Implementation of adaptive integration method for free energy calculations in molecular systems

Christopher A. Mirabzadeh[1] and F. Marty Ytreberg[1,2,3]

[1] Department of Physics, University of Idaho, Moscow, ID, United States of America
[2] Institute for Modeling Collaboration and Innovation, University of Idaho, Moscow, ID, United States of America
[3] Institute for Bioinformatics and Evolutionary Studies, University of Idaho, Moscow, ID, United States of America

ABSTRACT

Estimating free energy differences by computer simulation is useful for a wide variety of applications such as virtual screening for drug design and for understanding how amino acid mutations modify protein interactions. However, calculating free energy differences remains challenging and often requires extensive trial and error and very long simulation times in order to achieve converged results. Here, we present an implementation of the adaptive integration method (AIM). We tested our implementation on two molecular systems and compared results from AIM to those from a suite of other methods. The model systems tested here include calculating the solvation free energy of methane, and the free energy of mutating the peptide GAG to GVG. We show that AIM is more efficient than other tested methods for these systems, that is, AIM results converge to a higher level of accuracy and precision for a given simulation time.

INTRODUCTION

Measuring free energy differences using computer simulations can be computationally expensive, yet is useful for many different applications (see e.g., *Steinbrecher & Labahn, 2010*; *Chodera et al., 2011*; *Mobley et al., 2012*; *Zhan et al., 2013*; *Miller et al., 2014*; *Petukh, Li & Alexov, 2015*; *Zhan & Ytreberg, 2015*; *Miller et al., 2016*; *Cournia, Allen & Sherman, 2017*; *Hossain et al., 2019*; *Aminpour, Montemagno & Tuszynski, 2019*). Specific examples include determining protein conformational preferences, virtual screening for drug design or drug discovery (*Steinbrecher & Labahn, 2010*; *Chodera et al., 2011*; *Zhan & Ytreberg, 2015*; *Śledź & Caflisch, 2018*; *Aminpour, Montemagno & Tuszynski, 2019*; *Zhang et al., 2019*). Of specific relevance to the current study is that free energy calculations allow prediction of how amino acid mutations may modify protein-protein binding (*Zhan et al., 2013*; *Miller et al., 2014*; *Petukh, Li & Alexov, 2015*; *Miller et al., 2016*; *Geng et al., 2019*). We are particularly interested in developing and implementing efficient methods for calculating free energy differences and using them to understand how amino acid mutations modify protein–protein and protein-substrate interactions.

Corresponding author
F. Marty Ytreberg,
ytreberg@uidaho.edu

For this study, we have implemented the adaptive integration method (AIM) introduced by *Fasnacht, Swendsen & Rosenberg (2004)* for use in the GROMACS (*Berendsen, Van der Spoel & Van Drunen, 1995*) molecular dynamics simulation package, and have compared results to a suite of other methods. In previous studies, AIM was shown to provide high quality, precise and efficient estimates of binding free energies (*Ytreberg, Swendsen & Zuckerman, 2006*; *Kaus, Arrar & McCammon, 2014*; *Kaus & Mccammon, 2015*). We focus on alchemical free energy calculation where a system is transformed from one state to another via an unphysical pathway. The progress along the pathway that connects the two states is defined by the parameter λ. For this study, we are interested in two ways to explore λ space—both of which have the goal of obtaining equilibrium sampling of system configurations at discrete λ values along the pathway. The first way is to treat λ as a system variable that can be biased and sampled. A class of such methods, termed generalized ensemble, use an extended Hamiltonian to sample λ (*Bitetti-Putzer, Yang & Karplus, 2003*). For example, λ-dynamics (*Kong & Brooks, 1996*; *Knight & Brooks, 2009*; *Knight & Brooks, 2011*) treats λ as a dynamic particle in the system with fictitious mass. By contrast, AIM uses Metropolis Monte Carlo to sample λ space (*Fasnacht, Swendsen & Rosenberg, 2004*). Monte Carlo moves between values of λ are based on running estimates of free energy differences; this is a key distinction from other methods and allows AIM to continuously improve the estimate for the free energy during the simulation. The second way to explore λ space is to perform standard molecular dynamics or Monte Carlo simulations at fixed values of λ, typically discarding some simulation time for equilibration. The configurational ensembles at each value of λ can then be used to estimate free energy differences (see e.g., *Lyubartsev, Førrisdahl & Laaksonen, 1996*; *Gonçalves & Stassen, 2004*; *Kofke, 2005*; *Shirts, Mobley & Chodera, 2007*; *Chodera & Shirts, 2011*; *Klimovich, Shirts & Mobley, 2015*). In order to provide comparisons for AIM to fixed λ methods, we used the Python tool alchemical-analysis.py (*Klimovich, Shirts & Mobley, 2015*), part of the Pymbar package (*Shirts & Chodera, 2008*). This tool estimates the free energy using a suite of methods such as the Bennett acceptance ratio, multistate Bennett acceptance ratio, thermodynamic integration and exponential averaging.

For the current study, we chose two molecular systems that have well-documented results and are important starting points for biomolecular free energy studies. First, we calculated the solvation free energy of methane. Simulations were performed and the free energies were calculated using the fixed λ methods provided by alchemical-analysis. Simulations were also performed using AIM and results compared to fixed λ simulations. Using the lessons learned from the methane system, we then calculated the free energy of mutating the peptide GAG to GVG in water. For both systems, we found that AIM produces free energy estimates that are within statistical uncertainty of fixed λ methods but with greater efficiency (i.e., more accurate for a given simulation time).

METHODS

All methods, code and simulation input files are available in the supplemental materials. For this study, we performed alchemical free energy simulations where the system is changed

from a reference state to an end state by constructing a reaction pathway that modifies, adds or removes atoms. Such alchemical simulations are non-physical, i.e., the simulation does not represent what could occur naturally. Since the free energy is a state variable, it is independent of the path taken, and we may provide any path we wish. To perform these simulations the reaction pathway is divided into many separate, non-physical, λ states between a reference state and an end state. The λ states represent the progress along the reaction pathway as the reference state transforms into the end state.

Like most methods used to calculate free energies we start from the identity,

$$F = U - TS, \tag{1}$$

where U is the potential energy, T is the temperature and S is the entropy of the system. For free energy differences we generalize the formulation of the change in free energy by separating calculations into two, non-overlapping, thermodynamic end states, A and B, at constant system temperature T,

$$\Delta F \equiv \Delta F_{A \to B} = F_B - F_A = \Delta U - T \Delta S. \tag{2}$$

ΔF is the change in free energy, ΔU is the change in potential energy and ΔS is the change in entropy of the system. According to statistical mechanics, the free energy difference between the two end states, A and B, of the system is the log of the ratio of the configurational partition functions (see discussion in *Chipot & Pohorille (2007)*),

$$\Delta F = -k_B T \ln \frac{Z[U_B(\vec{x})]}{Z[U_A(\vec{x})]}. \tag{3}$$

Here, k_B is the Boltzmann constant and $Z[U(\vec{x})]$ is the configurational partition function for the energy states $U_A(\vec{x})$ and $U_B(\vec{x})$, where $\vec{x}$ is the vector of configuration coordinates. The configurational partition function is given by

$$Z[U(\vec{x})] = \int \exp(-\beta U(\vec{x})) d\vec{x}, \tag{4}$$

and $\beta = \frac{1}{k_B T}$.

Computationally, we calculate free energy differences between end states by performing molecular dynamics simulations along a reaction pathway of intermediate states, defined by λ, such that, $0 \leq \lambda \leq 1$.

This pathway connects the two end states of the system. In the case of poor overlap, where the end states may be separated by a high energy barrier, $|U_B - U_A| \gg k_B T$, this pathway mitigates the otherwise very slow convergence of free energy estimates (*Shirts, Mobley & Chodera, 2007*). Care should be taken when choosing intermediate states such that there is adequate overlap in the conformation space between the end states (*Shirts, Mobley & Chodera, 2007*; *Klimovich, Shirts & Mobley, 2015*). For our simulations the number of λ values and time per λ were chosen through extensive trial and error (more on this below).

The method of exponential averaging (DEXP, IEXP) (*Zwanzig, 1954*) starts from Eq. (3) above and then adding and subtracting $\exp(-\beta U(\vec{x}))$ from the integral in the configurational partition function of the numerator we end up with the final relationship,

$$\Delta F_{ij} = -k_B T \ln \langle \exp(-\beta \Delta U_{ij}(\vec{x})) \rangle_{\lambda_i}. \tag{5}$$

where ΔF_{ij} is the free energy between λ_i and λ_j and $\langle \cdot \rangle_{\lambda_i}$ represents an average of the equilibrium configuration for λ_i. Unlike some other methods, exponential averaging has an exact solution since it is only used to evaluate the difference between two states. However, it is the least efficient method and should not be used if difference in potential energies are much larger than $k_B T$ (*Shirts & Pande, 2005*). In addition, exponential averaging can be noisy, biased and dependent on the tails of the distribution of λ states (*Bruckner & Boresch, 2011*; *Shirts & Pande, 2005*).

For thermodynamic integration (TI) we estimate the free energy by first looking at the derivative of Eq. (1) with respect to λ,

$$\frac{\partial F}{\partial \lambda} = \left\langle \frac{\partial U}{\partial \lambda} \right\rangle_{\lambda}. \tag{6}$$

This differential equation, Eq. (6), can then be integrated to give,

$$\Delta F = \int_{\lambda=0}^{1} \left\langle \frac{\partial U_{\lambda}(\vec{x})}{\partial \lambda} \right\rangle_{\lambda} d\lambda \tag{7}$$

where the $\langle \cdot \rangle_{\lambda}$ notation represents the ensemble average at a given intermediate state, λ. The free energy is estimated by numerically integrating Eq. (7) after running equilibrium simulations at each intermediate λ state. Since numerical integration is required, TI can be biased by the chosen method of integration. Some of that bias can be removed by using cubic-spline interpolation or more complex integration estimators(*Shirts & Pande, 2005*; *Shyu & Ytreberg, 2009*).

The Bennett (*Bennett, 1976*) and multistate (*Shirts & Chodera, 2008*) Bennett acceptance ratio (BAR and MBAR) methods are far more efficient than exponential averaging and are commonly used to avoid the shortcomings of other methods (*Shirts & Pande, 2005*; *Ytreberg, Swendsen & Zuckerman, 2006*). BAR and MBAR typically achieve the same statistical precision as TI with fewer λ states unless the integrand for TI is very smooth (*Shirts & Mobley, 2013*; *Ytreberg, Swendsen & Zuckerman, 2006*). The complete derivation can be found in Bennett's paper (*Bennett, 1976*) but the premise is; for sufficiently large samples n_i of U_i and n_j of U_j,

$$\Delta F(i \rightarrow j) = k_B T \ln \frac{\langle f(\Delta U_{ij} + C) \rangle_j}{\langle f(\Delta U_{ji} - C) \rangle_i} + C. \tag{8}$$

C is a shift constant,

$$C = k_B T \ln \frac{n_j}{n_i}, \tag{9}$$

and $f(x)$ is the Fermi function,

$$f(x) = \frac{1}{1 + \exp(\beta x)}. \tag{10}$$

Equation (8) is the ratio of canonical averages of two different potentials U_i and U_j acting on the same configuration space meaning it requires information from two neighboring states. However, this limitation is not too much of a concern with a trivial coordinate

transformation or when using dummy coordinates in alchemical simulations. MBAR, an extension of BAR, differs in that it takes data from more than two states hence the name "multistate".

AIM is similar to TI in that numerical integration of Eq. (7) is performed; the key difference is how the averages $\langle \partial U / \partial \lambda \rangle_\lambda$ are obtained. AIM uses Metropolis Monte Carlo to move in λ space and ordinary running averages are calculated at each λ value. In AIM, a random move from λ_{old} to λ_{new} is accepted with probability

$$\min\{1, \exp(-\beta(U_{new}) - U_{old}) + \beta(F_{new} - F_{old})\} \tag{11}$$

where $U_{new} - U_{old}$ is the difference in the potential energy for the old and new λ values. $F_{new} - F_{old}$ is the estimated free energy difference based on the current running averages of $\partial U / \partial \lambda$.

Implementation

AIM was implemented in GROMACS as an expanded ensemble calculation. That is, the Hamiltonian must be calculated along with its derivative, and an expanded ensemble step must be performed for every dynamics step. In GROMACS, nstexpanded is the number of integration steps between attempted λ moves changing the system Hamiltonian in expanded ensemble simulations. This value must be a multiple of nstcalcenergy, the number of steps before calculating the system energy, but can be greater or less than nstdhdl, the number of steps before calculating $\partial U / \partial \lambda$ (referred to as dHdλ in GROMACS documentation). For a detailed explanation of all technical terms see reference *Abraham et al. (2016)*. The GROMACS package was further altered to print out the $\partial U / \partial \lambda$ averages computed by AIM to the log file when AIM is used as the lmc-mover.

AIM requires the $\partial U / \partial \lambda$ value from every dynamics step to be stored regardless of whether a move in λ space is attempted. Since $\partial U / \partial \lambda$ is only calculated at each step where free energies are calculated, every nstdhdl step, we set nstexpanded = nstdhdl = nstcalcenergy = 1 for AIM simulations. This further implies that lmc-stats functions were not used during AIM simulations because those functions modify the Hamiltonian which is not needed for AIM.

For the implementation of AIM with GROMACS we follow the outline given in our previous study *Ytreberg, Swendsen & Zuckerman (2006)*.

1. Start the simulation from an equilibrated configuration at $\lambda=0$ and perform one molecular dynamics step.
2. Randomly choose a trial move in λ space. For example, if our λ spacing is 0.05, a move from $\lambda = 0.35$ to 0.4 or 0.3 may be attempted but not to 0.45.
3. Calculate the difference in potential energy between the trial and current λ values.
4. Estimate the free energy difference between the trial and current λ values using the running averages of $\partial U / \partial \lambda$ and the trapezoidal rule.
5. Accept λ trial with probability given in Eq. (11).
6. If the move is accepted then λ is updated to the trial value, otherwise the simulations stays at the current λ.
7. The running average of $\partial U / \partial \lambda$ is updated.

Simulation details

The first system used here, methane in water, is detailed in systematic studies of force fields and the free energies of hydration of amino acid side chain analogs (*Sun et al., 1992*; *Lyubartsev, Førrisdahl & Laaksonen, 1996*; *Chodera & Shirts, 2011*; *Paliwal & Shirts, 2011*). For the GAG to GVG mutation the PMX (*Gapsys et al., 2015*) software package was used to construct the tri-peptide mutation. Using PMX, we generated the hybrid protein structure and topology for simulations of the chosen mutation, alanine to valine.

All simulations described in this paper were performed using the molecular dynamics package GROMACS 5.1.4. The simulations were carried out at 300 K and solvated in a dodecahedron box with TIP3P waters. The methane molecule was parameterized using the OPLS (Optimized Potential for Liquid Simulations) force field (*Jorgensen, Maxwell & Tirado-Rives, 1996*). The OPLS force field was chosen for this study because it is known to perform well on small molecules (*Shirts et al., 2003*). In future studies, we anticipate using AIM on protein systems where other force fields are more appropriate such as AMBER (*Salomon-Ferrer, Case & Walker, 2013*) and CHARMM (*Mackerell, Banavali & Foloppe, 2001*). Since all molecular dynamics force fields have similar form and number of parameters, it is expected that the performance of AIM would not depend on the force field chosen. For the GAG to GVG mutations, Na+ and Cl- ions were added to keep the simulation box neutral and reach a physiologically relevant 150 mM salt concentration.

For both systems, energy minimization was performed using steepest descent for 1,000 steps. The system was then equilibrated using simulated annealing for 1,000 ps to heat the system from 100 K to 300 K. For production simulations, electrostatic interactions were handled by Reaction field with a cut-off of 0.9 nm, Potential-shift-Verlet modifier and Verlet cutoff scheme. Van der Waals interactions were handled by twin range cutoffs with neighbor list cutoff of 1.15 nm and van der Waals cutoff of 0.9 nm. The bonds involving hydrogens were constrained with the Shake algorithm, allowing for a 2 fs time step. Long range dispersion corrections for energy and pressure were applied. For the free energy calculations, softcore scaling was used with parameters `sc-power=1`, `sc-r-power=6` and `sc-alpha=0.5`. In addition, the van der Waals and Coulomb interactions were separately turned on or off as a function of λ. That is, one is held fixed as a function of λ while the other changes. For the methods processed with alchemical-analysis.py we ran fixed λ simulations. That is, an equal amount of simulation time was spent at each λ value. For AIM we ran expanded ensemble simulations where we alternate between taking molecular dynamics steps and attempting trial moves in λ space. That is, for AIM the amount of time spent at each λ value is determined by the algorithm.

In order to determine the best distribution of intermediate λ states we followed a simple strategy: (i) Conduct short simulations with a small set of intermediates. (ii) Generate a plot comparing slope values between AIM and fixed λ (iii) Determine the locations of curvature in the estimate of the free energy. (iv) Increase the density of intermediate states in locations of high curvature. (v) Repeat until all areas of high curvature have been well explored. We note that these steps should be performed for any method where the slope of the free energy is used to calculate free energies, including both TI and AIM. Similar
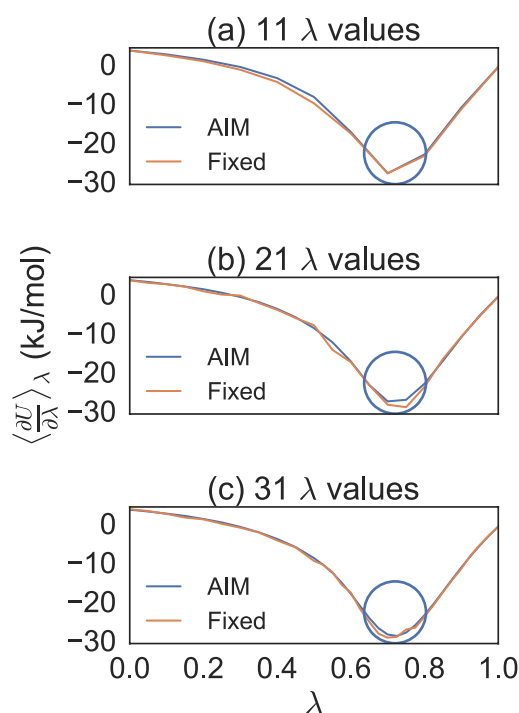
Figure 1 **Different λ densities for methane solvation free energy calculations.** Eight trial simulations of 100 ps per λ for 11, 21 and 31 λ values. This shows how the number of λ values were chosen to effectively compare AIM to fixed λ simulations. The circles indicate the region where the λ density needed to be increased.

steps would be performed for methods such as DEXP, IEXP, BAR and MBAR to ensure that energy differences are not too large between λ values.

RESULTS

Methane

After conducting short simulations, generating plots to determine locations of high curvature and increasing λ density in those regions, we averaged eight trial simulations of 100 ps per λ for separate λ distributions (see Fig. 1). We found, by progressively increasing the λ density between $\lambda = 0.5$ and $\lambda = 1.0$, that a distribution of 31 λ values gave us a dense enough distribution to properly compare AIM to fixed λ methods for the methane simulations.

Figure 2 is a violin plot to visualize the distribution and probability densities over the eight trials for each method as a function of simulation time per value of λ. A violin plot combines a box plot and a density plot to show the shape of the distribution around the mean. The thick black bar in the center represents the interquartile range, the white dot is the median and the thin black line going vertically through the middle represents the upper and lower adjacent values. Reading a violin plot is similar to reading a density plot. The thicker parts represent high frequency values and the thinner parts represent low
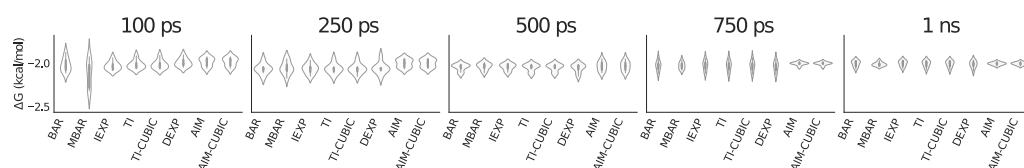
Figure 2 **Violin plot showing methane solvation results for 31 λ values averaged over eight trials.** A violin plot combines a box plot and a density plot to visualize the distribution and probability density. The graphic shows all methods have similarly converged at 1 ns per λ. AIM and AIM-CUBIC converge earlier than other methods at 750 ps per λ.

frequency values. The advantage of a violin plot over a box plot is that we are able to view the underlying distribution of the data.

In Fig. 2 at 100 ps per λ most methods have similar standard deviations of around 0.05 kcal/mol (visualized by the height of the violin shape in the figure), but the slower convergence of MBAR in this case leads to a larger standard deviation of around 0.16 kcal/mol. By 750 ps AIM has converged to a smaller standard deviation of around 0.02 kcal/mol compared to the other methods at around 0.05 kcal/mol.

GAG to GVG Mutation

For the GAG to GVG mutation we first tested a distribution of 41 λ values averaged over 8 trial simulations of 1 ps and 100 ps per λ; see Fig. 3. By reviewing the smoothness of the function we concluded that 41 λ values was sufficient. Figure 4 shows the convergence of the free energy for GAG to GVG over time for each method. At just 1 ps per λ value the AIM estimates are within less than 1.0 kcal/mol of the converged (1 ns per λ) result with a standard deviation of around 0.5 kcal/mol. All other methods are more than 2.0 kcal/mol from this converged result with larger standard deviations of around 1.0 kcal/mol. At 100 ps all estimates are less than 0.1 kcal/mol from the converged result, but AIM estimates have a standard deviation of around 0.1 kcal/mol compared to other methods at 0.4 kcal/mol. All methods have similarly converged at 1 ns per λ.

DISCUSSION

In the limit of infinite sampling, all rigorous methods (i.e., statistical mechanics-based methods), performed properly with the same force-field and parameters, will yield the same result within uncertainty. Often, it is of interest to define accuracy by comparing results to experimental data. However, given that the purpose of this study is to compare various computational methods, we define accuracy by comparing results to the value upon which all methods converge. For fixed λ simulations the sampling time is typically the same for each λ state. Sampling time must be increased whenever convergence has not been achieved. However, if bias is introduced by using an insufficient number of λ values in regions of high curvature, increased sampling leads to radical convergence problems (*Shyu & Ytreberg, 2009*; *Steinbrecher & Labahn, 2010*). If the curvature of the underlying free energy slope values is large, averaging over a state space that is not dense enough
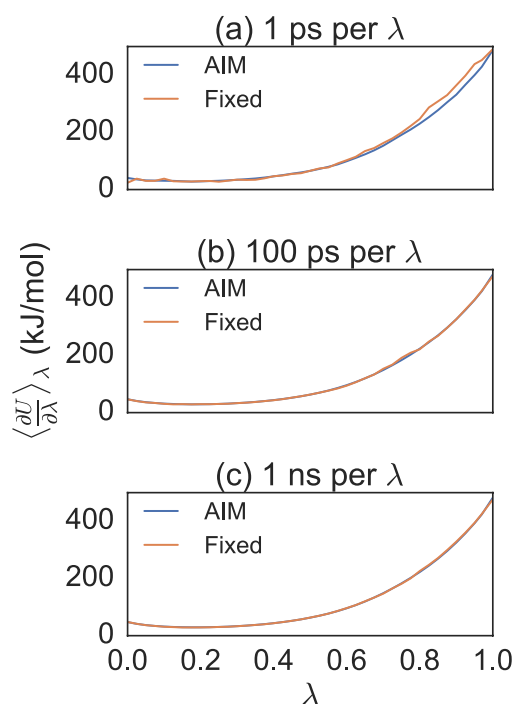
Figure 3   Different simulation times for alanine to valine mutation free energy calculations. Eight trial simulations of 41 λ values at 1 ps, 100 ps and 1 ns per λ. Note the smoothness of AIM versus fixed λ simulations. AIM requires less samples than fixed λ simulations to smooth the free energy function.
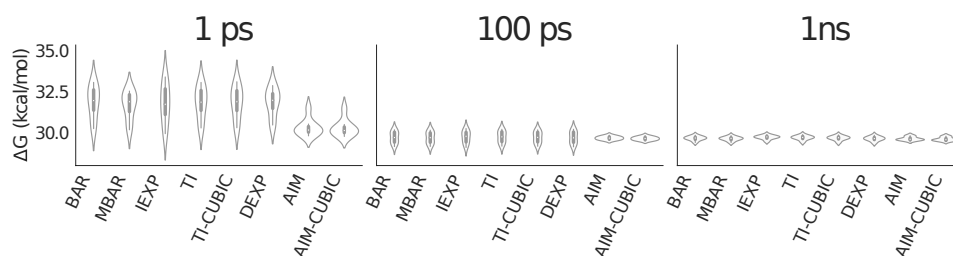
Figure 4   Violin plot showing alanine to valine mutation results for 41 λ values averaged over eight trials. The graphic shows all methods have similarly converged at 1 ns per λ. AIM and AIM-CUBIC converge more rapidly than other methods and are mostly converged at 100 ps per λ.

to fully describe the state function propagates this bias requiring significantly increased sampling time to achieve convergence. For TI, the bias will persist even for infinite sampling. In addition, increasing sampling time may not be realistic when dealing with limited computational resources. *Paliwal & Shirts (2011)* make a detailed argument to why convergence may not be possible for all systems due to hard limitations in computational resources.

In particular, both TI and AIM are calculating the same slope averages and should agree very well for simple systems and reasonably long simulation times. However, before

convergence is reached, due to the fact that AIM spends more time in some regions, we should not expect the approximation of AIM to exactly match TI with similar sampling time until the number of λ values has been sufficiently increased in high curvature regions. Once we have properly chosen the λ values then reasonably long simulations will lead to highly similar results between these two methods.

AIM is able to estimate the free energy for the amino acid mutation within 1.0 kcal/mol at a total simulation time of only 41 ps. This is quite remarkable since ±1.0 kcal/mol is typically the range that is desired for mutation studies. Of course, further studies using a broader range of amino acid change are needed, but it suggests that AIM may be suitable for quick estimation. We believe the reason that AIM performs so well in such cases is due to the Monte Carlo sampling that allows AIM to more efficiently sample λ space compared to fixed λ simulations.

The reader may note that AIM violates detailed balance since the acceptance criterion contains the free energy estimates that are updated continuously. AIM does however obey detailed balance asymptotically. As simulation time increases, the average free energy differences between λ values reach an equilibrium and detailed balance is satisfied. Once this equilibrium is attained the algorithm will sample all λ values equally, that is, the histogram of the number of configurations will become flat as a function of λ.

CONCLUSION

In this report we have implemented the adaptive integration method (AIM) for calculating free energy differences in GROMACS and applied it to two molecular systems. We have shown agreement within statistical uncertainty between AIM and a suite of fixed λ methods for methane solvation and an GAG to GVG mutation. We have also shown that AIM is more efficient than the other tested methods. That is, for a given amount of simulation time, AIM has a higher level of accuracy and precision. We anticipate these findings will extend to larger, more complex systems. Future studies will be performed to test whether this is the case.

Further, we found that running longer simulations with too few intermediate λ states generated results that were inconsistent between methods. The density and sampling convergence of the λ states directly influences the agreement between all the tested methods. Since some states will contribute disproportionately to the variance of the estimate, we found that generating short test simulations of different λ densities before attempting longer simulations is advisable.

Funding

Support for this research was provided by the National Science Foundation (DEB 1521049 and OIA 1736253) and the Center for Modeling Complex Interactions sponsored by the National Institutes of Health (P20 GM104420). Computer resources were provided in part by the Institute for Bioinformatics and Evolutionary Studies Computational Resources

Core sponsored by the National Institutes of Health (P30 GM103324). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Grant Disclosures
The following grant information was disclosed by the authors:
National Science Foundation: DEB 1521049, OIA 1736253.
The Center for Modeling Complex Interactions sponsored by the National Institutes of Health: P20 GM104420.
Institute for Bioinformatics and Evolutionary Studies Computational Resources Core sponsored by the National Institutes of Health: P30 GM103324.

Competing Interests
The authors declare there are no competing interests.

Author Contributions
- Christopher A. Mirabzadeh conceived and designed the experiments, performed the experiments, analyzed the data, performed the computation work, prepared figures and/or tables, authored or reviewed drafts of the paper, and approved the final draft.
- F. Marty Ytreberg conceived and designed the experiments, analyzed the data, prepared figures and/or tables, authored or reviewed drafts of the paper, and approved the final draft.

Data Availability
   All code used to generate the results, including scripts and modified GROMACS code are available in the Supplemental Files.

Supplemental Information
Supplemental information for this article can be found online at http://dx.doi.org/10.7717/peerj-cs.264#supplemental-information.

REFERENCES

Abraham M, Van der Spoel D, Lindahl E, Hess B. 2016. GROMACS user manual version 5.1.4. *Available at http//:www.gromacs.org* .

Aminpour M, Montemagno C, Tuszynski JA. 2019. An overview of molecular modeling for drug discovery with specific illustrative examples of applications. *Molecules* **24(9)**:Article 1693 DOI 10.3390/molecules24091693.

Bennett CH. 1976. Efficient estimation of free energy differences from Monte Carlo Data. *Journal of Computational Physics* **22**:245–268 DOI 10.1016/0021-9991(76)90078-4.

Berendsen HJ, Van der Spoel D, Van Drunen R. 1995. GROMACS: a message-passing parallel molecular dynamics implementation. *Computer Physics Communications* **91(1–3)**:43–56 DOI 10.1016/0010-4655(95)00042-E.

Bitetti-Putzer R, Yang W, Karplus M. 2003. Generalized ensembles serve to improve the convergence of free energy simulations. *Chemical Physics Letters* **377(5–6)**:633–641 DOI 10.1016/S0009-2614(03)01057-1.

Bruckner S, Boresch S. 2011. Efficiency of alchemical free energy simulations. I. A practical comparison of the exponential formula, thermodynamic integration, and Bennett's acceptance ratio method. *Journal of Computational Chemistry* **32(7)**:1303–1319 DOI 10.1002/jcc.21713.

Chipot C, Pohorille A. 2007. *Free energy calculations.* New York: Springer.

Chodera JD, Mobley DL, Shirts MR, Dixon RW, Branson K, Pande VS. 2011. Alchemical free energy methods for drug discovery: progress and challenges. *Current Opinion in Structural Biology* **21(2)**:150–160 DOI 10.1016/j.sbi.2011.01.011.

Chodera JD, Shirts MR. 2011. Replica exchange and expanded ensemble simulations as Gibbs sampling: Simple improvements for enhanced mixing. *Journal of Chemical Physics* **135(19)**:0–15 DOI 10.1063/1.3660669.

Cournia Z, Allen B, Sherman W. 2017. Relative binding free energy calculations in drug discovery: recent advances and practical considerations. *Journal of Chemical Information and Modeling* **57(12)**:2911–2937 DOI 10.1021/acs.jcim.7b00564.

Fasnacht M, Swendsen RH, Rosenberg JM. 2004. Adaptive integration method for Monte Carlo simulations. *Physical Review E* **69(5)**:056704-1–056704-15 DOI 10.1103/PhysRevE.69.056704.

Gapsys V, Michielssens S, Seeliger D, De Groot BL. 2015. pmx: automated protein structure and topology generation for alchemical perturbations. *Journal of Computational Chemistry* **36(5)**:348–354 DOI 10.1002/jcc.23804.

Geng C, Xue LC, Roel-Touris J, Bonvin AM. 2019. Finding the $\Delta\Delta G$ spot: are predictors of binding affinity changes upon mutations in protein–protein interactions ready for it? *Wiley Interdisciplinary Reviews: Computational Molecular Science* **9(5)**:e1410 DOI 10.1002/wcms.1410.

Gonçalves PFB, Stassen H. 2004. Calculation of the free energy of solvation from molecular dynamics simulations. *Pure and Applied Chemistry* **76(1)**:231–240 DOI 10.1351/pac200476010231.

Hossain S, Kabedev A, Parrow A, Bergström CA, Larsson P. 2019. Molecular simulation as a computational pharmaceutics tool to predict drug solubility, solubilization processes and partitioning. *European Journal of Pharmaceutics and Biopharmaceutics* **137(February)**:46–55 DOI 10.1016/j.ejpb.2019.02.007.

Jorgensen WL, Maxwell DS, Tirado-Rives J. 1996. Development and testing of the OPLS all-atom force field on conformational energetics and properties of organic liquids. *Journal of the American Chemical Society* **118(45)**:11225–11236 DOI 10.1021/ja9621760.

Kaus JW, Arrar M, McCammon JA. 2014. Accelerated adaptive integration method. *Journal of Physical Chemistry B* **118(19)**:5109–5118 DOI 10.1021/jp502358y.

Kaus JW, Mccammon JA. 2015. Enhanced ligand sampling for relative protein-ligand binding free energy calculations. *Journal of Physical Chemistry B* **119(20)**:6190–6197 DOI 10.1021/acs.jpcb.5b02348.

Klimovich PV, Shirts MR, Mobley DL. 2015. Guidelines for the analysis of free energy calculations. *Journal of Computer-Aided Molecular Design* **29**:397–411 DOI 10.1007/s10822-015-9840-9.

Knight JL, Brooks CL. 2009. λ—dynamics free energy simulation methods. *Journal of Computational Chemistry* **30(11)**:1692–1700 DOI 10.1002/jcc.21295.

Knight JL, Brooks CL. 2011. Multisite λ dynamics for simulated structure—activity relationship studies. *Journal of Chemical Theory and Computation* **7(9)**:2728–2739 DOI 10.1021/ct200444f.

Kofke DA. 2005. Free energy methods in molecular simulation. *Fluid Phase Equilibria* **228–229**:41–48 DOI 10.1016/j.fluid.2004.09.017.

Kong X, Brooks CL. 1996. Λ—dynamics: a new approach to free energy calculations. *The Journal of Chemical Physics* **105(6)**:2414–2423 DOI 10.1063/1.472109.

Lyubartsev AP, Førrisdahl OK, Laaksonen A. 1996. Calculations of solvation free energies by expanded ensemble method. In: *2nd international conference on natural gas hydrates. Toulouse (France), June 2–6, 1996, (6)*. 311–318.

Mackerell AD, Banavali N, Foloppe N. 2001. Development and current status of the CHARMM force field for nucleic acids. *Biopolymers (Nucleic Acid Sciences* **56**:257–265.

Miller CR, Johnson EL, Burke AZ, Martin KP, Miura TA, Wichman HA, Brown CJ, Ytreberg FM. 2016. Initiating a watch list for Ebola virus antibody escape mutations. *PeerJ* **4**:e1674 DOI 10.7717/peerj.1674.

Miller CR, Lee KH, Wichman HA, Ytreberg FM. 2014. Changing folding and binding stability in a viral coat protein: a comparison between substitutions accessible through mutation and those fixed by natural selection. *PLOS ONE* **9(11)**:e112988 DOI 10.1371/journal.pone.0112988.

Mobley DL, Liu S, Cerutti DS, Swope WC, Rice JE. 2012. Alchemical prediction of hydration free energies for SAMPL. *Journal of Computer-Aided Molecular Design* **26(5)**:551–562 DOI 10.1007/s10822-011-9528-8.

Paliwal H, Shirts MR. 2011. A benchmark test set for alchemical free energy transformations and its use to quantify error in common free energy methods. *Journal of Chemical Theory and Computation* **7(12)**:4115–4134 DOI 10.1021/ct2003995.

Petukh M, Li M, Alexov E. 2015. Predicting binding free energy change caused by point mutations with knowledge-modified MM/PBSA method. *PLOS Computational Biology* **11(7)**:1–23 DOI 10.1371/journal.pcbi.1004276.

Salomon-Ferrer R, Case DA, Walker RC. 2013. An overview of the Amber biomolecular simulation package. *Wiley Interdisciplinary Reviews: Computational Molecular Science* **3(2)**:198–210 DOI 10.1002/wcms.1121.

Shirts MR, Chodera JD. 2008. Statistically optimal analysis of samples from multiple equilibrium states. *Journal of Chemical Physics* **129(12)**:124105-1–124105-10 DOI 10.1063/1.2978177.

Shirts MR, Mobley DL. 2013. An introduction to best practices in free energy calculations. In: Monticelli L, Salonen E, eds. *Biomolecular simulations. Methods*

in molecular biology (Methods and Protocols), vol. 924. Totowa: Humana Press, 271–311 DOI 10.1007/978-1-62703-017-5.

Shirts MR, Mobley DL, Chodera JD. 2007. Chapter 4 alchemical free energy calculations: ready for prime time? *Annual Reports in Computational Chemistry* **3**(**February 2016**):41–59 DOI 10.1016/S1574-1400(07)03004-6.

Shirts MR, Pande VS. 2005. Comparison of efficiency and bias of free energies computed by exponential averaging, the Bennett acceptance ratio, and thermodynamic integration. *Journal of Chemical Physics* **122**(**14**):144107-1–144107-16 DOI 10.1063/1.1873592.

Shirts MR, Pitera JW, Swope WC, Pande VS. 2003. Extremely precise free energy calculations of amino acid side chain analogs: comparison of common molecular mechanics force fields for proteins. *Journal of Chemical Physics* **119**(**11**):5740–5761 DOI 10.1063/1.1587119.

Shyu C, Ytreberg FM. 2009. Reducing the bias and uncertainty of free energy estimates by using regression to fit thermodynamic integration data. *Journal of Computational Chemistry* **30**(**14**):2297–2304 DOI 10.1002/jcc.21231.

Śledź P, Caflisch A. 2018. Protein structure-based drug design: from docking to molecular dynamics. *Current Opinion in Structural Biology* **48**:93–102 DOI 10.1016/j.sbi.2017.10.010.

Steinbrecher T, Labahn A. 2010. Towards accurate free energy calculations in ligand protein-binding studies. *Current Medicinal Chemistry* **17**(**8**):767–785 DOI 10.2174/092986710790514453.

Sun Y, Spellmeyer D, Pearlman DA, Kollman P. 1992. Simulation of the solvation free energies for methane, ethane, and propane and corresponding amino acid dipeptides: a critical test of the bond-PMF correction, a new set of hydrocarbon parameters, and the gas phase-water hydrophobicity scale. *Journal of the American Chemical Society* **114**(**17**):6798–6801 DOI 10.1021/ja00043a027.

Ytreberg FM, Swendsen RH, Zuckerman DM. 2006. Comparison of free energy methods for molecular systems. *Journal of Chemical Physics* **125**(**18**):1–11 DOI 10.1063/1.2378907.

Zhan YA, Wu H, Powell AT, Daughdrill GW, Ytreberg FM. 2013. Impact of the K24N mutation on the transactivation domain of p53 and its binding to murine double-minute clone 2. *Proteins: Structure, Function and Bioinformatics* **81**(**10**):1738–1747 DOI 10.1002/prot.24310.

Zhan YA, Ytreberg FM. 2015. The cis conformation of proline leads to weaker binding of a p53 peptide to MDM2 compared to trans. *Archives of Biochemistry and Biophysics* **575**:22–29 DOI 10.1016/j.abb.2015.03.021.

Zhang H, Liao L, Saravanan KM, Yin P, Wei Y. 2019. DeepBindRG: a deep learning based method for estimating effective protein—ligand affinity. *PeerJ* **7**:e7362 DOI 10.7717/peerj.7362.

Zwanzig RW. 1954. High-temperature equation of state by a perturbation method. I. Nonpolar gases. *Journal of Chemical Physics* **22**(**8**):1420–1426 DOI 10.1063/1.1740409.