0.91$ simulation. Panels c and d are the average value of R in each λ simulation. The vertical bars are 95% confidence intervals of the average. Panels e and f show the $\partial U / \partial \lambda$ autocorrelation time for each λ simulation.

Table 5. Convergence of 5'-Adenosine Monophosphate N1 Deprotonation Free Energy with Respect to the Number of λ Window Simulations^a

N_λ	PI-TI	PI-TI3	PI-BAR	PI-MBAR
10	-77.46 ± 0.02	-77.34 ± 0.02	-77.44 ± 0.02	-77.49 ± 0.02
12	-77.44 ± 0.02	-77.36 ± 0.02	-77.43 ± 0.01	-77.50 ± 0.02
16	-77.40 ± 0.01	-77.36 ± 0.01	-77.40 ± 0.01	-77.47 ± 0.01
24	-77.36 ± 0.01	-77.33 ± 0.01	-77.35 ± 0.01	-77.42 ± 0.01
32	-77.33 ± 0.01	-77.32 ± 0.01	-77.33 ± 0.01	-77.38 ± 0.01

^aValues are in kcal/mol. There are 5579 water molecules and no NaCl counterions.

The 0.14 M NaCl simulations differ only by substituting enough Na^+ and Cl^- ions in the place of water molecules to achieve a physiological ion concentration. The $\lambda = 0$ and $\lambda = 1$ states contain the same number of Na^+ and Cl^- ions. Each transformation was performed with 12 λ -windows ($\lambda = i/11$ for $i \in [0,11]$), and 100 ns of production statistics were produced for each simulation. The convergence of ΔG_{AMP} with respect to the number of λ windows is shown in Table 5.

Comparison between Long MD Runs and Many Short MD Runs. The $\text{p}K_a$ values shown in Table 6 are averages from 16 independent estimates of ΔG_{dsRNA} . Unlike the $\text{p}K_a$ values shown in 2, the results shown in Table 6 are produced from a series of short simulations. Each estimate re-equilibrates every λ -window for 10 ns in the NVT ensemble, starting from either the $\lambda = 0$ or $\lambda = 1$ state. After equilibration, 10 ns of NVT production statistics

are performed for analysis, and the $\text{p}K_a$ is calculated from eq 43. The listed $\text{p}K_a$ value is the average of the 16 estimates:

$$\langle \text{p}K_a \rangle = \frac{1}{16} \sum_{i=1}^{16} \text{p}K_{a,i} \quad (48)$$

The \pm values in Table 6 are standard errors of the mean:

$$\sigma_{\langle \text{p}K_a \rangle} = \sqrt{\frac{\sigma_{\{\text{p}K_{a,i}\}}^2 + \sum_{i=1}^{16} \sigma_{\text{p}K_{a,i}}^2}{16}} \quad (49)$$

where $\sigma_{\{\text{p}K_{a,i}\}}$ is the standard deviation of the 16 estimates, and the summation accounts for the error of each estimate (eq 44). Table 6 also lists the maximum and minimum $\text{p}K_{a,i}$ values from the 16 estimates. The $\langle \Delta \text{p}K_a \rangle$ values and errors are analogous to eqs 45

Table 6. Distribution of pK_a Values from 16 Simulations, Each Providing 10 ns of Production Statistics

τ_{HREM}		PI-TI	PI-TI3	PI-BAR	PI-MBAR
$pK_a(\text{SS})$					
NA	Avg.	6.71 ± 0.17	6.67 ± 0.17	6.70 ± 0.11	6.73 ± 0.11
	Min.	6.11	6.07	6.10	6.15
	Max.	7.19	7.12	7.19	7.24
5 ps	Avg.	6.86 ± 0.14	6.85 ± 0.14	6.86 ± 0.10	6.88 ± 0.11
	Min.	6.22	6.18	6.21	6.24
	Max.	7.34	7.39	7.35	7.32
$pK_a(\text{SW})$					
NA	Avg.	5.92 ± 0.21	5.88 ± 0.21	5.91 ± 0.13	5.93 ± 0.14
	Min.	5.20	5.18	5.20	5.22
	Max.	6.37	6.32	6.37	6.39
5 ps	Avg.	6.04 ± 0.12	6.02 ± 0.12	6.04 ± 0.09	6.06 ± 0.10
	Min.	5.60	5.56	5.59	5.63
	Max.	6.37	6.37	6.37	6.38
$pK_a(\text{WW})$					
NA	Avg.	5.56 ± 0.18	5.54 ± 0.19	5.56 ± 0.11	5.57 ± 0.11
	Min.	5.12	5.11	5.11	5.13
	Max.	5.96	5.98	5.96	5.93
5 ps	Avg.	5.53 ± 0.12	5.53 ± 0.13	5.53 ± 0.09	5.53 ± 0.10
	Min.	5.10	5.05	5.09	5.12
	Max.	5.91	5.91	5.90	5.90
$\langle \Delta pK_a(\text{SS}) \rangle = \langle pK_a(\text{SS}) \rangle - \langle pK_a(\text{WW}) \rangle$					
NA	Avg.	1.14 ± 0.25	1.13 ± 0.25	1.14 ± 0.15	1.16 ± 0.16
5 ps	Avg.	1.34 ± 0.18	1.32 ± 0.19	1.34 ± 0.14	1.35 ± 0.15
$\langle \Delta pK_a(\text{SW}) \rangle = \langle pK_a(\text{SW}) \rangle - \langle pK_a(\text{WW}) \rangle$					
NA	Avg.	0.35 ± 0.28	0.34 ± 0.28	0.35 ± 0.17	0.36 ± 0.18
5 ps	Avg.	0.51 ± 0.17	0.49 ± 0.17	0.51 ± 0.13	0.53 ± 0.14

Table 7. Effect of Polarized MM Charges on pK_a Values

Model	Residues	pK_a			ΔpK_a	
		SS	SW	WW	SS-WW	SW-WW
Expt.		8.10 ± 0.06	7.28 ± 0.08	6.51 ± 0.04	1.59 ± 0.07	0.77 ± 0.09
ff14SB		6.84 ± 0.11	5.90 ± 0.06	5.49 ± 0.07	1.35 ± 0.13	0.40 ± 0.09
Shifted pK_a values						
ref 125	4+	6.87 ± 0.11	5.97 ± 0.06	5.50 ± 0.07	1.37 ± 0.13	0.47 ± 0.09
15%	4	7.49 ± 0.11	6.70 ± 0.06	6.25 ± 0.07	1.24 ± 0.13	0.46 ± 0.09
15%	4,17	8.40 ± 0.11	7.56 ± 0.06	7.03 ± 0.07	1.37 ± 0.13	0.53 ± 0.09
15%	3-5,16-18	8.23 ± 0.11	7.29 ± 0.06	6.58 ± 0.07	1.65 ± 0.13	0.71 ± 0.09
15%	1-20	7.84 ± 0.12	7.03 ± 0.08	6.50 ± 0.08	1.34 ± 0.14	0.53 ± 0.11
LDA	3-5,16-18	8.58 ± 0.11	7.30 ± 0.06	6.63 ± 0.07	1.96 ± 0.13	0.67 ± 0.09
GGA	3-5,16-18	8.29 ± 0.11	7.12 ± 0.06	6.35 ± 0.07	1.94 ± 0.13	0.77 ± 0.09
Hyb	3-5,16-18	8.62 ± 0.11	7.42 ± 0.06	6.64 ± 0.07	1.98 ± 0.13	0.78 ± 0.09
HF	3-5,16-18	8.77 ± 0.12	7.77 ± 0.07	7.01 ± 0.07	1.76 ± 0.14	0.76 ± 0.10
pK_a shifts relative to ff14SB						
ref 125	4+	0.03 ± 0.00	0.08 ± 0.00	0.01 ± 0.00	0.02 ± 0.01	0.07 ± 0.01
15%	4	0.65 ± 0.01	0.81 ± 0.01	0.75 ± 0.01	-0.10 ± 0.01	0.05 ± 0.01
15%	4,17	1.55 ± 0.01	1.66 ± 0.01	1.53 ± 0.01	0.02 ± 0.02	0.13 ± 0.02
15%	3-5,16-18	1.39 ± 0.02	1.40 ± 0.02	1.09 ± 0.02	0.30 ± 0.03	0.31 ± 0.02
15%	1-20	1.00 ± 0.04	1.14 ± 0.04	1.01 ± 0.04	-0.01 ± 0.05	0.13 ± 0.06
LDA	3-5,16-18	1.74 ± 0.02	1.40 ± 0.02	1.14 ± 0.02	0.61 ± 0.02	0.27 ± 0.02
GGA	3-5,16-18	1.45 ± 0.02	1.23 ± 0.01	0.86 ± 0.01	0.59 ± 0.02	0.37 ± 0.02
Hyb	3-5,16-18	1.78 ± 0.02	1.53 ± 0.02	1.15 ± 0.02	0.63 ± 0.02	0.38 ± 0.02
HF	3-5,16-18	1.93 ± 0.03	1.87 ± 0.03	1.52 ± 0.02	0.41 ± 0.04	0.36 ± 0.04

and 46. The $pK_{a,i}$ and $\sigma_{pK_{a,i}}$ values summarized in Table 6 can be found in the Supporting Information.

2.6. Improving pK_a Prediction with QM/MM Charge Fitting. The calculated A4 pK_a values shown in Tables 2 and 6 are about 1.2 pK_a units smaller than the experimental data. Table 7

summarizes how the pK_a values change when different MM charge sets are used. Our intention is to examine how sensitive the pK_a values are to the local charge environment by using different charges for the A4 nucleobase, the A4-C17 base pair, and the surrounding bases to gain an understanding of the degree to which one might expect the pK_a values to shift as a result.

If alternate charge sets do not significantly shift the pK_a values toward the experimental values, then some other explanation would be needed to rationalize the discrepancy between our MM results and the experimental data. Given this context, we are not advocating for general use any particular charge set based on the limited amount of data explored in this work.

The “Model” column in Table 7 labels the charge set. All charge sets used in Table 7 are provided in the Supporting Information. For example, the “ff14SB” charges are the force field parameters used in Table 2, and the listed pK_a values are the average PI-TI3 results from the four simulations listed in Table 2. The “Residues” column lists the nucleobase residue indices to which the model charges are applied. In other words, all atoms use ff14SB charges *except* for the explicitly listed nucleobases. The rows marked “4+” use the model charges for the protonated A4 nucleobase, but not the native A4 nucleobase. The rows marked “4” use the model charges for both the protonated and neutral A4 nucleobases. The rows marked “4,17” use the model charges for the A4-C17 basepair. The rows marked “3–5,16–18” include the bases immediately above and below A4-C17. The rows marked “1–20” include all nucleobases in the dsRNA.

The “ref 125” charge set uses the protonated adenine nucleobase charges developed in ref 125. The “15%” charge set uniformly polarizes the ff14SB nucleobase charges by a factor of 1.15, that is,

$$q_{a,15\%} = \langle q \rangle + 1.15(q_{a,\text{ff14SB}} - \langle q \rangle) \quad (50)$$

where $q_{a,\text{ff14SB}}$ is the ff14SB charge of nucleobase atom a , $\langle q \rangle$ is the average atomic nucleobase charge, and $q_{a,15\%}$ is the “15%” polarized charge of atom a . Equation 50 merely increases the magnitude of each nucleobase charge without having changed the nucleobase’s net charge. The “LDA” (SVWN3/6-31G*), “GGA” (PBE/6-31G*), “Hyb” (PBE0/6-31G*), and “HF” (HF/6-31G*) charges are Restrained Electrostatic Potential (RESP) fits¹²⁶ to *ab initio* QM/MM calculations. RESP fits are normally performed from gas phase calculations using HF/6-31G* because Hartree–Fock theory tends to “over-polarize” molecules and coincidentally happens to provide charges that mimic the polarization found in solution. The RESP fits performed in this work are not based upon gas phase calculations. Instead, we perform QM/MM calculations of the dsRNA molecule and perform RESP fits of the QM region using the density matrix that has been polarized by the MM environment. One should, therefore, expect HF/6-31G* to yield charges that are far too polarized because not only does HF overpolarize the charges in the gas-phase, the density matrix is explicitly polarized by the MM environment as well. In total, we performed 2880 RESP fits. RESP fits were performed for residues 3, 4, 5, 16, 17, and 18 in each of the six sequences (native and protonated-A4 SS, SW, WW sequences). Each RESP fit was performed with 80 snapshots taken from the 160 ns of MM production simulations. The nucleobase atomic charges were then averaged to yield a single set of U, G, C, A, and protonated-A charges. To perform one RESP fit, we selected a nucleobase as the QM region. Because the MM nucleobase charges do not sum to an integer value, the excess nucleobase charge was uniformly distributed to the nucleotide sugar ring while performing the QM calculation. Similarly, after performing the RESP fit, the link atom charge was summed in to the heavy-atom charge, and the RESP charges were uniformly shifted to preserve the original MM net nucleobase charge.

To obtain the pK_a values with the alternate set of model charges, we do not use the procedure described by eqs 42 and 43. Instead, we calculate the (PI-TI3) free energy change of transforming the

ff14SB charges (the MM charges) to the alternate model charges (the MM’ charges) for the dsRNA and AMP protonated and deprotonated states. The pK_a shift associated with polarizing the charges is given by eq 51, and the pK_a estimate using the polarized charges is given by eq 52.

$$\Delta pK_{a,\text{MM} \rightarrow \text{MM}'} = [RT \ln(10)]^{-1} \times \left(\Delta G_{\text{dsRNA},\text{MM} \rightarrow \text{MM}'}^{\text{protonated}} - \Delta G_{\text{dsRNA},\text{MM} \rightarrow \text{MM}'}^{\text{native}} + \Delta G_{\text{AMP},\text{MM} \rightarrow \text{MM}'}^{\text{native}} - \Delta G_{\text{AMP},\text{MM} \rightarrow \text{MM}'}^{\text{protonated}} \right) \quad (51)$$

$$pK_{a,\text{MM}'} = pK_{a,\text{MM}} + \Delta pK_{a,\text{MM} \rightarrow \text{MM}'} \quad (52)$$

The free energy of the MM \rightarrow MM’ transformations converge rapidly with time because the proton is neither appearing nor disappearing. We found it sufficient to perform each MM \rightarrow MM’ calculation from 40 ns of production statistics using 12 λ -windows.

3. RESULTS AND DISCUSSION

3.1. Comparisons between TI Methods. The $\text{Mg}^{2+} \rightarrow \text{Ca}^{2+}$ transformations shown in Figures 1–3 and Table 1 are used to compare 1-step and 3-step PI-TI methods to Std.-TI and SC-TI. Specifically, Table 1 and Figure 3 demonstrate that the PI-TI free energy difference between the two states agrees with the Std.-TI and SC-TI calculations; however, Figures 1 and 2 illustrate that the methods differ by the behavior of their intermediate λ -paths.

The Std.-TI method linearly mixes the end-state potential energies, so the neutral $\text{Mg}^0 \rightarrow \text{Ca}^0$ Std.-TI potential energies shown in Figure 1a are stationary with respect to λ wherever the LJ potentials of the two end-states intersect. As one proceeds from $\lambda = 0$ to $\lambda = 0.17$, the repulsive wall of the Std.-TI LJ potential quickly approaches that of the larger Ca^{2+} ion. Consequently, the potentials shown in Figure 1d do not uniformly transform between the end-states.

Within the scale from -0.2 to 0.2 kcal/mol, the neutral atom SC-TI (Figure 1b) and PI-TI (Figure 1c) potentials behave similarly; however, when viewed on a larger scale, one finds that the repulsive wall of the intermediate SC-TI λ -potentials are not as steep as those produced by PI-TI. The steepness of the LJ potentials account for the dramatic difference between the potentials shown in Figure 1e and f. The Std.-TI and PI-TI methods produce the same discharge and recharge free energies (Table 1); however, the slope of their $\langle \partial U / \partial \lambda \rangle_\lambda$ profiles differ in Figure 2a and b because $U_{\text{Std.-TI}}(\lambda)$ is a linear function of λ , whereas $U_{\text{PI-TI}}(\lambda)$ is a quadratic function of charge (and thus λ). In this particular example, where only 1 atom is changing its charge, the energy’s quadratic behavior is exhibited by the charge-neutralizing background plasma correction. That is, the electrostatic potential is linearly shifted by the system’s net charge, which is changing as a function of λ (eq 25). The observed shape of the $\langle \partial U / \partial \lambda \rangle_\lambda$ profiles also depends on the ensemble distribution. For example, the derivative $\partial U_{\text{Std.-TI}} / \partial \lambda$ is a constant with respect to λ , but the ensemble average $\langle \partial U_{\text{Std.-TI}} / \partial \lambda \rangle_\lambda$ has a significant nonzero slope. The ensemble distributions produced by charge-changing perturbations dominate the overall shape of the profiles, so the Std.-TI and PI-TI profiles both appear to be relatively linear.

The PI-TI 3-step LJ transformation displays a near-linear $\langle \partial U / \partial \lambda \rangle_\lambda$ profile in Figure 2c because the PI-TI intermediate potentials uniformly shift and scale the end-state potentials (Figure 1c). In contrast, the Std.-TI method is better described as showing exponential behavior due to the sudden change in the position of the repulsive wall (Figure 1a). The behavior of the SC-TI method in Figure 1c is more complicated because the

SC-TI method does not attempt to directly transform the Mg^{2+} to a Ca^{2+} . Instead, Mg^{2+} is effectively transformed to “nothing”, while Ca^{2+} is “grown” into the system from a vacuum.

Table 1 shows how the net free energies vary as a function of the number of λ -states, N_λ . Previous work has noted that parameter-interpolated alchemical pathways can lead to $\langle \partial U / \partial \lambda \rangle_\lambda$ profiles that are more linear than those produced from Std.-TI.¹¹ If the profiles were sufficiently linear, then it may be possible to perform PI-TI transformations with fewer intermediate λ -states than what Std.-TI would require to achieve the same accuracy. The 1-step PI-TI3 results are nearly independent of N_λ , whereas the 1-step Std.-TI3 results are the most sensitive. The 3-step variants of Std.-TI3, SC-TI3, and PI-TI3 methods display similar sensitivities to one another because their dominant sources of error occur in the discharge and recharge steps, which produce $\langle \partial U / \partial \lambda \rangle_\lambda$ profiles that are approximately linear for all three methods. In other words, parameter-interpolated pathways can certainly lead to $\langle \partial U / \partial \lambda \rangle_\lambda$ profiles that are more linear; however, the benefit of the improved linearity is diminished if the transformation is divided into three steps: discharge, LJ change, and recharge. If the transformation is divided into steps, then the improved linearity is largely confined to the uncharged LJ change, which often makes a relatively small contribution to the net free energy.

3.2. Double-Stranded RNA pK_a Benchmark Simulations. *Comparison between TI, BAR, and MBAR.* In Table 5, we observed differences in the TI, BAR, and MBAR calculations of the AMP reference system—particularly when few λ -windows are used. The simulation results summarized in Table 2 were evaluated with $N_\lambda = 12$; however, the pK_a and ΔpK_a values computed from the various methods are in close agreement because the transformations in the dsRNA and AMP systems are sufficiently similar that systematic errors are likely canceled. The largest pK_a difference between PI-TI, PI-TI3, PI-BAR, and PI-MBAR in Table 2 is only 0.04, which is within the standard error of the calculations.

The calculated pK_a values are approximately from 1.0 to 1.3 units smaller than the experiment values. In contrast, the calculated ΔpK_a values differ from experiments only by 0.2 to 0.3 units.

Comparison between HREM and Standard MD Results. Figure 6 shows that the standard MD simulations have larger error estimates than the HREM simulations. Furthermore, the errors are not particularly sensitive to the chosen time-interval between Hamiltonian exchange attempts. By exchanging λ -window Hamiltonians, one reduces the $\partial U / \partial \lambda$ autocorrelation times (Table 4). This increases the number of statistically independent samples and reduces the estimated error. Specifically, HREM reduces the maximum autocorrelation time by a factor of 1/3 or less.

Figure 7 illustrates that the large autocorrelation times observed in the non-HREM simulations is due to infrequent flipping between the two A4-C17 base pair orientations shown in Figure 5. The largest non-HREM τ_{cor} value occurs in the $\lambda = 0.91$ simulation (Figure 7e). The $\lambda = 0.91$ simulation also has the largest uncertainty in the average value of R (Figure 7c). The large uncertainty is produced from the infrequent transition between the two orientations (Figure 7a). The use of HREM (Figure 7b, d, and f) increases the frequency of the transitions, lowers the uncertainty in $\langle R \rangle$, and decreases the $\partial U / \partial \lambda$ τ_{cor} value.

3.3. Validation of Benchmark pK_a Calculations. *Free Energy Dependence on Simulation Cell Size and Ion-Environment.* The best strategy for calculating free energy differences in periodic, condensed phase systems is a subject of debate when the transformation involves a change in net charge.^{1,118–120} For example, the excess charge in one end-state can be accounted

for through the Ewald uniform background correction (as is done in the present work) or by distributing a neutralizing charge to other atoms to mimic a changed ion atmosphere. The pK_a calculations performed in the present work are made relative to a reference system (AMP), which should act to cancel systematic errors from the final result.¹²⁷ Our main concern is to verify that the computed free energies are converged with respect to system size because the Ewald uniform background plasma correction has a volume dependence. Table 3 shows that the AMP free energies converge to within 0.02 kcal/mol once the lattice vectors are larger than 47.6 Å. We completely removed the NaCl salt environment to further test the sensitivity of the result on ion atmosphere, and the free energy change differs only by 0.09 kcal/mol. We conclude that the discrepancy between our benchmark pK_a calculations and experiment values are not due to our strategy of using the Ewald uniform background correction to model systems with a net charge.

Table 5 tests the convergence of the AMP free energy with respect to the number of simulated λ -windows, N_λ . The PI-TI3 method is nearly independent of N_λ . The PI-TI and PI-BAR methods converge to the PI-TI3 value for $N_\lambda \gtrsim 24$. The PI-MBAR method is shifted from PI-BAR by approximately 0.05 kcal/mol but approaches the PI-TI3 result. The variations observed in Table 5 are an order of magnitude smaller than the differences between our benchmark pK_a calculations and the experiment values.

Comparison between Long MD Runs and Many Short MD Runs. The standard error estimate described by eq 40 assumes that the simulations have sampled all relevant areas of phase space. That is, eq 40 does not account for the error associated with the simulation's finite sampling.¹²⁸ For this reason, the simulations summarized in Figure 6 were run for 160 ns, which is more than 50 times longer than the maximum $\partial U / \partial \lambda$ autocorrelation time found in any λ -window simulation. Indeed, the simulation profiles shown in Figure 6 reasonably agree with one another after 100 ns. An alternative approach for estimating free energy averages and errors is to obtain a set of estimates from a series of short simulations starting from different initial conditions.¹²⁹ It has been suggested that this “ensemble average approach”^{121–124} may enhance reproducibility of equilibrium thermodynamic properties relative to single, long simulations, although this result is likely case dependent. Table 6 summarizes the distribution of pK_a values obtained from 16 independent TI calculations, each simulation providing 10 ns of production statistics. The initial conditions differ by having re-equilibrated each simulation for 10 ns with a different thermostat random number seed. We used 16 estimates so that the aggregate amount of production statistics (160 ns) matched our long simulations.

The pK_a averages shown in Table 6 agree with the long simulations within the reported errors. The minimum and maximum pK_a values span a range of 1 pK_a unit, which should make the reader skeptical of any single result obtained from a 10 ns simulation. Furthermore, the short simulations do not appear to significantly benefit from using HREM in the present example. We suspect that this is due, in part, to the fact that the short simulations severely underestimate the true $\partial U / \partial \lambda$ autocorrelation time. Moreover, it should be emphasized that the double-stranded RNA construct surrounding the mismatched A-C base pair is relatively stable and thus might not be reflective of RNA molecules with more complex tertiary structure. The maximum pK_a values shown in Table 6 are from 0.6 to 0.9 pK_a units smaller than the experiment values.

3.4. Improving pK_a Prediction with QM/MM Charge Fitting. Table 7 suggests that the 1.0 to 1.3 pK_a unit discrepancy

between our ff14SB MM results and the experiment values is due to the nearby nucleobase atomic charge values. If we merely replace the protonated-A4 atomic charges with an alternative charge set developed in ref 125, we obtain ostensibly similar pK_a results. The series of 15% polarized charges shown in Table 7 indicate that enhancing the polarization of the A4-C17 base pair and the immediately surrounding bases causes from a 1.1 to 1.4 pK_a unit shift toward the experiment values. Polarizing all other bases by 15% has much less of an effect by comparison.

All *ab initio* derived charges are applied to the residues 3–5, 16–18 in Table 7 because the surrounding residues have the greatest effect on the pK_a value. Furthermore, we wanted to limit the size of the “TI region” to prevent our pK_a predictions from becoming infected by subtle changes that may occur in distant, irrelevant areas of the dsRNA which would need to be canceled in both the native and protonated dsRNA transformations. The HF charges yield pK_a values that are larger than the experimental results, which we expected because the HF charges should be overpolarized by having performed the RESP fit in the dsRNA environment. The GGA (PBE/6-31G*) charges match experiments better than the LDA (SVWN3/6-31G*) or hybrid (PBE0/6-31G*) charges. To get a more general feel for how well the DFT-derived charge sets perform, we average the LDA, GGA, and hybrid-DFT pK_a values: 8.50 ± 0.14 (SS), 7.29 ± 0.09 (SW), and 6.54 ± 0.10 (WW). The averages are in good agreement with the experimental values. This suggests that emerging fragment-based *ab initio* methods may be useful for deriving MM charges for improved pK_a prediction.^{130–133} Furthermore, it may be beneficial to perform multiple pK_a calculations using several charge sets to incorporate the uncertainty of the model charges within the final result.

4. CONCLUSIONS

We develop and test a PI-TI method for thermodynamic integration free energy simulations on GPUs. The primary motivation for using the parameter-interpolated thermodynamic integration method is to allow propagation of the dynamics of a system with a standard MM potential energy, as opposed to the conventional linear alchemical interpolated potential energy. By doing so, one can immediately take advantage of GPU-accelerated MD software for performing the dynamics. Furthermore, additional program functionality may then become available for use, such as Hamiltonian replica exchange MD algorithms in AMBER. The TI gradient and MBAR energy analysis of the statistically independent trajectory frames can be efficiently post-processed in parallel with minimal overhead relative to the evaluation of the MD simulations themselves. Our ability to perform the dynamics with standard MM potential energy surfaces and then post-process the trajectories is made possible by having reformulated the reciprocal-space PME contribution to the TI gradient through differential chain-rule relationships. This approach differs from previous single-topology implementations which either do not support PME electrostatics with TI, e.g., GROMOS,^{52,53} or resort to a dual-topology-like Hamiltonian-mixing of two PME reciprocal-space calculations, e.g., GROMACS.^{26,34,55} As a consequence, the PI-TI algorithm does not require a specific single-topology TI implementation to propagate the dynamics, and it affords the opportunity to use GPU-accelerated software without interfering with sophisticated hardware optimizations that have been tuned to improve standard MD simulation performance. Furthermore, the AMBER community has established literature indicating that the use of two PME calculations for performing dual-topology TI slows the CPU simulation performance by 25% relative to standard MD simulations⁵⁷ and slows the GPU perfor-

mance by 30%.³⁰ Thus, the PI-TI algorithm described in the present work offers a massive performance improvement relative to the approach implemented within AMBER. The PI-TI method is suitable for transformations that involve a change in parameters, such as metal ion transformations, pK_a calculations, but it is not a general replacement for using soft-core TI to decouple or exchange whole ligands. Nonetheless, the PI-TI method may be generalized so as to be used with soft-core potentials and GPU-accelerated capability as implementations of those potentials become available.

In this work, we have found that parameter-interpolated pathways can lead to TI gradient profiles that are significantly more linear than those produced by the standard TI method. Therefore, fewer intermediate λ -states may be required to obtain accurate free energy predictions, depending on the transformation.

The PI-TI method afforded us the opportunity to use HREM to perform Monte Carlo exchanges between the λ -states during dynamics. We found that HREM reduces the $\partial U/\partial \lambda$ autocorrelation times and leads to smaller free energy error estimates.

We calculated the pK_a values of small double stranded RNA sequences and made comparison with experimental values. We performed our simulations with the GPU-accelerated version of PMEMD to extend our production to 160 ns, and we found that approximately 100 ns of statistics were necessary to obtain converged, reproducible results to within 0.25 pK_a units. The calculated pK_a shifts between the sequences reasonably agree with the experimental results; however, the absolute pK_a calculations are all too small by about 1.2 units. Furthermore, we examined how various MM charge sets affect the pK_a values and found good agreement with experiments when the surrounding nucleobase charges are replaced with DFT-derived RESP charges, where the DFT density matrix is polarized in a QM/MM environment. This result suggests that fragment-based *ab initio* methods may play a useful role in refining the quality of the electrostatic environment and lead to improved pK_a prediction.

■ ASSOCIATED CONTENT

Supporting Information

The Supporting Information is available free of charge on the ACS Publications website at DOI: 10.1021/acs.jctc.7b01175.

Tables of nucleobase charges and tables listing the full set of pK_a results summarized in Table 6. (PDF)

■ AUTHOR INFORMATION

Corresponding Author

*E-mail: Darrin.York@rutgers.edu.

ORCID

Darrin M. York: 0000-0002-9193-7055

Notes

The authors declare no competing financial interest.

■ ACKNOWLEDGMENTS

The authors are grateful for financial support provided by the National Institutes of Health (No. GM107485 and GM62248). Computational resources were provided by the Office of Advanced Research Computing (OARC) at Rutgers, The State University of New Jersey, the National Institutes of Health under Grant No. S10OD012346, and by the Extreme Science and Engineering Discovery Environment (XSEDE), which is supported by National Science Foundation (Grants No. ACI-1548562 and No. OCI-1053575).

REFERENCES

- (1) Lin, Y.-L.; Aleksandrov, A.; Simonson, T.; Roux, B. An Overview of Electrostatic Free Energy Computations for Solutions and Proteins. *J. Chem. Theory Comput.* **2014**, *10*, 2690–2709.
- (2) Hansen, N.; van Gunsteren, W. F. Practical Aspects of Free-Energy Calculations: A Review. *J. Chem. Theory Comput.* **2014**, *10*, 2632–2647.
- (3) Chodera, J.; Mobley, D.; Shirts, M.; Dixon, R.; Branson, K.; Pande, V. Alchemical free energy methods for drug discovery: progress and challenges. *Curr. Opin. Struct. Biol.* **2011**, *21*, 150–160.
- (4) Christ, C.; Mark, A.; van Gunsteren, W. Basic Ingredients of Free Energy Calculations: A Review. *J. Comput. Chem.* **2010**, *31*, 1569–1582.
- (5) Resat, H.; Mezei, M. Studies on free energy calculations. I. Thermodynamic integration using a polynomial path. *J. Chem. Phys.* **1993**, *99*, 6052–6061.
- (6) Bruckner, S.; Boresch, S. Efficiency of alchemical free energy simulations. II. Improvements for thermodynamic integration. *J. Comput. Chem.* **2011**, *32*, 1320–1333.
- (7) Zwanzig, R. W. High-temperature equation of state by a perturbation method. I. Nonpolar gases. *J. Chem. Phys.* **1954**, *22*, 1420–1426.
- (8) Bennett, C. H. Efficient estimation of free energy differences from Monte Carlo data. *J. Comput. Phys.* **1976**, *22*, 245–268.
- (9) Shirts, M. R.; Chodera, J. D. Statistically optimal analysis of samples from multiple equilibrium states. *J. Chem. Phys.* **2008**, *129*, 124105.
- (10) Kirkwood, J. G. Statistical mechanics of fluid mixtures. *J. Chem. Phys.* **1935**, *3*, 300–313.
- (11) Cross, A. J. A comment on Hamiltonian parametrization in Kirkwood free energy calculations. *Ann. N. Y. Acad. Sci.* **1986**, *482*, 89–90.
- (12) Mruzik, M. R.; Abraham, F. F.; Pound, G. M.; Schreiber, D. E. A Monte Carlo study of ion-water clusters. *J. Chem. Phys.* **1976**, *64*, 481–491.
- (13) Mezei, M.; Swaminathan, S.; Beveridge, D. L. Ab Initio calculation of the Free Energy of Liquid Water. *J. Am. Chem. Soc.* **1978**, *100*, 3255–3256.
- (14) Berens, P. H.; Mackay, D. H. J.; White, G. M.; Wilson, K. R. Thermodynamics and quantum corrections from molecular dynamics for liquid water. *J. Chem. Phys.* **1983**, *79*, 2375–2389.
- (15) Christ, C. D.; Fox, T. Accuracy assessment and automation of free energy calculations for drug design. *J. Chem. Inf. Model.* **2014**, *54*, 108–120.
- (16) Woods, C. J.; Essex, J. W.; King, M. A. The development of replica-exchange-based free-energy methods. *J. Phys. Chem. B* **2003**, *107*, 13703–13710.
- (17) Meng, Y.; Dashti, D.; Roitberg, A. E. Computing alchemical free energy differences with Hamiltonian replica exchange molecular dynamics (H-REMD) simulations. *J. Chem. Theory Comput.* **2011**, *7*, 2721–2727.
- (18) Jiang, W.; Hodoscek, M.; Roux, B. Computation of absolute hydration and binding free energy with free energy perturbation distributed replica-exchange molecular dynamics. *J. Chem. Theory Comput.* **2009**, *5*, 2583–2588.
- (19) Gil-Ley, A.; Bussi, G. Enhanced Conformational Sampling Using Replica Exchange with Collective-Variable Tempering. *J. Chem. Theory Comput.* **2015**, *11*, 1077–1085.
- (20) Stone, J. E.; Phillips, J. C.; Freddolino, P. L.; Hardy, D. J.; Trabuco, L. G.; Schulten, K. Accelerating molecular modeling applications with graphics processors. *J. Comput. Chem.* **2007**, *28*, 2618–2640.
- (21) Harvey, M. J.; Giupponi, G.; Fabritiis, G. D. ACEMD: Accelerating Biomolecular Dynamics in the Microsecond Time Scale. *J. Chem. Theory Comput.* **2009**, *5*, 1632–1639.
- (22) Götz, A.; Williamson, M. J.; Xu, D.; Poole, D.; Le Grand, S.; Walker, R. C. Routine microsecond molecular dynamics simulations with AMBER on GPUs. 1. Generalized Born. *J. Chem. Theory Comput.* **2012**, *8*, 1542–1555.
- (23) Salomon-Ferrer, R.; Götz, A. W.; Poole, D.; Le Grand, S.; Walker, R. C. Routine microsecond molecular dynamics simulations with AMBER on GPUs. 2. Explicit solvent Particle Mesh Ewald. *J. Chem. Theory Comput.* **2013**, *9*, 3878–3888.
- (24) Le Grand, S.; Götz, A. W.; Walker, R. C. SPFP: Speed without compromise—A mixed precision model for GPU accelerated molecular dynamics simulations. *Comput. Phys. Commun.* **2013**, *184*, 374–380.
- (25) Eastman, P.; Pande, V. S. Efficient nonbonded interactions for molecular dynamics on a graphics processing unit. *J. Comput. Chem.* **2010**, *31*, 1268–1272.
- (26) Pronk, S.; Páll, S.; Schulz, R.; Larsson, P.; Bjelkmar, P.; Apostolov, R.; Shirts, M. R.; Smith, J. C.; Kasson, P. M.; van der Spoel, D.; Hess, B.; Lindahl, E. GROMACS 4.5: a high-throughput and highly parallel open source molecular simulation toolkit. *Bioinformatics* **2013**, *29*, 845–854.
- (27) Kutzner, C.; Páll, S.; Fechner, M.; Esztermann, A.; de Groot, B. L.; Grubmüller, H. Best bang for your buck: GPU nodes for GROMACS biomolecular simulations. *J. Comput. Chem.* **2015**, *36*, 1990–2008.
- (28) Arthur, E. J.; Brooks, C. L. Efficient implementation of constant pH molecular dynamics on modern graphics processors. *J. Comput. Chem.* **2016**, *37*, 2171–2180.
- (29) Bergdorf, M.; Kim, E. T.; Rendleman, C. A.; Shaw, D. E. *Desmond/GPU Performance as of November 2014*; Technical Report DESRESTR-2014-01, 2014.
- (30) Lee, T.-S.; Hu, Y.; Sherborne, B.; Guo, Z.; York, D. M. Toward Fast and Accurate Binding Affinity Prediction with pmemdGTT: An Efficient Implementation of GPU-Accelerated Thermodynamic Integration. *J. Chem. Theory Comput.* **2017**, *13*, 3077–3084.
- (31) Wang, K.; Chodera, J.; Yang, Y.; Shirts, M. R. Identifying ligand binding sites and poses using GPU-accelerated Hamiltonian replica exchange molecular dynamics. *J. Comput.-Aided Mol. Des.* **2013**, *27*, 989.
- (32) Donnini, S.; Tegeler, F.; Groenhof, G.; Grubmüller, H. Constant pH molecular dynamics in explicit solvent with λ -dynamics. *J. Chem. Theory Comput.* **2011**, *7*, 1962–1978.
- (33) Eastman, P.; Swails, J.; Chodera, J. D.; McGibbon, R. T.; Zhao, Y.; Beauchamp, K. A.; Wang, L.-P.; Simmonett, A. C.; Harrigan, M. P.; Stern, C. D.; Wiewiora, R. P.; Brooks, B. R.; Pande, V. S. OpenMM 7: Rapid development of high performance algorithms for molecular dynamics. *PLoS Comput. Biol.* **2017**, *13*, 1005659–1005659.
- (34) Peng, X.; Zhang, Y.; Chu, H.; Li, G. Free energy simulations with the AMOEBA polarizable force field and metadynamics on GPU platform. *J. Comput. Chem.* **2016**, *37*, 614–622.
- (35) Eastman, P.; Pande, V. S. OpenMM: A Hardware Independent Framework for Molecular Simulations. *Comput. Sci. Eng.* **2010**, *12*, 34–39.
- (36) Eastman, P.; Friedrichs, M. S.; Chodera, J. D.; Radmer, R. J.; Bruns, C. M.; Ku, J. P.; Beauchamp, K. A.; Lane, T. J.; Wang, L.-P.; Shukla, D.; Tye, T.; Houston, M.; Stich, T.; Klein, C.; Shirts, M. R.; Pande, V. S. OpenMM 4: A Reusable, Extensible, Hardware Independent Library for High Performance Molecular Simulation. *J. Chem. Theory Comput.* **2013**, *9*, 461–469.
- (37) Boresch, S.; Bruckner, S. Avoiding the van der Waals endpoint problem using serial atomic insertion. *J. Comput. Chem.* **2011**, *32*, 2449–2458.
- (38) Jorgensen, W. L.; Ravimohan, C. Monte Carlo simulation of differences in free energies of hydration. *J. Chem. Phys.* **1985**, *83*, 3050–3054.
- (39) Jorgensen, W. L.; Tirado-Rives, J. Molecular Modeling of Organic and Biomolecular Systems Using BOSS and MCPRO. *J. Comput. Chem.* **2005**, *26*, 1689–1700.
- (40) Eastman, P.; Swails, J.; Chodera, J. D.; McGibbon, R. T.; Zhao, Y.; Beauchamp, K. A.; Wang, L. P.; Harrigan, M. P.; Stern, C. D.; Wiewiora, R. P.; Brooks, B. R.; Pande, V. S.; Simmonett, A. C. OpenMM 7: Rapid development of high performance algorithms for molecular dynamics. *PLoS Comput. Biol.* **2017**, *13*, e1005659.
- (41) Harger, M.; Li, D.; Wang, Z.; Dalby, K.; Lagardère, L.; Piquemal, J.-P.; Ponder, J.; Ren, P. Tinker-OpenMM: Absolute and relative alchemical free energies using AMOEBA on GPUs. *J. Comput. Chem.* **2017**, *38*, 2047–2055.
- (42) Grossfield, A.; Ren, P.; Ponder, J. W. Ion Solvation Thermodynamics from Simulation with a Polarizable Force Field. *J. Am. Chem. Soc.* **2003**, *125*, 15671–15682.
- (43) Ren, P.; Ponder, J. W. Temperature and Pressure Dependence of the AMOEBA Water Model. *J. Phys. Chem. B* **2004**, *108*, 13427–13437.

- (44) Zhang, C.; Bell, D.; Harger, M.; Ren, P. Polarizable Multipole-Based Force Field for Aromatic Molecules and Nucleobases. *J. Chem. Theory Comput.* **2017**, *13*, 666–678.
- (45) Mezei, M.; Beveridge, D. L. Computer simulation of chemical and biomolecular systems. *Ann. N. Y. Acad. Sci.* **1986**, *482*, 1–27.
- (46) Pearlman, D. A. A Comparison of Alternative Approaches to Free Energy Calculations. *J. Phys. Chem.* **1994**, *98*, 1487–1493.
- (47) Pearlman, D. A.; Kollman, P. A. A new method for carrying out free energy perturbation calculations: Dynamically modified windows. *J. Chem. Phys.* **1989**, *90*, 2460–2470.
- (48) Pearlman, D. A.; Case, D. A.; Caldwell, J. W.; Ross, W. R.; Cheatham, T., III; DeBolt, S.; Ferguson, D.; Seibel, G.; Kollman, P. AMBER, a package of computer programs for applying molecular mechanics, normal mode analysis, molecular dynamics and free energy calculations to simulate the structure and energetic properties of molecules. *Comput. Phys. Commun.* **1995**, *91*, 1–41.
- (49) Case, D. A.; Cheatham, T. E., III; Darden, T.; Gohlke, H.; Luo, R.; Merz, K. M.; Onufriev, A.; Simmerling, C.; Wang, B.; Woods, R. J. The Amber biomolecular simulation programs. *J. Comput. Chem.* **2005**, *26*, 1668–1688.
- (50) Essmann, U.; Perera, L.; Berkowitz, M. L.; Darden, T.; Lee, H.; Pedersen, L. G. A smooth particle mesh Ewald method. *J. Chem. Phys.* **1995**, *103*, 8577–8593.
- (51) AMBER 2016 Reference Manual. <http://ambermd.org/doc12/Ambler16.pdf> (accessed January 2018).
- (52) Kunz, A.-P. E.; Allison, J. R.; Geerke, D. P.; Horta, B. A. C.; Hunenberger, P. H.; Riniker, S.; Schmid, N.; van Gunsteren, W. F. New functionalities in the GROMOS biomolecular simulation software. *J. Comput. Chem.* **2012**, *33*, 340–353.
- (53) Eichenberger, A. P.; Allison, J. R.; Dolenc, J.; Geerke, D. P.; Horta, B. A. C.; Meier, K.; Oostenbrink, C.; Schmid, N.; Steiner, D.; Wang, D.; van Gunsteren, W. F. GROMOS++ Software for the Analysis of Biomolecular Simulation Trajectories. *J. Chem. Theory Comput.* **2011**, *7*, 3379–3390.
- (54) Van Der Spoel, D.; Lindahl, E.; Hess, B.; Groenhof, G.; Mark, A. E.; Berendsen, H. J. C. GROMACS: Fast, Flexible, and Free. *J. Comput. Chem.* **2005**, *26*, 1701–1718.
- (55) Christen, M.; Hünenberger, P. H.; Bakowies, D.; Baron, R.; Bürgi, R.; Geerke, D. P.; Heinz, T. N.; Kastenholz, M. A.; Kräutler, V.; Oostenbrink, C.; Peter, C.; Trzesniak, D.; van Gunsteren, W. F. The GROMOS Software for Biomolecular Simulation: GROMOS05. *J. Comput. Chem.* **2005**, *26*, 1719–1751.
- (56) Plimpton, S. Fast Parallel Algorithms for Short-Range Molecular Dynamics. *J. Comput. Phys.* **1995**, *117*, 1–19.
- (57) Kaus, J. W.; Pierce, L. T.; Walker, R. C.; McCammon, J. A. Improving the Efficiency of Free Energy Calculations in the Amber Molecular Dynamics Package. *J. Chem. Theory Comput.* **2013**, *9*, 4131–4139.
- (58) Abrams, C.; Bussi, G. Enhanced Sampling in Molecular Dynamics Using Metadynamics, Replica-Exchange, and Temperature-Acceleration. *Entropy* **2014**, *16*, 163.
- (59) Li, G.; Zhang, X.; Cui, Q. Free energy perturbation calculations with combined QM/MM potentials complications, simplifications, and applications to redox potential calculations. *J. Phys. Chem. B* **2003**, *107*, 8643–8653.
- (60) Shyu, C.; Ytreberg, F. M. Reducing the bias and uncertainty of free energy estimates by using regression to fit thermodynamic integration data. *J. Comput. Chem.* **2009**, *30*, 2297–2304.
- (61) Shyu, C.; Ytreberg, F. M. Accurate estimation of solvation free energy using polynomial fitting techniques. *J. Comput. Chem.* **2011**, *32*, 134–141.
- (62) de Ruiter, A.; Boresch, S.; Oostenbrink, C. Comparison of thermodynamic integration and Bennett acceptance ratio for calculating relative protein-ligand binding free energies. *J. Comput. Chem.* **2013**, *34*, 1024–1034.
- (63) Hummer, G. Fast-growth thermodynamic integration: Error and efficiency analysis. *J. Chem. Phys.* **2001**, *114*, 7330–7337.
- (64) Hummer, G. Fast-growth Thermodynamic Integration: Results for Sodium Ion Hydration. *Mol. Simul.* **2002**, *28*, 81–90.
- (65) Hess, B.; Peter, C.; Ozal, T.; van der Vegt, N. F. A. Fast-Growth Thermodynamic Integration: Calculating Excess Chemical Potentials of Additive Molecules in Polymer Microstructures. *Macromolecules* **2008**, *41*, 2283–2289.
- (66) Kong, X.; Brooks, C. L., III λ -dynamics: A new approach to free energy calculations. *J. Chem. Phys.* **1996**, *105*, 2414–2423.
- (67) Guo, Z.; Brooks, C. L. Rapid Screening of Binding Affinities: Application of the λ -Dynamics Method to a Trypsin-Inhibitor System. *J. Am. Chem. Soc.* **1998**, *120*, 1920–1921.
- (68) Guo, Z.; Brooks, C. L., III; Kong, X. Efficient and Flexible Algorithm for Free Energy Calculations Using the λ -Dynamics Approach. *J. Phys. Chem. B* **1998**, *102*, 2032–2036.
- (69) Banba, S.; Guo, Z.; Brooks, C. L., III Efficient sampling of ligand orientations and conformations in free energy calculations using the λ -dynamics method. *J. Phys. Chem. B* **2000**, *104*, 6903–6910.
- (70) Damodaran, K. V.; Banba, S.; Brooks, C. L., III Application of multiple topology λ -dynamics to a host-guest system: β -cyclodextrin with substituted benzenes. *J. Phys. Chem. B* **2001**, *105*, 9316–9322.
- (71) Knight, J. L.; Brooks, C. L., III Lambda-dynamics free energy simulation methods. *J. Comput. Chem.* **2009**, *30*, 1692–1700.
- (72) Steinbrecher, T.; Joung, I.; Case, D. A. Soft-Core Potentials in Thermodynamic Integration: Comparing One- and Two-Step Transformations. *J. Comput. Chem.* **2011**, *32*, 3253–3263.
- (73) Simonson, T. Free energy of particle insertion. *Mol. Phys.* **1993**, *80*, 441–447.
- (74) Maye, P. V.; Mezei, M. Calculation of the free energy of solvation of the Li^+ and Na^+ ions in water chloroform. *J. Mol. Struct.: THEOCHEM* **1996**, *362*, 317–324.
- (75) Mezei, M. Polynomial path for the calculation of liquid state free energies from computer simulations tested on liquid water. *J. Comput. Chem.* **1992**, *13*, 651–656.
- (76) Steinbrecher, T.; Mobley, D. L.; Case, D. A. Nonlinear scaling schemes for Lennard-Jones interactions in free energy calculations. *J. Chem. Phys.* **2007**, *127*, 214108.
- (77) Pitera, J. W.; van Gunsteren, W. F. A comparison of non-bonded scaling approaches for free energy calculations. *Mol. Simul.* **2002**, *28*, 45–65.
- (78) Beutler, T. C.; Mark, A. E.; van Schaik, R. C.; Gerber, P. R.; van Gunsteren, W. F. Avoiding singularities and numerical instabilities in free energy calculations based on molecular simulations. *Chem. Phys. Lett.* **1994**, *222*, 529–539.
- (79) Zacharias, M.; Straatsma, T. P.; McCammon, J. A. Separation-shifted scaling, a new scaling method for Lennard-Jones interactions in thermodynamic integration. *J. Chem. Phys.* **1994**, *100*, 9025–9031.
- (80) Tappura, K.; Lahtela-Kakkonen, M.; Teleman, O. A New Soft-Core Potential Function for Molecular Dynamics Applied to the Prediction of Protein Loop Conformations. *J. Comput. Chem.* **2000**, *21*, 388–397.
- (81) Hornak, V.; Simmerling, C. Development of softcore potential functions for overcoming steric barriers in molecular dynamics simulations. *J. Mol. Graphics Modell.* **2004**, *22*, 405–413.
- (82) Abraham, M. J.; Murtola, T.; Schulz, R.; Páll, S.; Smith, J. C.; Hess, B.; Lindahl, E. GROMACS: High performance molecular simulations through multi-level parallelism from laptops to supercomputers. *SoftwareX* **2015**, *1–2*, 19–25.
- (83) Gapsys, V.; Seeliger, D.; de Groot, B. L. New Soft-Core Potential Function for Molecular Dynamics Based Alchemical Free Energy Calculations. *J. Chem. Theory Comput.* **2012**, *8*, 2373–2382.
- (84) Naden, L. N.; Pham, T. T.; Shirts, M. R. Linear Basis Function Approach to Efficient Alchemical Free Energy Calculations. 1. Removal of Uncharged Atomic Sites. *J. Chem. Theory Comput.* **2014**, *10*, 1128–1149.
- (85) Naden, L. N.; Shirts, M. R. Linear basis function approach to efficient alchemical free energy calculations. 2. Inserting and deleting particles with Coulombic interactions. *J. Chem. Theory Comput.* **2015**, *11*, 2536–2549.
- (86) Sellers, M. S.; Lisal, M.; Brennan, J. K. Exponential-six potential scaling for the calculation of free energies in molecular simulations. *Mol. Phys.* **2015**, *113*, 45–54.

- (87) Scott, W. R. P.; Hünenberger, P. H.; Tironi, I. G.; Mark, A. E.; Billeter, S. R.; Fennen, J.; Torda, A. E.; Huber, T.; Krüger, P.; van Gunsteren, W. F. The GROMOS biomolecular simulation program package. *J. Phys. Chem. A* **1999**, *103*, 3596–3607.
- (88) Darden, T.; York, D.; Pedersen, L. Particle mesh Ewald: An $N \log(N)$ method for Ewald sums in large systems. *J. Chem. Phys.* **1993**, *98*, 10089–10092.
- (89) Giese, T. J.; Panteva, M. T.; Chen, H.; York, D. M. Multipolar Ewald methods, I: Theory, accuracy, and performance. *J. Chem. Theory Comput.* **2015**, *11*, 436–450.
- (90) Giese, T. J.; York, D. M. Ambient-Potential Composite Ewald Method for ab Initio Quantum Mechanical/Molecular Mechanical Molecular Dynamics Simulation. *J. Chem. Theory Comput.* **2016**, *12*, 2611–2632.
- (91) Tang, K. T.; Toennies, J. P. A simple theoretical model for the van der Waals potential at intermediate distances. I. Spherically symmetric potentials. *J. Chem. Phys.* **1977**, *66*, 1496–1506.
- (92) Tang, K. T.; Toennies, J. P. An improved simple model for the van der Waals potential based on universal damping functions for the dispersion coefficients. *J. Chem. Phys.* **1984**, *80*, 3726–3736.
- (93) Tang, K. T.; Toennies, J. P. The van der Waals potentials between all the rare gas atoms from He to Rn. *J. Chem. Phys.* **2003**, *118*, 4976–4983.
- (94) Li, P.; Merz, K. M., Jr. Taking into account the ion-induced dipole interaction in the nonbonded model of ions. *J. Chem. Theory Comput.* **2014**, *10*, 289–297.
- (95) Panteva, M. T.; Giambasu, G. M.; York, D. M. Force field for Mg^{2+} , Mn^{2+} , Zn^{2+} , and Cd^{2+} ions that have balanced interactions with nucleic acids. *J. Phys. Chem. B* **2015**, *119*, 15460–15470.
- (96) Panteva, M. T.; Giambasu, G. M.; York, D. M. Comparison of structural, thermodynamic, kinetic and mass transport properties of Mg^{2+} ion models commonly used in biomolecular simulations. *J. Comput. Chem.* **2015**, *36*, 970–982.
- (97) Ren, P.; Ponder, J. W. Polarizable atomic multipole water model for molecular mechanics simulation. *J. Phys. Chem. B* **2003**, *107*, 5933–5947.
- (98) Wheatley, R. J.; Price, S. L. An overlap model for estimating the anisotropy of repulsion. *Mol. Phys.* **1990**, *69*, 507–533.
- (99) Elking, D. M.; Cisneros, G. A.; Piquemal, J.-P.; Darden, T. A.; Pedersen, L. G. Gaussian Multipole Model (GMM). *J. Chem. Theory Comput.* **2010**, *6*, 190–202.
- (100) Giese, T. J.; York, D. M. Charge-dependent model for many-body polarization, exchange, and dispersion interactions in hybrid quantum mechanical/molecular mechanical calculations. *J. Chem. Phys.* **2007**, *127*, 194101.
- (101) Van Vleet, M. J.; Misquitta, A. J.; Stone, A. J.; Schmidt, J. R. Beyond Born-Mayer: Improved Models for Short-Range Repulsion in ab Initio Force Fields. *J. Chem. Theory Comput.* **2016**, *12*, 3851–3870.
- (102) Klauda, J. B.; Wu, X.; Pastor, R. W.; Brooks, B. R. Long-range Lennard-Jones and electrostatic interactions in interfaces: application of the isotropic periodic sum method. *J. Phys. Chem. B* **2007**, *111*, 4393–4400.
- (103) Wu, X.; Brooks, B. R. Isotropic periodic sum: a method for the calculation of long-range interactions. *J. Chem. Phys.* **2005**, *122*, 044107.
- (104) Maier, J. A.; Martinez, C.; Kasavajhala, K.; Wickstrom, L.; Hauser, K. E.; Simmerling, C. ff14SB: Improving the Accuracy of Protein Side Chain and Backbone Parameters from ff99SB. *J. Chem. Theory Comput.* **2015**, *11*, 3696–3713.
- (105) Horn, H. W.; Swope, W. C.; Pitera, J. W.; Madura, J. D.; Dick, T. J.; Hura, G. L.; Head-Gordon, T. Development of an improved four-site water model for biomolecular simulations: TIP4P-Ew. *J. Chem. Phys.* **2004**, *120*, 9665–9678.
- (106) Li, P.; Roberts, B. P.; Chakravorty, D. K.; Merz, K. M., Jr. Rational design of Particle Mesh Ewald compatible Lennard-Jones parameters for + 2 metal cations in explicit solvent. *J. Chem. Theory Comput.* **2013**, *9*, 2733–2748.
- (107) Case, D. A.; Betz, R. M.; Cerutti, D. S.; Cheatham, T. E., III; Darden, T. A.; Duke, R. E.; Giese, T. J.; Gohlke, H.; Goetz, A. W.; Homeyer, N.; Izadi, S.; Janowski, P.; Kaus, J.; Kovalenko, A.; Lee, T. S.; LeGrand, S.; Li, P.; Lin, C.; Luchko, T.; Luo, R.; Madej, B.; Mermelstein, D.; Merz, K. M.; Monard, G.; Nguyen, H.; Nguyen, H. T.; Omelyan, I.; Onufriev, A.; Roe, D. R.; Roitberg, A.; Sagui, C.; Simmerling, C. L.; Botello-Smith, W. M.; Swails, J.; Walker, R. C.; Wang, J.; Wolf, R. M.; Wu, X.; Xiao, L.; Kollman, P. A. *AMBER 2016*; University of California, San Francisco, 2016.
- (108) Towns, J.; Cockerill, T.; Dahan, M.; Foster, I.; Gaither, K.; Grimshaw, A.; Hazlewood, V.; Lathrop, S.; Lifka, D.; Peterson, G. D.; Roskies, R.; Scott, J. R.; Wilkens-Diehr, N. XSEDE: Accelerating Scientific Discovery. *Comput. Sci. Eng.* **2014**, *16*, 62–74.
- (109) Ryckaert, J. P.; Ciccotti, G.; Berendsen, H. J. C. Numerical Integration of the Cartesian Equations of Motion of a System with Constraints: Molecular Dynamics of n-Alkanes. *J. Comput. Phys.* **1977**, *23*, 327–341.
- (110) Bogusz, S.; Cheatham, T. E., III; Brooks, B. R. Removal of pressure and free energy artifacts in charged periodic systems via net charge corrections to the Ewald potential. *J. Chem. Phys.* **1998**, *108*, 7070–7084.
- (111) Turq, P.; Lantelme, F.; Friedman, H. L. Brownian dynamics: Its application to ionic solutions. *J. Chem. Phys.* **1977**, *66*, 3039–3044.
- (112) Klimovich, P. V.; Shirts, M. R.; Mobley, D. L. Guidelines for the analysis of free energy calculations. *J. Comput.-Aided Mol. Des.* **2015**, *29*, 397–411.
- (113) Chodera, J. D.; Swope, W. C.; Pitera, J. W.; Seok, C.; Dill, K. A. Use of the weighted histogram analysis method for the analysis of simulated and parallel tempering simulations. *J. Chem. Theory Comput.* **2007**, *3*, 26–41.
- (114) Shirts, M. R.; Bair, E.; Hooker, G.; Pande, V. S. Equilibrium free energies from nonequilibrium measurements using maximum-likelihood methods. *Phys. Rev. Lett.* **2003**, *91*, 140601.
- (115) Paliwal, H.; Shirts, M. R. A benchmark test set for alchemical free energy transformations and its use to quantify error in common free energy methods. *J. Chem. Theory Comput.* **2011**, *7*, 4115–4134.
- (116) Wilcox, J. L.; Bevilacqua, P. C. pK_a shifting in double-stranded RNA is highly dependent upon nearest neighbors and bulge positioning. *Biochemistry* **2013**, *52*, 7470–7476.
- (117) Izatt, R. M.; Christensen, J. J.; Rytting, J. H. Sites and thermodynamic quantities associated with proton and metal ion interaction with ribonucleic acid, deoxyribonucleic acid, and their constituent bases, nucleosides, and nucleotides. *Chem. Rev.* **1971**, *71*, 439–481.
- (118) Kastenzholz, M. A.; Hünenberger, P. H. Computation of methodology-independent ionic solvation free energies from molecular simulations II. The hydration free energy of the sodium cation. *J. Chem. Phys.* **2006**, *124*, 224501.
- (119) Kastenzholz, M. A.; Hünenberger, P. H. Computation of methodology-independent ionic solvation free energies from molecular simulations. I. The electrostatic potential in molecular liquids. *J. Chem. Phys.* **2006**, *124*, 124106.
- (120) Rocklin, G. J.; Mobley, D. L.; Dill, K. A.; Hünenberger, P. H. Calculating the binding free energies of charged species based on explicit-solvent simulations employing lattice-sum methods: An accurate correction scheme for electrostatic finite-size effects. *J. Chem. Phys.* **2013**, *139*, 184103.
- (121) Caves, L. S.; Evanseck, J. D.; Karplus, M. Locally accessible conformations of proteins: multiple molecular dynamics simulations of crambin. *Protein Sci.* **1998**, *7*, 649–666.
- (122) Loccisano, A. E.; Acevedo, O.; DeChancie, J.; Schulze, B. G.; Evanseck, J. D. Enhanced sampling by multiple molecular dynamics trajectories: carbonmonoxy myoglobin 10 micros A0->A(1-3) transition from ten 400 ps simulations. *J. Mol. Graphics Modell.* **2004**, *22*, 369–376.
- (123) Likic, V. A.; Gooley, P. R.; Speed, T. P.; Strehler, E. E. A statistical approach to the interpretation of molecular dynamics simulations of calmodulin equilibrium dynamics. *Protein Sci.* **2005**, *14*, 2955–2963.
- (124) Elofsson, A.; Nilsson, L. How consistent are molecular dynamics simulations? Comparing structure and dynamics in reduced and oxidized Escherichia coli thioredoxin. *J. Mol. Biol.* **1993**, *233*, 766–780.

(125) Heldenbrand, H.; Janowski, P. A.; Giambaşu, G.; Giese, T. J.; Wedekind, J. E.; York, D. M. Evidence for the role of active site residues in the hairpin ribozyme from molecular simulations along the reaction path. *J. Am. Chem. Soc.* **2014**, *136*, 7789–7792.

(126) Bayly, C. I.; Cieplak, P.; Cornell, W. D.; Kollman, P. A. A Well-Behaved Electrostatic Potential Based Method Using Charge Restraints for Deriving Atomic Charges: The RESP Model. *J. Phys. Chem.* **1993**, *97*, 10269–10280.

(127) Simonson, T.; Hummer, G.; Roux, B. Equivalence of M- and P-Summation in Calculations of Ionic Solvation Free Energies. *J. Phys. Chem. A* **2017**, *121*, 1525–1530.

(128) Zuckerman, D. M.; Woolf, T. B. Overcoming finite-sampling errors in fast-switching free-energy estimates: extrapolative analysis of a molecular system. *Chem. Phys. Lett.* **2002**, *351*, 445–453.

(129) Coveney, P. V.; Wan, S. On the calculation of equilibrium thermodynamic properties from molecular dynamics. *Phys. Chem. Chem. Phys.* **2016**, *18*, 30236–30240.

(130) Giese, T. J.; York, D. M. Quantum mechanical force fields for condensed phase molecular simulations. *J. Phys.: Condens. Matter* **2017**, *29*, 383002.

(131) Han, J.; Truhlar, D. G.; Gao, J. Optimization of the explicit polarization (X-Pol) potential using a hybrid density functional. *Theor. Chem. Acc.* **2012**, *131*, 1161–1161.

(132) Wang, Y.; Sosa, C. P.; Cembran, A.; Truhlar, D. G.; Gao, J. Multilevel X-Pol: A Fragment-Based Method with Mixed Quantum Mechanical Representations of Different Fragments. *J. Phys. Chem. B* **2012**, *116*, 6781–6788.

(133) Gao, J.; Wang, Y. Communication: Variational many-body expansion: Accounting for exchange repulsion, charge delocalization, and dispersion in the fragment-based explicit polarization method. *J. Chem. Phys.* **2012**, *136*, 071101.