



universität
wien

DISSERTATION / DOCTORAL THESIS

Titel der Dissertation / Title of the Doctoral Thesis

„Alchemical Free Energy Calculations: Enhancements and Applications “

verfasst von / submitted by

Johannes Karwounopoulos, B.Sc. M.Sc.

angestrebter akademischer Grad / in partial fulfilment of the requirements for the degree of
Doktor der Naturwissenschaften (Dr. rer. nat.)

Wien, 2023 / Vienna 2023

Studienkennzahl lt. Studienblatt /
degree programme code as it appears on the student
record sheet:

UA 796 605 419

Dissertationsgebiet lt. Studienblatt /
field of study as it appears on the student record sheet:

Chemie

Betreut von / Supervisor:

Univ.-Prof. Mag. Dr. Stefan Boresch

Acknowledgements

I want to thank all the people whom I met during the last three years of my Ph.D.

First, I want to thank Stefan for being such a wonderful supervisor. I want to thank you for all the support and encouragement over the past few years. You are an inspiring scientist, and I am honored that you shared so much of your knowledge and experience with me. You gave me all the freedom to experience things on my own, but I always felt assured that I could rely on your guidance whenever I needed it.

Next, I want to thank Marcus. Working together on numerous projects was truly enjoyable, and your enthusiasm and creativity consistently served as a great source of inspiration. Despite we did not often meet in person, you were always available for me, and I learned a great deal from our interactions.

I want to thank Åsmund for our very nice collaboration. I met you as a very curious person who is always interested in learning something new. Engaging in the exchange of ideas, thoughts, and knowledge with you was enjoyable and infectious.

A special thanks goes to my colleagues András and Florian who accompanied me for the last three years. You both made each workday enjoyable, and as a team, there was no problem, bug, or challenge that we couldn't overcome together, thank you for everything!

Continuing, I would like to express my gratitude to all the members and students of the Institute of Computational Biological Chemistry whom I had the pleasure of meeting in the past few years: Alexander Grasser, Anna Picha, Christian Fellingner, Mazin Almarashi, Marion Sappl, Márta Gódcény, Klara Pribanova, and Sara Tkaczyk. I am grateful for the opportunity to have met each of you and shared time with you.

The positive atmosphere at the Institute owes much to the presence of Othmar Steinhauser and Christian Schröder. Sharing lunch and cake breaks with you both was consistently delightful. Thank you for creating such a nice atmosphere!

Finally, I want to thank my friends and family for supporting me during this journey. I am extremely thankful for my partner, Lisa, who supported and encouraged me during these years.

Zusammenfassung

Neben vielen anderen Anwendungsmöglichkeiten können molekulardynamische (MD) Simulationen verwendet werden, um freie Energieunterschiede zwischen verschiedenen Zuständen eines molekularen Systems zu berechnen. Dies ermöglicht die Vorhersage von Gleichgewichtskonstanten, Reaktionsraten und Bindungsaffinitäten. Bei der Verwendung von MD Methoden zur Berechnung von freien Energien können typischerweise drei mögliche Fehlerquellen auftreten. Diese wären: 1) unzureichendes Sampling, 2) Fehler beim Aufsetzen der Simulation, wie beispielsweise die Auswahl des falschen Tautomers, und 3) Schwachstellen in den Kraftfeldern. Es gibt mehrere Strategien zur Bewältigung dieser Probleme, wovon einige Möglichkeiten in dieser Arbeit genauer erörtert werden.

Zum Aufsetzen freier Energie Simulationen stehen eine wachsende Anzahl von Tools zur Verfügung. Diese Tools sind oft eng mit einem MD Programm verbunden, und nutzen meist speziellen Code um freie Energie Simulationen zu ermöglichen. Eine Ausnahme bildet `transformato`, das auf dem Common Core/Serial Atom Insertion (CC/SAI) Ansatz beruht. Dies erfordert keinen speziellen Code für die alchemische Transformation und kann theoretisch mit jedem MD Programm verwendet werden.

Hier stellen wir zunächst die Erweiterung von `transformato` zur Berechnung von freien relativen Bindungsenergieunterschieden (engl. relative binding free energies, RBFE) vor. Dieser Ansatz wurde durch die Berechnung von RBFEs für 76 Ligandenpaare aus etablierten Datensätzen validiert. Unsere Ergebnisse zeigen eine gute Übereinstimmung mit experimentellen Daten (engl. Mean Absolute Error (MAE) von 0,85 kcal/mol).

Darüber hinaus wurde `transformato` erweitert, um die Berechnung absoluter freier Solvatationsenergien (engl. absolute solvation free energy, ASFE) zu ermöglichen. Wir haben ASFEs für einen Großteil der Moleküle der FreeSolv Datenbank (621 von 642) berechnet. Dabei wurde ein MAE von 1,12 kcal/mol erhalten. Der Vergleich mit dem Experiment ermöglicht es des Weiteren, Stärken und Schwächen des CGenFF Kraftfelds zu erkennen.

Um das Problem unzureichender Kraftfelder anzugehen, werden kontinuierlich Anstrengungen unternommen, um Kraftfeldparameter zu verfeinern und besser an experimentelle Beobachtungen anzupassen. Trotz dieser Fortschritte gibt es Situationen, in denen Kraftfelder allein möglicherweise nicht ausreichen. Um dies zu beheben, werden Ansätze wie kombinierte Quantenmechanik/Molekulardynamik (QM/MM) Simulationen verwendet. Besonders von Interesse sind dabei Ansätze, bei denen ein Machine-Learning (ML) Potential verwendet wird, um QM Energien zu reproduzieren. Wir haben einen in-

direkten freien Energiepfad angewendet, um die intramolekulare Energie einiger Moleküle der vorherigen ASFE Simulationen auf eine ML Beschreibung (in unserem Fall das ANI-2x ML Potential) zu korrigieren. Wird diese Korrektur auf eine große Anzahl an Molekülen angewandt, lässt sich kaum eine Verbesserung im Vergleich zum Experiment beobachten. Für einige Moleküle, die jedoch besonders schlecht abschneiden kann durch die indirekte Korrektur eine kleine Verbesserung bei der Vorhersage von ASFEs beobachtet werden.

Abstract

Among many other applications, molecular dynamics (MD) simulations can be used to calculate the free energy differences between different states of a molecular system, allowing the prediction of equilibrium constants, reaction rates, and binding affinities. There are three sources of error in MD-based free energy simulations (FES). These are 1) insufficient sampling, 2) errors in the simulation setup, such as incorrect tautomer selection, and 3) inadequacies of the force fields. There are several strategies for dealing with these problems, some of which are discussed in this thesis.

To avoid incorrect simulation setups, a growing number of tools are available to assist in setting up FES. They are often closely tied to an MD engine and rely on special purpose code. An exception is the recently developed `transformato` tool based on the common core / serial-atom-insertion (CC/SAI) approach, which does not require any special code for alchemical transformations and thus can, theoretically, be used with any MD engine.

Here, we first report the extension of `transformato` to support the calculation of relative binding free energies (RBFE). This approach was validated by calculating the RBFEs for 76 ligand pairs from established datasets. Our results show reasonable agreement with experimental data (mean absolute error (MAE) of 0.85 kcal/mol).

Furthermore, the capabilities of `transformato` have been extended to allow the computation of absolute solvation free energies (ASFE). We calculated ASFEs for most molecules in the FreeSolv database (621 out of 642). An MAE of 1.12 kcal/mol was obtained. The critical comparison to experiment makes it possible to discern strengths and weaknesses of the CHARMM general force field (CGenFF).

To address the problem of inadequate force fields, various efforts are being made to refine force field parameters to better match experimental observations. However, there are situations where force field-based simulations alone may not suffice. In such cases, approaches such as combined Quantum Mechanics/ Molecular Mechanics (QM/MM) simulations can be used to incorporate the higher accuracy of quantum chemical methods. Recently, approaches, where a machine learning (ML) potential is used to reproduce QM energies, have become increasingly popular. We applied an indirect free energy pathway, correcting the intramolecular energy of the compounds from the previous ASFE simulations to an ML description (in our case, the ANI-2x machine learning potential). While we did not see an overall improvement when applying the approach to a large number of molecules, for some poorly performing compounds the predicted solvation free energy was in better agreement with the experimental value.

Contents

1	Introduction	13
1.1	Molecular dynamics simulations with classical force fields	13
1.2	Free energy calculations	15
1.2.1	Free energy perturbation	15
1.2.2	Bennett Acceptance Ratio	16
1.2.3	Thermodynamic Integration	17
1.3	Alchemical pathways	18
1.3.1	Dummy atoms	19
1.3.2	Soft core potentials	20
1.3.3	Serial-atom-insertion method	21
1.4	Indirect free energy calculations	22
1.5	The <code>transformato</code> Python package	24
1.5.1	Conceptual basis of <code>transformato</code>	24
1.5.2	Connecting a ligand to its common core	26
1.5.3	The role(s) of <code>transformato</code> within an FES workflow	28
2	Publications	29
2.1	Relative binding free energy calculations with <code>transformato</code>: A molecular dynamics engine-independent tool <i>Front. Mol. Biosci.</i> 9:954638 (2022)	31
2.2	Calculations of Absolute Solvation Free Energies with <code>transformato</code>—Application to the FreeSolv Database Using the CGenFF Force Field <i>J. Chem. Theory Comput.</i> 19, 5988-59998 (2023)	49
2.3	Insights and Challenges in Correcting Force Field Based Solvation Free Energies Using A Neural Network Potential <i>ChemRxiv.</i> (2023)	63
3	Discussion	85
3.1	Summary	85
3.2	Further/ongoing enhancements of <code>transformato</code>	86
3.3	Concluding discussion and outlook	89
	Bibliography	91
	Appendix	95
A	Supplementary information <i>Front. Mol. Biosci.</i> 9:954638 (2022) . . .	95

B	Supplementary information <i>J. Chem. Theory Comput.</i> 19, 5988-59998 (2023)	111
C	Supplementary information <i>ChemRxiv.</i> (2023)	125

Abbreviations

ASFE	absolute solvation free energy
BAR	Bennett acceptance ratio
CC	Common Core
CGenFF	CHARMM General Force Field
FEP	free energy perturbation
FES	free energy simulation
LJ	Lennard-Jones
LRC	long range correction
MBAR	multistate Bennett acceptance ratio
MD	molecular dynamics
ML	machine learning
MM	molecular mechanics
PME	Particle Mesh Ewald
QM	quantum mechanics
RBFE	relative binding free energy
RSFE	relative solvation free energy
SAI	Serial-Atom-Insertion
TI	Thermodynamic Integration

1 Introduction

1.1 Molecular dynamics simulations with classical force fields

Binding affinity prediction is becoming an important step in the lead optimization stage of drug discovery projects. Especially relative binding free energy (RBFE) predictions, which estimate the binding affinity of a congeneric series of ligands, have become a powerful tool in recent years.¹⁻⁶ The calculation of free energy differences, which is the focus of this thesis, uses molecular dynamics (MD) simulations as the underlying sampling method.

MD simulations study the dynamics of (bio)molecular systems at atomic resolution, following atomic motions over time. Atoms are represented by their positions $\{\vec{r}_i\}$ within a three-dimensional box. For each particle i at a specific time t , a force $\vec{F}_i$ is calculated using Newton's law of motion:

$$\{\vec{F}_i\} = m_i \cdot \vec{a}_i \quad (1.1)$$

From the positions $\{\vec{r}_i\}$ of and the forces $\{\vec{F}_i(t)\}$ acting on the particles at the time t as well as their velocities at the time t , the position at the next time step $\{\vec{r}(t + \Delta t)\}$ can be calculated. Typically, forces are calculated with the help of force fields,⁷⁻¹⁰ which represent a simplified description to calculate the forces and thus the potential energy (U_{total}) of a system of interest.

One distinguishes between intramolecular forces (U_{bonded}) and intermolecular forces ($U_{\text{non-bonded}}$).

$$U_{\text{total}} = U_{\text{bonded}} + U_{\text{non-bonded}} \quad (1.2)$$

Intramolecular forces are again subdivided into:

$$U_{\text{bonded}} = U_{\text{bond}} + U_{\text{angle}} + U_{\text{dihedral}} + U_{\text{improper}} + U_{\text{cmap}} \quad (1.3)$$

Bond and angle interactions are usually described by a harmonic potential. The functional form of the dihedrals varies between the different force fields. U_{cmap} is a term used only in specific force fields, for example in the CHARMM force field family.¹¹ The bonded energy terms need to be calculated only within a molecule.

Computationally more challenging are the non-bonded interactions ($U_{\text{non-bonded}}$) which need to be calculated between atoms of different molecules or within molecules where

the atoms are more than three bonds apart:

$$U_{\text{non-bonded}} = U_{\text{electrostatic}} + U_{\text{van der Waals}} \quad (1.4)$$

The van der Waals interaction between two atoms (ij) consists of a repulsive term at short distances and an attractive term at longer distances and is usually treated with a Lennard-Jones (LJ) potential.

$$U_{\text{non-bonded}} = \underbrace{\sum_i \sum_j 4\epsilon_{ij} \left[\left(\frac{\sigma_{ij}}{r_{ij}} \right)^{12} - \left(\frac{\sigma_{ij}}{r_{ij}} \right)^6 \right]}_{\text{Lennard-Jones}} + \underbrace{\sum_i \sum_j \frac{q_i q_j}{4\pi\epsilon_0 r_{ij}}}_{\text{electrostatic}} \quad (1.5)$$

The parameter σ_{ij} corresponds to the distance between the two atoms i, j , where U_{LJ} is zero, and ϵ_{ij} is the depth of the potential well. Electrostatic interactions between atoms are described using a Coulomb potential with the partial atomic charges q_i and q_j for the two atoms i and j .

In theory, the non-bonded interactions need to be calculated between each pair of atoms in a simulation box; this would increase the simulation time drastically. LJ interactions decrease rapidly with increasing distance, which allows the introduction of cut-off distances. Interactions are only considered for atom pairs below this distance (typically 12 Å). In most approaches, the interaction is not turned off abruptly, but smoothly by employing a switching function. In addition, long-range effects beyond the cut-off distance can be accounted for analytically by making use of long range correction (LRC) approaches.

Electrostatic interactions are long-ranged, so a cut-off as for LJ would introduce a notable error in the simulation. To avoid using a cut-off, the so-called Ewald summation can be applied. In this approach, the electrostatic interactions are divided into two contributions: The direct (real space) part, which accounts for the short-range interactions between the atoms and which is evaluated directly using a cut-off distance, and the indirect (reciprocal) part to account for the long-ranged part of the electrostatic interactions.¹² Today, usually an accelerated variant of the Ewald summation, the so-called, Particle Mesh Ewald (PME) is used. Here, the point charges are replaced by a grid-based charge distribution, making it possible to use Fast Fourier Transformation to calculate the reciprocal space part extremely efficiently.^{13,14}

1.2 Free energy calculations

As mentioned, the so-called free energy simulation (FES) are one important application of MD simulations. The Helmholtz free energy of a molecular system with a constant number of particles N , in the NVT ensemble, is given by:

$$A = -k_B T \ln Z \quad (1.6)$$

Here, k_B is the Boltzmann constant, T is the temperature and Z is the partition function. In most applications, the Gibbs (G) rather than the Helmholtz free energy A is of interest. Since the statistical mechanical derivations outlined below are clearer in the NVT ensemble, we describe the techniques used in this work in terms of $(\Delta)A$. Furthermore, at room temperature and pressure, differences between ΔA and ΔG are miniscule.

The canonical partition function Z provides information on how the energy among all different possible microstates i of a system at a given temperature is distributed:

$$Z = \sum_i \exp\left(-\frac{U_i}{k_B T}\right) \quad (1.7)$$

In general, the absolute free energy of a system can neither be calculated nor measured experimentally. Thus, the goal is the calculation of the free energy difference between two states of a molecular system.¹⁵ In other words, we want to compute the free energy difference (ΔA) between an initial state (0) and a final state (1)

$$\Delta A_{0 \rightarrow 1} = A_1 - A_0 = -k_B T \ln \frac{Z(1)}{Z(0)} \quad (1.8)$$

where the second equality follows from Equation 1.6.

The direct application of Equation 1.8 would necessitate the evaluation of the partition functions at state 0 and 1, respectively, which is impossible except for toy systems. This apparent obstacle can be circumvented by various reformulations of Equation 1.8, which avoid the need to evaluate the full partition functions.

1.2.1 Free energy perturbation

One technique, referred to as free energy perturbation (FEP), alternatively known as Exponential Averaging (EXP) or the Zwanzig relationship,¹⁶ makes it possible to estimate the free energy difference between two states 0 and 1 as follows:

$$\Delta A_{0 \rightarrow 1} = -k_B T \ln \left\langle \exp \left[-\frac{U(1) - U(0)}{k_B T} \right] \right\rangle_0 \quad (1.9)$$

or,

$$\Delta A_{1 \rightarrow 0} = -k_B T \ln \left\langle \exp \left[-\frac{U(0) - U(1)}{k_B T} \right] \right\rangle_1 \quad (1.10)$$

The angular brackets $\langle \rangle$ indicate an ensemble average, which can be approximated using MD (or Monte Carlo) simulations. The subscripts 0 and 1 indicate the state at which the coordinate frames used to compute the ensemble average were sampled. Note that in Equation 1.9 sampling is only needed at the initial state (0), and the free energy difference $\Delta A_{0 \rightarrow 1}$ is obtained by reevaluating the configurations sampled at this state with the potential energy functions for state 1. Naturally, the same procedure can also be conducted in the reverse direction: by sampling state 1 and then reevaluating the configurations using the potential energy functions of state 0 (see Equation 1.10).

However, if the potential energy surface of the two states is too dissimilar, the convergence of Equations 1.9 and 1.10 is poor. As pointed out, e.g., in Ref.,¹⁵ a sufficient number of configurations sampled at state 0 need to be low-energy configurations at state 1, and vice versa. This requirement means that the distribution of energy differences $\Delta U_{0 \rightarrow 1} = U(1) - U(0)$ in the forward direction, $p(\Delta U_{0 \rightarrow 1})$ and the backward distribution $p(-\Delta U_{1 \rightarrow 0})$ must have considerable overlap. The distribution of forward and backward energy differences can be viewed as a one-dimensional representation of the phase space overlap between states 0 and 1.¹⁵

In the case of poor convergence, one can divide the calculation of the free energy difference between the two end states 0 and 1 into multiple smaller steps. By introducing multiple intermediate steps connecting the two end states 0 and 1, the similarity and thus the phase space overlap between the two end states is increased. The necessary intermediate states are introduced with the help of the so-called coupling parameter λ , controlling the mixing of the end states. In the simplest possible form, linear mixing is used, i.e.,

$$U(\lambda) = \lambda U(1) + (1 - \lambda)U(0). \quad (1.11)$$

The free energy difference of interest is then obtained as follows:

$$\Delta A = \sum_{l=1}^{n-1} \Delta A(\lambda_l \rightarrow \lambda_{l+1}) \quad (1.12)$$

Here $U(\lambda_0) = U(0)$, the potential energy function of the initial state, and $U(\lambda_n) = U(1)$ that of the final state.

1.2.2 Bennett Acceptance Ratio

Bennett acceptance ratio (BAR) can be considered an extension of the FEP approach.¹⁷ This method requires sampling at both end states 0 and 1:

$$\Delta A = k_B T \left(\ln \frac{\langle f(\Delta U_{1 \rightarrow 0} + C) \rangle_1}{\langle f(\Delta U_{0 \rightarrow 1} - C) \rangle_0} \right) + C \quad (1.13)$$

where $f(x)$ denotes the Fermi function $f(x) = (1 + \exp(\frac{x}{k_B T}))^{-1}$ and

$$C = k_B T \ln \frac{Z_0 N_1}{Z_1 N_0} \quad (1.14)$$

Here, N_0 and N_1 represent the number of data points for computing the ensemble averages in the forward and backward directions, respectively. One starts with an estimate for C ; Equation 1.13 is then iterated until the condition

$$\langle f(\Delta U_{1 \rightarrow 0} + C) \rangle_1 = \langle f(\Delta U_{0 \rightarrow 1} + C) \rangle_0 \quad (1.15)$$

is fulfilled. With C thus determined, one can calculate the free energy difference as follows:

$$\Delta A_{0 \rightarrow 1} = -k_B T \ln \frac{N_1}{N_0} + C \quad (1.16)$$

As pointed out in the previous subsection, the convergence of FEP relies on a substantial overlap between the configuration space of each state. The BAR method is more robust in this respect; in principle, overlap between $p(\Delta U_{0 \rightarrow 1})$ and $p(-\Delta U_{1 \rightarrow 0})$ is sufficient to guarantee a correct free energy difference.¹⁵ Even though BAR is comparatively less dependent on overlap compared to FEP, in most practical applications there is no phase space overlap between the end states. In such cases, intermediate states can be inserted, as just outlined for FEP.

The multistate Bennett acceptance ratio (MBAR) method is an extension of BAR.¹⁸ While BAR combines samples from two states, MBAR takes data from all i intermediate states, from which one constructs an $i \times i$ matrix of energy differences between the respective intermediate states. The MBAR method is capable of using all this information and computes the free energy difference $\Delta A_{0 \rightarrow 1}$ in a single step.

1.2.3 Thermodynamic Integration

In contrast to FEP, BAR, and MBAR, Thermodynamic Integration (TI) directly utilizes the coupling parameter λ . This approach considers λ as an additional, continuous variable on which the potential energy function depends, i.e., $U = U(\{\vec{r}_i\}, \lambda)$.¹⁹ By employing the fundamental theorem of calculus, the free energy difference can be expressed as follows:

$$\Delta A = A(1) - A(0) = \int_0^1 \frac{dA(\lambda)}{d\lambda} d\lambda = -k_B T \int_0^1 \left[\frac{d \ln Z(\lambda)}{d\lambda} \right] d\lambda \quad (1.17)$$

finally resulting in:

$$\Delta A = \int_0^1 d\lambda \left\langle \frac{\partial U(\lambda)}{\partial \lambda} \right\rangle_\lambda \quad (1.18)$$

The integration over λ has to be carried out numerically. One performs multiple simulations at discrete λ values ($\lambda_0 = 0 < \lambda_1 < \lambda_2 < \dots < \lambda_n = 1$) to compute the ensemble average $\left\langle \frac{\partial U(\lambda)}{\partial \lambda} \right\rangle_\lambda$ for each λ_i , followed by the application of a numerical quadrature method to estimate the integral. Often, the trapezoidal rule is used; more recent studies show that integration based on a cubic spline approximation of the integrand is more accurate.^{20–22}

1.3 Alchemical pathways

With the previously introduced methods (FEP, BAR and TI), one can estimate free energy differences ΔA or ΔG between states. By combining these with suitable thermodynamic pathways, one can calculate the quantities of interest, such as (relative) solvation and binding free energy differences. Since in practice one is mostly interested in Gibbs free energy differences, we now switch the notation from ΔA to ΔG . As a first example of the use of thermodynamic cycles, we consider the absolute solvation free energy (ASFE) (see the left box in Figure 1.1), which provides information about the affinity of a solute for a particular solvent environment.^{23–25} Instead of directly calculating ΔG^{solv} , an alchemical pathway can be followed, where all non-bonded interactions of the solute are turned off in both the gas phase ($\Delta G_{L_1}^{\text{gas}}$) and the solvent phase ($\Delta G_{L_1}^{\text{aq}}$), a process known as annihilation.²⁶

$$\Delta G^{\text{solv}} = \Delta G_{L_1}^{\text{gas}} - \Delta G_{L_1}^{\text{aq}} \quad (1.19)$$

However, for large molecules, this approach can be challenging. Furthermore, it may be sufficient to know (and hence calculate) the relative solvation free energy (RSFE), $\Delta \Delta G_{L_1 \rightarrow L_2}^{\text{solv}}$ (see the right box in Figure 1.1).²⁷ Here, the different solvation behavior of two (slightly) different molecules is determined. For this, one compound is mutated into the other one, once in the solvent phase and once in the gas phase. Again, by following an alchemical pathway, the RSFE can be obtained,

$$\Delta \Delta G_{L_1 \rightarrow L_2}^{\text{solv}} = \Delta G_{L_1 \rightarrow L_2}^{\text{aq}} - \Delta G_{L_1 \rightarrow L_2}^{\text{gas}} \quad (1.20)$$

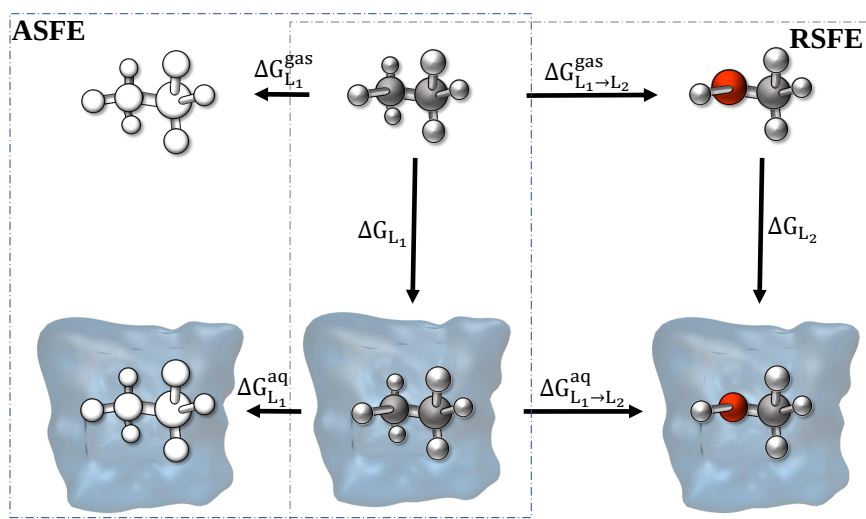


Figure 1.1 Thermodynamic cycles used to compute ASFE and RSFE. Left box: alchemical route for computing the ASFE of ethane through two annihilation steps in solution and in the gas phase. Right box: Computation of the RSFE when mutating the ethane molecule into a methanol molecule along the alchemical pathways in the gas phase and in solution.

Solvation free energies, whether absolute or relative, are of significant interest in understanding the solubility of solutes and are sometimes employed in force field development.²⁸ For practical applications related to drug design and molecular modeling, binding free energies are more essential.^{1,29,30}

Again, alchemical pathways can be used to avoid the explicit computation of two absolute free energy differences for two ligands, if one is mostly interested in their RBE difference. Using an analogous cycle as shown for RSFE above (right side of Figure 1.1), one carries out alchemical mutations between L_1 and L_2 in solution (ligand) and when bound to the receptor (complex).

$$\Delta\Delta G_{L_1 \rightarrow L_2}^{\text{bind}} = \Delta G_{L_1 \rightarrow L_2}^{\text{complex}} - \Delta G_{L_1 \rightarrow L_2}^{\text{ligand}} \quad (1.21)$$

A large-scale practical example of this approach is the subject of Section 2.1. To compute free energy differences with the help of these alchemical pathways, it is often necessary to use modifications in the MD setup. Some techniques are explained in greater detail in the following sections (1.3.1, 1.3.2, 1.3.3).

1.3.1 Dummy atoms

During the calculation of relative free energy differences, when mutating one molecule into another, the number of atoms in the initial and final state of the system may be different. However, simulations are carried out in the canonical ensemble, where the number of atoms must remain constant. To address this issue, the notion of so-called dummy atoms is introduced. Dummy atoms are atoms that are included in the simulation but only participate in bonded interactions and have no non-bonded interactions (e.g., they have a partial charge of zero). They are essential when alchemically transforming molecules to keep the number of atoms present in the simulation system constant.²¹ Two main approaches are used to handle dummy atoms: the single topology and the dual topology approach.

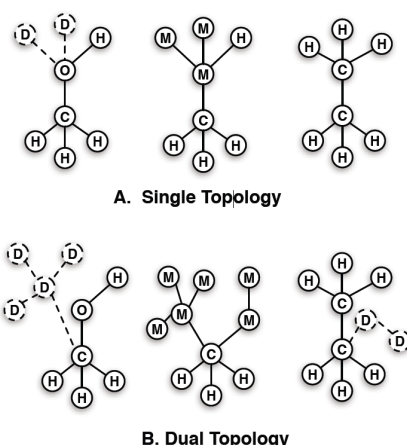


Figure 1.2 Difference between (A) the single topology and (B) the dual topology approach. D stands for non-interacting dummy atoms, M stands for nonphysical intermediate atoms. In the single topology approach (A), dummy atoms are present at least at one of the two end states to ensure that the number of atoms is identical for both states. In the dual topology approach (B), atoms belonging to one physical end state are present at both end states, either as physical or as dummy atoms. Figure taken from here http://www.alchemistry.org/wiki/Constructing_a_Pathway_of_Intermediate_States and mentioned by Mey et al. in Reference [26]

In the single topology approach, each molecule is described by its own topology. If the number of atoms differs between the two molecules, the molecule with fewer atoms is supplemented with additional dummy atoms to maintain a consistent number of atoms, the other end state can either consist entirely of physical atoms (see Figure 1.2A) or include dummy atoms at the endstate as well. In the dual topology approach (Figure 1.2B), at each of the two end states, atoms not present in the other end state are present as additional dummy atoms. This approach avoids mutating atoms, as done in the single topology approach, because they are only turned on or off. However, the dual topology approach requires more dummy atoms in total. In principle, both approaches yield the same results, and the choice between them often depends on the specific simulation code used.²⁶

1.3.2 Soft core potentials

When adding or deleting an atom in the presence of a solvent, multiple problems can occur for calculating the LJ potential, as introduced as part of the non-bonded interactions in Equation 1.5. When computing the free energy difference between two states 0 and 1 following a coupling parameter λ , several problems can arise.³¹ At $\lambda=0$, a non-interacting particle may be at the same position as a fully interacting particle. This leads to a division by zero ($r_{ij}=0$). However, since all major simulation programs contain provisions guarding against crashes caused by divisions by zero, particles sitting on top of each other have no discernible side effects in practice. A more severe problem arises when λ is close to the end state where the particle is decoupled. Miniscule changes

in distance between the almost-dummy atom and a physical atom can cause dramatic, abrupt changes in their interaction energy, which leads to problems when integrating the equations of motion.^{32,33} An additional complication arises in TI. At the endpoint where one or more particles have no interactions with the environment, the integrand $\left\langle \frac{\partial U(\lambda)}{\partial \lambda} \right\rangle_\lambda$ may become singular and, in particular, for a linear dependence on λ , the integration diverges.³¹

To circumvent these issues, it is common to introduce a soft-core potential, which is dependent on the coupling parameter λ . A soft-core (SC) modified LJ potential for two particles i and j can look as follows:³⁴

$$E_{\text{LJ}}^{\text{SC}} = \lambda 4\epsilon_{ij} \left(\frac{\sigma^{12}}{(r_{ij}^2 + (1-\lambda)\delta)^6} - \frac{\sigma^6}{(r_{ij}^2 + (1-\lambda)\delta)^3} \right) \quad (1.22)$$

Here, δ is a positive constant, for which Zacharias et al. suggested a value of $5 \text{ \AA}^2$.³⁴ At $\lambda=0$, even if $r_{ij}=0$ there will be never a division by zero in Equation 1.22. Furthermore, for λ close to zero, the term $(r_{ij}^2 + (1-\lambda)\delta)$ does not become severely small; hence, all forces remain bounded. Using a functional form as in Equation 1.22, the integrand $\left\langle \frac{\partial U(\lambda)}{\partial \lambda} \right\rangle_\lambda$ remains finite. Depending on the MD engine, various functional forms of soft-core potentials are used.^{32,35,36}

1.3.3 Serial-atom-insertion method

Soft core potentials, as just described, are the usual workaround to avoid the van der Waals endpoint problem. An alternative is the so-called Serial-Atom-Insertion (SAI) method.³⁷ In this approach, one does not turn off all LJ interactions simultaneously as a function of λ . Instead, the LJ interactions of the affected atoms are turned off one by one. This means that an atom is either fully interacting or fully turned off, and the problematic case where LJ interactions are almost turned off never arises. One can view SAI as a sequence of particle insertions/deletions, a well-understood problem.³⁸ One should note that in SAI λ is not a continuous variable anymore, i.e., SAI can only be used together with FEP or BAR/MBAR.

1.4 Indirect free energy calculations

The vast majority of FES use force fields to describe inter- and intramolecular interactions. While force fields are computationally very efficient and broadly applicable, they are based on approximations and empirical data. While they are highly optimized for describing proteins and biomolecules, small organic molecules can be challenging. Thus, approaches partially relying on more accurate quantum mechanics (QM) descriptions of interactions have been developed. Specifically, in QM/MM hybrid methods^{39–42} a small part of a system, e.g., the ligand, is described by QM, while the environment, e.g., the protein and surrounding water, are described at the force field level (molecular mechanics (MM)). It is possible to employ QM/MM methodologies for the calculation of free energy differences directly.^{43,44} However, this is computationally demanding and faces practical challenges as some concepts essential for free energy simulations do not readily apply to QM/MM descriptions, such as the treatment of dummy atoms or the incorporation of soft core potentials.

To overcome these limitations, indirect QM/MM free energy simulations have been developed.^{45–49} They use a computationally cheaper description of the potential energy (e.g., a molecular mechanics force field) to perform the (alchemical) free energy calculation. The free energy estimate obtained is then corrected with a more accurate but also more expensive description of the potential energy (e.g., a QM/MM potential).^{50,51} A typical cycle is depicted in Figure 1.3. In the example shown, the solvation free energy $\Delta G_{\text{high}}^{\text{solv}}$ at a high level of theory, e.g., a QM/MM description of interactions is obtained as

$$\Delta G_{\text{high}}^{\text{solv}} = \Delta G_{\text{low}}^{\text{solv}} - \Delta G_{\text{low} \rightarrow \text{high}}^{\text{gas}} + \Delta G_{\text{low} \rightarrow \text{high}}^{\text{aq}} \quad (1.23)$$

Here, $\Delta G_{\text{low}}^{\text{solv}}$ is the solvation free energy difference at the low level of theory, typically MM, which can be computed utilizing all needed "alchemical" tricks. $\Delta G_{\text{low} \rightarrow \text{high}}^{\text{gas}}$ and $\Delta G_{\text{low} \rightarrow \text{high}}^{\text{aq}}$ are the needed corrections from the low to a high level of theory in the gas phase and in aqueous solution. Indirect approaches as shown in the cycle in Figure 1.3 are also referred to as end state corrections.

The challenge of indirect QM/MM FES is to obtain converged free energy differences between the two levels of theory, $\Delta G_{\text{low} \rightarrow \text{high}}^{\text{gas}}$ and $\Delta G_{\text{low} \rightarrow \text{high}}^{\text{aq}}$ in Figure 1.3.⁵² When attempting to compute a correction free energy $\Delta G_{\text{low} \rightarrow \text{high}}$ by FEP, one needs to compute the energy differences $\Delta U_{\text{low} \rightarrow \text{high}}$ for configurations sampled at the low level of theory. In most of the cases, the bond lengths and angles sampled at the low level of theory are "subtly wrong" at the high level of theory, leading to large fluctuations of $\Delta U_{\text{low} \rightarrow \text{high}}$.⁴⁹ In such cases, the exponential average in Equation 1.9 converges extremely slowly or not at all.⁵³ To address the convergence issue, more robust techniques such as nonequilibrium work (NEW) methods have been employed to connect the MM and QM/MM levels of theory.⁵¹

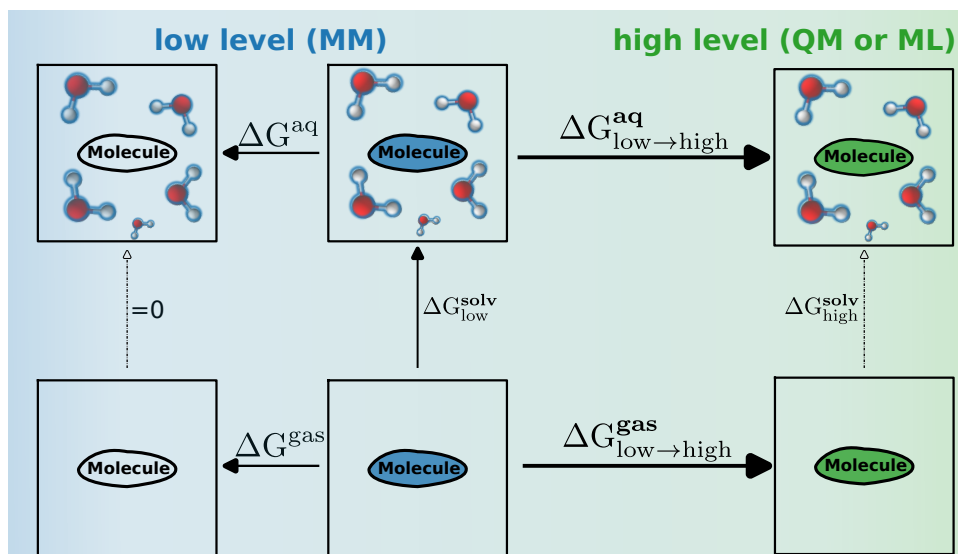


Figure 1.3 Indirect free energy pathway for correcting ASFEs from a low to a high level of theory. The left (blue) side represents the traditional approach for calculating ASFE based on a force field (MM). The MM based free energy $\Delta G_{\text{low}}^{\text{solv}}$ is obtained from the free energy difference in the gas phase and in aqueous solution ($\Delta G^{\text{gas}} - \Delta G^{\text{aq}}$; cf. Figure 1.1). In the indirect approach (right-hand-side depicted in green), free energy differences between the two levels of theories are calculated in both environments (aqueous solution and gas phase). The difference between these two legs in the thermodynamic pathway accounts for the end state correction ($-\Delta G_{\text{low} \rightarrow \text{high}}^{\text{gas}} + \Delta G_{\text{low} \rightarrow \text{high}}^{\text{aq}}$).

ML potentials in indirect cycles Neural network potentials (NNPs), also referred to as machine learning (ML) potentials, offer a tempting tradeoff between accuracy and speed compared to both molecular mechanics and QM methods.⁵⁴ One notable example of NNPs is the family of ANI potentials, first introduced in 2017.^{55,56} ANI-2x has gained significant popularity in computational chemistry for various applications like conformer generation, and chemical property prediction.^{57,58} It has already been employed in free energy simulations.^{59,60} In the context of an indirect free energy cycle, one can apply the ANI potential as the target (high level) potential to correct free energies previously calculated at a low level of theory (see Figure 1.3).^{59,61}

Despite their demonstrated utility, NNPs do have several limitations. Typically, only a limited number of elements are supported. E.g., ANI-2x only handles H, C, N, O, F, S, and Cl.⁵⁶ The ANI family of NNPs also can only be used for neutral molecules. Furthermore, while couplings between NNPs and MM force fields analogous to QM/MM approaches exist, these only provide the equivalent of mechanical embedding.⁶² This means that the interactions within the high-level region are described by the NNP, but interactions between this region and the environment are described by MM.

1.5 The `transformato` Python package

Wieder et al. introduced an open-source Python-based tool, called `transformato`, that assists in setting up large-scale FES. It uses the CC/SAI approach to compute alchemical free energy differences. The validity of the approach was confirmed through computing a series of RSFE and comparing the `transformato` results to the results of an established method.⁶³ In this section, we first outline the approach taken by `transformato` and how it differs from conventional alchemical FES. Then, we briefly describe how it was extended to support the calculation of RBEF and ASFE; full details are provided in Section 2.1 and 2.2, respectively.

1.5.1 Conceptual basis of `transformato`

The thermodynamic cycle drawn in straight lines in Figure 1.4A is the standard approach to computing RBEFs. To avoid the calculation of the absolute binding affinities (vertical arrows), one follows the alchemical horizontal arrows, transforming ligand L_1 into L_2 , both in the protein–ligand complex (top) and in solution (bottom). This is completely analogous to the calculation of RSFE vs. ASFE; (cf. Section 1.3 above), and the RBEF is obtained according to:

$$\Delta\Delta G_{L_1 \rightarrow L_2}^{\text{bind}} = \Delta G_{L_1 \rightarrow L_2}^{\text{complex}} - \Delta G_{L_1 \rightarrow L_2}^{\text{ligand}} \quad (1.24)$$

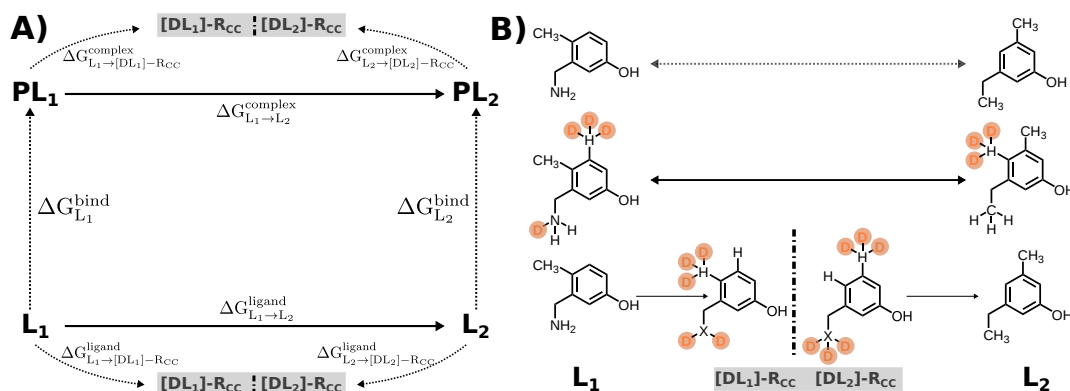


Figure 1.4 A) Alchemical pathways to calculate RBEFs. Traditional approaches calculate the free energy differences of alchemical transformations of L_1 into L_2 (horizontal, straight arrows). In the CC/SAI approach as implemented in `transformato` (dashed, curved arrows), free energy differences are calculated between the end states (L_1 and L_2) and their respective Common Core ([DL₁]-R_{CC} and [DL₂]-R_{CC}). B) Upper panel: Example of an alchemical transformation between two ligands. Middle panel: the dummy atoms needed to mutate L_1 into L_2 following the traditional approach (and single topology) of the pathway shown in A are shown explicitly and highlighted in salmon. Lower panel: The common core intermediates and their respective dummy atoms for carrying out the mutation in the CC/SAI approach.

In Figure 1.4B two hypothetical ligands, L_1 and L_2 , are drawn, the dashed arrow

indicates that L_1 has to be mutated alchemically into L_2 , and vice versa (top panel). Since the ligands do not have the same number of atoms, dummy atoms are required to carry out the transformation in practice. The middle panel illustrates a possible setup in the single topology framework; note the presence of dummy atoms colored in salmon at both endpoints. Despite their differences, the two ligands are quite similar; it is obvious to see the commonalities, i.e., their Common Core (CC).

Since we are carrying out alchemical transformations anyway, we do not have to worry whether intermediate states are physically meaningful. The CC can be reached from L_1 and L_2 , respectively, by changing all atoms not belonging to the CC into dummy atoms. This is illustrated in the bottom panel of Figure 1.4, again dummy atoms are colored in salmon. The number of dummy atoms in the CCs reached from L_1 and L_2 , i.e., $[DL_1]-R_{CC}$ and $[DL_2]-R_{CC}$, are not the same. Another difference/ambiguity is the atom labeled X, which in L_1 was a nitrogen atom, and in L_2 a carbon atom. Finally, the partial charges of the CC part in L_1 and L_2 may be different.

Alchemically transforming one CC directly into another, i.e., $[DL_1]-R_{CC} \rightarrow [DL_2]-R_{CC}$, is not possible, as the two intermediates differ in the number of atoms. It is straightforward to ensure that the partial charges of the non-dummy atoms in the two CCs are identical; furthermore, all force field parameters related to the X atoms can be set to the same values. Once this is done, $[DL_1]-R_{CC}$ and $[DL_2]-R_{CC}$, indeed, differ only in the number of dummy atoms.

In the CC/SAI approach taken by `transformato`, these two intermediate states are inserted in both alchemical legs of the thermodynamic cycle, shown in Figure 1.4A. Recently, Fleck et al. showed that if treated correctly, the contributions of dummy atoms cancel exactly from the two legs of a thermodynamic cycle.²¹ Thus, while the partition functions of $[DL_1]-R_{CC}$ and $[DL_2]-R_{CC}$ are not identical because of the different numbers of atoms, this difference is identical in the protein–ligand environment (complex) and in aqueous solution (ligand). In other words, instead of transforming L_1 into L_2 directly (straight horizontal arrows in Figure 1.4), one can insert the CCs and go from L_1 to $[DL_1]-R_{CC}$ and from L_2 to $[DL_1]-R_{CC}$, as indicated in the curved paths in Figure 1.4. When using the CC/SAI approach as implemented by `transformato`, one computes the RBE between each ligand, L_1 , L_2 , and the CC structure derived from it ($\Delta\Delta G_{L_1 \rightarrow [DL_1]-R_{CC}}^{\text{bind}}$ and $\Delta\Delta G_{L_2 \rightarrow [DL_2]-R_{CC}}^{\text{bind}}$). Each of these auxiliary free energy differences is calculated as follows:

$$\Delta\Delta G_{L_i \rightarrow [DL_i]-R_{CC}}^{\text{bind}} = \Delta G_{L_i \rightarrow [DL_i]-R_{CC}}^{\text{complex}} - \Delta G_{L_i \rightarrow [DL_i]-R_{CC}}^{\text{ligand}}, \quad i = 1, 2 \quad (1.25)$$

The RBE of interest is then obtained according to:

$$\Delta\Delta G_{L_1 \rightarrow L_2}^{\text{bind}} = \Delta\Delta G_{L_1 \rightarrow [DL_1]-R_{CC}}^{\text{bind}} - \Delta\Delta G_{L_2 \rightarrow [DL_2]-R_{CC}}^{\text{bind}} \quad (1.26)$$

In the CC/SAI framework, Equation 1.26 replaces Equation 1.24 of the conventional approach.

The correctness of the CC/SAI approach relies (1) on the cancellation of dummy atom contributions from double free energy differences,²¹ and (2) the equivalence of the CCs, [DL₁]-R_{CC} and [DL₂]-R_{CC}. In other words, it is the user's responsibility that the CCs reached from the two end states are identical, except for the number of dummy atoms. How this is done in the `transformato` framework is outlined below and described in detail in Section 2.1. The comparison of the conventional approach to computing, e.g., a RBFE and `transformato` illustrated in Figure 1.4 highlights another notable feature. When converting L₁ directly into L₂ (middle panel in Figure 1.4B), dummy atoms may be needed at both end states. By contrast, when using `transformato` the physical end states are free from dummy atoms; these are only present at the nonphysical intermediate CC states (bottom panel in Figure 1.4B).

1.5.2 Connecting a ligand to its common core

Figure 1.5 illustrates how the free energy transforming a ligand to its CC is carried out in practice. The transformation outlined in Figure 1.5A is one of the systems described in Section 2.1. All atoms that are part of the CC are highlighted in green, and dummy atoms are highlighted in salmon. The mutation procedure consists of several stages. This is illustrated in Figure 1.5B. In the initial stage (I), the charge of all non-CC atoms is linearly set to zero, typically in three or four intermediate states.

Next (stage II), the LJ interactions for all hydrogen atoms in the non-CC region are scaled to zero. Generally, one intermediate state is sufficient to achieve this. Subsequently, in stage (III), the LJ interactions of heavy atoms within the non-CC region are scaled to zero, one atom at a time. Here, the SAI approach is used, where LJ interactions are either fully active or entirely deactivated (see Section 1.3.3). In the next stage (IV), the last non-CC atom is transformed into a so-called "junction" atom. It resembles a dummy atom (e.g., it always has zero charge), but it has partial LJ interactions. Its purpose is to attach the other dummy atoms via a so-called "terminal junction", for which the separation of dummy atom and physical atom degrees of freedom can always be carried out exactly.^{21,63} The final stage (V) adjusts the CC obtained from L₁ to match the CC reached from L₂. In the specific example, this means that the charge distribution needs to be adapted, as well as the parameters of the bond from the junction atom to the CC.

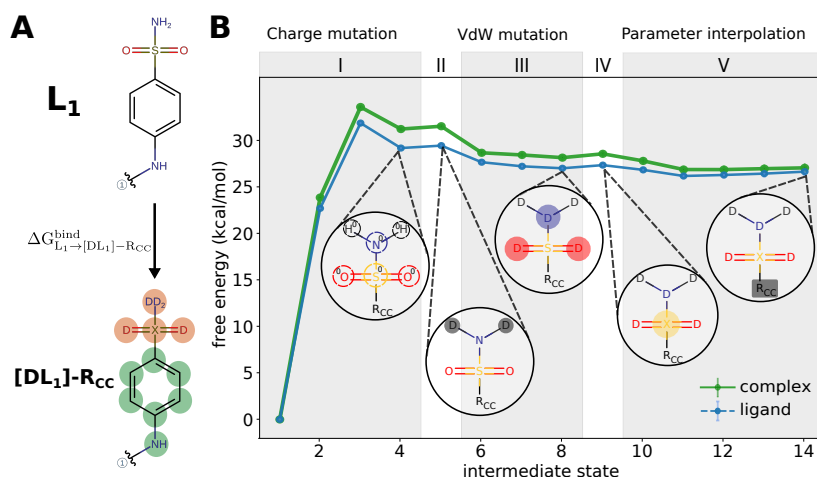


Figure 1.5 Detailed scheme for mutating a molecule into its CC: **A)** Starting structure (top) and corresponding CC structure (bottom). Dummy atoms are depicted in salmon, CC atoms in green. **B)** The intermediate states necessary to transform L_1 to its CC. The x-axis displays the consecutive numbering of the intermediate states, with state 1 corresponding to the native, physical molecule, i.e., L_1 . In this particular example, 14 such states are needed. The process involves the gradual deactivation of interactions in five stages (I–V), as indicated by alternating shaded and non-shaded regions. First, (I), the charges of all atoms not belonging to the CC are turned off, followed by the simultaneous deactivation of LJ interactions for all non-CC hydrogen atoms (II). Subsequently, the LJ interactions of each non-CC heavy atom are turned off individually, one atom at a time. Finally, (IV), the last non-CC atom is converted into a so-called "junction" atom, denoted as X. In the last stage (V), parameter interpolation is executed so that the CC of L_1 is adjusted to match the CC reached from L_2 . This final stage (V) is only needed for one of the two ligands of a transformation pair. All stages are performed in both environments (ligand and complex).

In addition to adapting `transformato` for the calculation of RBE, we also extended the workflow to make possible the computation of ASFEs. The annihilation of the solute's nonbonded interactions can be viewed as the transformation of a physical molecule to a CC containing zero atoms. In practice, this means that when `transformato` is used to set up ASFEs, no search for a suitable CC is necessary. On the other hand, special care is needed to ensure convergence when removing the LJ interactions of all heavy atoms (stage (III) in Figure 1.5B). The specific modifications of the `transformato` workflow when computing ASFEs are described in Section 2.2.

1.5.3 The role(s) of `transformato` within an FES workflow

For each molecular system subjected to an FES, initial coordinates in both environments (gas phase and solution for solvation free energies; solution and complex for binding free energies) are required. Specifically, for RBFЕ calculations, the ligand needs to be positioned within the binding pocket of the target ligand. These starting coordinates are then processed further with CHARMM-GUI.^{64,65} All capabilities of CHARMM-GUI can be used, including the flexibility to adjust ion concentrations, model missing residues, and more, facilitating the creation of a system solvated within a water box. To avoid having to process multiple very similar systems (e.g., the protein and the N ligands of interest) via an online server, we developed a small Python utility called `macha`,¹ which facilitates and automates the local recalculation of analogous setup steps of the CHARMM-GUI workflow.

The systems prepared by CHARMM-GUI (or `macha`) are the starting for `transformato` itself. The steps described next can be scripted, or interactive manipulation through a Jupyter notebook interface⁶⁶ is possible. First, in any calculation of relative free energy differences, `transformato` searches for suitable CCs. Next, `transformato` determines the necessary number of intermediate states for the five stages described above (cf. Figure 1.5B). For each of these intermediates, full self-contained input is created (in particular, all necessary force field parameters for dummy atoms are generated) and saved to a separate directory. The MD simulation required for sampling of each of the intermediate states can therefore be submitted immediately to the available computational resources without requiring further interaction by the user. Currently, these simulations can be run using either CHARMM^{67,68} or OpenMM.^{69,70} Each MD run of an intermediate state saves coordinates to disk at regular intervals. These need to be post-processed to obtain the energy differences to the respective other states. The necessary setup for this is also orchestrated by `transformato`. Finally, the matrix of energy differences (cf. Section 1.2.2) is evaluated by the `pymbar`¹⁸ module to calculate the free energy difference of interest.

¹<https://github.com/akaupang/macha>

2 Publications

In this Chapter, the three publications to which I contributed as the main author are reproduced. First, a brief introduction to and summary of each publication is given, followed by the publications themselves. The supplementary information of each original publication is provided in full in the Appendix (Appendix A, Appendix B, Appendix C). The first two publications extend the applicability of `transformato`, starting from the proof of concept work by Wieder et al.,⁶³ in which the methodology was carefully validated. The third publication combines force-field based FES carried out with `transformato` with end state corrections to an ML/MM level of theory.

Relative binding free energy calculations with transformato: A molecular dynamics engine-independent tool (DOI: 10.3389/fmolb.2022.954638)

The adaptations of the `transformato` tool to enable the computation of RBE are reported. We describe in detail how to apply the CC/SAI approach, on which `transformato` is based, to obtain relative binding affinities. The theory section of this part complements the derivations presented in Section 1.5. We present RBE results for five well-established model systems, JNK1 (c-Jun N-terminal kinases), CDK2 (Cyclin-dependent kinase 2), TYK2 (Non-receptor tyrosine-protein kinase), FXa (factor Xa), and GAL3 (galactin-3), which have been used as benchmarks previously.^{30,71,72} In total, we computed 76 relative binding affinities. We compared our results to the experimental binding data (RMSE of 1.17 kcal/mol, Pearson correlation coefficient of 0.73), as well as the results of earlier computational studies. The agreement with experiment is comparable to that of earlier work.^{30,72-74}

To further validate the `transformato` approach, we computed RBEs for selected pairs using different CCs, demonstrating that the results do not depend on the choice of the intermediate state. Currently, `transformato` supports OpenMM,^{69,70} CHARMM,^{67,68} and CHARMM/OpenMM as backends to carry out the underlying MD simulations. Several RBEs were calculated using OpenMM as well as CHARMM/OpenMM and found to be in excellent agreement. Most results reported were obtained using Hydrogen Mass Repartitioning (HMR),^{75,76} having ascertained that the 4 fs time step has no or little effects on the computed RBEs. The combination of `transformato` with HMR results in a performant tool that provides an alternative to costly commercial products.

Calculations of Absolute Solvation Free Energies with Transformato—Application to the FreeSolv Database Using the CGenFF Force Field (DOI: 10.1021/acs.jctc.3c00691)

Here, we describe the second functionality enhancement of `transformato` compared to the initial implementation,⁶³ enabling the calculations of ASFEs. Section 1.5 only provided a brief outline; here, the full details of the SAI-based protocol to calculate ASFEs using `transformato` are presented. The approach is first validated for 21 compounds, for which we compare results obtained with `transformato` to those obtained with a well-established method, the PERT module of CHARMM.⁶⁸ This small dataset was also used for additional tests, in particular, to determine an optimal treatment of LJ interactions, as well as applying adequate LJ long-range corrections (LRC). Next, we calculated ASFEs for 621 molecules of the FreeSolv database.⁷⁷ The 21 compounds for which we do not report ASFEs fall into two categories. For 10 molecules, the automated parameterization with CHARMM General Force Field (CGenFF)^{78,79} failed; for the remaining molecules, it became quickly clear that the derived parameters were incorrect. Results were compared to the experimental solvation free energies (RMSE of 1.76 kcal/mol), as well as to ASFEs computed with the GAFF force field.⁸⁰ The strengths and weaknesses of the CGenFF force field for various functional groups were analyzed.

Insights and Challenges in Correcting Force Field Based Solvation Free Energies Using ANI-2x (DOI: 10.26434/chemrxiv-2023-8jgjq)

Not all results obtained in the ASFE calculations with `transformato` using the CGenFF force field were in good agreement with the experimental data. The performance of GAFF was found to be comparable and suffer from similar weaknesses. To improve the predictive performance of ASFE calculations, we explore an indirect correction to a higher level of theory (also known as end state correction); cf. Section 1.4. Specifically, the solutes are reweighted to an ML potential, ANI-2x.⁵⁶ Interactions between the MM and the ML region are described classically; this corresponds to mechanical embedding in QM/MM simulations.⁶² We calculated the necessary free energy difference between the force field-based MM description and the ML description by nonequilibrium switching simulations. To streamline this process, we implemented an automated workflow for rapidly calculating the end state correction for large numbers of molecules, which can be easily combined with `transformato`.

We report results for two sets of data; 156 ASFEs based on our earlier results obtained with `transformato` and CGenFF, and 589 ASFEs obtained with the OpenFF 2.0 force-field. Overall, the improvements were only marginal, suggesting that a more advanced coupling between the MM and the ML part of the system is needed. Furthermore, we confirmed that in 90% of the cases investigated, unidirectional and bidirectional nonequilibrium protocols gave comparable results.

Section 2.1:

Relative binding free energy calculations with transformato: A molecular dynamics engine-independent tool

Front. Mol. Biosci. 9:954638 (2022)

Authors: Johannes Karwounopoulos, Marcus Wieder and Stefan Boresch

Contributions: Conceptualization: JK, MW, and SB; methodology: JK, MW, and SB; software: JK and MW; investigation: JK, MW, and SB; writing: JK, MW, and SB; supervision: MW and SB



OPEN ACCESS

EDITED BY

Germano Heinzelmann,
Federal University of Santa Catarina,
Brazil

REVIEWED BY

Son Tung Ngo,
Ton Duc Thang University, Vietnam
Ulf Ryde,
Lund University, Sweden

*CORRESPONDENCE

Marcus Wieder,
marcus.wieder@univie.ac.at
Stefan Boresch,
stefan@mdy.univie.ac.at

SPECIALTY SECTION

This article was submitted to Biological
Modeling and Simulation,
a section of the journal
Frontiers in Molecular Biosciences

RECEIVED 27 May 2022

ACCEPTED 18 July 2022

PUBLISHED 06 September 2022

CITATION

Karwounopoulos J, Wieder M and
Boresch S (2022), Relative binding free
energy calculations with transformato:
A molecular dynamics engine-
independent tool.
Front. Mol. Biosci. 9:954638.
doi: 10.3389/fmolb.2022.954638

COPYRIGHT

© 2022 Karwounopoulos, Wieder and
Boresch. This is an open-access article
distributed under the terms of the
[Creative Commons Attribution License](#)
(CC BY). The use, distribution or
reproduction in other forums is
permitted, provided the original
author(s) and the copyright owner(s) are
credited and that the original
publication in this journal is cited, in
accordance with accepted academic
practice. No use, distribution or
reproduction is permitted which does
not comply with these terms.

Relative binding free energy calculations with transformato: A molecular dynamics engine-independent tool

Johannes Karwounopoulos^{1,2}, Marcus Wieder^{3*} and
Stefan Boresch^{1*}

¹Faculty of Chemistry, Institute of Computational Biological Chemistry, University of Vienna, Vienna, Austria, ²Vienna Doctoral School of Chemistry (DoSCHEM), University of Vienna, Vienna, Austria,

³Department of Pharmaceutical Sciences, Faculty of Life Sciences, University of Vienna, Vienna, Austria

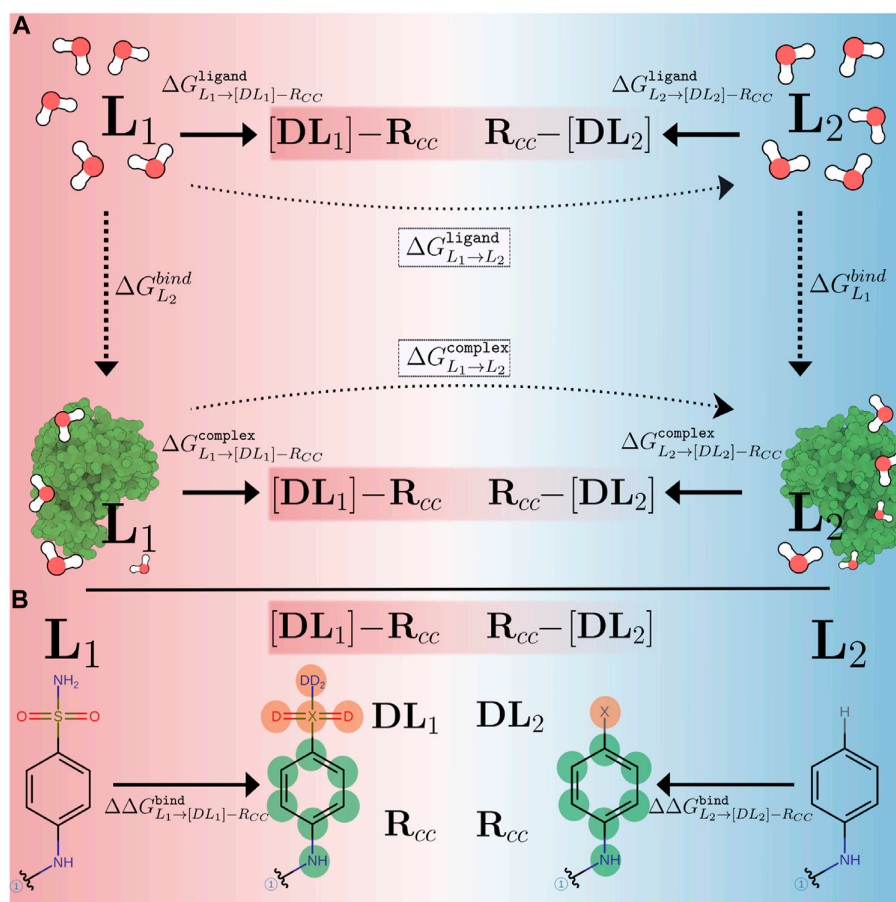
We present the software package transformato for the setup of large-scale relative binding free energy calculations. Transformato is written in Python as an open source project (<https://github.com/wiederm/transformato>); in contrast to comparable tools, it is not closely tied to a particular molecular dynamics engine to carry out the underlying simulations. Instead of alchemically transforming a ligand L_1 directly into another L_2 , the two ligands are mutated to a common core. Thus, while dummy atoms are required at intermediate states, in particular at the common core state, none are present at the physical endstates. To validate the method, we calculated 76 relative binding free energy differences $\Delta\Delta G_{L_1 \rightarrow L_2}^{bind}$ for five protein–ligand systems. The overall root mean squared error to experimental binding free energies is 1.17 kcal/mol with a Pearson correlation coefficient of 0.73. For selected cases, we checked that the relative binding free energy differences between pairs of ligands do not depend on the choice of the intermediate common core structure. Additionally, we report results with and without hydrogen mass reweighting. The code currently supports OpenMM, CHARMM, and CHARMM/OpenMM directly. Since the program logic to choose and construct alchemical transformation paths is separated from the generation of input and topology/parameter files, extending transformato to support additional molecular dynamics engines is straightforward.

KEYWORDS

free energy, molecular dynamics simulation, binding affinity, automated setup, open source, python (programming language)

1 Introduction

The accurate prediction of relative protein–ligand binding affinities is one of the major tasks in computer-aided drug design projects, especially during lead optimization. A group of methods often referred to as alchemical free energy simulations has become a versatile tool in this area, e.g., (Deflorian et al., 2020; Majellaro et al., 2020; Mortier et al., 2020; Schindler et al., 2020; Zhang C.-H. et al., 2021). While relative binding free energy (RBE) simulations have been successful in reproducing and predicting experimental

**FIGURE 1**

(A) Comparison of alchemical paths used in traditional setups (dotted, horizontal arrows) and in the CC/SAI approach (thick, horizontal arrows) implemented in transformato to compute relative binding free energy differences. Free energies are calculated relative to non-physical intermediate states, the common core $[DL_i]-R_{CC}$ ($i = 1, 2$), connecting the two ligands (L_1, L_2). Here $[DL_i]$ represents the atoms of each ligand not in the CC region and R_{CC} indicates the CC region itself. (B) The mutation path for calculating $\Delta \Delta G_{L_1 \rightarrow L_2}^{\text{bind}}$ between a pair of ligands taken from the CDK2 dataset. The atoms that differ between the two ligands and, thus, are not present in the CC, are highlighted in color; these are transformed into dummy atoms (reddish circles). The common core is indicated by the green circles.

results, their application in drug design projects is still far from routine. They require significant computing resources and are comparatively slow; setting up the simulations and analysis procedures is difficult and tedious, even for experts. To make their utilization easier, several front ends to biomolecular simulation packages have been developed to help set up RBEF simulations (Seeliger and de Groot, 2010; Gapsys et al., 2015; Loeffler et al., 2015; Wang et al., 2015; Zhang H. et al., 2021; Petrov, 2021). Here, we present a related tool, transformato, which, in contrast to most other tools, is not dependent on a particular simulation program. Transformato is a Python package that automates the setup and calculation of relative solvation and binding free energy calculations using the common core/serial-atom-insertion (CC/SAI) approach (Wieder et al.,

2022). The CC/SAI approach avoids the need for special-purpose code (mixing of energy terms, soft-core potentials, etc.), making it possible to carry out RBEF calculations with standard molecular dynamics (MD) engines. Specifically, transformato is not restricted to a specific MD program; the code currently supports CHARMM and OpenMM.

1.1 Introduction to transformato—the common core/serial-atom-insertion approach

Figure 1 illustrates the traditional and the CC/SAI approach, implemented in transformato, to compute the RBEF difference

between two ligands L_1 and L_2 . In both cases, one avoids the direct calculation of the binding free energies $\Delta G_{L_1}^{bind}$ and $\Delta G_{L_2}^{bind}$ (vertical dashed arrows in Figure 1A) and considers instead the alchemical transformation of L_1 into L_2 , which is carried out for the free ligand in water ($\Delta G_{L_1 \rightarrow L_2}^{ligand}$) and for the ligand in complex with the receptor ($\Delta G_{L_1 \rightarrow L_2}^{complex}$). Since the free energy is a state function, the quantity of interest, the RBE difference $\Delta \Delta G_{L_1 \rightarrow L_2}^{bind} = \Delta G_{L_2}^{bind} - \Delta G_{L_1}^{bind}$, can also be calculated as $\Delta \Delta G_{L_1 \rightarrow L_2}^{bind} = \Delta G_{L_1 \rightarrow L_2}^{complex} - \Delta G_{L_1 \rightarrow L_2}^{ligand}$ (Tembre and McCammon, 1984).

In the traditional approach, indicated by the dotted arrows in Figure 1A, this is done in a single transformation. The ligand L_1 is “morphed” into L_2 by gradually scaling the force field parameters associated with all atoms that are different in the two ligands along a non-physical coordinate—the so-called coupling parameter λ . For each environment (the ligand in aqueous solution, protein–ligand complex in aqueous solution), several simulations, typically 10–20, are performed at different values of λ . Such transformations can be set up using either the single topology or dual topology paradigm (Pearlman, 1994). One important practical detail is that the number of atoms must not change. Since in most cases the two ligands do not consist of exactly the same number of atoms, so-called dummy atoms need to be introduced along the alchemical paths (single topology). In dual topology, all groups are present simultaneously and interactions with the remainder of the system are turned on/off as needed. Dummy atoms (single topology), as well as atoms in non-interacting groups (dual topology), are particles that stay connected to the physical molecule by bonded terms, but do not take part in any Lennard-Jones (LJ) or Coulomb interactions (Mey et al., 2020; Fleck et al., 2021). One can show that the presence of dummy atoms has no influence on double free energy differences, such as the RBE differences considered here, but some care concerning their treatment is required (Fleck et al., 2021). The correct handling of dummy atoms or non-interacting groups in an automated manner is one of the challenges in large-scale FES.

In the common core (CC) approach, rather than alchemically transforming L_1 into L_2 directly, one defines a suitable common substructure present in both ligands, which we refer to as the CC. It is not necessary that the CC corresponds to a physical molecule (Wieder et al., 2022). The atoms of a ligand that are not part of the CC, are mutated to dummy atoms (hereafter, referred to as *non-CC atoms*). Starting from each of the physical endstates $L = \{L_1, L_2\}$, we compute the free energy difference between the ligand and the common core $[DL]-R_{CC}$, as indicated in Figure 1A. Here $[DL]$ indicates the non-CC atoms transformed into dummy atoms at the CC state, and R_{CC} denotes the interacting atoms belonging to the CC. This is done for the free ligand solvated in water $\Delta G_{L \rightarrow [DL]-R_{CC}}^{ligand}$, as well as for the protein–ligand complex $\Delta G_{L \rightarrow [DL]-R_{CC}}^{complex}$. Thus, for each of the ligands, we can compute a RBE difference with respect to a CC:

$$\Delta \Delta G_{L \rightarrow [DL]-R_{CC}}^{bind} = \Delta G_{L \rightarrow [DL]-R_{CC}}^{complex} - \Delta G_{L \rightarrow [DL]-R_{CC}}^{ligand} \quad (1)$$

When mutating the non-CC atoms of two ligands to dummy atoms, the CCs reached from L_1 and L_2 , respectively, are not necessarily identical. Most importantly, if the physical endstates L_1 and L_2 consist of different numbers of atoms, then the corresponding CCs will contain different numbers of dummy atoms. However, if these dummy atoms are treated correctly, they do *not* influence double free energy differences (Fleck et al., 2021); in other words, $\Delta \Delta G_{L \rightarrow [DL]-R_{CC}}^{bind}$ of Eq. 1 is not affected by the number of dummy atoms present at the CC endpoint. Provided that the remaining differences between the CCs, if any, are accounted for (see below and Section 2.4 for further details), the RBE difference $\Delta \Delta G_{L_1 \rightarrow L_2}^{bind}$ between the two ligands in the CC framework is obtained as:

$$\Delta \Delta G_{L_1 \rightarrow L_2}^{bind} = \Delta \Delta G_{L_1 \rightarrow [DL_1]-R_{CC}}^{bind} - \Delta \Delta G_{L_2 \rightarrow [DL_2]-R_{CC}}^{bind} \quad (2)$$

The approach just outlined can of course be extended to more than two ligands. If a suitable CC is chosen, then all pairwise RBE differences between N ligands can be obtained from just N calculations of $\Delta \Delta G_{L_i \rightarrow [DL_i]-R_{CC}}^{bind}$ (cf. Eq. 1).

In Figure 1B we show a specific alchemical transformation studied in this work to illustrate these general considerations. The two ligands (L_1 , L_2) are shown on the left and right-hand side, respectively. The black (non-colored) parts of the structures represent the CCs. The regions that are different between the two ligands are drawn in color, with their atoms labeled in boldface. For L_1 , there are six non-CC atoms that need to be transformed into dummy atoms, while for L_2 , only a single hydrogen atom needs to be mutated. The electrostatic and LJ interactions of the non-CC atoms with the environment (solvent and/or protein) and the atoms belonging to the CC are turned off using the serial atom insertion (SAI) approach (Boresch and Bruckner, 2011; Wieder et al., 2022). The detailed methodology and sequence of steps used are described in full detail in *Methods* (Section 2.4). Although the different number of dummy atoms has no influence on $\Delta \Delta G_{L_1 \rightarrow L_2}^{bind}$ as computed according to Eq. 2 (cf. above), some other force field parameters of the 2 CCs are different. In order to make the CCs identical and to close the thermodynamic cycle, additional modifications need to be applied to one of the ligands, specifically its CC, during a final stage. In the example shown in Figure 1B, the partial charges of the atoms in the phenyl ring, to which the dummy atom(s) are connected, are slightly different since the charge distribution in the physical ligands differs. A more detailed description of the modifications needed to ensure that the CCs are identical is given in Section 2.4.

1.2 Goals of this work

We recently used transformato to compute relative solvation free energy differences and demonstrated that one can obtain

high accuracy and precision with the CC/SAI approach (Wieder et al., 2022). In this work we report first results of RBFE calculations obtained with transformato as the setup tool. Specifically, we carried out 76 pairwise mutations from well established datasets (Homeyer and Gohlke, 2013; Wang et al., 2015; Gapsys et al., 2020) and compare our results both to the experimental reference data, as well as to the earlier computational results obtained by other groups. We either directly compare $\Delta\Delta G^{bind}$ values for pairs of ligands, or, by using one of the experimental binding affinities, express our results as absolute binding free energies, as was done in some of the reference studies (Wang et al., 2015; Hu et al., 2016; Song et al., 2019). In addition to the overall performance of transformato, we also investigated a number of additional aspects. First, for selected ligand pairs we computed RBFE differences using different CCs, thus proving the self-consistency of the CC/SAI approach. Second, while we used OpenMM (Eastman et al., 2017) as the underlying MD program in all of our calculations, we repeated a subset of alchemical transformations with CHARMM as the computational backend (Brooks et al., 2009). Third, we achieve computational efficiency by the consequent use of hydrogen mass reweighting (HMR) (Hopkins et al., 2015) in the underlying MD simulations. The use of HMR for RBFE simulations was recently studied and discussed by Zhang H. et al. (2021). To validate that HMR can be safely used in FES, we report results obtained with and without HMR for the same subset of alchemical transformations used in the OpenMM to CHARMM comparison. Finally, while we use the CHARMM family of force fields (see *Methods*), in one case we explored the effect of using different charge models.

2 Methods

2.1 Choice of datasets

To validate the CC/SAI approach as implemented in transformato for RBFE calculations, we selected five benchmark applications for which experimental binding affinities are known and that have been studied extensively in previous work. Three of these were taken from Wang et al. (2015), i.e., JNK1 (Supplementary Figure S1), CDK2 (Supplementary Figure S2), and TYK2 (Supplementary Figure S3). In addition, we investigated the FXa system (Supplementary Figure S4), first studied by Homeyer and Gohlke (2013) and reevaluated by Hu et al. (2016). Finally, we chose a set of inhibitors of galactin-3 (GAL3, Supplementary Figure S5) studied by Gapsys et al. (2020) and earlier by Manzoni and Ryde (2018). For these five protein–ligand datasets, we compare our results to the experimentally determined ΔG^{bind} values, as well as to calculated values reported in the respective literature. For the JNK1, FXa and GAL3 datasets we calculated $\Delta\Delta G_{L_1 \rightarrow L_2}^{bind}$ for the ligand pairs used in the original studies (Wang et al., 2015;

Hu et al., 2016; Gapsys et al., 2020). For the CDK2 and TYK2 dataset, we employed a different approach and mutated each ligand to a CC resembling the respective smallest ligand. Thus, for JNK1, FXa, GAL3 multiple CCs were required to compute the specific RBFE differences, whereas in the second case (CDK2, TYK2) only a single CC was needed. All comparisons to the study by Gapsys et al. (2020) are based on their results obtained with the CHARMM Generalized force field (CGenFF), the force field used in this work (see below); we indicate this by the abbreviation pmx/CGenFF. An overview of the systems studied and the previous computational results we compare to is shown in Table 1.

2.2 Dataset preparation

For CDK2, TYK2, and JNK1 structural information for the protein and the ligands was taken from the supporting information of Wang et al. (2015). Similarly, protein structure files for the GAL3 dataset were obtained from Gapsys et al. (2020). In both cases we used Maestro (Release 2021-3, Schrödinger, LLC, New York, NY, 2021) to prepare starting coordinates for the protein–ligand complexes.

The protein–ligand structures for the FXa system were generated as described by Homeyer and Gohlke (2013) by starting from the high resolution crystal structure of the Factor Xa-L51a complex (PDB code: 2RA0). The ligand present in this PDB entry (L51a) served as the template to model structures for the other ligands of the series (L51b–L51k). We used the protonation state of the ligands suggested by Hu et al. (2016). For each modelled FXa protein–ligand structure we performed a short minimization of the ligand in the binding site with the built-in minimizer of Maestro.

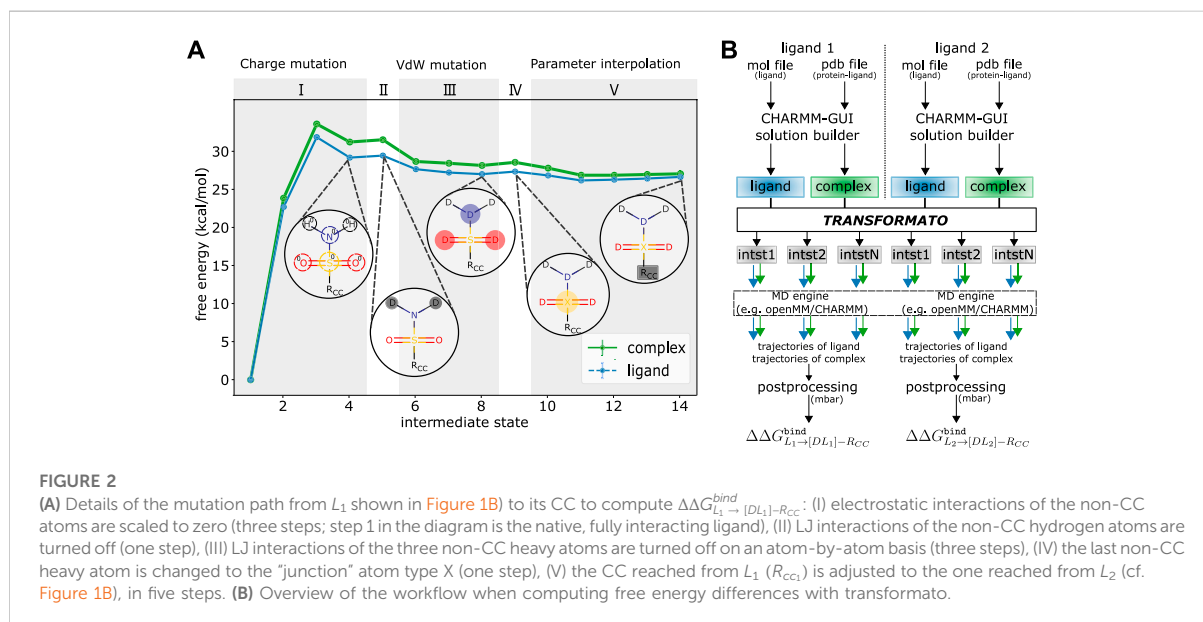
No attempts were made to optimize the protonation state of the proteins. All Asp, Glu, Lys and Arg residues were assumed to be in their charged state, and all histidines were set to neutral with the proton on the δ -nitrogen.

2.3 CHARMM-GUI preparation

The PDB files of the ligand and of the protein–ligand complexes were uploaded to the Solution Builder functionality of CHARMM-GUI (Jo et al., 2008; Lee et al., 2016) to create input files for the ligand in water (“ligand”) and for the ligand bound to the protein (“complex”). This is the first step of the workflow depicted in Figure 2B. All systems were made electrically neutral by adding a suitable number of potassium and chloride ions (JNK1, GAL3, CDK2, and TYK2) or calcium and chloride ions for FXa, see Hu et al. (2016). Ligand parameters were generated with CGenFF (v2.5) (Vanommeslaeghe et al., 2010; Vanommeslaeghe et al., 2012; Vanommeslaeghe and

TABLE 1 Overview of systems and previous computational studies.

System	PDB ID	No. of	Program	Force field (+ charge assignment)		References
				ligand	protein	
JNK1	2GMX	21	FEP+	OPLS2.1	OPLS2.1	Wang et al. (2015)
			AMBER-TI	RESP charges/GAFF 1.8	ff14SB	Song et al. (2019)
			pmx/CGenFF	CGenFF v3.0.1 (v4.1)	CHARMM36m	Gapsys et al. (2020)
FXa	2RA0	11	AMBER-TI	AM1-BCC/GAFF	ff12SB	Hu et al. (2016)
GAL3	5E89	8	pmx/CGenFF	CGenFF v3.0.1 (v4.1)	CHARMM36m	Gapsys et al. (2020)
CDK2	1H1Q	14	FEP+	OPLS2.1	OPLS2.1	Wang et al. (2015)
			AMBER-TI	RESP charges/GAFF 1.8	ff14SB	He et al. (2020)
			pmx/CGenFF	CGenFF v3.0.1 (v4.1)	CHARMM36m	Gapsys et al. (2020)
TYK2	4GIH	15	FEP+	OPLS2.1	OPLS2.1	Wang et al. (2015)
			AMBER-TI	RESP charges/GAFF 1.8	ff14SB	Song et al. (2019)
			pmx/CGenFF	CGenFF v3.0.1 (v4.1)	CHARMM36m	Gapsys et al. (2020)



MacKerell, 2012; Yu et al., 2012; Gutiérrez et al., 2016). The CHARMM36m force field (Huang et al., 2016) was used for the proteins, and TIP3P was employed as the water model (Jorgensen et al., 1983). Furthermore, during setup with CHARMM-GUI we prepared the systems for HMR. Following Hopkins et al. (2015), hydrogen masses were multiplied by a factor of 3 while the masses of the heavy atoms they are bound to were lowered accordingly to maintain a constant molecular mass. This allowed us to use a time-step of 4 fs (all bond lengths to hydrogens constrained), thus considerably lowering the computational cost of the

subsequent MD simulations. A subset of mutations was repeated without HMR; here a time-step of 1 fs was used and the ligands were fully flexible.

At the end of the preparatory steps with CHARMM-GUI just outlined, one obtains two folders for each protein–ligand system; one for the ligand, one for the protein–complex. Each contains all input and parameter files to run MD simulations of the respective system. Each set of initial coordinates was equilibrated for 125 ps under NVT conditions with weak position restraints applied to the heavy atoms of the protein [force constant of 400 kJ/mol/nm² on the protein backbone atoms and 40 kJ/mol/nm² on atoms in

the protein side-chain; these are the CHARMM-GUI recommended defaults (Lee et al., 2016)]. These inputs and coordinate files for the native systems serve as the basis from which transformato constructs the mutations paths between pairs of ligands and input files for the simulation of intermediate states (see Figure 2B).

2.4 Practical aspects of transformato

For the mutation of L_1 to L_2 , transformato first identifies the maximum common substructure, which forms the basis for the CCs $[DL_i]-R_{CC}$ of Figure 1A. For each ligand transformato constructs an alchemical path along which the atoms not belonging to the maximum common substructure are mutated to dummy atoms using the SAI approach. At this point, however, the CCs reached from L_1 and L_2 , respectively, are not necessarily identical (cf. the Introduction). Thus, for one of the ligands transformato generates the required, additional steps for the transformation $[DL_1]-R_{CC_1} \rightarrow [DL_1]-R_{CC_2}$. For each intermediate step along the alchemical path $L_i \rightarrow [DL_i]-R_{CC}$ ($i = 1, 2$), transformato creates all necessary files for running the MD simulations; thus, each of these states can be sampled independently. During the MD simulations coordinates are saved to disk; from these trajectories the energies at the respective other states, needed by the (multi-state) Bennett acceptance ratio method (MBAR) (Shirts and Chodera, 2008), are extracted in a post-processing step (see below).

We illustrate the above by describing in detail the steps required to transform L_1 shown in Figure 1B to its CC. All steps are illustrated in Figure 2A. The exact number and sequence of intermediate steps needed for this particular transformation is depicted; in general, it depends on the details of the mutation (see below). During the first stage (I) the electrostatic interaction of the non-CC atoms are scaled linearly to zero. In all transformations considered in this work, three intermediate states were used. Based on our experience, depending on the polarity and number of atoms, up to five intermediates may be necessary. To ensure that the overall charge of the system remains unchanged as the charges of the non-CC atoms are turned off, a compensating partial charge is added to the real atom of the CC to which the non-CC atoms are connected. While this stage corresponds to a linear dependence of the partial charges on a continuous coupling parameter λ as in traditional approaches, transformato, nevertheless, generates self-contained input files for each intermediate state.

Next, the LJ interactions of the non-CC atoms are turned off using the SAI approach (Boresch and Bruckner, 2011). First (stage II), the LJ interactions of all non-CC hydrogen atoms are turned off. In a single step. For this stage, a single step is always sufficient, regardless of the number of hydrogen atoms. Then (stage III), the LJ interactions of the non-CC heavy atoms are turned off atom-by-atom. In this work, cf. Figure 2, in each step

the LJ interaction of only a single heavy atom was turned off. We strongly advise against turning off the interactions of more than two heavy atoms simultaneously to ensure sufficient overlap between neighboring states. The treatment of the “last” non-CC heavy atom (stage IV), i.e., the atom directly connected to an atom in the CC region, is special; cf. Wieder et al. (2022). Rather than mutating it to a dummy atom (no LJ interactions), this “junction” atom X retains some LJ-interactions (note, though, that its partial charge is zero). By means of this junction atom, all dummy atoms are attached to the CC via a “terminal junction.” Fleck et al. (2021) demonstrated that this guarantees that any contributions from dummy atoms cancel from the double free energy differences of interest. Transformato ensures that in any transformation from L_1 to L_2 , the junction atoms X of the CCs have identical parameters. Since state (IV) is only a change in LJ interactions, one intermediate step is always sufficient.

Finally (V), the parameters of the CC of L_1 are modified so that they become identical to the parameters of the CC reached from L_2 . In the specific example (see Figure 1B), two types of parameter changes are required: 1) since the charge distributions in the phenyl rings of the two ligands are not the same in the real molecules, these need to be made identical at the CC endpoint. 2) The parameters of the bond between the junction atom X and the physical CC-atom it is connected to have to be made identical. At the end of stage (IV), in L_1 the C-X bond is described by the parameters of the C-S bond of the native ligand, whereas in L_2 , the parameters are those of a C-H bond. During stage V, the partial charges and bonded parameters involving X of the CC reached from L_1 are simultaneously modified linearly in five steps (see Figure 2A). The number of intermediate states is rather system dependent; the most critical factor affecting convergence is the change of the equilibrium bond-length from the last physical atom to the junction atom X; the change in bond-length per step should be ≤ 0.125 Å. During stage (V) there is a strict one to one correspondence between interacting atoms; all alchemical changes are carried out in the single topology paradigm by parameter mixing. As during all earlier stages, transformato writes a complete set of input/parameter files for each of these intermediate states.

More extensive differences between the CCs of L_1 and L_2 are permitted. The main requirement for the CCs, as used by transformato, is that there needs to be an unambiguous correspondence between each of the atoms. If, e.g., the common substructure search is based on element identity, then two atoms are considered to correspond to each other if they have the same element, even if they have different hybridization states or atom types. In this case, additional parameter changes need to be applied during stage (V) that are essential to ensure the validity of Eq. 2. For a given ligand pair, stages (I)–(IV) need to be carried out for each of the ligands, whereas stage (V) is only required for one of them. In our specific example (Figure 1B), the CC reached from L_2 is considered the endstate that must be also reached starting from L_1 to close the thermodynamic cycle.

2.5 Details of the MD simulations

As shown in Figure 2B and described in the previous section, transformato generates all necessary input files to perform MD simulations for each intermediate state, both for the ligand in solution, as well as for the full protein–ligand complex. All simulations were performed using the OpenMM software package (version 7.5) (Eastman et al., 2017). A subset of the mutations was also carried out using CHARMM/OpenMM (version c47a1) (Brooks et al., 2009). For each intermediate state a Langevin dynamics simulation of 5 ns length was carried out at 303.15 K; the friction coefficient was set to 1/ps. All simulations, except the short equilibration runs with position restraints described earlier, were carried out under constant pressure conditions. The pressure was controlled using a Monte Carlo barostat (Chow and Ferguson, 1995; Åqvist et al., 2004). Waters were kept rigid throughout the simulation using the SETTLE (Miyamoto and Kollman, 1992) (OpenMM) or SHAKE algorithm (Ryckaert et al., 1977) (CHARMM/OpenMM). Coulomb interactions were calculated using the particle-mesh Ewald (PME) method (Essmann et al., 1995). LJ interactions were switched smoothly to zero between 10 Å and 12 Å using the CHARMM force-switching function (Steinbach and Brooks, 1994). Production runs for each of the intermediate states were started from the respective restart file generated during the equilibration (see Section 2.3). Prior to each production run, the coordinates were optimized using the L-BFGS algorithm in OpenMM or the steepest descent and adopted basis Newton-Raphson minimizer in CHARMM/OpenMM. Simulations of each state were repeated three times with different random initial velocities.

2.6 Calculation of relative binding free energy differences ($\Delta\Delta G_{L_1 \rightarrow L_2}^{bind}$)

During each of the MD simulations described in the previous section, coordinates were written to disk every 250 steps. Using HMR and a time-step of 4 fs, each trajectory, therefore, contained 5,000 frames. The first 25% of each trajectory were considered as equilibration and discarded; the remaining coordinate sets were used to recompute the energies at all other intermediate states. All scripts for this post-processing are generated by transformato, which then invokes the MBAR functionality of pymbar (Shirts and Chodera, 2008) to compute the free energy differences $\Delta G_{L_i \rightarrow [DL_i]-R_{CC}}^{ligand}$ and $\Delta G_{L_i \rightarrow [DL_i]-R_{CC}}^{complex}$ ($i = 1, 2$). For each intermediate state λ and each configuration sample x , the reduced potential $u(x, \lambda)$ was computed to form the $N \times K$ matrix of inputs, where N is the number of snapshots used and K is the number of alchemical states λ_k for a given transformation. Finally, for both ligands L_1 and L_2 we obtained $\Delta\Delta G_{L_i \rightarrow [DL_i]-R_{CC}}^{bind}$, from which we computed the RBFE difference $\Delta\Delta G_{L_1 \rightarrow L_2}^{bind}$ according to Eq. 2. Since each set of simulations was repeated three times, using different independent initial velocities (cf. above), we obtained three statistically independent free energy

differences. We used these to estimate the statistical error; when the directly computed free energy differences were combined/processed further (see below), Gaussian error propagation was used.

2.7 Expressing the results as absolute binding free energies (ΔG^{bind})

To validate a tool such as transformato, one needs to compare both to other computational methods, as well as to experimental data. Transformato leads directly to RBFE differences $\Delta\Delta G_{L_1 \rightarrow L_2}^{bind}$. These can be compared across different computational methods easily only for identical pairs of ligands. As described in Section 2.1, for three model applications (JNK1, FXa and GAL3) we computed exactly the same RBFE differences for which results were reported by others.

Comparison of the results obtained with different methods and, in particular, comparison to experimental data, is much easier using absolute binding free energy differences ΔG^{bind} . For this reason, in the past several authors have expressed results of RBFE calculations in terms of ΔG^{bind} (Wang et al., 2015; Song et al., 2019; He et al., 2020). There are several options for post-processing $\Delta\Delta G_{L_1 \rightarrow L_2}^{bind}$, such as the “cycle closure” (Wang et al., 2015) or “centered RMSE” (Gathiaka et al., 2016) approaches, which attempt to improve the overall results and make the corresponding RMSE and MAE lower (Song et al., 2019). In this work we used the simplest possible approach: for each system we chose one ligand L_1 and considered its experimental $\Delta G_{L_1}^{bind}$ as the reference value. With the help of this reference ligand/value, we computed the “absolute” binding free energies ΔG_i^{bind} ($i = 2, \dots, n$, with n being the total number of ligands considered for this system) for all other ligands according to $\Delta G_{L_i}^{bind} = \Delta\Delta G_{L_i \rightarrow L_1}^{bind} - \Delta G_{L_1}^{bind}$. If there is no direct connection between L_1 and some L_j , the free energy can be calculated relative to another ligand L_j for which $\Delta G_{L_j}^{bind}$ could be derived directly, according to $\Delta G_{L_i}^{bind} = \Delta\Delta G_{L_i \rightarrow L_j}^{bind} - \Delta G_{L_j}^{bind}$. In such cases, we always picked the shortest path available. If there were several equivalent, shortest paths, we calculated the average value along all of them. Following these steps, we re-expressed our results as ΔG^{bind} for all ligands of the five model applications.

Two computational studies (Hu et al., 2016; Gapsys et al., 2020) to which we compare to did not report ΔG^{bind} values. In these cases, we utilized their tabulated RBFE differences to obtain absolute binding free energy differences as just outlined. For the JNK1, FXa and GAL3 systems we computed RBFE differences for exactly the ligand pairs described in the literature; therefore, we could re-compute any missing ΔG^{bind} values exactly as we did for our own results. The absolute binding free energy differences computed in this manner, together with the path used to obtain them, are listed in Supplementary Tables S1–S3. For CDK2 and TYK2, on the other hand, we used a single CC, so our results cannot be directly compared to those reported by Gapsys et al. (2020). In this case,

TABLE 2 Comparison of a) relative and b) absolute binding free energy differences calculated with transformato, pmx/CGenFF, FEP+ and AMBER-TI with experiment. The root mean squared error (RMSE), the mean absolute error (MAE), both in kcal/mol, as well as the Pearson's correlation coefficient R , and the Spearman's rank correlation coefficient ρ are listed. For the $\Delta\Delta G^{bind}$ results obtained with transformato in a) we also report bootstrapped error estimates for RMSE, MAE, R and ρ .

a) $\Delta\Delta G^{bind}$

Transformato					pmx/CGenFF ^a		FEP + ^b		AMBER-TI	
system	RMSE	MAE	R	ρ	RMSE	MAE	RMSE	MAE	RMSE	MAE
overall	1.18 [0.98; 1.38]	0.87 [0.72; 1.02]	0.57 [0.36; 0.71]	0.48 [0.29; 0.64]						
JNK1	0.91 [0.64; 1.17]	0.68 [0.51; 0.86]	0.34 [-0.02; 0.66]	0.32 [-0.02; 0.63]	0.95	0.68	1.02	0.78	1.45 ^c	1.15 ^c
FXa	1.23 [0.77; 1.69]	1.01 [0.70; 1.41]	0.83 [0.40; 0.95]	0.71 [0.22; 0.99]					1.11 ^d	0.72 ^d
GAL3	0.58 [0.39; 0.73]	0.50 [0.32; 0.68]	0.76 [0.08; 0.94]	0.57 [-0.17; 0.96]	0.61	0.54				
CDK2	1.12 [0.69; 1.46]	0.80 [0.46; 1.17]	0.63 [0.28; 0.88]	0.59 [0.08; 0.92]	1.13	0.84	1.16	0.95	1.16 ^c	0.94 ^c
TYK2	1.74 [1.26; 2.15]	1.37 [0.95; 1.83]	0.42 [-0.16; 0.72]	0.21 [-0.02; 0.63]	1.61	1.33	0.95	0.74	1.29 ^c	1.07 ^c

b) ΔG^{bind}

system	Transformato			pmx/CGenFF ^a			FEP + ^b			AMBER-TI		
	RMSE	MAE	R	RMSE	MAE	R	RMSE	MAE	R	RMSE	MAE	R
overall	1.17	0.85	0.73									
JNK1	0.81	0.57	0.60	0.81	0.57	0.66	1.14	1.06	0.85	1.45 ^c	1.07 ^c	0.22 ^c
FXa	1.04	0.92	0.76							0.96 ^d	0.66 ^d	0.83 ^d
GAL3	0.90	0.69	0.59	0.50	0.43	0.90						
CDK2	1.12	0.80	0.61	1.14	0.89	0.41	0.95	0.82	0.52	0.84 ^c	0.72 ^c	0.74 ^c
TYK2	1.74	1.37	0.42	1.87	1.61	0.53	0.58	0.46	0.88	1.27 ^c	1.07 ^c	0.49 ^c

^a Gapsys et al. (2020), ^b Wang et al. (2015), ^c Song et al. (2019), ^d Hu et al. (2016), ^e He et al. (2020).

we processed the RBE differences reported by Gapsys et al. (2020) using exactly the same procedure outlined above for our own results: we searched for the shortest path connecting the reference ligand (and its experimental ΔG^{bind}) to all other ligands (see Supplementary Tables S4, S5 for full details). For the results of transformato we do not need to search for shortest pathways since all RBE differences are calculated with respect to the same ligand (we were using a single CC) which was chosen as the reference ligand L_1 (see Supplementary Tables S6, S7).

3 Results and discussion

3.1 Overview of results for all systems

In total, we computed 76 RBE differences $\Delta\Delta G^{bind}_{L_i \rightarrow L_j}$, which we also re-expressed as $\Delta G^{bind}_{L_i}$. All values, together

with the corresponding free energy differences from the earlier studies we compare to (Wang et al., 2015; Hu et al., 2016; Song et al., 2019; Gapsys et al., 2020; He et al., 2020), can be found machine-readable form in the files summary_ddG.csv and summary_dG.csv provided in the SI. The statistical analyses of these data can be found in the Jupyter notebook RBE_transformato_workbook.ipynb available as SI. A summary of the results is shown in Table 2.

As listed in Table 2, the RMSE ($\Delta\Delta G^{bind}$) of all 76 $\Delta\Delta G^{bind}_{L_i \rightarrow L_j}$ results compared to the experimental binding affinities reported by Wang et al. (2015) is 1.18 kcal/mol and the MAE ($\Delta\Delta G^{bind}$) 0.87 kcal/mol. The Pearson's correlation coefficient R using all data is 0.57, while Spearman's ρ is 0.48. Seven out of the total $\Delta\Delta G^{bind}_{L_i \rightarrow L_j}$ had an unsigned error (UE) greater than 2 kcal/mol (9.2%). This number is in line with expectation, assuming an underlying Gaussian distribution of the error with RMSE ($\Delta\Delta G^{bind}$) = 1.18. For five mutations, the UE was between

1.5 and 2.0 kcal/mol (6.6%), and in 15.8% of the cases the UE was between 1.0 and 1.5 kcal/mol. Thus, 68.4% of the computed RBFE differences were within ± 1 kcal/mol of their corresponding experimental value. Plots depicting the correlation between computed and experimental $\Delta\Delta G_{L_1 \rightarrow L_2}^{bind}$ results for each of the five model systems are shown in [Supplementary Figure S6](#). Whenever $\Delta\Delta G_{L_1 \rightarrow L_2}^{bind}$ results for the same alchemical transformations were reported in previous computational work, the respective data are also included [Supplementary Figure S6](#).

For the absolute binding free energies ΔG^{bind} (see [Table 2](#)), the performance indicators are very similar to those obtained for $\Delta\Delta G^{bind}$. The RMSE and MAE are almost identical, with the correlation between the results computed by transformato and experiment slightly better ($R = 0.73$). [Supplementary Figure S7](#); [Supplementary Table S8](#) in the SI provide an alternative viewpoint on the quality of the overall results. [Supplementary Figure S7](#) shows a histogram of the deviations between computed and experimental results $\Delta G_{i,exp}^{bind} - \Delta G_i^{bind}$. In [Supplementary Table S8](#) the percentage of cases falling within selected ranges of maximal absolute deviation is listed and compared to the expected values drawn from a Gaussian distribution centered about zero ($\mu = 0$, i.e., assuming the absence of systematic deviations) with a standard deviation $\sigma = \text{RMSE}(\Delta G^{bind}) = 1.17$ kcal/mol. The plot of this Gaussian is superposed on the histogram (orange, dashed line) in [Supplementary Figure S7](#). Overall, the bell curve fits the data reasonably well. However, one can clearly discern that all results in poor agreement with experiment (results with an UE > 2.5 kcal/mol) are too negative ([Supplementary Figure S7](#)) and that their occurrence is slightly higher than expected from a strict Gaussian distribution of errors ([Supplementary Table S8](#)). This observation is reflected by the second Gaussian function (solid orange line) plotted in [Supplementary Figure S7](#). This curve was obtained from a fit of the histogram data to a Gaussian; while σ remains mostly unchanged, the value of μ obtained from the fit is ≈ -0.4 kcal/mol. This suggests a small systematic deviation of the computed free energy differences.

As one can see in [Supplementary Figure S7](#) (black, hatched histogram), a very similar trend can be discerned in the pmx/CGenFF results by [Gapsys et al. \(2020\)](#). Since in both studies the same force field was used, this may be indicative of errors resulting from the parameterization. Further, as one can see from the performance indicators (RMSE, MAE, R) in [Table 2](#) for each of the five systems, our results for TYK2 are much poorer than for the other four. Indeed, most of the results in poor agreement with experiment were obtained for TYK2 (see below). This is also the case for the pmx/CGenFF results ([Gapsys et al., 2020](#)).

3.2 Detailed ΔG^{bind} results for each system

The agreement between computed [this work, pmx/CGenFF ([Gapsys et al., 2020](#)), FEP+ ([Wang et al., 2015](#)), and AMBER-TI ([Hu et al., 2016](#); [Song et al., 2019](#); [He et al., 2020](#))] and experimentally determined binding affinities is plotted in [Figure 3](#). The graphs complement the statistical descriptors in [Table 2](#).

3.2.1 JNK1

JNK1 is the first of three systems for which we closely followed the transformation paths described in the literature ([Wang et al., 2015](#)) to compute RBFE differences $\Delta\Delta G_{L_1 \rightarrow L_2}^{bind}$. In terms of RMSE and MAE (see [Table 2](#)) the best results were obtained with transformato and pmx/CGenFF. On the other hand, the highest correlation was obtained with FEP+. Looking at [Figure 3](#), one easily discerns one transformato result which is in very poor agreement with experiment. This is ligand 18652, and the underlying $\Delta\Delta G_{L_1 \rightarrow L_2}^{bind}$ calculation causing the error is the transformation of 18631 $\rightarrow$ 18652 (cf. [Supplementary Figure S1](#)). This is a relatively complicated mutation in which several large functional groups need to be turned off to reach the CC state. Specifically, this is an alchemical transformation which, in general, we would try to avoid, but which we carried out to follow the mutation paths used by [Wang et al. \(2015\)](#). Analyzing the raw data, we noted that the phase space overlaps between several intermediate states were poor when using the standard protocol. We, therefore, repeated the full calculation with 10 instead of 5 ns of sampling per state. The longer simulation protocol improved the result dramatically; the $\Delta\Delta G_{L_1 \rightarrow L_2}^{bind}$ for the transformation of 18631 $\rightarrow$ 18652 changed from $+1.6 \pm 0.48$ to -0.11 ± 0.32 kcal/mol, lowering the deviation from experiment from -2.8 to -1.1 kcal/mol. If we include this improved result, our correlation metrics increase (e.g., for Pearson's R from 0.34 to 0.5 for $\Delta\Delta G_{L_1 \rightarrow L_2}^{bind}$ and from 0.6 to 0.74 for ΔG^{bind}). However, our focus was on the quality of the results that can be obtained with transformato using a computationally affordable protocol. Furthermore, no other alchemical transformation had a similarly poor overlap, so we did not recompute any other $\Delta\Delta G_{L_1 \rightarrow L_2}^{bind}$ values with the longer simulation protocol.

3.2.2 FXa

FXa is one of two systems not studied by [Wang et al. \(2015\)](#). We computed the $\Delta\Delta G_{L_1 \rightarrow L_2}^{bind}$ values reported by [Homeyer and Gohlke \(2013\)](#), but used the protonation states suggested by [Hu et al. \(2016\)](#); hence, we only compare to the latter results. AMBER has a slightly lower RMSE (ΔG^{bind}) of 0.96 kcal/mol and a MAE (ΔG^{bind}) of 0.66 kcal/mol, compared to our values of 1.04 and 0.92 kcal/mol. Pearson's R is also better, 0.83 (AMBER) vs. 0.76 (this work). Overall, however, the results of transformato for this system are quite satisfactory. The FXa system was also used for

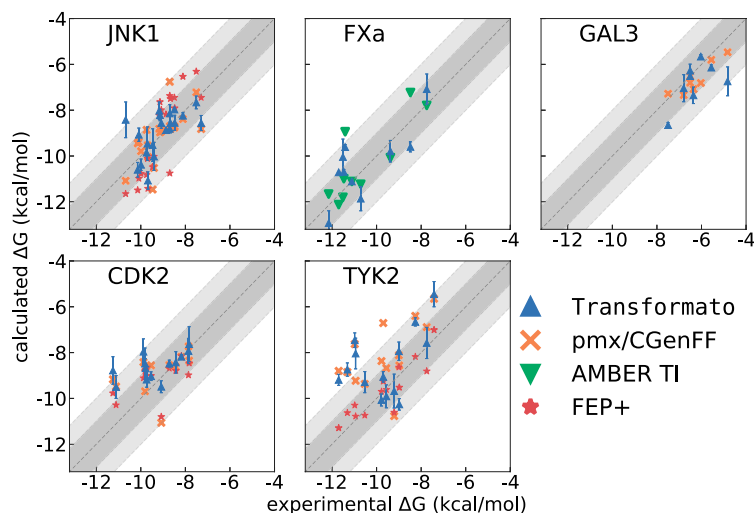


FIGURE 3

ΔG^{bind} calculated with transformato (blue triangles) compared to results obtained by pmx/CGenFF (for JNK1, GAL3, CDK2, TYK2) marked as orange crosses, by FEP+ (for JNK1, CDK2, TYK2) marked as red stars, and by AMBER (FXa) marked as green triangles. The respective RMSE and MAE values are listed in Table 2.

experiments with different charge assignments; these results are not included in Table 2 or Figure 3; see Section 3.4.1 below.

3.2.3 GAL3

The GAL3 system was first studied by Manzoni and Ryde (2018); we followed the mutation paths described there, but compare our results to those obtained with pmx/CGenFF (Gapsys et al., 2020). Our RMSE (ΔG^{bind}) = 0.90 kcal/mol and MAE (ΔG^{bind}) = 0.69 kcal/mol are quite good, but higher than those for pmx/CGenFF. Further, Pearson's R for pmx/CGenFF is noticeably higher. However, for the $\Delta\Delta G_{L_1 \rightarrow L_2}^{bind}$ results, the performance of transformato and pmx/CGenFF is more similar (see Table 2). This is also the case for Pearson's R , which for pmx/CGenFF is 0.82 (see RBFE_transformato_workbook.ipynb in the SI) and 0.76 for transformato (Table 2).

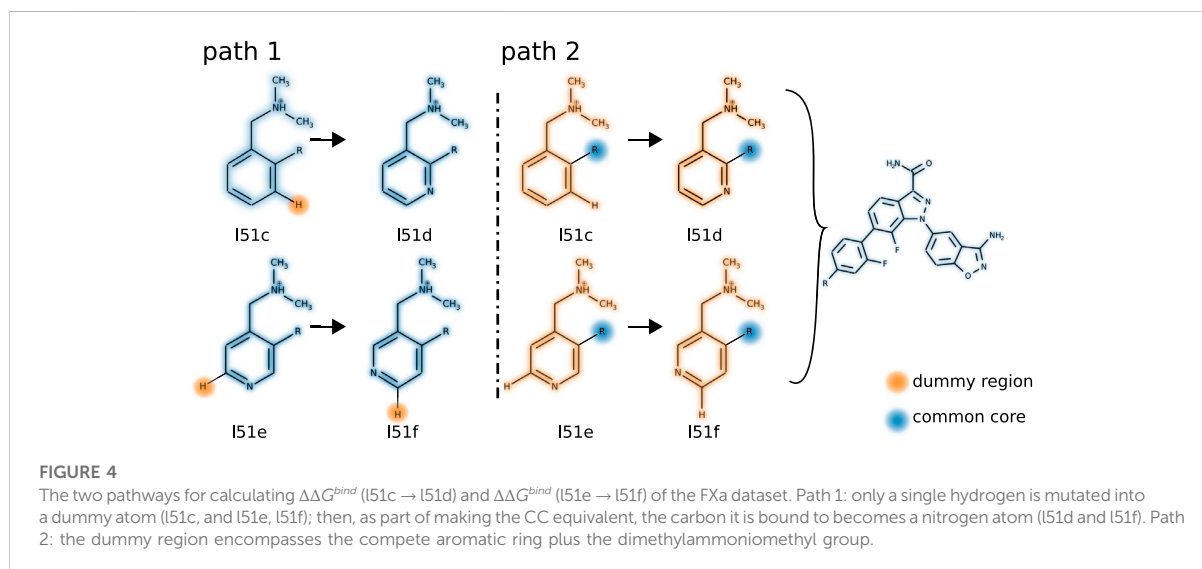
3.2.4 CDK2

As described in Methods, for CDK2 and TYK2 we employed a single CC; i.e., we followed different mutation paths than those used in the studies we compare to. For CDK2 our CC was based on the smallest ligand of the set (1h1q, see Supplementary Figure S2). Using transformato in this manner takes advantage of the CC/SAI approach; in total only 14 alchemical mutations were required for the 14 ligands studied; this includes the minor changes required for 1h1q $\rightarrow$ CC. For CDK2, the RMSE (ΔG^{bind}) and MAE (ΔG^{bind}) values were reported by all methods (Table 2). The RMSE values lie within a range of 0.3 kcal/mol across the different methods,

with the spread of the MAE (0.15 kcal/mol) being even narrower. All methods also have an acceptable Pearson's R . The similarity in performance across all methods can also be seen in Figure 3.

3.2.5 TYK2

As for CDK2, we used a single CC (ejm_31, Supplementary Figure S3), again in contrast to the previously reported approaches, in which more than one mutation path was considered for most of the ligands. As one sees in Table 2 most programs perform relatively poorly for TYK2, the single exception being FEP+ with an RMSE (ΔG^{bind}) = 0.58 and MAE (ΔG^{bind}) = 0.46 kcal/mol. Transformato, pmx/CGenFF and AMBER-TI perform significantly worse; the highest deviations were obtained with transformato and pmx/CGenFF. The RMSE (ΔG^{bind}) for transformato is 1.74 kcal/mol and 1.87 kcal/mol for pmx/CGenFF, and the MAE (ΔG^{bind}) is also high: 1.37 kcal/mol for transformato and 1.61 kcal/mol for pmx/CGenFF. AMBER-TI performs better with an RMSE (ΔG^{bind}) of 1.27 kcal/mol and an MAE (ΔG^{bind}) of 1.07 kcal/mol. This is reflected in Figure 3, where one sees that for transformato, four ΔG^{bind} values lie outside the ± 2 kcal/mol threshold; this is also the case for pmx/CGenFF. By contrast, for FEP+, only one value lies outside the ± 1 kcal/mol threshold. As summarized in Section 3.1 above, only seven $\Delta\Delta G_{L_1 \rightarrow L_2}^{bind}$ results deviated from the experimental value by more than 2 kcal/mol; four of these were obtained for TYK2. Clearly, for this system, our results are inferior to those obtained with FEP+ (Wang et al., 2015). However, since the performance of pmx/CGenFF for TYK2 is



similarly poor (Gapsys et al., 2020), the CGenFF force field used in both cases might be a contributing factor. We return to the potential role of parameterization in Section 3.4.2.

3.3 Additional validation

3.3.1 Use of alternative common cores

The results presented so far demonstrate the utility of transformato to set up large-scale free energy simulations. To validate the CC/SAI approach further, we calculated two RBFE differences for the FXa model system (I51c $\rightarrow$ I51d, I51e $\rightarrow$ I51f) with very different choices for the CCs. The details are shown in Figure 4. As one sees, the mutation consists of either transforming a phenyl ring into a pyridyl group (I51c $\rightarrow$ I51d), or changing the nitrogen position in a pyridyl group relative to the other substituents (I51e $\rightarrow$ I51f). Along path 1 (Figure 4, left), we include the aromatic ring in the CC. Thus, only a single hydrogen needs to be transformed into a dummy atom. However, the 2 CCs reached from the two initial states are quite different. Thus, the final stage (stage V, cf. Section 2.4 and Figure 2), transforming the CC reached from one initial state into the other, is involved: the atom types of C and N need to be swapped, which entails the changes of the force field parameters of all associated bonded terms.

The alternative (path 2, Figure 4, center) consists in transforming the full phenyl/pyridyl ring to dummy atoms. In this case, the dimethylammoniomethyl group present in all four ligands needs to be switched off as well, since one cannot have two disjunct CCs attached to a non-interacting dummy region. Path 2 would be transformato's default mode. However,

for the summary reported in Table 2, we used the results obtained along path 1.

The comparison of the detailed results obtained with the two transformations is shown in Table 3. The agreement of $\Delta\Delta G_{L_1 \rightarrow L_2}^{bind}$ along the two paths is quite good, although the statistical error along path 2 is significantly larger. This is to be expected, as for each transformation to the CC an aromatic ring plus the dimethylammoniomethyl moiety needed to be mutated to dummy atoms. In practical work with transformato, one would avoid mutations such as those needed for path 2; nevertheless, it is satisfying to see that acceptable results can be obtained even along such a non-optimal path.

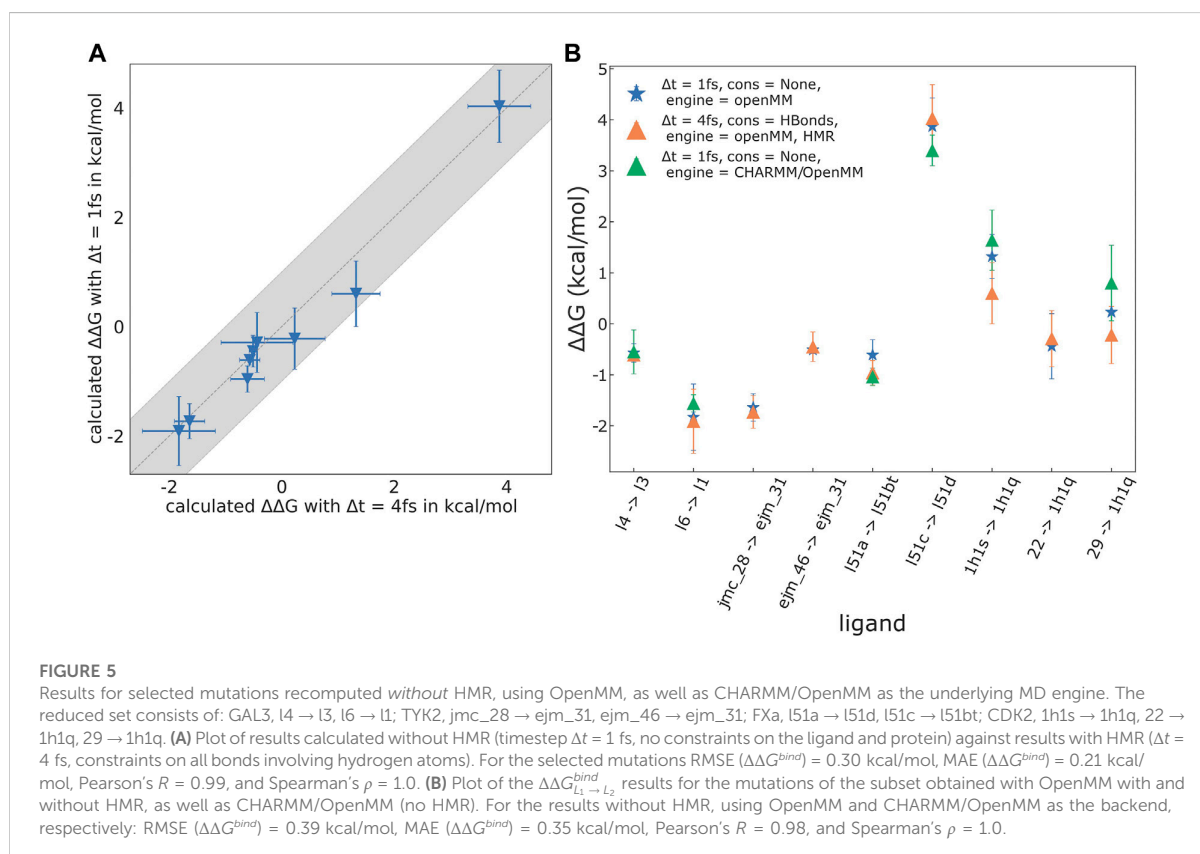
3.3.2 Re-computation of selected results with CHARMM/OpenMM and without HMR

All results reported so far were obtained with OpenMM as the underlying MD engine and employing HMR. Recently, Zhang H. et al. (2021) carried out free energy simulations with different time-steps, and with and without HMR. Using CHARMM-GUI's Free Energy Calculator and AMBER-TI as the free energy tool, they observed good agreement between results obtained with all protocols. We nevertheless wanted to validate these findings for the CC/SAI approach, with a special focus on the correct usage of constraints.

We chose nine ligand pairs from the JNK1, FXa, GAL3 and CDK2 model systems and recomputed the respective $\Delta\Delta G_{L_1 \rightarrow L_2}^{bind}$ values without HMR and with a "safe" time-step of 1 fs. In addition, we also computed these free energy differences using CHARMM/OpenMM as the MD engine. Two of the ligand pairs contain halogens, for which virtual sites ("lone-pairs") are

TABLE 3 $\Delta\Delta G_{L_1 \rightarrow L_2}^{bind}$ for the mutations I51c $\rightarrow$ I51d, I51e $\rightarrow$ I51f in the FXa model system along two different paths (see also Figure 4).

Mutation	$\Delta\Delta G_{L_1 \rightarrow L_2}^{bind}$ (path 1)	$\Delta\Delta G_{L_1 \rightarrow L_2}^{bind}$ (path 2)	$\Delta\Delta G_{L_1 \rightarrow L_2}^{bind}$ (exp)
I51c $\rightarrow$ I51d	4.03 ± 0.7	4.21 ± 1.1	3.36
I51e $\rightarrow$ I51f	-0.87 ± 0.5	-0.79 ± 1.5	-2.32



employed in the CGenFF force field (Gutiérrez et al., 2016). While this poses no difficulty for OpenMM, the use of lone-pairs (regardless of whether they are part of the mutation itself!) seems not supported by CHARMM/OpenMM, so these two $\Delta\Delta G_{L_1 \rightarrow L_2}^{bind}$ values could only be computed with OpenMM.

Two plots depicting the results and a detailed list of the mutations studied are shown in Figure 5. We find excellent agreement between the results obtained using OpenMM with and without HMR, with an RMSE/MAE of 0.32/0.23 kcal/mol, respectively. The same is the case for the results obtained with CHARMM/OpenMM compared to the OpenMM results without HMR; here, RMSE ($\Delta\Delta G^{bind}$) = 0.39 and MAE ($\Delta\Delta G_{L_1 \rightarrow L_2}^{bind}$) = 0.35 kcal/mol. Thus, HMR is an excellent tool to speed up free energy simulations, with our settings

increasing the throughput by a factor of four (time-step $\Delta t = 4$ fs rather than 1 fs; cf. Methods).

When using HMR with a 4 fs time-step, constraints on all bonds containing hydrogen atoms are required to ensure stable simulations. If the equilibrium bond length of a constrained bond is changed during an alchemical mutation, the work exerted by the constraint must be properly accounted for (Boresch and Karplus, 1998). Zhang H. et al. (2021) correctly warn about this difficulty. Presently, transformato avoids this pitfall as follows: First, in the CC/SAI approach one rarely needs to change bond lengths. Second, when such an alchemical mutation is needed, as for the bonds to the junction atom X (cf. Methods), the special atom types used for dummy atoms, as well as for X, are not recognized by OpenMM as hydrogen atoms. For the CHARMM

TABLE 4 $\Delta\Delta G_{L_1 \rightarrow L_2}^{bind}$ in kcal/mol for the alchemical transformations of I51a into the two tautomeric forms of ligand I51b/I51bt of the FXa dataset. TF denotes results obtained with transformato. Values for $\Delta\Delta G_{L_1 \rightarrow L_2}^{AMBER}$ and $\Delta\Delta G_{L_1 \rightarrow L_2}^{exp}$ were taken from Hu et al. (2016).

Mutation	$\Delta\Delta G_{L_1 \rightarrow L_2}^{TF}$	$\Delta\Delta G_{L_1 \rightarrow L_2}^{AMBER}$	$\Delta\Delta G_{L_1 \rightarrow L_2}^{exp}$
I51a $\rightarrow$ I51b (CGenFF charges)	-1.23 ± 0.3		
I51a $\rightarrow$ I51b (AMBER charges)	0.76 ± 0.3	1.44 ± 0.2	
I51a $\rightarrow$ I51bt (CGenFF charges)	-1.07 ± 0.3		-1.45
I51a $\rightarrow$ I51bt (AMBER charges)	-0.56 ± 0.1	-1.19 ± 0.1	

backend, the same can be achieved through atom selections when setting up constraints. Thus, for the currently supported use cases, transformato handles constraints and changes of bond lengths arising in alchemical mutations correctly.

3.4 Influences of force field and charge assignment

3.4.1 Charge assignment and tautomerism

One of the ligands in the FXa dataset can exist in two tautomeric forms, labelled I51b and I51bt by Hu et al. (2016). The computed $\Delta\Delta G_{L_1 \rightarrow L_2}^{bind}$ for the transformation I51a $\rightarrow$ I51b deviated by nearly 3 kcal/mol from the experimental result, whereas the computational result for I51a $\rightarrow$ I51bt agreed well with experiment (Hu et al., 2016); see also Table 4. By contrast, our calculations give very similar free energy differences for both tautomeric states, -1.23 kcal/mol and -1.07 kcal/mol, respectively, both in good agreement with the experimental value of -1.45 kcal/mol (see Table 4).

Since on the one hand tautomers and ionization states are a source of error in free energy simulations (Cournia et al., 2021) and the disagreement between the two calculational results is surprisingly large, we decided to investigate this further. It seems reasonable to focus on electrostatic interactions. Therefore, we made the partial charges of the ligand more similar to the parameterization used by Hu et al. (2016), i.e., the AMBER force field family, atomic charges prepared with the AM1-BCC approach (Jakalian et al., 2000). We replaced the charges from ParamChem/CGenFF with charges from the ACPYPE server (Silva and Vranken, 2012). When repeating the RBFE calculations for the two transformations (I51a $\rightarrow$ I51b/t) with these chimeric parameters (CGenFF force field, but AM1-BCC charges), the results were much closer to the values reported by Hu et al. (2016), with a positive sign for the mutation I51a $\rightarrow$ I51b and a negative one for the favored mutation I51a $\rightarrow$ I51bt (see Table 4).

While mixing parameters as described above is *not* recommended, the results make clear the origin of the rather different free energy predictions obtained in this particular case by transformato and AMBER (Hu et al., 2016), respectively. The rather dissimilar results for the two charge models highlight the

difficulty in relying on additive force fields to predict the preferred tautomeric state.

3.4.2 Observations concerning the influence of the force field

The data in Table 2 and the plots shown in Figure 3 demonstrate the good overall agreement between results obtained by transformato with results of previous studies in the literature. For four out of five model applications, the computed binding free energy differences are quite comparable between different programs and force fields. As already described in Sections 3.1 and 3.2, the single exception are the results for TYK2. Here, transformato performed worse than FEP+ (Wang et al., 2015), and our results showed high deviations compared to experiment. However, TYK2 seems a challenging system for computational methods. The results obtained with AMBER-TI (Song et al., 2019) are already in poorer agreement with experiment, with Pearson's $R = 0.49$ compared to $R = 0.88$ for FEP+. The RMSE (ΔG^{bind}) and MAE (ΔG^{bind}) of pmx/CGenFF are similarly high as the values obtained with transformato (see Table 2). Recently, it was shown that for TYK2 a refinement of alchemical free energy differences by a physics-based machine learning potential improved the agreement with experiment significantly (Rufa et al., 2020). The comparable performance of transformato and of the results by Gapsys et al. (2020) may indicate shortcomings in the CGenFF parameters of the ligand in the case of the TYK2 application, as this force field was used in both studies.

The ParamChem/CGenFF parameterization procedure reports two “penalty scores,” one for the partial charges assigned, and one for the bonded parameters. These scores indicate the amount of analogy that could be established during the parameter assignment. High values do not automatically mean a poor quality of the parameters, but they are a warning to the user that further inspection and optimization may be needed.¹ Looking at the penalty scores reported for the sets of ligands used in this work, we find, e.g., for CDK2, a system for which transformato's agreement with experiment is very

¹ See, e.g., the frequently asked questions at <https://cgenff.umaryland.edu/commonFiles/faq.php>

good, maximal charge penalties between 21 and 43, and maximal bonded parameter penalties between 45 and 89. For TYK2, on the other hand, the largest charge penalty is 142. Furthermore, for the ligands containing a cyclopropyl moiety (ejm46, jmc23, jmc27, jmc28, jmc30; see [Supplementary Figure S3](#)) the maximal parameter penalties range from 114 to 225. These are exactly the ligands for which we obtained the largest deviations from the experimental binding free energy differences ($UE > 2.5$ kcal/mol) and that contribute to the systematic offset of ≈ -0.4 kcal/mol for the average deviation between results obtained with transformato and experiment (cf. [Section 3.1](#) and [Supplementary Figure S7](#)). While the penalty scores for the three other systems, JNK1, FXa, and GAL3, are higher than those for CDK2, they are lower than those for TYK2.

The cyclopropyl-containing ligands of the TYK2 dataset have high parameter penalties, as just mentioned, but acceptable charge penalties, in particular for the atoms of the cyclopropyl moiety itself. This is not surprising, as cyclopropane is part of the training set, so reference partial charges for this group are available. However, the partial charges are exactly the same as those for normal alkyl groups. According to the FreeSolv database ([Mobley and Guthrie, 2014](#)), the solvation free energies of propane and cyclopropane are +2.0 and +0.75 kcal/mol, respectively. In previous work ([Fleck et al., 2021](#)), we obtained +2.33 kcal/mol for the solvation free energy of propane using the CGenFF parameters, a value in good agreement with experiment. To probe the quality of the cyclopropane parameters, we therefore computed its solvation free energy difference, following the protocol by [Fleck et al. \(2021\)](#), and obtained $\Delta G_{solv} = +2.17 \pm 0.07$ kcal/mol. Thus, the CGenFF parameters make cyclopropane more hydrophobic than it should be. We note in passing that the solvation free energies using AMBER parameters reported in the FreeSolv database are +2.50 and +2.48 kcal/mol for propane and cyclopropane, respectively, i.e., quite comparable to the CHARMM results. While all of the above is “circumstantial evidence,” the high parameter penalties for the cyclopropyl-containing ligands, combined with the rather poor agreement between the computed and experimental solvation free energy of cyclopropane itself, suggest that insufficiencies of the force field may indeed at least partially responsible for the poor performance observed for TYK2.

4 Concluding discussion

In the study introducing transformato ([Wieder et al., 2022](#)), we demonstrated that one can achieve very high accuracy with the CC/SAI approach for (relative) solvation free energy differences. Our present results demonstrate that the approach can be extended to large-scale RBFE calculations and that, overall, the agreement with related approaches ([Wang et al., 2015](#); [Song et al., 2019](#); [Gapsys et al., 2020](#); [He et al., 2020](#); [Petrov, 2021](#)) is good.

Transformato is developed as an open source project (see Code and Data Availability below), and it has reached a state in which it is possible for others to use/test it. Obviously, this is work in progress and several aspects of transformato can and should be improved. For JNK1, we found one mutation, for which phase space overlaps between some neighboring states were low. While in this particular case the best solution would be to avoid this specific mutation path, such situations should be flagged automatically. Some of our results, in particular those for the TYK2 dataset, highlight the sensitivity of the accuracy of RBFE calculations to the force field parameters. Other sources of systematic errors that can arise in free energy simulations of protein–ligand systems include, e.g., differences in binding modes of ligands not captured by the simulations, handling of trapped waters and choosing the correct ionization/tautomerization states; for additional details we refer the reader to [Mey et al. \(2020\)](#) and [Cournia et al. \(2021\)](#) and the references listed therein. One methodological aspect more specific to the CC/SAI approach is the following: Situations may arise in which the CC is significantly smaller than the physical ligand; hence, it may be bound much less tightly and move around in the binding pocket. Complications resulting from this unwanted flexibility can be prevented by adding restraints to keep the CC in a position and orientation with respect to the receptor that resembles the physical protein–ligand complex. We did not detect indications of such problems in the results reported here, but work towards implementing such restraints is currently ongoing.

The CC/SAI approach is well suited for situations in which a large portion of the ligands under consideration is identical and when the structural differences are located in one or more functional groups. Inefficiencies may arise if the CC is significantly smaller than either of the two ligands. In such a case, for both ligands, a large number of atoms needs to be mutated into dummy atoms, which not only increases the computational cost, but also lowers accuracy and precision. Such is the situation encountered for path 2 in [Section 3.3.1](#), as reflected by the noticeably higher statistical uncertainty for the results obtained along this route. The example, however, also highlights that the automatic CC assignment of transformato can be overridden, as was done along path 1. In such cases, stage (V) (cf. [Section 2.4](#)) is more involved, but transformato supports the necessary setup.

The CC/SAI approach implemented in transformato completely separates the logistics of setting up the alchemical mutation from the underlying MD simulations. One benefit of this is performance; the plain MD simulations can use the fastest available code paths. On average, using OpenMM, a full calculation of $\Delta\Delta G_{L_1 \rightarrow L_2}^{bind}$ took approximately 1 day on a single mid-range, consumer grade GPU (NVIDIA RTX3070). As the required MD simulations are completely independent, they can be distributed between several computers with linear

speedup. Further, adding support for additional MD programs should not be difficult. Maybe even more important, the full functionality of the underlying MD engine can be used. E.g., with OpenMM, we can use the CGenFF parameters that place lone-pairs on the halogens (Gutiérrez et al., 2016)². Further extending transformato to the Drude polarizable force field, given the existing support in OpenMM (Huang et al., 2018) and CHARMM-GUI (Kognole et al., 2021), should be straightforward.

Setting up alchemical transformations with transformato relies on CHARMM-GUI (Jo et al., 2008; Lee et al., 2016); in other words, in principle, RBFE differences can be computed for any system that can be handled by CHARMM-GUI, such as membrane proteins, etc. Thus, particularly for academic users, transformato provides a low-cost/low-barrier entry to large-scale RBFE calculations.

5 Code and data availability

- Python package used in this work (release v0.2): <https://github.com/wiederm/transformato>.

Data availability statement

The datasets presented in this study can be found in online repositories. The names of the repository/repositories and accession number(s) can be found in the article/Supplementary Material.

Author contributions

Conceptualization: JK, MW, and SB; methodology: JK, MW, and SB; software: JK and MW; investigation: JK, MW, and SB; writing—original draft: JK, MW, and SB; writing—review and editing: JK, MW, and SB; funding acquisition: MW and SB; resources: MW and SB; supervision: MW and SB.

References

- Åqvist, J., Wennerström, P., Nervall, M., Bjelic, S., and Brandsdal, B. O. (2004). Molecular dynamics simulations of water and biomolecules with a Monte Carlo constant pressure algorithm. *Chem. Phys. Lett.* 384, 288–294. doi:10.1016/j.cplett.2003.12.039
- Boresch, S., and Bruckner, S. (2011). Avoiding the van der Waals endpoint problem using serial atomic insertion. *J. Comput. Chem.* 32, 2449–2458. doi:10.1002/jcc.21829
- Boresch, S., and Karplus, M. (1998). The role of bonded terms in free energy simulations: 1. theoretical analysis. *J. Phys. Chem. A* 103, 103–118. doi:10.1021/jp981628n
- 2 Supporting lone-pairs required a patch to parmed (Shirts et al., 2017), which is available from <https://github.com/JohannesKarwou/ParmEd>. It should be available in the next official parmed release.

Funding

MW acknowledges support from a FWF Erwin Schroedinger Postdoctoral Fellowship J 4245-N28. SB acknowledges grant P-31024-N28 of the Austrian Science Foundation (FWF).

Acknowledgments

MW is grateful for the support of Thierry Langer, who donated significant computational resources to perform parts of the calculations/simulations. We thank Åsmund Kaupang for many helpful discussions and the careful reading of the manuscript. We also would like to thank our internship students, Márta Gódney, Sonja Berger, Alexandra Katholnig, Aleksander Szarzynski and Adrian Daniel Vasiiu.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Supplementary material

The Supplementary Material for this article can be found online at: <https://www.frontiersin.org/articles/10.3389/fmolb.2022.954638/full#supplementary-material>

Brooks, B. R., Brooks, C. L., Mackerell, A. D., Nilsson, L., Petrella, R. J., Roux, B., et al. (2009). CHARMM: The biomolecular simulation program. *J. Comput. Chem.* 30, 1545–1614. doi:10.1002/jcc.21287

Chow, K. H., and Ferguson, D. M. (1995). Isothermal-isobaric molecular dynamics simulations with Monte Carlo volume sampling. *Comput. Phys. Commun.* 91, 283–289. doi:10.1016/0010-4655(95)00059-O

Cournia, Z., Chipot, C., Roux, B., York, D. M., and Sherman, W. (2021). "Free energy methods in drug discovery—introduction," in *Free energy methods in drug discovery: Current state and future directions* (Washington, United States: American Chemical Society), 1. –38. doi:10.1021/bk-2021-1397.ch001

Deflorian, F., Perez-Benito, L., Lenselink, E. B., Congreve, M., van Vlijmen, H. W. T., Mason, J. S., et al. (2020). Accurate prediction of GPCR ligand binding affinity with free energy perturbation. *J. Chem. Inf. Model.* 60, 5563–5579. doi:10.1021/acs.jcim.0c00449

- Eastman, P., Swails, J., Chodera, J. D., McGibbon, R. T., Zhao, Y., Beauchamp, K. A., et al. (2017). OpenMM 7: Rapid development of high performance algorithms for molecular dynamics. *PLoS Comput. Biol.* 13, e1005659. doi:10.1371/JOURNAL.PCBI.1005659
- Essmann, U., Perera, L., Berkowitz, M. L., Darden, T., Lee, H., and Pedersen, L. G. (1995). A smooth particle mesh Ewald method. *J. Chem. Phys.* 103, 8577–8593. doi:10.1063/1.470117
- Fleck, M., Wieder, M., and Boresch, S. (2021). Dummy atoms in alchemical free energy calculations. *J. Chem. Theory Comput.* 17, 4403–4419. doi:10.1021/ACS.JCTC.0C01328
- Gapsys, V., Michielssens, S., Seeliger, D., and de Groot, B. L. (2015). pmx: Automated protein structure and topology generation for alchemical perturbations. *J. Comput. Chem.* 36, 348–354. doi:10.1002/jcc.23804
- Gapsys, V., Pérez-Benito, L., Aldeghi, M., Seeliger, D., van Vlijmen, H., Tresadern, G., et al. (2020). Large scale relative protein ligand binding affinities using non-equilibrium alchemy. *Chem. Sci.* 11, 1140–1152. doi:10.1039/c9sc03754c
- Gathiaka, S., Liu, S., Chiu, M., Yang, H., Stuckey, J. A., Kang, Y. N., et al. (2016). D3R grand challenge 2015: Evaluation of protein–ligand pose and affinity predictions. *J. Comput. Aided. Mol. Des.* 30, 651–668. doi:10.1007/S10822-016-9946-8
- Gutiérrez, I. S., Lin, F.-Y., Vanommeslaeghe, K., Lemkul, J. A., Armacost, K. A., Brooks, C. L., et al. (2016). Parametrization of halogen bonds in the CHARMM general force field: Improved treatment of ligand–protein interactions. *Bioorg. Med. Chem.* 24, 4812–4825. doi:10.1016/j.bmc.2016.06.034
- He, X., Liu, S., Lee, T.-S., Ji, B., Man, V. H., York, D. M., et al. (2020). Fast, accurate, and reliable protocols for routine calculations of protein–ligand binding affinities in drug design projects using AMBER GPU-TI with ff14SB/GAFF. *ACS Omega* 5, 4611–4619. doi:10.1021/acsomega.9b04233
- Homeyer, N., and Gohlke, H. (2013). FEW: A workflow tool for free energy calculations of ligand binding. *J. Comput. Chem.* 34, 965–973. doi:10.1002/jcc.23218
- Hopkins, C. W., Le Grand, S., Walker, R. C., and Roitberg, A. E. (2015). Long-time-step molecular dynamics through hydrogen mass repartitioning. *J. Chem. Theory Comput.* 11, 1864–1874. doi:10.1021/ct5010406
- Hu, Y., Sherborne, B., Lee, T. S., Case, D. A., York, D. M., and Guo, Z. (2016). The importance of protonation and tautomerization in relative binding affinity prediction: a comparison of AMBER TI and Schrödinger FEP. *J. Comput. Aided. Mol. Des.* 30, 533–539. doi:10.1007/s10822-016-9920-5
- Huang, J., Lemkul, J. A., Eastman, P. K., and MacKerell, A. D. (2018). Molecular dynamics simulations using the drude polarizable force field on GPUs with OpenMM: Implementation, validation, and benchmarks. *J. Comput. Chem.* 39, 1682–1689. doi:10.1002/jcc.25339
- Huang, J., Rauscher, S., Nawrocki, G., Ran, T., Feig, M., De Groot, B. L., et al. (2016). CHARMM36m: an improved force field for folded and intrinsically disordered proteins. *Nat. Methods* 14, 71–73. doi:10.1038/nmeth.4067
- Jakalian, A., Bush, B. L., Jack, D. B., and Bayly, C. I. (2000). Fast, efficient generation of high-quality atomic charges. AM1-BCC model: I. Method. *J. Comput. Chem.* 21, 132–146. doi:10.1002/(sici)1096-987x(20000130)21:2<132::aid-jcc5>3.0.co;2-p
- Jo, S., Kim, T., Iyer, V. G., and Im, W. (2008). CHARMM-GUI: A web-based graphical user interface for CHARMM. *J. Comput. Chem.* 29, 1859–1865. doi:10.1002/jcc.20945
- Jorgensen, W. L., Chandrasekhar, J., Madura, J. D., Impey, R. W., and Klein, M. L. (1983). Comparison of simple potential functions for simulating liquid water. *J. Chem. Phys.* 79, 926–935. doi:10.1063/1.445869
- Kognole, A. A., Lee, J., Park, S.-J., Jo, S., Chatterjee, P., Lemkul, J. A., et al. (2021). CHARMM-GUI Drude prep for molecular dynamics simulation using the classical Drude polarizable force field. *J. Comput. Chem.* 43, 359–375. doi:10.1002/jcc.26795
- Lee, J., Cheng, X., Swails, J. M., Yeom, M. S., Eastman, P. K., Lemkul, J. A., et al. (2016). CHARMM-GUI input generator for NAMD, GROMACS, AMBER, OpenMM, and CHARMM/OpenMM simulations using the CHARMM36 additive force field. *J. Chem. Theory Comput.* 12, 405–413. doi:10.1021/ACS.JCTC.5B00935
- Loeffler, H. H., Michel, J., and Woods, C. (2015). FESetup: Automating setup for alchemical free energy simulations. *J. Chem. Inf. Model.* 55, 2485–2490. doi:10.1021/acs.jcim.5b00368
- Majellaro, M., Jespers, W., Crespo, A., Núñez, M. J., Novio, S., Azuaje, J., et al. (2020). 3, 4-dihydropyrimidin-2(1H)-ones as antagonists of the human A_{2B} adenosine receptor: Optimization, structure–activity relationship studies, and enantiospecific recognition. *J. Med. Chem.* 64, 458–480. doi:10.1021/acs.jmedchem.0c01431
- Manzoni, F., and Ryde, U. (2018). Assessing the stability of free-energy perturbation calculations by performing variations in the method. *J. Comput. Aided. Mol. Des.* 32, 529–536. doi:10.1007/S10822-018-0110-5
- Mey, A. S., Allen, B. K., Macdonald, H. E. B., Chodera, J. D., Hahn, D. F., Kuhn, M., et al. (2020). Best practices for alchemical free energy calculations [article v1.0]. *Living J. Comput. Mol. Sci.* 2, 18378. doi:10.33011/livecoms.2.1.18378
- Miyamoto, S., and Kollman, P. A. (1992). Settle: An analytical version of the SHAKE and RATTLE algorithm for rigid water models. *J. Comput. Chem.* 13, 952–962. doi:10.1002/jcc.540130805
- Mobley, D. L., and Guthrie, J. P. (2014). FreeSolv: a database of experimental and calculated hydration free energies, with input files. *J. Comput. Aided. Mol. Des.* 28, 711–720. doi:10.1007/s10822-014-9747-x
- Mortier, J., Friberg, A., Badock, V., Moosmayer, D., Schroeder, J., Steigemann, P., et al. (2020). Computationally empowered workflow identifies novel covalent allosteric binders for KRAS^{G12C}. *ChemMedChem* 15, 827–832. doi:10.1002/cmdc.201900727
- Pearlman, D. A. (1994). A comparison of alternative approaches to free energy calculations. *J. Phys. Chem.* 98, 1487–1493. doi:10.1021/j100056a020
- Petrov, D. (2021). Perturbation free-energy toolkit: An automated alchemical topology builder. *J. Chem. Inf. Model.* 61, 4382–4390. doi:10.1021/ACS.JCIM.1C00428
- Rufa, D. A., Macdonald, H. E. B., Fass, J., Wieder, M., Grinaway, P. B., Roitberg, A. E., et al. (2020). Towards chemical accuracy for alchemical free energy calculations with hybrid physics-based machine learning/molecular mechanics potentials. bioRxiv doi:10.1101/2020.07.29.227959
- Ryckaert, J. P., Cicotti, G., and Berendsen, H. J. (1977). Numerical integration of the cartesian equations of motion of a system with constraints: molecular dynamics of n-alkanes. *J. Comput. Phys.* 23, 327–341. doi:10.1016/0021-9991(77)90098-5
- Schindler, C. E. M., Baumann, H., Blum, A., Böse, D., Buchstaller, H.-P., Burgdorf, L., et al. (2020). Large-scale assessment of binding free energy calculations in active drug discovery projects. *J. Chem. Inf. Model.* 60, 5457–5474. doi:10.1021/acs.jcim.0c00900
- Seeliger, D., and de Groot, B. L. (2010). Protein thermostability calculations using alchemical free energy simulations. *Biophys. J.* 98, 2309–2316. doi:10.1016/j.bpj.2010.01.051
- Shirts, M. R., and Chodera, J. D. (2008). Statistically optimal analysis of samples from multiple equilibrium states. *J. Chem. Phys.* 129, 124105. doi:10.1063/1.2978177
- Shirts, M. R., Klein, C., Swails, J. M., Yin, J., Gilson, M. K., Mobley, D. L., et al. (2017). Lessons learned from comparing molecular dynamics engines on the SAMPL5 dataset. *J. Comput. Aided. Mol. Des.* 31, 147–161. doi:10.1007/s10822-016-9977-1
- Silva, D., and Vranken, B. F. (2012). ACPYPE-AnteChamber PYthon Parser interface. *BMC Res. Notes* 5, 367. doi:10.1186/1756-0500-5-367
- Song, L. F., Lee, T.-S., Zhu, C., York, D. M., and Merz, K. M. (2019). Using AMBER18 for relative free energy calculations. *J. Chem. Inf. Model.* 59, 3128–3135. doi:10.1021/acs.jcim.9b00105
- Steinbach, P. J., and Brooks, B. R. (1994). New spherical-cutoff methods for long-range forces in macromolecular simulation. *J. Comput. Chem.* 15, 667–683. doi:10.1002/jcc.540150702
- Tembre, B. L., and McCammon, J. A. (1984). Ligand-receptor interactions. *Comput. Chem.* 8, 281–283. doi:10.1016/0097-8485(84)85020-2
- Vanommeslaeghe, K., Hatcher, E., Acharya, C., Kundu, S., Zhong, S., Shim, J., et al. (2010). CHARMM general force field: A force field for drug-like molecules compatible with the CHARMM all-atom additive biological force fields. *J. Comput. Chem.* 31, 671–690. doi:10.1002/JCC.21367
- Vanommeslaeghe, K., and MacKerell, A. D. (2012). Automation of the CHARMM general force field (CGenFF) I: Bond perception and atom typing. *J. Chem. Inf. Model.* 52, 3144–3154. doi:10.1021/CI300363C
- Vanommeslaeghe, K., Raman, E. P., and MacKerell, A. D. (2012). Automation of the CHARMM general force field (CGenFF) II: Assignment of bonded parameters and partial atomic charges. *J. Chem. Inf. Model.* 52, 3155–3168. doi:10.1021/CI3003649
- Wang, L., Wu, Y., Deng, Y., Kim, B., Pierce, L., Krilov, G., et al. (2015). Accurate and reliable prediction of relative ligand binding potency in prospective drug discovery by way of a modern free-energy calculation protocol and force field. *J. Am. Chem. Soc.* 137, 2695–2703. doi:10.1021/ja512751q
- Wieder, M., Fleck, M., Braunsfeld, B., and Boresch, S. (2022). Alchemical free energy simulations without speed limits. A generic framework to calculate free energy differences independent of the underlying molecular dynamics program. *J. Comput. Chem.* 43, 1151–1160. doi:10.1002/JCC.26877
- Yu, W., He, X., Vanommeslaeghe, K., and MacKerell, A. D. (2012). Extension of the CHARMM general force field to sulfonyl-containing compounds and its utility in biomolecular simulations. *J. Comput. Chem.* 33, 2451–2468. doi:10.1002/JCC.23067
- Zhang, C.-H., Stone, E. A., Deshmukh, M., Ippolito, J. A., Ghahremanpour, M. M., Tirado-Rives, J., et al. (2021a). Potent noncovalent inhibitors of the main protease of SARS-CoV-2 from molecular sculpting of the drug peramppanel guided by free energy perturbation calculations. *ACS Cent. Sci.* 7, 467–475. doi:10.1021/acscentsci.1c00039
- Zhang, H., Kim, S., Giese, T. J., Lee, T.-S., Lee, J., York, D. M., et al. (2021b). CHARMM-GUI free energy calculator for practical ligand binding free energy simulations with AMBER. *J. Chem. Inf. Model.* 61, 4145–4151. doi:10.1021/acs.jcim.1c00747

Section 2.2:

Calculations of Absolute Solvation Free Energies with transformers– Application to the FreeSolv Database Using the CGenFF Force Field

J. Chem. Theory Comput. 19, 5988-59998 (2023)

Authors: Johannes Karwounopoulos, Åsmund Kaupang, Marcus Wieder and Stefan Boresch

Contributions: Conceptualization: JK, MW, and SB; methodology: JK, AK, MW, and SB; software: JK and AK; investigation: JK, AK, MW and SB; writing: JK, AK, MW, and SB; supervision: MW and SB

Calculations of Absolute Solvation Free Energies with **Transformato**—Application to the FreeSolv Database Using the CGenFF Force Field

Johannes Karwounopoulos, Åsmund Kaupang, Marcus Wieder, and Stefan Boresch*

Cite This: *J. Chem. Theory Comput.* 2023, 19, 5988–5998

Read Online

ACCESS |



Metrics & More

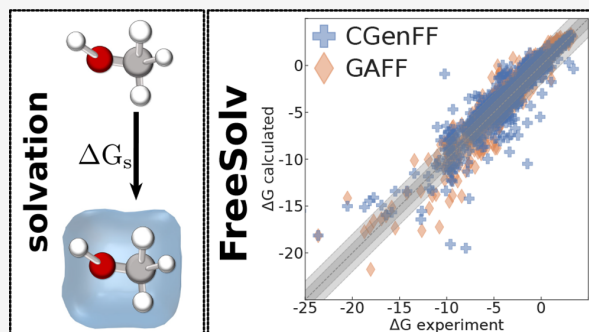


Article Recommendations



Supporting Information

ABSTRACT: We recently introduced **transformato**, an open-source Python package for the automated setup of large-scale calculations of relative solvation and binding free energy differences. Here, we extend the capabilities of **transformato** to the calculation of absolute solvation free energy differences. After careful validation against the literature results and reference calculations with the PERT module of CHARMM, we used **transformato** to compute absolute solvation free energies for most molecules in the FreeSolv database (621 out of 642). The force field parameters were obtained with the program **cgenff** (v2.5.1), which derives missing parameters from the CHARMM general force field (CGenFF v4.6). A long-range correction for the Lennard-Jones interactions was added to all computed solvation free energies. The mean absolute error compared to the experimental data is 1.12 kcal/mol. Our results allow a detailed comparison between the AMBER and CHARMM general force fields and provide a more in-depth understanding of the capabilities and limitations of the CGenFF small molecule parameters.



INTRODUCTION

Alchemical free energy simulations are quickly becoming a routine method in the toolbox of computational chemists.^{1–3} Their predictive capacity depends on (i) the accuracy of the force field used, (ii) the extent of sampling of the relevant regions of phase space, and (iii) the correct setup of the underlying molecular dynamics (MD) simulations.^{2,4} In the case of discrepancies between computed and experimentally measured free energy differences, it is often difficult, if not impossible, to determine the cause of the erroneous results. Especially when studying protein–ligand affinities, incorrect computed binding free energy differences can be caused by either of these three sources of error.

Because biological processes take place in aqueous solution, solvation by water and the accurate calculation of hydration free energies is relevant. The correct representation of a molecule's interactions with water is a prerequisite for the computational prediction of transfer free energies, i.e., partition coefficients between a polar and non-polar phase, as well as of binding free energy differences. In contrast to proteins, exhaustive sampling is often feasible for small to mid-sized organic molecules. Similarly, the challenge of accurately assigning protomeric states is somewhat simplified compared to the protein environment, given that there are typically few accessible states under physiologically relevant conditions, and their environmental and dynamical dependencies are more

readily modeled in isotropic settings. Nevertheless, attention should be paid to the choice of protonation and tautomeric state. Therefore, absolute solvation free energy (ASFE) calculations have served as a sensitive force field accuracy measure.^{5,6}

Since the beginning of the century, several studies of increasing scope explored the quality of force fields by comparing the results of ASFE calculations to experiments. Three early examples are the calculation of ASFEs of the amino acid side chain analogues.^{7–9} This work was followed by competitive ASFE prediction challenges involving an increasing number of small organic compounds.^{5,10–12} Already in 2008, Mobley and co-workers computed ASFEs for 504 neutral molecules in implicit solvent. In a subsequent study, they used the AMBER general force field (GAFF)¹³ to calculate ASFEs in explicit solvent for the same set of 504 neutral small organic molecules.¹⁴ In 2009, Shivakumar et al.¹⁵ reported the ASFEs of 239 neutral ligands, a test set they also used in later work.^{16,17} In 2011, the Automated force field

Received: June 23, 2023

Published: August 24, 2023



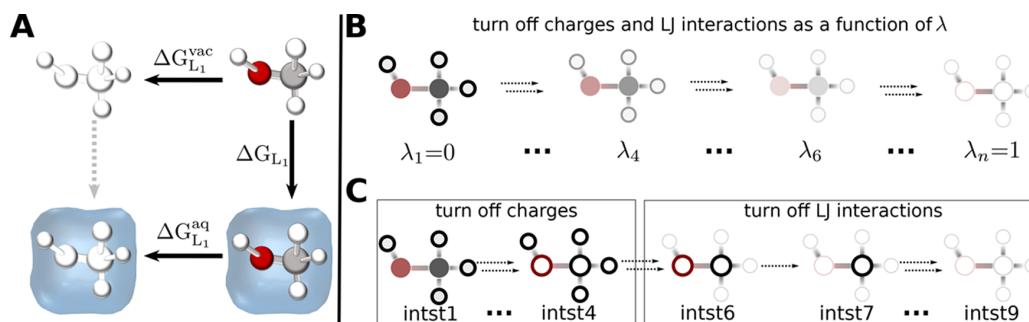


Figure 1. Transformato uses the pathway shown in A for calculating ASFEs. Instead of following the vertical line (ΔG_{L1}) directly, one follows the two horizontal lines, turning off charges and LJ interactions of the solute, once in solution (aq.) and once in vacuum (vac). In B, a standard approach for calculating ASFEs as, e.g., implemented in the PERT module of CHARMM, is sketched, where non-bonded interactions (charges and LJ interactions) are scaled to zero simultaneously as a function of a continuous coupling parameter λ , typically in $n = 11$ or $n = 21$ steps. In C, the sequence of steps taken by transformato is illustrated, using methanol as the example. During the first four intermediate states (intst1–intst4), the partial charges of the solute are scaled to zero. Afterward, the LJ interactions are scaled to zero, first for all hydrogen atoms in two steps, indicated by the two arrows connecting intst4 with intst6, then for each heavy atom, one after another. The LJ interaction of the last heavy atom is removed in two states.⁴⁸

Topology Builder (ATB) and repository, a web server providing topologies and parameters compatible with the GROMOS force field family,¹⁸ was used to estimate the ASFE in water for 190 molecules, including the amino acid side chain analogues and various organic molecules from the SAMPL challenges.¹⁹ The calculation of ASFEs served to validate and refine the ATB as described in subsequent studies.^{20,21} In 2018, Boulanger et al.²² introduced General Automated Atomic Model Parameterization (GAAMP) to calculate ASFE for 426 compounds. Similarly to the ATB tool, GAAMP creates charges and parameters based on quantum mechanical calculations, which can be used together with either the GAFF or CHARMM force field.

Calculations as just described require experimental reference data. In 2014, Mobley and Guthrie²³ established the FreeSolv database. It contains experimentally determined and computed ASFEs for 642 small organic, neutral molecules. The calculations reported in ref 23 were carried out with GROMACS 3.3.1,^{24,25} using the GAFF force field,¹³ explicit water (TIP3P²⁶), and AM1-BCC^{27,28} charges. Updates were reported in 2017.²⁹ Another source of experimental data is the Minnesota Solvation database,³⁰ which also contains solvation free energies for non-aqueous solvents. The late J. Peter Guthrie started the compilation of an even larger collection of experimental solvation free energies of small molecules.³¹ Despite the relatively small size of the molecules in the FreeSolv database compared to typical drugs, the chemical space covered by the database is quite extensive.^{23,29} This makes the database suitable for evaluating the performance of force fields in realistic scenarios involving drug-like molecules.

The FreeSolv database frequently serves as the source of experimental reference values. One recent example is work by Riquelme et al.³² who recalculated the entire FreeSolv database with polarized Hirshfeld charges, obtaining a root mean squared error (RMSE) of 2.0 kcal/mol for the whole set. Dodda et al.³³ calculated the ASFE for a subset of 426 molecules of the FreeSolv testing different charge models together with the OPLS-AA force field.³⁴ Computational approaches are not limited to free energy methods based on molecular dynamics (MD). Quantum chemical calculations combined with implicit solvent models are known to predict solvation free energies well.^{35,36} Recently, excellent agreement

between computed and experimental ASFE values was obtained using molecular density functional theory.^{37,38} Lately, the FreeSolv database is also used in the field of machine learning to develop and validate models for predicting molecular properties related to solvation and hydration.^{39,40}

Large-scale free energy simulations require automated setups. We recently developed and presented a tool, called transformato,^{41,42} for calculating relative solvation and relative binding free energies using the common-core/serial-atom-insertion approach⁴³ in a semi-automated manner. Given their importance, we extended the functionality of transformato to the computation of ASFEs. To the best of our knowledge, no systematic study for the compounds in the FreeSolv database has been carried out using the CHARMM general force field (CGenFF).^{44–48} The calculation of the ASFEs for all molecules in the FreeSolv database using CGenFF, therefore, is not only a large-scale test of the new ASFE functionality of transformato but also of wider interest concerning the strengths and weaknesses of this widely used force field.

Specifically, we proceeded as follows. First, we used transformato to calculate the ASFEs for 21 compounds and compared the results to values obtained with the PERT alchemical free energy functionality of CHARMM.⁴⁹ Additional test/validation calculations were carried out to choose the treatment of Lennard-Jones (LJ) interactions and the handling of long-range corrections (LRCs) for the LJ interactions. We then computed the ASFEs of 621 (out of the 642) compounds in the FreeSolv database; see Results and Discussion section for the details on why we were unable to compute ASFEs for some molecules. The results for essentially the complete FreeSolv database are compared to both the experimental values and the results obtained with the GAFF force field. In light of the range of chemical functionalities covered by the FreeSolv database, we briefly analyze the relative strengths and weaknesses of CGenFF and GAFF in their description of a selection of functional groups.

METHODS

Customizing Transformato for ASFE Simulations. Computing Relative Free Energy Differences with Transformato. We developed transformato for the calcu-

lation of relative free energy differences. When computing a relative solvation free energy difference between two solutes,⁴¹ the alchemical transformation from the initial to the final state in vacuum and in aqueous solution passes through an intermediate state, the so-called “common core” (CC). *transformato* determines a suitable CC by searching for the maximum common substructure of the two molecules. All non-CC atoms are mutated into non-interacting dummy atoms in a stepwise procedure. First, their charges are scaled to zero while maintaining the overall charge of the solute. Next, the LJ interactions of these atoms are removed one by one using the so-called “serial-atom-insertion” (SAI) method.⁴³ When turning off the LJ interactions of heavy atoms not present in the CC, each atom is turned off in a separate simulation step. *transformato* generates all necessary input files (topology, custom parameters for the dummy atoms) so that plain MD simulations can be carried out. No special code, such as soft-core potentials or energy/parameter mixing, is required, making *transformato*, in principle, independent of the underlying MD program. In practice, OpenMM⁵⁰ is the only fully supported backend presently; CHARMM can be used with some restrictions as well. During each MD simulation, trajectories are saved; these are post-processed to compute the energy differences to all other intermediate states. Finally, the multistate Bennett’s acceptance ratio method (MBAR) as implemented in *pymbar*⁵¹ is used to obtain free energy differences from these data.⁴⁰ For the full details, we refer the reader to refs 41 and 42.

Implementation of ASFE in *Transformato*. To calculate the ASFE ΔG_{L_i} , *transformato* uses the usual thermodynamic cycle shown in Figure 1A. Instead of calculating ΔG_{L_i} directly, the approach used by *transformato* follows the horizontal arrows, i.e., all non-bonded interactions of the solute are turned off in the gas phase ($\Delta G_{L_i}^{\text{vac}}$) and in solution ($\Delta G_{L_i}^{\text{aq}}$) (an approach referred to as annihilation⁵²). The ASFE of interest, ΔG_{L_i} , is obtained according to $\Delta G_{L_i} = \Delta G_{L_i}^{\text{vac}} - \Delta G_{L_i}^{\text{aq}}$; see Figure 1A.

In the case of absolute free energy differences, no CC is needed as *all* non-bonded interactions of the solute are turned off. Thus, *transformato* carries out SAI as described by Boresch and Bruckner⁴³ in a fully automated manner. The traditional approach to calculating such free energy differences, e.g., with the PERT module of CHARMM,⁴⁹ is sketched in Figure 1B. The gradual removal of the interactions as a function of a continuous coupling parameter (Figure 1B) should be compared with the *transformato*/SAI workflow, depicted in panel C of Figure 1: First, the electrostatic interactions of the molecule are turned off. This is achieved by scaling the partial charges of all atoms linearly to zero. By default, four intermediate states are employed, scaling the partial charges of the solute atoms by 1.0, 0.6, 0.3, and 0.0. Next, the LJ interactions of all solute hydrogen atoms are scaled to zero in two steps, scaling r_{min} and ϵ by 0.5 and 0.0. Subsequently, the LJ interactions of the heavy atoms are scaled to zero, one by one, using a single step for each atom. Each heavy atom is thus turned off in a separate intermediate state, except for the last atom, in which case, by default, two intermediate states are used to scale the LJ interactions to zero. As in the case of relative free energy differences, *transformato* generates all needed files so that for each intermediate state, one can perform a plain MD simulation.

Coordinates are saved to disk and are analyzed using the (multistate) Bennett acceptance ratio method (MBAR).⁵¹

Workflow—Simulation Details. The FreeSolv database^{23,29} was used as provided on GitHub (<https://github.com/MobleyLab/FreeSolv>). For each provided SMILES string in the file *database.txt*, a PDB file was created using the Python extension of Open Babel (Pybel).⁵³ Missing solute parameters were generated with a stand-alone version of *cgenff* (v2.5.1), which is based on version 4.6 of the CHARMM general force field (CGenFF).^{44–48} The solutes were placed in cubic simulation boxes with a side length of ≥ 26 Å, which is sufficiently large to be commensurate with the default CHARMM cut-off of 12 Å. Depending on the size of the molecule, the box length may be considerably larger to ensure adequate solvation. The initial side-length of the cubic box, as well as the number of water molecules in the box, are listed for each solute in the Supporting Information. These initial steps were automated with a small utility written in Python, called *macha* (<https://github.com/akaupang/macha>), which utilizes CHARMM scripts generated by CHARMM-GUI,^{54,55} as templates. The *macha* utility wraps these scripts/tools in a package that enables the automatic processing of multiple input molecules into solvated systems that can be simulated with CHARMM or OpenMM. Once basic inputs for simulating a system in the gas phase and in aqueous solution were generated, we invoked *transformato* to create all intermediate states from full solute–solvent interactions to a solute without any non-bonded interactions as described above. Then, the simulations were run using OpenMM (v7.7).⁵⁰ For each intermediate state, a Langevin dynamics simulation of 5 ns was carried out at 303.15 K; the friction coefficient was set to 1/ps. All simulations were carried out under constant pressure conditions, using a Monte Carlo barostat.^{56,57} Waters were kept rigid throughout the simulation employing the SETTLE⁵⁸ algorithm; the time step was set to 1 fs. Coulomb interactions were calculated using the particle-mesh Ewald (PME) method.⁵⁹ LJ interactions were switched smoothly to zero between 10 and 12 Å with the standard switching function of OpenMM (see eq 1 in the SI). Several simulations were repeated using the force switching function of CHARMM (“vswitch”),⁶⁰ which can be mimicked using OpenMM; see below for additional details concerning the treatment of LJ interactions. Before each production run, the system was minimized for 500 steps using the L-BFGS minimizer of OpenMM. Simulations of each state were repeated four times with different random initial velocities.

Post-Processing the Intermediate States. During each of the MD simulations, coordinates were saved to disk every 500 steps, resulting in 10,000 frames per trajectory. The first 25% of each trajectory was discarded as equilibration; the remaining coordinates were used to recompute the energies at all intermediate states. All post-processing was automated by *transformato*, which then invokes the MBAR functionality of *pymbar*⁵¹ to compute the free energy differences $\Delta G_{L_i}^{\text{vac}}$ and $\Delta G_{L_i}^{\text{aq}}$ (cf. Figure 1A). For each intermediate state k and for each configuration sample x , the reduced potential $u(x,k)$ was computed, resulting in a $N \times K$ matrix, where $N = 7500$ is the number of snapshots used and K is the number of intermediate states $k = 1, \dots, K$ for a given transformation. The exact number of intermediate states used for each molecule is listed in the Supporting Information. Each set of simulations was repeated four times, using different independent initial

velocities (cf. above), so we obtained four statistically independent free energy differences. We used these four values to estimate the statistical error.

Reference Calculations Using the PERT Module of CHARMM. We utilized the same topology and coordinates (PSF- and CRD files) as for the `transformato` runs. All calculations were carried out with version c47a1 of CHARMM.⁴⁹ The non-bonded interactions of the solute were turned off in 21 equidistant λ -states. Since we also removed intramolecular non-bonded interactions, a gas phase correction was needed. We set the time step to 1 fs; SHAKE⁶¹ was applied to the waters only. The PSSP soft-core potential was used to avoid LJ endpoint problems.⁴⁹ In the gas phase, neither LJ nor electrostatic interactions were truncated. In aqueous solution, LJ interactions were switched smoothly to zero between 10 and 12 Å using the potential-based CHARMM switching function.⁶² Electrostatic interactions were computed by PME⁵⁹ ($\kappa = 0.34 \text{ Å}^{-1}$; depending on the box size, a $24 \times 24 \times 24$ or $32 \times 32 \times 32$ grid was used for the fast Fourier transforms). In the gas phase, 2 ns simulations were performed (the first 10% was discarded as burn-in), while in solution, we performed 1 ns simulation (again, the first 10% of the simulation was discarded) per λ -state. All calculations were repeated five times. Free energy differences were computed by thermodynamic integration.⁶³ The $\langle dU/d\lambda \rangle_\lambda$ averages were computed on the fly and extracted from the output files, fitted to a spline function, which was integrated analytically; see Fleck et al.⁶⁴ for additional details.

Calculation of LJ Long-Range Correction (LRC). In all simulations in the aqueous phase, the LJ potential was switched off between 10 and 12 Å. Thus, any non-polar, attractive interactions beyond the cut-off radius of 12 Å were omitted. Shirts and co-workers showed that such a truncation can affect binding free energies and outlined how missing dispersion interactions can be accounted for.⁶⁵ In constant volume simulations (NVT), the isotropic LRC for the LJ interactions, which should be sufficient in the case of solute–solvent systems, is a constant term that can be calculated analytically for each molecule after the simulation has been carried out.

For NPT ensembles, the situation is a bit more complex. Shirts et al.⁶⁵ describe several strategies for correcting binding free energy simulations. To account for the fluctuation of the box size during a simulation, at the very least, multiple snapshots of a trajectory have to be analyzed. For each of the 621 compounds, for which we computed the ASFE, we proceeded as follows. In each case, we ran two additional constant pressure MD simulations, one for the fully interacting solute–solvent system, and the other for the water box containing the same number of water molecules as for the native system but with the solute removed. All simulation conditions were identical to what was described above. The simulation length was 3 ns; the first nanosecond was discarded as equilibration. Snapshots were saved every 5 ps (5,000 steps), and their potential energy computed with and without the LRC as outlined in the OpenMM documentation (http://docs.openmm.org/7.7.0/userguide/theory/02_standard_forces.html). Thus, we obtained LRCs ΔE_{LRC} for the solute–solvent system and its corresponding water box. By averaging over the 400 snapshots, we obtained averaged values for the solute–solvent system ($\langle \Delta E_{\text{LRC}}^{\text{full}} \rangle$) and for the corresponding water box ($\langle \Delta E_{\text{LRC}}^{\text{water}} \rangle$). The difference $\langle \Delta E_{\text{LRC}}^{\text{full}} \rangle - \langle \Delta E_{\text{LRC}}^{\text{water}} \rangle$ is our estimate for the omitted LJ LRC of the solute–solvent

interactions. Since these energy differences are relatively noisy, we repeated the above procedure three times, using the average over the three repetitions as the LRC and the standard deviation as its error estimate.

Since an a posteriori correction as just described omits the influence of the LJ long-range interactions on the virial, we ascertained that corrections obtained in this manner are sufficient as follows. We selected nine compounds for which we repeated the full ASFE calculation as set up by `transformato` with OpenMM's LRC option turned on. ASFES obtained in this manner were compared to the values using the approximate a posteriori LRC. In all cases, both approaches gave LRC contributions that agreed within the respective error bars; see Figure S1 in SI.

RESULTS AND DISCUSSION

Validation of Transformato. The ASFES of 21 molecules were used to validate the correctness of the results obtained with `transformato`. Eighteen of these were chosen because we had computed their ASFES with the PERT module of CHARMM in earlier work.^{41,64} Initially, we assumed that the minor differences in simulation setup and the use of earlier versions of `cgenff` in our previous work would be irrelevant, but see below. Four out of these 18 molecules are not part of the FreeSolv database, and their experimental ASFES are not known. In addition, we recomputed the ASFES with the PERT module of CHARMM for three molecules from the FreeSolv database, for which the values computed with `transformato` deviated significantly from the experimental results. All 21 compounds are depicted in Figure S2.

The initial comparison of the results, i.e., ASFES obtained with `transformato` and the values computed by Fleck et al.⁶⁴ and Wieder et al.⁴¹ is shown in Figure S3. The RMSE of 0.79 kcal/mol and the mean absolute error (MAE) of 0.46 kcal/mol were surprisingly high. In Figure S3, one sees that most values agree well but that four `transformato` results deviate by more than 1 kcal/mol from the literature values. One of the deviating ASFES was obtained for cyclohexa-2,5-dien-1-one from the Fleck et al.⁶⁴ data set (green data points and green box in Figure S3). The three other compounds are 2-methylfuran, 2-cyclopentylindole, and 7-cyclopentylindole, calculated initially by Wieder et al.⁴¹ (red data points and red box in Figure S3).

While the simulation setup in the earlier studies was very similar to what is described in **Workflow—Simulation Details**, we realized that force field parameters from different `cgenff` versions can be quite dissimilar, e.g., for cyclohexa-2,5-dien-1-one, we noted that the partial charges used by Fleck et al.,⁶⁴ derived with `cgenff` (v2.2), were quite different from those obtained in this study with `cgenff` (v2.5.1). Thus, we recomputed the ASFE with `transformato` using the older charges; this reduced the deviation to less than 0.5 kcal/mol. We found similar discrepancies in the parameters, primarily in the partial charges, for the other three problematic molecules. For these cases, we recomputed the ASFES with PERT as described above, using the `cgenff` (v2.5.1) force field parameters. We also inspected the partial charges of all other compounds; these were either identical or differed by no more than ± 0.02 e.

Furthermore, we utilized PERT to compute the ASFES for three molecules from the FreeSolv database, for which we detected significant discrepancies between the `trans-`

formato results and the experimental values. The comparison between ASFEs computed with *transformato* and recomputed free energies using PERT is shown in Figure 2.

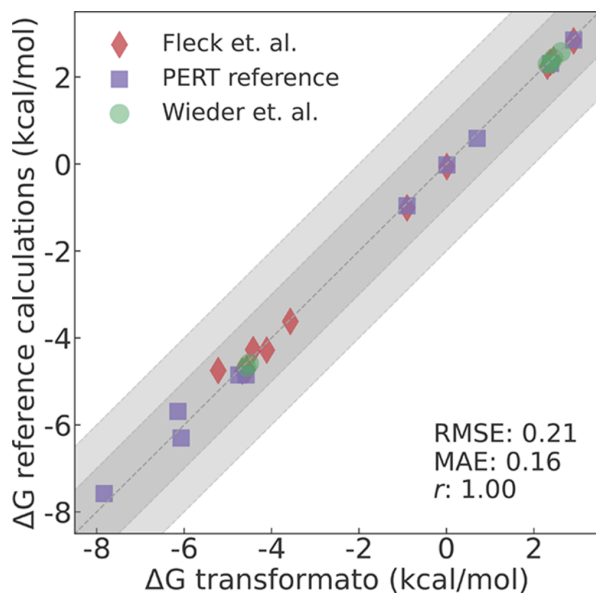


Figure 2. ASFEs computed using *transformato* either plotted against values extracted from Wieder et al.⁴¹ (green circles) and Fleck et al.⁶⁴ (red diamonds), or against values recalculated using PERT (purple squares). The overall RMSE and MAE were 0.21 and 0.16 kcal/mol, respectively.

The RMSE for this refined comparison involving 21 ASFEs was 0.21 kcal/mol, and the MAE was 0.16 kcal/mol. Given that the statistical uncertainty of the computed ASFEs for the 21 molecules is ± 0.10 kcal/mol or larger, the agreement between the PERT reference and the *transformato* results is excellent. The excellent agreement between PERT and *transformato* results indicates that the initial discrepancies were not caused by *transformato* but by differences in the versions of the employed force field (mostly in the partial charges).

Treatment of LJ Interactions. While most MD programs treat electrostatic interactions by PME, they often provide several options on how to truncate LJ interactions smoothly at the cut-off distance, e.g., in CHARMM, two switching functions are available, the original potential-based (VSWI, see eq 2 in the SI),⁶² and the newer force-based one (VFSW).⁶⁰ While VFSW should be used with the current family of CHARMM force fields,⁶⁶ the traditional alchemical free energy module PERT of CHARMM only supports VSWI when soft cores are used.⁴⁹ OpenMM natively supports a potential-based switching function, which we will call “OMMvswi.” In addition, CHARMM-GUI provides a custom force routine for VFSW, which we refer to as “OMMvswi.”⁵⁴ While the functional forms of VSWI and OMMvswi are different, the resulting shapes of the tapering functions are very similar. Therefore, we used OMMvswi in the validation calculations just described. To understand the effects of this particular switching function for LJ interactions on the ASFEs, we used *transformato* to (re)compute them with OMMvswi. In Figure 3, we show the results for which experimental solvation free energies are available. One sees that

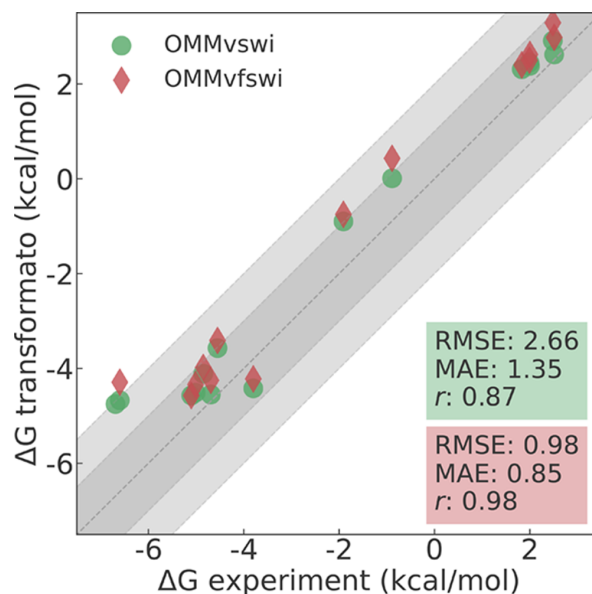


Figure 3. ASFEs calculated once with the OpenMM default switching function (OMMvswi, green dots) and once with the force-switching function (OMMvswi, red diamonds), plotted against the experimental values. Data points for which no experimental values are available are omitted in this plot but are shown in Figure S4.

the differences between the OMMvswi (green circles) and OMMvswi results (red diamonds) are small. In both cases, most ASFEs are too positive compared to experiment. With OMMvswi, an RMSE of 0.98 kcal/mol and an MAE of 0.85 kcal/mol were obtained. These values reduced slightly to an RMSE of 0.79 kcal/mol and an MAE of 0.66 kcal/mol when using the OMMvswi function (Figure 3). In the SI (Figure S4), we plot the ASFEs obtained with the two treatments of LJ interactions directly against each other. In this plot, we also included solvation free energies of the compounds for which no experimental data are available. The data can be fitted to a regression line $y = 0.98x + 0.25$ (plotted in Figure S4). While the slope is very close to unity, the OMMvswi results are systematically shifted toward more positive values by +0.25 kcal/mol. A closer examination shows that the difference between the two treatments of LJ interactions increases slightly with the size of the solute, in line with what we observed for the LRC of the LJ interactions (see below).

Thus, at least for the subset of compounds studied, OMMvswi gave results in slightly better agreement with experiment. We, therefore, decided to keep OMMvswi as the truncation method for LJ interactions in the calculations of ASFEs for the full FreeSolv data set. Since we are applying the LRC for the LJ interactions (cf. Calculation of LJ Long-Range Correction (LRC)), the choice of the switching function for LJ interactions used during the simulations should have only negligible effects on the computed ASFEs. In the future, such ambiguities could be avoided using LJPME.^{67,68}

Absolute Solvation Free Energies for the FreeSolv Data Set. Difficulties, Challenges, and Failures. With the automated procedure outlined in the Methods section, we obtained ASFEs, which we considered converged and free from major problems, for 621 molecules out of the 642 entries in the FreeSolv database. In 10 cases, the cgenff program failed to parametrize the molecules. These are typically small, simple

molecules, such as the carbontetrahalogenides CX₄, ammonia, or formaldehyde; the list of all 10 molecules can be found in the SI (Table S1). While it would be straightforward to assign force field parameters for these compounds manually, it is presently not possible in an automated manner. Since we want to describe an automated workflow, we did not manually incorporate these molecules even though they might be interesting as test molecules for later force field refinements. The one exception was methane, which also cannot be handled by the *cgenff* program, since we had force field parameters available in our reference data set (see Validation of Transformato). We also excluded 11 organophosphorodithioates, all of which have a sulfur-phosphorus motif that seems to be handled incorrectly by *cgenff*/CGenFF. An example of these compounds is shown in Figure S5, together with the Mobley IDs for the other 10 compounds. While we obtained ASFEs for these molecules, the deviation from the experimental values was large in all cases. Upon inspecting the generated force field parameters, we noted that the phosphorus–sulfur double bond was parametrized identically to the P–S single bond (cf. Figure S5), which makes little sense.

In nine out of the remaining 621 cases, we obtained standard deviations between the four individual runs of more than $k_B T \approx 0.6$ kcal/mol. Upon closer inspection, we noticed that overlap was missing between some adjacent intermediate states. Recalculating these nine ASFEs with 10 ns production time per intermediate state, instead of the default 5 ns, improved the overlap between neighboring states and in all but one cases reduced the standard deviation significantly. For a detailed list of these compounds and the 5 vs 10 ns results, see Table S2.

Influence of the LRC. Since we calculated the LRCs as a separate correction, we could analyze how it influenced the overall agreement with the experimental values. The LRC contribution to the ASFE is always negative, ranging from -0.1 kcal/mol for small molecules up to -1.2 kcal/mol for large ones (see Figure S6). We already noted for the validation set that computed solvation free energies tend to be too positive, so the LRC on average improves the agreement with experiment slightly. Indeed, applying the correction improved the overall RMSE by 0.1 kcal/mol and the MAE by 0.2 kcal/mol, compared to the ASFE without LRC. In the remainder of the manuscript, all ASFEs include the LRC. The uncorrected ASFEs for all compounds can be found in the Supporting Information.

Comparison between Experiment and Previous Computational Studies. Our results for 621 out of the 642 molecules in the FreeSolv database have an RMSE of 1.76 [1.52, 2.02] kcal/mol and an MAE of 1.12 [1.02, 1.23] kcal/mol compared to the experimental solvation free energies. The 95% confidence interval is given in brackets, and the bootstrapping procedure is described in the Supporting Information. The Pearson and Spearman correlation coefficients are 0.9 [0.88, 0.92] and 0.91 [0.89, 0.93], respectively. The results are plotted in Figure 4 (blue crosses), which also displays the computational results reported in the FreeSolv database (orange diamonds). The detailed results can be found in the Supporting Information.

Our results are in slightly poorer agreement with experiment than the computational results obtained with GAFF.²³ Using the data as presently reported in the FreeSolv database, the RMSE and MAE for GAFF for the respective 621 molecules

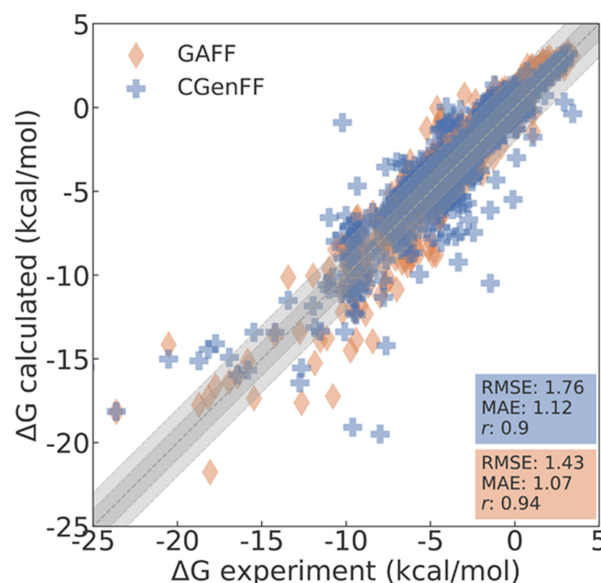


Figure 4. Comparison of the ASFEs for the 621 molecules investigated in this study compared to experimental values from the FreeSolv database.²³ Results obtained with CGenFF and transformato are marked as blue crosses; results by Mobley and Guthrie²³ using the GAFF force field are displayed as orange diamonds.

are 1.43 [1.31, 1.55] and 1.07 [1.00, 1.15] kcal/mol, respectively. The Pearson and Spearman correlation coefficients both are 0.94 [0.93, 0.95], which is also marginally better. As one can see in Figure 4, our results contain data points that deviate massively from their respective experimental ASFEs. In Table 1, we list the number of molecules

Table 1. Number (and Percentage) of Molecules with Large Deviations from the Experimental Results

deviation [kcal/mol]	>2.0	>3.0	>4.0	>6.0
CGenFF	85 (14%)	41 (7%)	20 (3%)	7 (1%)
GAFF ²³	62 (10%)	26 (4%)	13 (2%)	4 (0.5 %)

deviating by more than a certain threshold from the experimental result. We also include the corresponding numbers for the GAFF results. The numbers in Table 1 confirm that we have poor agreement for more molecules compared to Mobley and Guthrie.²³

Aside from the computational results reported in the FreeSolv database, one related large-scale study is the work by Shivakumar et al.¹⁵ who reported ASFEs for 239 neutral molecules. They compared the commercial version of the CHARMM force field (CHARMM-MSI)⁶⁹ and the standard GAFF force field¹³ with different charge assignments (AM1-BCC/RESP/CHelpG) for the respective compounds. Unfortunately, their molecules are named differently than in this work and no SMILES strings are available, which makes an automated comparison impossible. Performing a spot check, we could identify 101 molecules that are present in both the FreeSolv database and Shivakumar et al.¹⁵ For this subset, our calculations yielded an RMSE of 1.38 kcal/mol and an MAE of 1.03 kcal/mol, which compares favorably to the results obtained with the commercial CHARMM force field

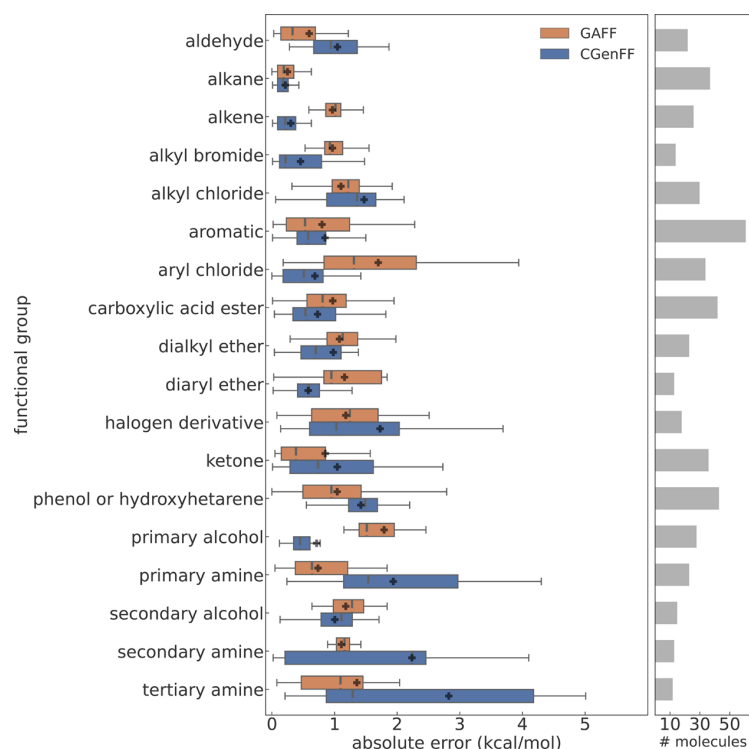


Figure 5. Box-like plot of the ASFE results for groups of molecules sharing the same chemical functionality; results are only shown when there are at least 10 molecules belonging to a group in the database. The classification into functional groups follows the original work by Mobley and Guthrie.²³ Left: the absolute error compared to the experiment, obtained with *transformato*/CGenFF (blue, this work) and GAFF²³ (orange) are shown in a box plot-like manner. For each calculated ASFE, the absolute deviation from its experimental value is calculated, grouped and assigned to the corresponding functional group. The black crosses indicate the mean absolute error (MAE) for a particular group. The whiskers indicate the molecules with minimal and maximal deviation from the experimental ASFE. The bars depict the range from the first to the third quantile of the MAE values. The median absolute error is indicated as a thin, dashed vertical line. Right: the gray bars indicate the number of molecules belonging to a group.

(CHARMM-MSI);⁶⁹ for these 101 molecules, their RMSE was 2.41 kcal/mol, and their MAE was 1.40 kcal/mol. Among the different force fields and charge models they used, a combination of AM1-BCC and the GAFF performed best, with an RMSE of 1.34 kcal/mol and an MAE of 1.05 kcal/mol. This is similar to the performance of the current *cgenff*/CGenFF combination used in this study.

The validation set for version 3.0 of the ATB^{19,20} contains a significant portion of the FreeSolv database, with 59 molecules excluded due to experimental uncertainties of 1 kcal/mol or more.²¹ This resulted in an MAE of 1.00 kcal/mol and an RMSE of 1.5 kcal/mol. When excluding compounds with experimental uncertainties greater than 1.0 kcal/mol, the performance of the CGenFF results presented here is comparable, with an MAE of 1.08 kcal/mol and an RMSE of 1.68 kcal/mol.

The largest too negative deviation was observed for cyanuric acid, where we missed the experimental ASFE (−18.06 kcal/mol) by 11.59 kcal/mol, with a calculated ASFE of −29.65 kcal/mol. In the opposite direction, we obtained the most positive wrong result for β -glucose, −15.51 kcal/mol instead of the experimental value of −25.47 kcal/mol. Both molecules were part of the SAMPL2 challenge⁵ and were among those compounds having the largest variation in results during the competition. Cyanuric acid may adopt multiple tautomeric forms. Initially, we simulated the *all-oxo* form based on the

SMILES code provided in the FreeSolv database as it is believed to be the dominant form in solution.^{70,71} However, Pérez-Manríquez et al.⁷² suggest that the aromatic enol form may partially exist in aqueous solution as well. Thus, we also calculated the ASFE for this tautomeric form, obtaining a value of −7.27 kcal/mol. For glucose, the organizers of the SAMPL2 challenge pointed out the high flexibility of the sugar and argued that its polarity, and hence, its ASFE, may change considerably upon a conformational change. Thus, in both cases, the origin of the large error may not be caused by the force field alone.

Performance for Different Functional Groups. We categorized the molecules based on their chemical functionalities using the *groups.txt* file from the FreeSolv repository available on GitHub (<https://github.com/MobleyLab/FreeSolv>). While many compounds in the data set exhibit a high degree of polyfunctionality, there are also large groups of monofunctional molecules. To avoid double-counting polyfunctional compounds, we utilized only the first category provided in the *groups.txt* file to assign the molecules to their respective categories. The assignment used is listed in the [Supporting Information](#). Proceeding in this manner results in some ambiguity, e.g., our category “amine” contains both aliphatic and aromatic compounds. We consider this acceptable for a quick survey, and the same criteria were applied to our results, as well as to those of Mobley and Guthrie.²³

Furthermore, since the number of molecules in the FreeSolv database is not too large to begin with, more detailed classification attempts quickly lead to categories consisting of only very few molecules. Figure 5 summarizes our analysis. For all functional groups, for which there are at least 10 molecules in the database, we plot the absolute error obtained with *transformato*/CGenFF (blue) against the GAFF results (orange).²³ Based on the MAE (black crosses in Figure 5), GAFF outperformed CGenFF for nine out of the 18 investigated functional groups; i.e., for these groups, the use of GAFF led to a lower MAE. Conversely, CGenFF yielded a lower MAE for the remaining nine functional groups. In Figure 5, one sees that CGenFF performs notably worse than GAFF for the primary, secondary, and tertiary amines, the latter having an MAE (black cross) of 2.6 kcal/mol; the corresponding RMSE was 3.2 kcal/mol. Overall, tertiary amines were the chemical functionality for which calculations with CGenFF presented the largest deviations from the experimental results. Note, though, that in terms of the median absolute error (gray dashed line), the performance of CGenFF is much closer to that of GAFF. Primary and secondary amines were the other two functional groups for which the MAE obtained with CGenFF was > 2 kcal/mol. For the halogen derivatives and alkyl chlorides, CGenFF also gave results in poor agreement with experiment (MAE > 1.5 kcal/mol). Examples of functional groups for which GAFF performed worse are aryl chlorides and primary alcohols. In these two cases, GAFF resulted in MAEs > 1.5 kcal/mol, significantly higher than the MAEs obtained with CGenFF. CGenFF also performs significantly better for diaryl ethers, whereas, for the dialkyl ethers, both force fields give similar results.

We conducted similar analyses with the other used metrics (RMSE and the Pearson correlation coefficients); see Figure S7. The results for the Pearson correlation coefficient contained one unexpected data point. For the diaryl ethers, the GAFF results show a negative Pearson's r , possibly indicating one or more erroneous entries in the database. It should be noted that for this group, CGenFF also has an r value of only 0.6. In line with the MAE results, the CGenFF r values for primary, secondary, and tertiary amines are low as well; for all other chemical groups $r(\text{CGenFF}) \geq 0.8$.

CONCLUSIONS

Advantages of *Transformato*. The present results, as well as those of refs 41 and 42, demonstrate the utility of *transformato* in setting up and carrying out large-scale free energy calculations. By relying on SAI, the underlying MD program does not need support for special purpose code, such as soft-core potentials. Not counting the initial validation and preliminary tests for some subsets, we were able to compute the 621 ASFEs in 4 weeks, utilizing on average 15 consumer-grade GPUs (the fastest ones being NVIDIA RTX2080 cards). Furthermore, since *transformato* provides self-contained inputs for each intermediate state, computations can be easily distributed across as many nodes as there are available. For a medium-sized molecule from the FreeSolv database with seven heavy atoms (e.g., toluene), 15 intermediate states are necessary. When running them in parallel with the local resources just described, a simulation of one intermediate state takes approximately 20 minutes. The post processing of the trajectories takes another 20 minutes; thus, on this small cluster, the calculation of an ASFE requires less than an hour of

wall time. A peculiarity of SAI is that the number of intermediate states depends on the size of the alchemical region of the solute, in particular, the number of non-hydrogen atoms. This contrasts with most standard free energy simulation protocols, in which a fixed number of intermediate λ -states, e.g., 11 or 21, is used for all alchemical transformations. At first glance, this is a downside since the larger a molecule, the longer it takes to compute its solvation free energy simply because more intermediate states are necessary. However, this intrinsic adaptiveness also has advantages. First, for smaller molecules, e.g., ethane or methanol, typical protocols with 21 λ -states are inefficient. Conversely, these 21 λ -states may be not enough for large solutes. In Figure 6, we

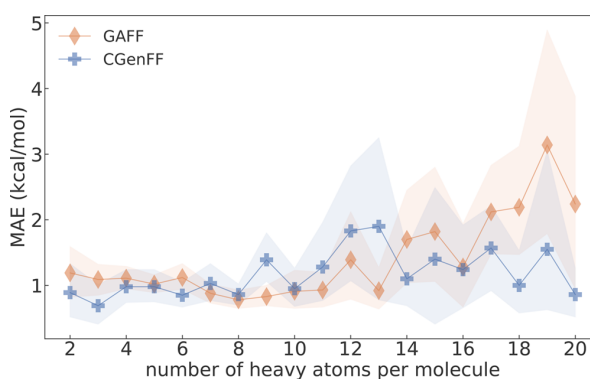


Figure 6. MAEs for the calculated ASFEs grouped by the number of heavy atoms. Values denoted GAFF were obtained using a standard lambda protocol with 21 lambda states. Values denoted CGenFF were obtained using *transformato*, with different numbers of intermediate states, depending on the size of each molecule. The shaded region surrounding the lines corresponds to the upper and lower bounds of the bootstrapped error for molecules that have the same number of heavy atoms.

plot the MAEs sorted by the number of solute heavy atoms. The present results (blue) are compared to the ASFEs from Mobley and Guthrie,²³ who used a 21 λ -states protocol. For molecules with 14 heavy atoms or more, the MAEs of the *transformato* results are lower than those of Mobley and Guthrie.²³ Thus, *transformato* automatically imposes less costly protocols for small(er) molecules and more expensive ones for large(r) compounds; the data in Figure 6 indicate that this extra effort is well spent.

cgenff/CGenFF for ASFE. Overall, the use of *cgenff*/CGenFF leads to a slightly worse agreement with experimental ASFEs compared to GAFF. As one can see from Table 1, this is caused by a relatively small number of compounds for which the computed ASFEs are off by 2 or more kcal/mol. There is no trivial way to relate a wrong free energy difference to particular force field parameters (though there are attempts such as “Time Machine” (<https://github.com/proteneer/timemachine>)), and a systematic failure analysis is out of scope for this study. At the same time, it was straightforward to identify a chemical functionality, in particular amines, for which CGenFF tends to perform poorly (cf. Figure 5). Although occasionally used,⁷³ the comparison of computed ASFEs to experimental data is not a routine part of parameter optimization for the additive CHARMM force field family. In light of this, we consider the agreement between computed and experimental ASFEs quite satisfactory.

Users of CGenFF should, however, keep some additional cautions in mind. Force field parameters generated by the *cgenff* program are different depending on the version used. Similarly, for a compound that is part of CGenFF's template set (i.e., which is explicitly present in the topology file *top_all136_cgenff.rtf*), one may obtain different partial charges when processing the molecule with *cgenff* compared to the charges found in the template topology file. Since the parameters generated by *cgenff* are based on a machine-learning model, the resulting assigned charges and bonded parameters can vary as the training set for the force field is extended.^{44,74} Keeping this in mind, the above observations are the expected behavior. Nevertheless, some differences in parameters we encountered during the validation phase were unexpectedly drastic, which caused some confusion, e.g., for 2-methylfurane, there are differences in partial charges of > 0.2 e between the two *cgenff*/CGenFF versions, changing the computed ASFE by almost 2 kcal/mol. The need for strict version control of parameters and keeping the version of *cgenff* constant during a project might need to be better communicated.

SUMMARY

In conclusion, the implementation of *transformato* establishes a scalable solution for calculating ASFEs. Separating the “alchemical setup and post-processing” from running the underlying MD simulations is beneficial as the two functionalities can be optimized independently. The computational efficiency of *transformato* is also a step toward democratizing access to extensive testing of force fields, empowering even smaller research entities with limited resources to perform ASFE calculations for the FreeSolv database. The results obtained offer insights in the limitations of the *cgenff*/CGenFF small molecule parameter set for calculating ASFEs and may serve as a starting point for further development of the CGenFF force field.

ASSOCIATED CONTENT

Data Availability Statement

All plots shown in this paper were produced using the Jupyter-notebook available on GitHub (<https://github.com/JohannesKarwou/notebooks/blob/main/freeSolvSummary.ipynb>). The notebook also contains the calculations of all statistics reported in this paper (RMSE, MAE, Pearson correlation, and Spearman's rank correlation), as well as the corresponding bootstrapped errors. *Transformato* version used in this work (release v0.3): <https://github.com/wiederm/transformato>. *Macha* version used in this work (release v0.0.1): <https://github.com/akaupang/macha>.

Supporting Information

The Supporting Information is available free of charge at <https://pubs.acs.org/doi/10.1021/acs.jctc.3c00691>.

Supporting figures and tables. Figure S1: comparison of two approaches to compute the LJ LRC; Figure S2: molecules for which ASFEs with comparable simulation setups and force field parameters have been reported in the literature; Figure S3: initial comparison between results from *transformato* (TF) and reference results from the literature before ensuring that identical force field parameters were used; Figure S4: comparison of ASFEs calculated with TF when using two different switching functions for the LJ interactions; Figure S5:

one of the 11 organophosphorodithioates for which we report no ASFEs (Mobley ID: 7754849); Figure S6: average long-range correction as a function of the number of atoms per molecule, including hydrogens, are shown as blue squares; Figure S7: expanded statistical measures for the results reported in this study (blue) and by Mobley and Guthrie; Table S1: all molecules for which the *cgenff* program failed to provide parameters; and Table S2: molecules, for which the standard deviation (std) of the ASFE (dG TF) was >0.6 kcal/mol when using the default simulation length per state of 5 ns (PDF)

Supporting spreadsheet. Entries labeled *dG (TF)* are the direct results calculated with *transformato*, entries labeled *dG (TF) lrc* are the long-range corrected ASFE, i.e., the final results on which our analysis and discussion are based; entry *std* lists the standard deviation between the four individual runs; file further provides the number of intermediate states used for each ASFE calculation (*nr inst*), and the initial box size and number of water molecules for each solute (*box length* and *nr waters*); and functional group used for the analysis of the respective molecule can be found in the *functional group* column (XLSX)

Force field parameters for each solute generated by *cgenff* (ZIP)

AUTHOR INFORMATION

Corresponding Author

Stefan Boresch – Faculty of Chemistry, Institute of Computational Biological Chemistry, University of Vienna, 1090 Vienna, Austria; orcid.org/0000-0002-2793-6656; Email: stefan@mdy.univie.ac.at

Authors

Johannes Karwounopoulos – Faculty of Chemistry, Institute of Computational Biological Chemistry, University of Vienna, 1090 Vienna, Austria; Vienna Doctoral School of Chemistry (DoSChem), University of Vienna, 1090 Vienna, Austria; orcid.org/0000-0002-0172-863X

Åsmund Kaupang – Department of Pharmacy, Section for Pharmaceutical Chemistry, University of Oslo, 0316 Oslo, Norway; orcid.org/0000-0003-4376-0774

Marcus Wieder – Department of Pharmaceutical Sciences, Pharmaceutical Chemistry Division, University of Vienna, 1090 Vienna, Austria; orcid.org/0000-0003-2631-8415

Complete contact information is available at: <https://pubs.acs.org/doi/10.1021/acs.jctc.3c00691>

Funding

Open Access is funded by the Austrian Science Fund (FWF).

Notes

The authors declare no competing financial interest.

ACKNOWLEDGMENTS

The authors would like to thank Charles L. Brooks III and Arghya Chakravorty for several fruitful discussions, as well as H. Lee Woodcock for feedback on an early version of this manuscript. This research was funded by National Institute of General Medical Sciences grant number 7601R01GM129519-0 and grant P-31024-N28 of the Austrian Science Fund (FWF).

REFERENCES

- (1) Cournia, Z.; Allen, B.; Sherman, W. Relative Binding Free Energy Calculations in Drug Discovery: Recent Advances and Practical Considerations. *J. Chem. Inf. Model.* **2017**, *57*, 2911–2937.
- (2) Chodera, J. D.; Mobley, D. L.; Shirts, M. R.; Dixon, R. W.; Branson, K.; Pande, V. S. Alchemical free energy methods for drug discovery: progress and challenges. *Curr. Opin. Struct. Biol.* **2011**, *21*, 150–160.
- (3) Williams-Noonan, B. J.; Yuriev, E.; Chalmers, D. K. Free Energy Methods in Drug Design: Prospects of “alchemical Perturbation” in Medicinal Chemistry. *J. Med. Chem.* **2018**, *61*, 638–649.
- (4) Lee, T. S.; Tsai, H. C.; Ganguly, A.; York, D. M. ACES: Optimized Alchemically Enhanced Sampling. *J. Chem. Theory Comput.* **2023**, *19*, 472–487.
- (5) Geballe, M. T.; Skillman, A. G.; Nicholls, A.; Guthrie, J. P.; Taylor, P. J. The SAMPL2 blind prediction challenge: Introduction and overview. *J. Comput.-Aided Mol. Des.* **2010**, *24*, 259–279.
- (6) Kashefolgheta, S.; Oliveira, M. P.; Rieder, S. R.; Horta, B. A.; Acree, W. E.; Hünenberger, P. H. Evaluating Classical Force Fields against Experimental Cross-Solvation Free Energies. *J. Chem. Theory Comput.* **2020**, *16*, 7556–7580.
- (7) Villa, A.; Mark, A. E. Calculation of the free energy of solvation for neutral analogs of amino acid side chains. *J. Comput. Chem.* **2002**, *23*, 548–553.
- (8) Shirts, M. R.; Pitner, J. W.; Swope, W. C.; Pande, V. S. Extremely precise free energy calculations of amino acid side chain analogs: Comparison of common molecular mechanics force fields for proteins. *J. Chem. Phys.* **2003**, *119*, 5740–5761.
- (9) Deng, Y.; Roux, B. Hydration of Amino Acid Side Chains: Nonpolar and Electrostatic Contributions Calculated from Staged Molecular Dynamics Free Energy Simulations with Explicit Water Molecules. *J. Phys. Chem. B* **2004**, *108*, 16567–16576.
- (10) Nicholls, A.; Mobley, D. L.; Guthrie, J. P.; Chodera, J. D.; Bayly, C. I.; Cooper, M. D.; Pande, V. S. Predicting Small-Molecule Solvation Free Energies: An Informal Blind Test for Computational Chemistry. *J. Med. Chem.* **2008**, *51*, 769–779.
- (11) Guthrie, J. P. A blind challenge for computational solvation free energies: Introduction and overview. *J. Phys. Chem. B* **2009**, *113*, 4501–4507.
- (12) Mobley, D. L.; Wymer, K. L.; Lim, N. M.; Guthrie, J. P. Blind prediction of solvation free energies from the SAMPL4 challenge. *J. Comput.-Aided Mol. Des.* **2014**, *28*, 135–150.
- (13) Wang, J.; Wolf, R. M.; Caldwell, J. W.; Kollman, P. A.; Case, D. A. Development and testing of a general amber force field. *J. Comput. Chem.* **2004**, *25*, 1157–1174.
- (14) Mobley, D. L.; Bayly, C. I.; Cooper, M. D.; Shirts, M. R.; Dill, K. A. Small molecule hydration free energies in explicit solvent: An extensive test of fixed-charge atomistic simulations. *J. Chem. Theory Comput.* **2009**, *5*, 350–358.
- (15) Shivakumar, D.; Deng, Y.; Roux, B. Computations of absolute solvation free energies of small molecules using explicit and implicit solvent model. *J. Chem. Theory Comput.* **2009**, *5*, 919–930.
- (16) Shivakumar, D.; Williams, J.; Wu, Y.; Damm, W.; Shelley, J.; Sherman, W. Prediction of absolute solvation free energies using molecular dynamics free energy perturbation and the opfs force field. *J. Chem. Theory Comput.* **2010**, *6*, 1509–1519.
- (17) Shivakumar, D.; Harder, E.; Damm, W.; Friesner, R. A.; Sherman, W. Improving the prediction of absolute solvation free energies using the next generation opfs force field. *J. Chem. Theory Comput.* **2012**, *8*, 2553–2558.
- (18) Oostenbrink, C.; Villa, A.; Mark, A. E.; Van Gunsteren, W. F. A biomolecular force field based on the free enthalpy of hydration and solvation: The GROMOS force-field parameter sets 53A5 and 53A6. *J. Comput. Chem.* **2004**, *25*, 1656–1676.
- (19) Malde, A. K.; Zuo, L.; Breeze, M.; Stroet, M.; Poger, D.; Nair, P. C.; Oostenbrink, C.; Mark, A. E. An Automated Force Field Topology Builder (ATB) and Repository: Version 1.0. *J. Chem. Theory Comput.* **2011**, *7*, 4026–4037. PMID: 26598349
- (20) Koziara, K. B.; Stroet, M.; Malde, A. K.; Mark, A. E. Testing and validation of the Automated Topology Builder (ATB) version 2.0: prediction of hydration free enthalpies. *J. Comput.-Aided Mol. Des.* **2014**, *28*, 221–233.
- (21) Stroet, M.; Caron, B.; Visscher, K. M.; Geerke, D. P.; Malde, A. K.; Mark, A. E. Automated Topology Builder Version 3.0: Prediction of Solvation Free Enthalpies in Water and Hexane. *J. Chem. Theory Comput.* **2018**, *14*, 5834–5845. PMID: 30289710
- (22) Boulanger, E.; Huang, L.; Rupakheti, C.; MacKerell, A. D. J.; Roux, B. Optimized Lennard-Jones Parameters for Druglike Small Molecules. *J. Chem. Theory Comput.* **2018**, *14*, 3121–3131. PMID: 29694035
- (23) Mobley, D. L.; Guthrie, J. P. FreeSolv: a database of experimental and calculated hydration free energies, with input files. *J. Comput.-Aided Mol. Des.* **2014**, *28*, 711–720.
- (24) Lindahl, E.; Hess, B.; van der Spoel, D. GROMACS 3.0: A package for molecular simulation and trajectory analysis. *J. Mol. Model.* **2001**, *7*, 306–317.
- (25) Van Der Spoel, D.; Lindahl, E.; Hess, B.; Groenhof, G.; Mark, A. E.; Berendsen, H. J. GROMACS: Fast, flexible, and free. *J. Comput. Chem.* **2005**, *26*, 1701–1718.
- (26) Jorgensen, W. L.; Chandrasekhar, J.; Madura, J. D.; Impey, R. W.; Klein, M. L. Comparison of simple potential functions for simulating liquid water. *J. Chem. Phys.* **1983**, *79*, 926–935.
- (27) Jakalian, A.; Bush, B. L.; Jack, D. B.; Bayly, C. I. Fast, Efficient Generation of High-Quality Atomic Charges. AM1-BCC Model: I. Method. *J. Comput. Chem.* **2000**, *21*, 132–146.
- (28) Jakalian, A.; Jack, D. B.; Bayly, C. I. Fast, efficient generation of high-quality atomic charges. AM1-BCC model: II. Parameterization and validation. *J. Comput. Chem.* **2002**, *23*, 1623–1641.
- (29) Duarte Ramos Matos, G.; Kyu, D. Y.; Loeffler, H. H.; Chodera, J. D.; Shirts, M. R.; Mobley, D. L. Approaches for Calculating Solvation Free Energies and Enthalpies Demonstrated with an Update of the FreeSolv Database. *J. Chem. Eng. Data* **2017**, *62*, 1559–1569.
- (30) Marenich, A. V.; Kelly, C. P.; Thompson, J. D.; Hawkins, G. D.; Chambers, C. C.; Giesen, D. J.; Winget, P.; Cramer, C. J.; Truhlar, D. G. *Minnesota Solvation Database Home Page*; Minnesota Solvation Database, University of Minnesota: Minneapolis, 2012.
- (31) Guthrie, J.; Mobley, D. *The Guthrie Hydration Free Energy Database of Experimental Small Molecule Hydration Free Energies*; Department of Pharmaceutical Sciences, UCI, 2018.
- (32) Riquelme, M.; Lara, A.; Mobley, D. L.; Verstraelen, T.; Matamala, A. R.; Vöhringer-Martinez, E. Hydration Free Energies in the FreeSolv Database Calculated with Polarized Iterative Hirshfeld Charges. *J. Chem. Inf. Model.* **2018**, *58*, 1779–1797.
- (33) Dodda, L. S.; Vilseck, J. Z.; Tirado-Rives, J.; Jorgensen, W. L. 1.14*CM1A-LBCC: Localized Bond-Charge Corrected CM1A Charges for Condensed-Phase Simulations. *J. Phys. Chem. B* **2017**, *121*, 3864–3870.
- (34) Jorgensen, W. L.; Tirado-Rives, J. Potential energy functions for atomic-level simulations of water and organic and biomolecular systems. *Proc. Natl. Acad. Sci. U. S. A.* **2005**, *102*, 6665–6670.
- (35) Ribeiro, R. F.; Marenich, A. V.; Cramer, C. J.; Truhlar, D. G. Prediction of SAMPL2 aqueous solvation free energies and tautomeric ratios using the SM8, SM8AD, and SMD solvation models. *J. Comput.-Aided Mol. Des.* **2010**, *24*, 317–333.
- (36) Klamt, A.; Diedenhofen, M. Blind prediction test of free energies of hydration with COSMO-RS. *J. Comput.-Aided Mol. Des.* **2010**, *24*, 357–360.
- (37) Luukkonen, S.; Belloni, L.; Borgis, D.; Levesque, M. Predicting Hydration Free Energies of the FreeSolv Database of Drug-like Molecules with Molecular Density Functional Theory. *J. Chem. Inf. Model.* **2020**, *60*, 3558.
- (38) Borgis, D.; Luukkonen, S.; Belloni, L.; Jeanmairet, G. Accurate prediction of hydration free energies and solvation structures using molecular density functional theory with a simple bridge functional. *J. Chem. Phys.* **2021**, *155*, No. 024117.

- (39) Zhang, Z. Y.; Peng, D.; Liu, L.; Shen, L.; Fang, W. H. Machine Learning Prediction of Hydration Free Energy with Physically Inspired Descriptors. *J. Phys. Chem. Lett.* **2023**, *14*, 1877–1884.
- (40) Wiercioch, M.; Kirchmair, J. DNN-PP: A novel Deep Neural Network approach and its applicability in drug-related property prediction. *Expert Syst. Appl.* **2023**, *213*, No. 119055.
- (41) Wieder, M.; Fleck, M.; Braunsfeld, B.; Boresch, S. Alchemical free energy simulations without speed limits. A generic framework to calculate free energy differences independent of the underlying molecular dynamics program. *J. Comput. Chem.* **2022**, *43*, 1151–1160.
- (42) Karwounopoulos, J.; Wieder, M.; Boresch, S. Relative binding free energy calculations with transformato: A molecular dynamics engine-independent tool. *Front. Mol. Biosci.* **2022**, *9*, No. 954638.
- (43) Boresch, S.; Bruckner, S. Avoiding the van der Waals endpoint problem using serial atomic insertion. *J. Comput. Chem.* **2011**, *32*, 2449–2458.
- (44) Vanommeslaeghe, K.; Hatcher, E.; Acharya, C.; Kundu, S.; Zhong, S.; Shim, J.; Darian, E.; Guvench, O.; Lopes, P.; Vorobyov, I.; Mackerell, A. D. CHARMM general force field: A force field for drug-like molecules compatible with the CHARMM all-atom additive biological force fields. *J. Comput. Chem.* **2010**, *31*, 671–690.
- (45) Yu, W.; He, X.; Vanommeslaeghe, K.; MacKerell, A. D. Extension of the CHARMM general force field to sulfonyl-containing compounds and its utility in biomolecular simulations. *J. Comput. Chem.* **2012**, *33*, 2451–2468.
- (46) Vanommeslaeghe, K.; Raman, E. P.; MacKerell, A. D. Automation of the CHARMM General Force Field (CGenFF) II: Assignment of Bonded Parameters and Partial Atomic Charges. *J. Chem. Inf. Model.* **2012**, *52*, 3155–3168.
- (47) Vanommeslaeghe, K.; MacKerell, A. D. Automation of the CHARMM general force field (CGenFF) I: Bond perception and atom typing. *J. Chem. Inf. Model.* **2012**, *52*, 3144–3154.
- (48) Gutiérrez, I. S.; Lin, F.-Y.; Vanommeslaeghe, K.; Lemkul, J. A.; Armacost, K. A.; Brooks, C. L.; MacKerell, A. D. Parametrization of halogen bonds in the CHARMM general force field: Improved treatment of ligand–protein interactions. *Bioorg. Med. Chem.* **2016**, *24*, 4812–4825.
- (49) Brooks, B. R.; Brooks, C. L.; Mackerell, A. D.; Nilsson, L.; Petrella, R. J.; Roux, B.; Won, Y.; Archontis, G.; Bartels, C.; Boresch, S.; Caffisch, A.; Caves, L.; Cui, Q.; Dinner, A. R.; Feig, M.; Fischer, S.; Gao, J.; Hodoseck, M.; Im, W.; Kuczera, K.; Lazaridis, T.; Ma, J.; Ovchinnikov, V.; Paci, E.; Pastor, R. W.; Post, C. B.; Pu, J. Z.; Schaefer, M.; Tidor, B.; Venable, R. M.; Woodcock, H. L.; Wu, X.; Yang, W.; York, D. M.; Karplus, M. CHARMM: The biomolecular simulation program. *J. Comput. Chem.* **2009**, *30*, 1545–1614.
- (50) 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*, No. 1005659.
- (51) Shirts, M. R.; Chodera, J. D. Statistically optimal analysis of samples from multiple equilibrium states. *J. Chem. Phys.* **2008**, *129*, 124105.
- (52) Mey, A. S. J. S.; Allen, B. K.; McDonald, H. E. B.; Chodera, J. D.; Hahn, D. F.; Kuhn, M.; Michel, J.; Mobley, D. L.; Naden, L. N.; Prasad, S.; Rizzi, A.; Scheen, J.; Shirts, M. R.; Tresadern, G.; Xu, H. Best Practices for Alchemical Free Energy Calculations. *Living J. Comput. Mol. Sci.* **2020**, *2*, 18378.
- (53) O’Boyle, N. M.; Morley, C.; Hutchison, G. R. Pybel: A Python wrapper for the OpenBabel cheminformatics toolkit. *Chem. Cent. J.* **2008**, *2*, S.
- (54) Lee, J.; Cheng, X.; Swails, J. M.; Yeom, M. S.; Eastman, P. K.; Lemkul, J. A.; Wei, S.; Buckner, J.; Jeong, J. C.; Qi, Y.; Jo, S.; Pande, V. S.; Case, D. A.; Brooks, C. L.; MacKerell, A. D.; Klauda, J. B.; Im, W. CHARMM-GUI Input Generator for NAMD, GROMACS, AMBER, OpenMM, and CHARMM/OpenMM Simulations Using the CHARMM36 Additive Force Field. *J. Chem. Theory Comput.* **2016**, *12*, 405–413.
- (55) Jo, S.; Kim, T.; Iyer, V. G.; Im, W. CHARMM-GUI: A web-based graphical user interface for CHARMM. *J. Comput. Chem.* **2008**, *29*, 1859–1865.
- (56) Åqvist, J.; Wennerström, P.; Nervall, M.; Bjelic, S.; Brandsdal, B. O. Molecular dynamics simulations of water and biomolecules with a Monte Carlo constant pressure algorithm. *Chem. Phys. Lett.* **2004**, *384*, 288–294.
- (57) Chow, K. H.; Ferguson, D. M. Isothermal-isobaric molecular dynamics simulations with Monte Carlo volume sampling. *Comput. Phys. Commun.* **1995**, *91*, 283–289.
- (58) Miyamoto, S.; Kollman, P. A. Settle: An analytical version of the SHAKE and RATTLE algorithm for rigid water models. *J. Comput. Chem.* **1992**, *13*, 952–962.
- (59) 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.
- (60) Steinbach, P. J.; Brooks, B. R. New spherical-cutoff methods for long-range forces in macromolecular simulation. *J. Comput. Chem.* **1994**, *15*, 667–683.
- (61) Ryckaert, J. P.; Ciccotti, G.; Berendsen, H. J. 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.
- (62) Brooks, B. R.; Brucoleri, R. E.; Olafson, B. D.; States, D. J.; Swaminathan, S.; Karplus, M. CHARMM: A Program for Macromolecular Energy, Minimization, and Dynamics Calculations*. *J. Comput. Chem.* **1983**, *4*, 187–217.
- (63) Kirkwood, J. G. Statistical Mechanics of Fluid Mixtures. *J. Chem. Phys.* **1935**, *3*, 300–313.
- (64) Fleck, M.; Wieder, M.; Boresch, S. Dummy Atoms in Alchemical Free Energy Calculations. *J. Chem. Theory Comput.* **2021**, *17*, 4403–4419.
- (65) Shirts, M. R.; Mobley, D. L.; Chodera, J. D.; Pande, V. S. Accurate and Efficient Corrections for Missing Dispersion Interactions in Molecular Simulations. *J. Phys. Chem. B* **2007**, *111*, 13052–13063.
- (66) Huang, J.; Rauscher, S.; Nawrocki, G.; Ran, T.; Feig, M.; De Groot, B. L.; Grubmüller, H.; MacKerell, A. D. CHARMM36m: an improved force field for folded and intrinsically disordered proteins. *Nat. Methods* **2017**, *14*, 71–73.
- (67) Leonard, A. N.; Simmonett, A. C.; Pickard, F. C.; Huang, J.; Venable, R. M.; Klauda, J. B.; Brooks, B. R.; Pastor, R. W. Comparison of Additive and Polarizable Models with Explicit Treatment of Long-Range Lennard-Jones Interactions Using Alkane Simulations. *J. Chem. Theory Comput.* **2018**, *14*, 948–958.
- (68) Wennberg, C. L.; Murtola, T.; Páll, S.; Abraham, M. J.; Hess, B.; Lindahl, E. Direct-Space Corrections Enable Fast and Accurate Lorentz-Berthelot Combination Rule Lennard-Jones Lattice Summation. *J. Chem. Theory Comput.* **2015**, *11*, S737–S746.
- (69) Momany, F. A.; Rone, R. Validation of the general purpose QUANTA 3.2/CHARMM force field. *J. Comput. Chem.* **1992**, *13*, 888–900.
- (70) Wahman, D. G. First acid ionization constant of the drinking water relevant chemical cyanuric acid from 5 to 35 °C. *Environ. Sci.: Water Res. Technol.* **2018**, *4*, 1522–1530.
- (71) Reva, I. Comment on “Density functional theory studies on molecular structure, vibrational spectra and electronic properties of cyanuric acid”. *Spectrochim. Acta, Part A* **2015**, *151*, 232–236.
- (72) Pérez-Manríquez, L.; Cabrera, A.; Sansores, L. E.; Salcedo, R. Aromaticity in cyanuric acid. *J. Mol. Model.* **2011**, *17*, 1311–1315.
- (73) Chen, I. J.; Yin, D.; MacKerell, A. D. Combined ab initio/empirical approach for optimization of Lennard-Jones parameters for polar-neutral compounds. *J. Comput. Chem.* **2002**, *23*, 199–213.
- (74) Orr, A. A.; Sharif, S.; Wang, J.; MacKerell, A. D. Preserving the Integrity of Empirical Force Fields. *J. Chem. Inf. Model.* **2022**, *62*, 3825–3831.

Section 2.3:

Insights and Challenges in Correcting Force Field Based Solvation Free Energies Using A Neural Network Potential

ChemRxiv. (2023)

Authors: Johannes Karwounopoulos, Zhiyi Wu, Sara Tkaczyk, Shuzhe Wang, Adam Baskerville, Kavindri Ranasinghe, Thierry Langer, Geoffrey Wood, Marcus Wieder, Stefan Boresch

Contributions: Conceptualization: JK, MW, and SB; methodology: JK, ZW, MW, and SB; software: JK, ZW and MW; investigation: JK, ZW, MW and SB; writing: JK, ZW, MW, and SB; supervision: MW and SB

Insights and Challenges in Correcting Force Field Based Solvation Free Energies Using A Neural Network Potential

Johannes Karwounopoulos ^{1,2}, Zhiyi Wu ⁵, Sara Tkaczyk ^{3,4}, Shuzhe Wang ⁵, Adam Baskerville ⁵, Kavindri Ranasinghe ⁵, Thierry Langer ³, Geoffrey Wood ⁵, Marcus Wieder ^{5, 6*}, Stefan Boresch ^{1*}

*For correspondence:

stefan@mdy.univie.ac.at (SB);
marcus.wieder@gmail.com (MW)

¹Faculty of Chemistry, Institute of Computational Biological Chemistry, University Vienna, Währingerstr. 17, 1090 Vienna, Austria; ²Vienna Doctoral School of Chemistry (DoSChem), University of Vienna, Währingerstr. 42, 1090 Vienna, Austria; ³Department of Pharmaceutical Sciences, Pharmaceutical Chemistry Division, University of Vienna, Josef-Holaubek-Platz 2, 1090 Vienna, Austria; ⁴Vienna Doctoral School of Pharmaceutical, Nutritional and Sport Sciences (PhaNuSpo), University of Vienna, Josef-Holaubek-Platz 2, 1090 Vienna, Austria; ⁵Exscientia plc, Schroedinger Building, Oxford, United Kingdom; ⁶Open Molecular Software Foundation, Davis, California 95616, United States

Abstract We present a comprehensive study investigating the potential gain in accuracy for calculating absolute solvation free energies (ASFE) using a neural network potential for the intramolecular energies. We calculated ASFE using the Open Force Field (OpenFF) and compared the results to previously calculated ASFEs employing the CHARMM General Force Field (CGenFF). By applying a nonequilibrium (NEQ) switching approach between the molecular mechanics (MM) description (either OpenFF or CGenFF) and the machine learning (ML)/MM level of theory (using ANI-2x as the ML potential), we attempt to improve the accuracy of the calculated ASFEs. The predictive performance of the results did not change when applying this approach to all 589 small molecules in the FreeSolv database that ANI-2x can describe. When selecting a subset of 156 molecules, focusing on compounds where the force fields performed poorly, we saw a slight improvement in the root-mean-square error (RMSE) and mean absolute error (MAE). The majority of our calculations utilized unidirectional NEQ protocols based on Jarzynski's equation. Additionally, we conducted bidirectional NEQ switching for the subset of 156 FreeSolv molecules. Notably, only a small fraction (10 out of 156) exhibited statistically significant discrepancies between unidirectional and bidirectional NEQ switching free energy estimates.

Introduction

It is hard to overestimate the importance of solvation for biological processes. The correct description of the interaction of water with biological macromolecules and small molecule substrates

is crucial to understanding ligand binding and predicting binding affinities. The accuracy of the molecular mechanical force field employed in calculations of free energy differences is one of the limitations of such methods. For this reason, several large-scale studies have focused on the computation of solvation free energy differences. The comparison of the predictions to experimental data helps identify the strengths and weaknesses of the force field used [4, 6, 7, 36].

Recently, we reported calculated solvation free energy differences for most of the compounds in the FreeSolv database using the CHARMM General force field (CGenFF) [21]. We have now completed analogous calculations with the OpenFF 2.0 force field, and these results are presented here. Overall, the agreement with the experimental data is acceptable (for OpenFF 2.0 root mean squared error (RMSE): 1.33 [1.24, 1.43] kcal/mol, mean absolute error (MAE): 1.01 [0.95, 1.07] kcal/mol; for CGenFF RMSE: 1.76 [1.52, 2.02] kcal/mol, MAE: 1.12 [1.02, 1.23] kcal/mol), but, in particular, for CGenFF, there is a sizable number of molecules with significant discrepancies between the computed and experimentally determined solvation free energies.

Neural network potentials (NNP) are a recent development that allows a more accurate description of intra- and intermolecular interactions at affordable cost. The use of NNPs in free energy simulations may, therefore, improve the accuracy of such calculations [1, 24, 35]. However, although NNPs are fast compared to quantum chemical calculations, they are significantly slower than classical mechanical force fields [14]. Furthermore, it is unknown how to apply certain "tricks" used in free energy simulations, such as soft-core potentials, in combination with NNPs. One can avoid both complications by indirect pathways, commonly used to compute free energy differences with Quantum Mechanics (QM)/MM hybrid potential energy functions [9, 16, 26, 27, 33, 41, 46]. Indirect free energy calculations use a computationally cheaper description of the potential energy (e.g., a molecular mechanics force field) and calculate the free energy contribution needed for changing to a more expensive description of the potential energy (e.g., a QM potential). The free energy difference between the different levels of theory can be calculated reliably using nonequilibrium switching techniques [17, 23, 37, 40, 44]. NNPs offer a tempting tradeoff between accuracy and speed compared to molecular mechanics and QM methods, which is why they can be applied as the high-level potential in these indirect cycles. An early example of using NNPs to refine classical free energy simulations is a study by Rufa et al. [35]. Recently, we investigated the convergence of the correction step required in indirect pathways, i.e., calculating the free energy difference between an MM and a machine learning (ML) representation of a system [40]. In both studies, the ANI-2x NNP was used [10, 39]. Simulations with ANI and hybrid ANI/MM simulations can be carried out efficiently using TorchANI [15] and NNPOPS [14].

The computational framework of reference [40] is suitable for use in the gas phase and aqueous solution. In this work, we explore whether MM→ML / MM→ML/MM corrections can improve the agreement of computed solvation free energies with the experiment. There are very efficient implementations of ANI in OpenMM; furthermore, mixing MM and ANI is straightforward [12, 15].

Therefore, the MM→ML/MM corrections can be integrated smoothly into our automated workflows to compute the solvation free energies with CGenFF and OpenFF 2.0 [21, 22]. Nevertheless, ANI/MM simulations are costly, so the focus is on these molecules that performed poorly with CGenFF, OpenFF 2.0, or both. From previous simulations, conducted exclusively at the MM level of theory, we identified the 100 compounds for each force field exhibiting the highest discrepancies compared to their experimental values (selection was limited to molecules with elemental composition covered by the ANI-2x training set (H, C, N, O, F, S, and Cl)). Thus, we created a dataset comprising 156 compounds, where 41 were among the worst-performing compounds for both force fields. The remaining compounds exhibited poor performance for CGenFF (56 compounds) or OpenFF exclusively (59 compounds).

Furthermore, one must consider the limitations of the ANI/MM hybrid potential function currently implemented in OpenMM-ML. As pointed out in [14], the coupling between MM and ANI is analogous to "mechanical embedding" in QM/MM simulations. Specifically, only the intramolecular interactions of the solute are described by the NNP, whereas the solute-solvent interactions remain classical. Therefore, one cannot expect improvements in all cases. Nevertheless, differences between the MM and ML/MM descriptions of the system should be discernible, especially for larger, more flexible solutes. These cases are of particular interest, as they can help identify shortcomings of the force field.

The remainder of the manuscript is organized as follows: We computed endstate corrections by analyzing nonequilibrium work distributions for the joined dataset of poor-performing compounds. This involved unidirectional transitioning from the force field-based description (either CGenFF or OpenFF) to the machine learning-based description (ANI-2x in both cases). In addition to that, we divided our findings into two separate sections: firstly, we determined the endstate correction for all 589 molecules within our OpenFF results, and secondly, we computed free energies using the more comprehensive two-sided Crook's approach for the 156 molecules found in the combined dataset, exclusively for the CGenFF results.

Theory

Absolute free energy calculations

We calculated ASFes using the alchemical pathway shown on the left side of Figure 1. Both protocols used in this work (cf. Methods) involve (at least partial) annihilation of the solute's non-bonded interactions, i.e., turning off both the non-bonded intramolecular interaction of the solute and its intermolecular interactions with its surroundings (solvent). The annihilation absolute free energy protocol requires a gas phase correction; the absolute solvation free energy of interest $\Delta G_{\text{MM} \rightarrow \text{MM}}^{\text{solv}}$ is obtained as the difference between the annihilation protocol in solution $\Delta G_{L_1}^{\text{aq}}$ and in the gas phase $\Delta G_{L_1}^{\text{gas}}$ [28].

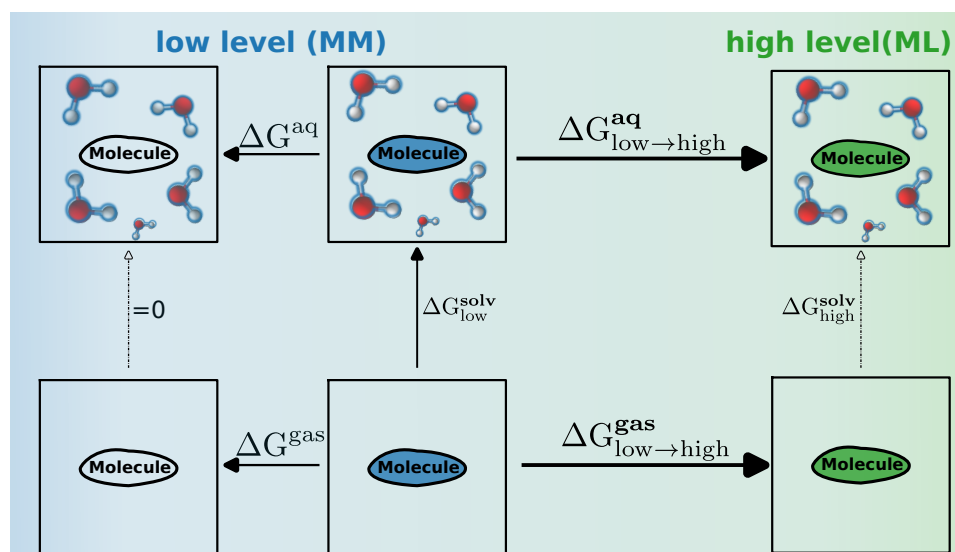


Figure 1. Free energy calculations between high and low levels of theory can be used to correct alchemical free energy calculations performed at a low level of theory. Left: The thermodynamic cycle used to compute an ASFE $\Delta G_{\text{low}}^{\text{solv}}$ at the MM level of theory using annihilation of the solute's non-bonded interactions [28]. Right: The indirect free energy cycle to correct $\Delta G_{\text{low}}^{\text{solv}}$ (in this work using either the CGenFF or the OpenFF force field) from the MM to the ML/MM level of theory.

Endstate correction

Free energy estimates from non-equilibrium work values (W) can be calculated using the Jarzynski equation [18] or the Crooks' fluctuation theorem [8]. The Jarzynski equation recovers the free energy estimate between a target and a reference distribution based on a non-equilibrium work process that starts at the reference and anneals to the target distribution. According to the Jarzynski equation, the free energy difference for the accumulated nonequilibrium work between two states 0 and 1 ($W_{0 \rightarrow 1}$) is obtained as follows:

$$\Delta G_{\text{Jar}}^{0 \rightarrow 1} = -k_B T \ln \left\langle \exp \left(\frac{-W_{0 \rightarrow 1}}{k_B T} \right) \right\rangle_0 \quad (1)$$

In our specific use case, state 0 indicates the potential energy function at the low level of theory, while 1 represents the same system at a higher level of theory. The subscript 0 in $\langle \rangle_0$ indicates that the NEQ switching simulations to obtain the work values $W_{0 \rightarrow 1}$ are started from equilibrium configurations sampled at the lower level of theory (state 0).

The Crooks' fluctuation theorem recovers the equilibrium free energy estimate between the initial and final state based on non-equilibrium work processes that transform the reference to the target potential and *vice versa*. Thus, one has to additionally carry out sampling at and NEQ switching simulations starting from the high level of theory, or, in other words, compute work

values in the $1 \rightarrow 0$ direction. The free energy between states 0 and 1 is then given by [8]:

$$\Delta G_{\text{Crooks}}^{0 \leftrightarrow 1} = k_B T \left(\ln \frac{\langle f(\Delta W_{1 \rightarrow 0} + C) \rangle_1}{\langle f(\Delta W_{0 \rightarrow 1} - C) \rangle_0} \right) + C \quad (2)$$

where $f(x)$ denotes the Fermi function $f(x) = (1 + \exp(\frac{x}{k_B T}))^{-1}$ and

$$C = k_B T \ln \frac{Q_0 N_1}{Q_1 N_0} \quad (3)$$

Here, Q represents the canonical partition function of the respective state (0 and 1), and N is the number of work values in the forward and backward direction, respectively. Equation 2 is typically solved by searching iteratively for the value of C for which the argument of the logarithm becomes unity and, hence, the first term in Equation 2 vanishes. The value of C found in this manner is essentially the sought free energy difference, cf. Equation 3.

Methods

Overview of calculations/workflow

We performed ASFE calculations for most of the compounds in the FreeSolv database [11, 31], a curated collection of experimental solvation free energies for 642 drug-like molecules. Compounds containing elements not covered in the ANI-2x training were removed, leaving 589 molecules for which an MM→ML/MM endstate correction can be carried out. ASFEs were computed using two independent workflows, which we label as protocols `EXS` and `UVIE`. Here, we focus on the commonalities; full details of each of the two protocols can be found in SI. The `UVIE` protocol used the CGenFF force field [42, 43] and `transformato` [45] to calculate ASFEs at the MM level of theory. The results obtained with this approach have been previously described [21]. In the `EXS` protocol, ASFEs at the MM level of theory were computed using `openmmtools` 0.23.0 [19] and the Open Force Field (OpenFF 2.0) [3]. While the ASFEs at the MM level of theory were calculated rather differently, the MM→ML/MM endstate corrections were carried out quite similarly in both approaches, though specific adaptations were necessary.

Using the `EXS` protocol, endstate corrections were computed for all 589 molecules, i.e., the complete subset of the FreeSolv database, excluding any compounds that contained elements not covered by the training set of ANI-2x. These corrections were calculated unidirectionally. MM→ML/MM corrections using the `UVIE` workflow were computed for a subset of the FreeSolv database (156 molecules), for which the force field results were in poor agreement with the experimental data (cf. the Introduction). These corrections were computed by uni- and bidirectional NEQ switching methods (cf. below).

Endstate correction with NNP

Equilibrium simulations

MM level of theory

For each compound for which endstate corrections were carried out, simulations were performed in the gas phase and solution. OpenMM 8.0 [12] was used with an integration time step of 1 fs. Molecules were solvated in rigid TIP3P [20] water using the SETTLE algorithm [29], simulations were performed under constant pressure conditions using a Monte Carlo barostat [2, 5]. The solutes themselves were fully flexible. Coulomb interactions were calculated using the particle-mesh Ewald (PME) method [13], Lennard-Jones (LJ) interactions were calculated using a cutoff of 12 Å using the default OpenMM switching function between 10 Å and 12 Å. Every simulation was performed on energy-minimized structures using the L-BFGS minimizer.

ML/MM level of theory

The difference between MM and ML/MM simulations is the treatment of the intramolecular energetics of the small molecule, which was calculated using the ANI-2x potential instead of the respective MM force field.[10] Specifically, the high-performance ANI-2x potential re-implementation, NNPOPS (v.0.4), was used in this work [14]. To interpolate between the ANI-2x and the force field, we used the `OpenMM-ML` package.[35].

Nonequilibrium switching simulations

5 ps NEQ switching simulations were initialized by randomly selecting 300 conformations (with replacement) from the equilibrium trajectories (either at the MM or ML/MM endstate). The NEQ protocol consisted of an alternating sequence of propagation and perturbation steps, in which the potential was slowly perturbed while propagating the coordinates. In each perturbation step, the coupling parameter $\lambda = t/\tau$ was used to scale the potential energy $U = (1 - \lambda)U_{\text{MM}} + \lambda U_{\text{ML/MM}}$ as a function of the current perturbation $t \in [0, \tau]$ and the total protocol length τ . Each propagation step consisted of a 1 fs integration step to propagate the conformation x from x_t to x_{t+1} . The work value along a particular trajectory up to time $t + 1$ is calculated using $W_t + U_{t+1}(x_{t+1}) - U_t(x_{t+1})$. Nonequilibrium switching simulations can be performed uni- and bidirectionally, i.e., employing either the Jarzynski or the Crooks equation. We used the exponential averaging (EXP) estimator and the BAR estimator (as implemented in `pymbar` [38]) to obtain free energies from unidirectional and, respectively, bidirectional NEQ switching simulations.

As illustrated in Figure 1, these calculations were performed both in aqueous solution and in the gas phase to obtain $\Delta G_{\text{MM} \rightarrow \text{ML/MM}}^{\text{aq}}$ and $\Delta G_{\text{MM} \rightarrow \text{ML}}^{\text{gas}}$. The free energy differences ($\Delta G_{\text{MM} \rightarrow \text{ML/MM}}$) between levels of theory and thus the endstate correction value is obtained by $\Delta G_{\text{MM} \rightarrow \text{ML/MM}}^{\text{corr}} = -\Delta G_{\text{MM} \rightarrow \text{ML}}^{\text{gas}} + \Delta G_{\text{MM} \rightarrow \text{ML/MM}}^{\text{aq}}$. Using $\Delta G_{\text{MM} \rightarrow \text{ML/MM}}^{\text{corr}}$, the corrected ASFE can be calculated $\Delta G_{\text{MM}}^{\text{solv}} + \Delta G_{\text{MM} \rightarrow \text{ML/MM}}^{\text{corr}}$.

Corrections using the `EXS` protocol were computed using only unidirectional NEQ switching, limiting the calculation of the free energy difference to the EXP estimator. By contrast, using protocol `UVIE`, also bidirectional NEQ calculations were performed, allowing the use of the Crooks equation. As mentioned above, corrections using the `UVIE` protocol were calculated only for the 156 molecule subset. Corrections obtained by Crooks' equation are denoted as $\Delta G_{\text{MM} \leftrightarrow \text{ML}/\text{MM}}^{\text{corr, Crooks}}$. Results are presented as the deviation of the computed result from the experimental reference value, i.e.,

$$\delta \Delta G_{\text{theory}} = \Delta G_{\text{theory}}^{\text{exp}} - \Delta G_{\text{theory}}^{\text{calc}}. \quad (4)$$

The superscripts *exp* and *calc* denote the experimental and calculated solvation free energy, respectively. The subscript *theory* stands for either OpenFF or CGenFF at the MM level, or the ANI2x corrected result, indicated either as OpenFF/ANI or CGenFF/ANI, respectively.

Multistate equilibrium free energy simulations

For 10 molecules out of the 156 molecule subset, we performed multistate equilibrium free energy simulations (MFES) using a ladder of 11 equidistant λ windows, varying the coupling parameter λ from 0 (MM-endstate) to 1 (ML-endstate). Sampling was performed for 5 ns from each equilibrium distribution, and 5,000 samples were collected. To ensure that the samples represent the stationary distribution, the initial 20% of each simulation was discarded, resulting in 4,000 samples per simulation. These were further pruned, and only every 5th sample was used. From the combined set of 11 alchemical states, consisting of 800 samples each (11 $\times$ 800 in total), connecting the MM and ML potentials, we calculated the free energy difference using the MBAR estimator, as implemented in the `pymbar` package [38].

Results and Discussion

ASFE results using OpenFF 2.0

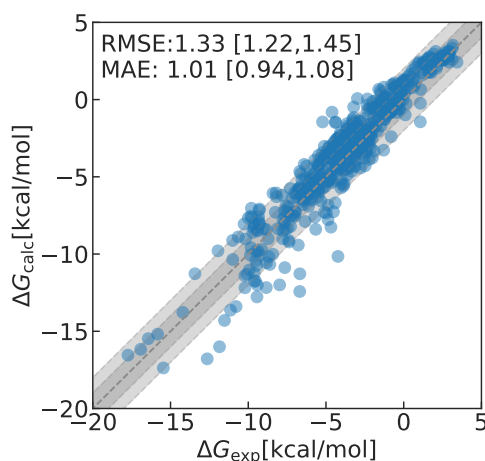


Figure 2. Absolute solvation free energy calculations using the OpenFF force field. ΔG_{calc} are the calculated values using protocol EXS, while experimental values (ΔG_{exp}) are taken from literature [31].

The performance of the classical ASFE results for 589 molecules of the FreeSolv database calculated with the EXS protocol (OpenFF 2.0) is good. A plot comparing experimental and calculated ASFEs is shown in Figure 2. Both RMSE: 1.33 [1.24, 1.43] kcal/mol and MAE: 1.01 [0.95, 1.07] kcal/mol are low. The RMSE and MAE obtained with OpenFF 2.0 are better than the values obtained with CGenFF (RMSE: 1.76 [1.52, 2.02] kcal/mol, MAE: 1.12 [1.02, 1.23] kcal/mol) [21] (protocol UVIE), as well as the values obtained with the AMBER general force field (GAFF) as reported in the FreeSolv database (RMSE: 1.54 [1.39; 1.70], MAE 1.11 [1.03; 1.19] kcal/mol) [11, 31].

In Figure S1 in the SI, we compare the ASFEs calculated with these three force fields by presenting kernel density estimates (KDE) [32, 34] of the deviation between experimental and calculated ASFE, $\delta\Delta G$. These plots highlight a weakness of all three force fields which is not apparent from the RMSE and MAE: in each case, the peak of the KDE is near -1 kcal/mol, indicating that on average ASFEs predicted by all three force fields are too hydrophobic by about 1 kcal/mol. The KDEs presented in Figure S1 in the SI are in line with earlier observations, see, e.g., [30].

Endstate corrections

Unidirectional correction of the OpenFF ASFEs to ANI-2x/OpenFF (589 compounds, protocol EXS)

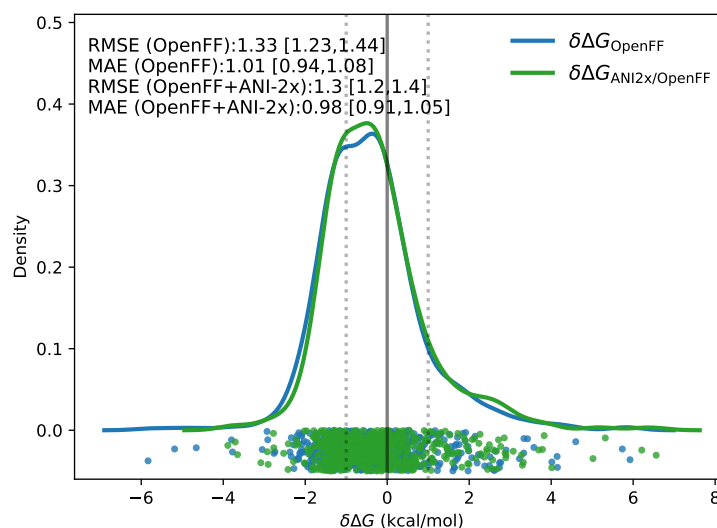


Figure 3. The KDE of $\delta\Delta G$ for the ML corrected free energy estimate shows no significant improvement for the 589 compounds of the FreeSolv database. Overlay of the KDE for $\delta\Delta G$ of the MM (blue line, ΔG_{OpenFF}) and the MM/ML (green line, $\Delta G_{\text{OpenFF/ANI2x}}$) free energy estimates (see Equation 4). Additionally, the plot shows dots indicating the individual deviations from the experimental values for the MM ASFE results (blue dots) and the ML/MM corrected ASFE results using the unidirectional correction according to protocol EXS (green dots). The gray dotted line displays the ideal behavior, $\delta\Delta G = 0$.

Using the EXS protocol, we computed unidirectional $\Delta G_{\text{MM} \rightarrow \text{ML/MM}}^{\text{corr}}$ corrections for the 589 (out of 642) molecules in the FreeSolv database that can be described by the ANI-2x NNP (cf. Methods). The results are summarized in Figure 3, where one sees that the RMSE and MAE practically do not change. Similarly, the correlation with the experiment remains unchanged (Pearson correlation is 0.95 in both cases, while the Spearman varies from 0.94 [0.93, 0.95] to 0.95 [0.94, 0.96]). All statistical descriptors (RMSE, MAE, etc.) discussed here and later in this manuscript are summarized in a supplementary file (*correctedASFE.csv*). Figure 3 superposes the KDE of $\delta\Delta G$ between the experimental and calculated MM and ML/MM corrected free energy estimates for the 589 molecules under investigation. One clearly sees that the effect of the MM $\rightarrow$ ML/MM correction is negligible.

The absolute value of the correction $\Delta G_{\text{MM} \rightarrow \text{ML/MM}}^{\text{corr}}$ for more than half of the molecules (294) is smaller than 0.5 kcal/mol. Focusing on the compounds for which the correction was larger than 0.5 kcal/mol, the predicted ASFE was in improved agreement with the experimental values in approximately 60% of the cases.

Comparison of corrections using different force fields (protocol `UVIE` and protocol `EXS`, 156 molecule subset)

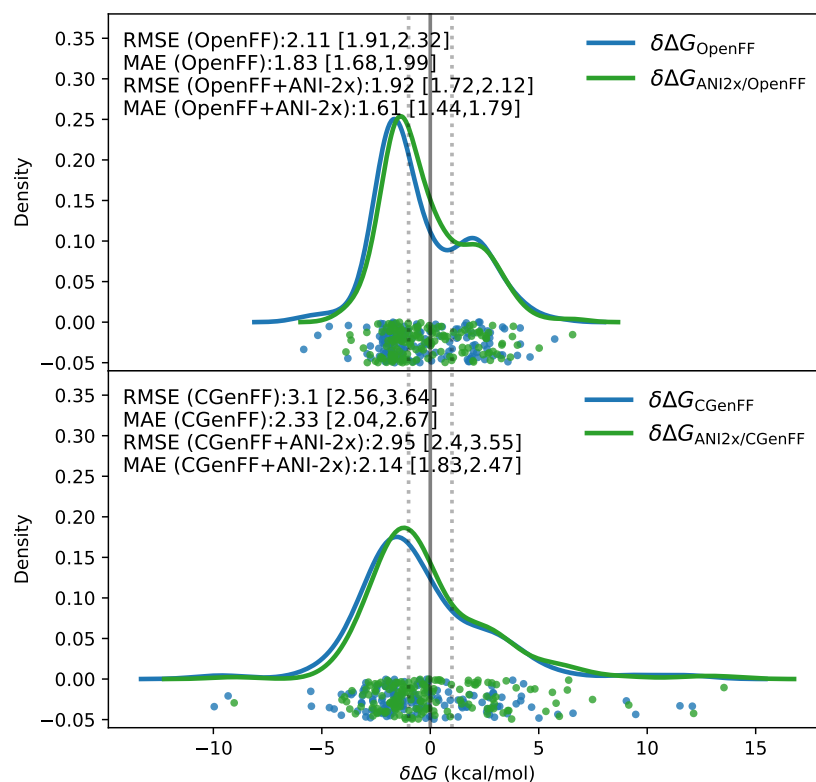


Figure 4. The KDE of $\delta\Delta G$ (Equation 4) for the 156 compound subset. Top: KDE of the errors obtained with OpenFF 2.0 (blue line) and for the ML corrected results (green line). Bottom: KDE of the errors for CGenFF (blue) and ML corrected results (green). Results for the individual molecules are shown as dots, blue for the MM, and green for the ML corrected values.

The results of the MM→ML/MM corrections for the subset of 156 molecules, for which the computed solvation energies are in poor agreement with experiment when using either the CGenFF or the OpenFF 2.0 force field, or both, are summarized in Figure 4. Using the `UVIE` protocol, the MAE was reduced by 0.15 kcal/mol, from 3.10 [2.56, 3.64] kcal/mol to 2.95 [2.40, 3.55] kcal/mol. While the Pearson correlation coefficient improved slightly from 0.77 [0.69, 0.83] to 0.80 [0.72, 0.87], there was no change in the Spearman's correlation (0.76 [0.66, 0.83] before and 0.76 [0.64, 0.83] after the correction). Applying the ML correction to the ASFes obtained with OpenFF (protocol `EXS`) gave a similar trend. The RMSE and MAE were reduced from 2.11 [1.91, 2.32] kcal/mol and 1.83 [1.68, 1.99] kcal/mol to 1.92 [1.72, 2.12] kcal/mol and 1.61 [1.44, 1.79] kcal/mol, respectively. The Pearson as well as the Spearman correlation improved slightly from 0.90 [0.87, 0.92] to 0.93 [0.90, 0.95] and from 0.91 [0.87, 0.93] to 0.92 [0.89, 0.94] respectively. While the numbers move slightly in the right directions, none of these improvements is statistically significant.

FF	corrections improving the result [%]	number of molecules with $ \Delta G_{\text{MM} \rightarrow \text{ML/MM}}^{\text{corr}} > 0.5 \text{ kcal/mol}$	thereof improving results [%]
CGenFF	62	37	68
OpenFF 2.0	63	46	67

Table 1. Comparison of improvements for the 156 compound subset

Table 1 provides some information in how many cases the ML correction ($\Delta G_{\text{MM} \rightarrow \text{ML/MM}}^{\text{corr}}$) improved the agreement with the experimental values. In both protocols, this was the case for slightly over 60% of the solutes. However, even for this reduced subset of molecules, the (absolute value of the) correction is $< 0.5 \text{ kcal/mol}$ in most cases; only for about 25% of the compounds (37 for CGenFF, 46 for OpenFF) it is larger than 0.5 kcal/mol . For these molecules, applying the MM $\rightarrow$ ML/MM correction improves the agreement with experiment in almost 70% of the cases, a slightly higher percentage compared to the entire 156 molecule subset.

Correction is dependent on the flexibility of the molecule

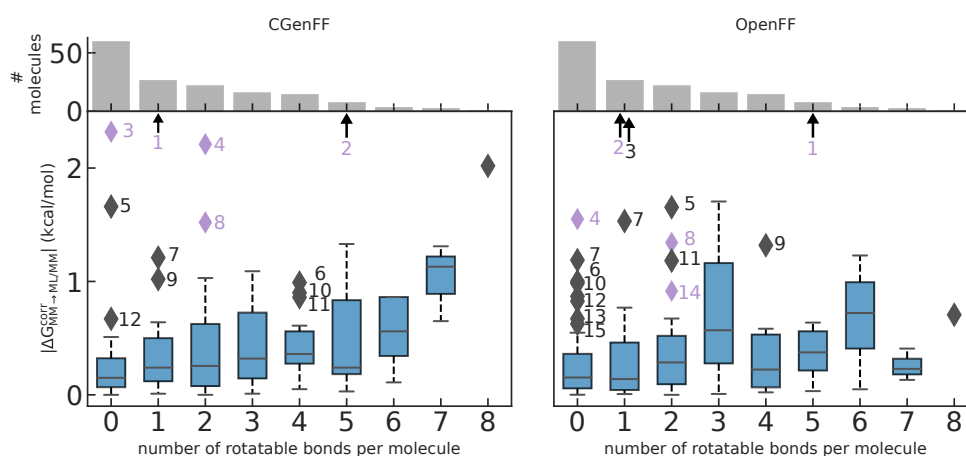


Figure 5. Unidirectional ML correction ($\Delta G_{\text{MM} \rightarrow \text{ML/MM}}^{\text{corr}}$) as a function of the number of rotatable bonds for the 156 compound subset using protocols UVIE (left side) and EXS (right side). The molecules are numbered independently for each force field, starting with one (1) for the compound having the highest correction value. All numbered molecules are shown in the SI (Figure S2 for protocol UVIE and Figure S3 for protocol EXS).

Since $\Delta G_{\text{MM} \rightarrow \text{ML/MM}}^{\text{corr}}$ is negligible in many, if not most cases, it is of some interest to understand when the correction of force field results to the ML/MM of theory is likely to be sizeable. An attempt is shown in Figure 5, where we group the corrections obtained for the 156 compound subset according to the number of rotatable bonds (as reported by the *CalcNumRotatableBonds* function in the *rdkit* functionality¹). The bars at the top of the plots indicate how many molecules have this num-

¹<https://www.rdkit.org/>

ber of rotatable bonds. This criterion is clearly far from perfect. Ring systems may be classified as having zero rotatable bonds, yet aliphatic rings are often highly flexible. The results are shown as box plots; all outliers (shown as diamonds) have numbers indicating the molecules in question. These molecules, together with their FreeSolv ID, are plotted in Figures S2 (protocol *UVIE*) and S3 (protocol *EXS*). Compounds that are outliers in both protocols are highlighted in purple.

Ignoring the outliers, there is a slight trend to larger corrections as the number of rotatable bonds increases. Some outliers with zero rotatable bonds are relatively flexible aliphatic rings (e.g., compounds with Mobley ID: 9534740 and 1896013, see Figures S2 and S3). Overall, these findings are not surprising, as the ML correction only changes the intramolecular interactions of the solutes. In contrast, the interactions between the solute and the water environment are calculated at the MM level of theory (mechanical embedding). Thus, for relatively rigid molecules (low number of rotatable bonds, aromatic rings), one cannot expect the $\Delta G_{\text{MM} \rightarrow \text{ML/MM}}^{\text{corr}}$ correction to be large. Conversely, the correction is not necessarily significant for flexible molecules. On the one hand, if the MM and the ML description of the solute intramolecular interactions lead to similar conformational preferences, there is little reason to expect large corrections. Furthermore, a minor correction might even result from a fortuitous error compensation. The MM and ML descriptions of the solute may well result in different conformational preferences. These, in principle, should lead to a large correction. However, the solute-solvent interactions, which are described classically throughout, may offset this effect.

Comparing unidirectional with bidirectional and MFES results

In protocol *UVIE*, we computed the $\text{MM} \rightarrow \text{ML/MM}$ correction not only unidirectionally, i.e., by using Jarzynski's equation, but also carried out equilibrium simulations at the ML/MM level of theory and backward switches in the $\text{ML/MM} \rightarrow \text{MM}$ direction. Hence, we also calculated $\Delta G_{\text{MM} \leftrightarrow \text{ML/MM}}^{\text{corr, Crooks}}$ using Crooks' equation. Therefore, we can systematically investigate the inaccuracy of using unidirectional vs. the more robust bidirectional NEQ methods. Out of the 156 compounds studied using the *UVIE* protocol, the deviation between the Jarzynski and Crooks results was larger than 1 kT only for 10 compounds, either in the gas phase or in aqueous solution, or both. Details are shown in Figure S4, which also presents several related characteristics for these 10 molecules. In three out of these ten cases, the deviation $|\Delta G_{\text{MM} \leftrightarrow \text{ML/MM}}^{\text{corr, Crooks}} - \Delta G_{\text{MM} \rightarrow \text{ML/MM}}^{\text{corr}}|$ is substantial, i.e., significantly larger than 1 kcal/mol.

Upon closer inspection, we noted that there was no overlap between the distributions of forward and backward work values in these three plus one additional case. When using the protocol *UVIE*, overlap between forward and backward work distributions was monitored routinely; these four molecules were the only ones for which there was no overlap. Since poor or no overlap between forward and backward work distributions raises doubts about the reliability of even the Crooks results, we also computed the correction free energy for these ten molecules by MFES. As

one can see in Figure S4, the difference between the MFES and Crooks results for the six molecules for which there is overlap is less or around 1 kT. For the solvent results (blue squares) of the four molecules without overlap, the deviation is larger (bottom panel of Figure S4).

On the one hand, these findings indicate that unidirectional methods are not sufficient in all cases, and in selected cases, even the bidirectional Crooks result may not be fully reliable. On the other hand, we observed problems only for 10 out of 156 molecules, i.e., significantly fewer than 10%. Replacing the unidirectional with the Crooks or MFES results would lead to only negligible changes in the overall RMSE, MAE, and correlation coefficients. One should also remember that the general use of bidirectional NEQ to compute the MM→ML/MM correction free energy would at least double the computational effort; the use of MFES would be too costly to apply to large sets of compounds.

Conclusion

Utilizing the ANI-2x potential, we employed an automated protocol to correct solvation free energies obtained with the OpenFF and CGenFF force fields. When applying the MM→ML/MM correction to the full subset of molecules that can be described by ANI-2x, the minuscule overall improvement in free energy is statistically not significant (see Figure 3). Focusing on the subset of molecules for which the error of the ASFE is highest using the MM protocols, we can observe some improvement and large corrections for a few molecules (see Figures 4 and 5). The majority of the results presented here were calculated using unidirectional NEQ protocols, i.e., Jarzinsky's equation. The ANI-2x potential is sufficiently fast so that we could also carry out bidirectional non-equilibrium switching and sampling for a sizeable subset of the FreeSolv (156 molecules, protocol UVIE). Only a small subset (10 out of 156) showed statistically relevant deviations between the free energy estimate based on the forward NEQ switching trajectory and the forward and reverse NEQ switching trajectories.

There are important lessons in these results: The currently available coupling between MM and ANI-2x corresponds to mechanical embedding in QM/MM [14, 35]. The description of the solute-solvent interactions remains classically at both levels of theory (MM and ML/MM). This explains why different results are obtained depending on the MM force field used, and why the corrections are tiny in most cases. As shown, large corrections are obtained primarily for large, flexible solutes. We, therefore, conclude that improving free energy estimates significantly will require advanced treatment of the interaction of the small molecule with its surroundings, i.e., moving beyond mechanical embedding [25, 47]. Given the limitations of mechanical embedding, using unidirectional methods to compute the MM→ML/MM corrections seems adequate. As an alternative to more advanced embeddings, the treatment of the entire system with the NNP should be considered; the performance of ANI-2x is undoubtedly sufficient to allow nanosecond simulations of solute-solvent systems consisting of the solute and up to a thousand solvent molecules. Obviously, direct

free energy simulations at the ML level of theory require testing that the NNP used reproduces condensed phase properties of aqueous solutions correctly. In addition, protocols for the annihilation or decoupling of the solute need to be developed to ensure that endpoint catastrophes are avoided.

Acknowledgments

The authors appreciate financial support from the National Institute of Health (1R01GM129519) and the Austrian Science Fund (FWF), grant P-31024-N28.

Supporting information

(1) In the supplementary information PDF file, the following additional details are provided:

We describe in detail how ASFEs were calculated on the MM level of theory and how the end-state corrections to the ANI-2x potential were applied for the two respective protocols (UVIE and EXS). In Figure S1 we show the error distribution of the MM calculated ASFEs for three different force fields. In Figure S2 and S3 we show the structures of the outliers observed in Figure 5 for the two investigated force fields. In Figure S4 we show several characteristics of the 10 compounds (of the 156 subset), for which the MM → ML/MM correction differed by more than 1 kT when computed by Jarzynski's and Crooks' equation.

(2) The ASFE on the MM and ML/MM level for the 589 compounds of the FreeSolv database calculated using EXS are summarized in the spreadsheet *correctedASFE_EXS_full_set.csv*

(3) The ASFE values on the MM and ML/MM level for the two force fields (OpenFF 2.0 and CGenFF) for the 156 compounds of the combined dataset calculated with protocol EXS and UVIE are summarized in the spreadsheet *correctedASFE_UVIE_EXS.csv*. In the case of the correction for *cgenff*, unidirectional values are indicated with "Jar", while bidirectional values are depicted with "Crooks".

Competing Interest Statement

The authors declare the following competing financial interest(s): SB is a consultant for Exscientia plc.

Code and data availability

- All plots shown in this paper were produced using the Jupyter-notebook available on GitHub (<https://github.com/JohannesKarwou/notebooks/blob/main/combinedDataset.ipynb>). The notebook also contains the calculations of all statistics reported in this paper (RMSE, MAE, Pearson correlation, and Spearman's rank correlation) and the corresponding bootstrapped errors.
- Python package used in this work (release v0.3): https://github.com/wiederm/endstate_correction

References

- [1] Akkus E, Tayfuroglu O, Yildiz M, Kocak A. Accurate Binding Free Energy Method from End-State MD Simulations. *J Chem Inf Model*. 2022 9; 62(17):4095–4106. doi: [10.1021/acs.jcim.2c00601](https://doi.org/10.1021/acs.jcim.2c00601).

- [2] **Åqvist J**, Wennerström P, Nervall M, Bjelic S, Brandsdal BO. Molecular dynamics simulations of water and biomolecules with a Monte Carlo constant pressure algorithm. *Chem Phys Lett*. 2004 Jan; 384(4-6):288–294. doi: [10.1016/j.cplett.2003.12.039](https://doi.org/10.1016/j.cplett.2003.12.039).
- [3] **Boothroyd S**, Behara PK, Madin OC, Hahn DF, Jang H, Gapsys V, Wagner JR, Horton JT, Dotson DL, Thompson MW, Maat J, Gokey T, Wang LP, Cole DJ, Gilson MK, Chodera JD, Bayly CI, Shirts MR, Mobley DL. Development and Benchmarking of Open Force Field 2.0.0: The Sage Small Molecule Force Field. *J Chem Theory Comput*. 2023 6; 19(11):3251–3275. doi: [10.26434/chemrxiv-2022-n2z1c-v2](https://doi.org/10.26434/chemrxiv-2022-n2z1c-v2).
- [4] **Chodera JD**, Mobley DL, Shirts MR, Dixon RW, Branson K, Pande VS. Alchemical free energy methods for drug discovery: progress and challenges. *Curr Opin Struct Biol*. 2011 Apr; 21(2):150–160. doi: [10.1016/j.sbi.2011.01.011](https://doi.org/10.1016/j.sbi.2011.01.011).
- [5] **Chow KH**, Ferguson DM. Isothermal-isobaric molecular dynamics simulations with Monte Carlo volume sampling. *Comput Phys Commun*. 1995 Sep; 91(1-3):283–289. doi: [10.1016/0010-4655\(95\)00059-o](https://doi.org/10.1016/0010-4655(95)00059-o).
- [6] **Cournia Z**, Allen B, Sherman W. Relative Binding Free Energy Calculations in Drug Discovery: Recent Advances and Practical Considerations. *J Chem Inf Model*. 2017 12; 57(12):2911–2937. doi: [10.1021/acs.jcim.7b00564](https://doi.org/10.1021/acs.jcim.7b00564).
- [7] **Cournia Z**, Chipot C, Roux B, York DM, Sherman W. Free Energy Methods in Drug Discovery—Introduction. In: *Free Energy Methods in Drug Discovery: Current State and Future Directions* American Chemical Society; 2021.p. 1–38. doi: [10.1021/bk-2021-1397.ch001](https://doi.org/10.1021/bk-2021-1397.ch001).
- [8] **Crooks GE**. Path-ensemble averages in systems driven far from equilibrium. *Phys Rev E*. 2000 3; 61(3):2361. doi: [10.1103/PhysRevE.61.2361](https://doi.org/10.1103/PhysRevE.61.2361).
- [9] **Cui Q**, Elstner M. “Multi-Scale” QM/MM Methods with Self-Consistent-Charge Density-Functional-Tight-Binding (SCC-DFTB). *Chall Adv Comput Che*. 2009; 7:173–196. doi: [10.1007/978-1-4020-9956-4_7](https://doi.org/10.1007/978-1-4020-9956-4_7).
- [10] **Devereux C**, Smith JS, Huddleston KK, Barros K, Zubatyuk R, Isayev O, Roitberg AE. Extending the Applicability of the ANI Deep Learning Molecular Potential to Sulfur and Halogens. *Journal of chemical theory and computation*. 2020 7; 16(7):4192–4202. doi: [10.1021/acs.jctc.0c00121](https://doi.org/10.1021/acs.jctc.0c00121).
- [11] **Duarte Ramos Matos G**, Kyu DY, Loeffler HH, Chodera JD, Shirts MR, Mobley DL. Approaches for Calculating Solvation Free Energies and Enthalpies Demonstrated with an Update of the FreeSolv Database. *J Chem Eng Data*. 2017 5; 62(5):1559–1569. doi: [10.1101/104281](https://doi.org/10.1101/104281).
- [12] **Eastman P**, Galvelis R, Peláez RP, Abreu CRA, Farr SE, Gallicchio E, Gorenko A, Henry MM, Hu F, Huang J, Krämer A, Michel J, Mitchell JA, Pande VS, Rodrigues JP, Rodriguez-Guerra J, Simmonett AC, Swails J, Zhang I, Chodera JD, et al. OpenMM 8: Molecular Dynamics Simulation with Machine Learning Potentials. *arxiv*. 2023 10; doi: [10.48550/ARXIV.2310.03121](https://doi.org/10.48550/ARXIV.2310.03121).
- [13] **Essmann U**, Perera L, Berkowitz ML, Darden T, Lee H, Pedersen LG. A smooth particle mesh Ewald method. *J Chem Phys*. 1995 Aug; 103(19):8577. doi: [10.1063/1.470117](https://doi.org/10.1063/1.470117).
- [14] **Galvelis R**, Varela-Rial A, Doerr S, Fino R, Eastman P, Markland TE, Chodera JD, De Fabritiis G. NNP/MM: Accelerating Molecular Dynamics Simulations with Machine Learning Potentials and Molecular Mechanics. *Journal of Chemical Information and Modeling*. 2023; 63(18):5701–5708. <https://doi.org/10.1021/acs.jcim.3c00773>, doi: [10.1021/acs.jcim.3c00773](https://doi.org/10.1021/acs.jcim.3c00773).

- [15] **Gao X**, Ramezanghorbani F, Isayev O, Smith JS, Roitberg AE. TorchANI: A Free and Open Source PyTorch-Based Deep Learning Implementation of the ANI Neural Network Potentials. *Journal of chemical information and modeling*. 2020 7; 60(7):3408–3415. doi: [10.1021/acs.jcim.0c00451](https://doi.org/10.1021/acs.jcim.0c00451).
- [16] **Heimdal J**, Ryde U. Convergence of QM/MM free-energy perturbations based on molecular-mechanics or semiempirical simulations. *Phys Chem Chem Phys*. 2012 8; 14(36):12592–12604. doi: [10.1039/C2CP41005B](https://doi.org/10.1039/C2CP41005B).
- [17] **Hudson PS**, Woodcock HL, Boresch S. Use of Nonequilibrium Work Methods to Compute Free Energy Differences between Molecular Mechanical and Quantum Mechanical Representations of Molecular Systems. *J Phys Chem Lett*. 2015 12; 6(23):4850–4856.
- [18] **Jarzynski C**. Nonequilibrium Equality for Free Energy Differences. *Phys Rev Lett*. 1997 4; 78(14):2690. doi: [10.1103/PhysRevLett.78.2690](https://doi.org/10.1103/PhysRevLett.78.2690).
- [19] **John Chodera**, Andrea Rizzi, Levi Naden, Kyle Beauchamp, Patrick Grinaway, Josh Fass, Iván Pulido, Alex Wade, Mike Henry, Gregory A Ross, Andreas Krämer, Hannah Bruce Macdonald, jaimergp, Bas Rustenburg, David W H Swenson, Ivy Zhang, Andy Simmonett, Mark J Williamson, hb0402, Jake Fennick, et al., choderalab/openmmtools: 0.23.1. Zenodo; 2023. <https://zenodo.org/records/8102771>, doi: [10.5281/ZENODO.8102771](https://doi.org/10.5281/ZENODO.8102771).
- [20] **Jorgensen WL**, Chandrasekhar J, Madura JD. Comparison of simple potential functions for simulating liquid water. *J Chem Phys*. 1983; 79(2):926–935. doi: [10.1063/1.445869](https://doi.org/10.1063/1.445869).
- [21] **Karwounopoulos J**, Kaupang Å, Wieder M, Boresch S. Calculations of Absolute Solvation Free Energies with Transformato – Application to the FreeSolv Database Using the CGenFF Force Field. *J Chem Theory Comput*. 2023 9; 19(17):5988–5998. doi: [10.1021/acs.jctc.3c00691](https://doi.org/10.1021/acs.jctc.3c00691).
- [22] **Karwounopoulos J**, Wieder M, Boresch S. Relative binding free energy calculations with transformato: A molecular dynamics engine-independent tool. *Front Mol Biosci*. 2022 9; 9:850. doi: [10.3389/fmolb.2022.954638](https://doi.org/10.3389/fmolb.2022.954638).
- [23] **Kearns FL**, Hudson PS, Woodcock HL, Boresch S. Computing converged free energy differences between levels of theory via nonequilibrium work methods: Challenges and opportunities. *J Comput Chem*. 2017 6; 38(16):1376–1388. doi: [10.1002/jcc.24706](https://doi.org/10.1002/jcc.24706).
- [24] **Kulichenko M**, Smith JS, Nebgen B, Li YW, Fedik N, Boldyrev AI, Lubbers N, Barros K, Tretiak S. The Rise of Neural Networks for Materials and Chemical Dynamics. *J Phys Chem Lett*. 2021 7; 12(26):6227–6243. doi: [10.1021/acs.jpclett.1c01357](https://doi.org/10.1021/acs.jpclett.1c01357).
- [25] **Lier B**, Poliak P, Marquetand P, Westermayr J, Oostenbrink C. BuRNN: Buffer Region Neural Network Approach for Polarizable-Embedding Neural Network/Molecular Mechanics Simulations. *J Phys Chem Lett*. 2022 5; 13(17):3812–3818. doi: [10.1021/acs.jpclett.2c00654](https://doi.org/10.1021/acs.jpclett.2c00654).
- [26] **Lu X**, Fang D, Ito S, Okamoto Y, Ovchinnikov V, Cui Q. QM/MM free energy simulations: recent progress and challenges. *Mol Simul*. 2016 9; 42(13):1056–1078. doi: [10.1080/08927022.2015.1132317](https://doi.org/10.1080/08927022.2015.1132317).
- [27] **Luzhkov V**, Warshel A. Microscopic models for quantum mechanical calculations of chemical processes in solutions: LD/AMPAC and SCAAS/AMPAC calculations of solvation energies. *J Comput Chem*. 1992 3; 13(2):199–213. doi: [10.1002/jcc.540130212](https://doi.org/10.1002/jcc.540130212).

- [28] **Mey ASJS**, Allen BK, McDonald HEB, Chodera JD, Hahn DF, Kuhn M, Michel J, Mobley DL, Naden LN, Prasad S, Rizzi A, Scheen J, Shirts MR, Tresadern G, Xu H. Best Practices for Alchemical Free Energy Calculations. *LiveCoMS*. 2020 12; 2(1):18378. doi: [10.33011/LIVECOMS.2.1.18378](https://doi.org/10.33011/LIVECOMS.2.1.18378).
- [29] **Miyamoto S**, Kollman PA. Settle: An analytical version of the SHAKE and RATTLE algorithm for rigid water models. *J Comput Chem*. 1992 Oct; 13(8):952–962. doi: [10.1002/jcc.540130805](https://doi.org/10.1002/jcc.540130805).
- [30] **Mobley DL**, Bayly CI, Cooper MD, Shirts MR, Dill KA. Small molecule hydration free energies in explicit solvent: An extensive test of fixed-charge atomistic simulations. *J Chem Theory Comput*. 2009 2; 5(2):350–358. doi: [10.1021/ct800409d](https://doi.org/10.1021/ct800409d).
- [31] **Mobley DL**, Guthrie JP. FreeSolv: a database of experimental and calculated hydration free energies, with input files. *J Comput Aided Mol Des*. 2014 Jun; 28(7):711–720. doi: [10.1007/s10822-014-9747-x](https://doi.org/10.1007/s10822-014-9747-x).
- [32] **Parzen E**. On Estimation of a Probability Density Function and Mode. *The Annals of Mathematical Statistics*. 1962 9; 33(3):1065–1076. <http://projecteuclid.org/euclid.aoms/1177704472>, doi: [10.1214/aoms/1177704472](https://doi.org/10.1214/aoms/1177704472).
- [33] **Reddy MR**, Singh UC, Erion MD. Use of a QM/MM-based FEP method to evaluate the anomalous hydration behavior of simple alkyl amines and amides: Application to the design of FBPase inhibitors for the treatment of type-2 diabetes. *J Am Chem Soc*. 2011 6; 133(21):8059–8061. doi: [10.1021/ja201637q](https://doi.org/10.1021/ja201637q).
- [34] **Rosenblatt M**. Remarks on Some Nonparametric Estimates of a Density Function. *The Annals of Mathematical Statistics*. 1956 9; 27(3):832–837. doi: [10.1214/aoms/1177728190](https://doi.org/10.1214/aoms/1177728190).
- [35] **Rufa DA**, Macdonald HEB, Fass J, Wieder M, Grinaway PB, Roitberg AE, Isayev O, Chodera JD. Towards chemical accuracy for alchemical free energy calculations with hybrid physics-based machine learning / molecular mechanics potentials. *bioRxiv*. 2020 Jul; doi: [10.1101/2020.07.29.227959](https://doi.org/10.1101/2020.07.29.227959).
- [36] **Schindler CEM**, Baumann H, Blum A, Böse D, Buchstaller HP, Burgdorf L, Cappel D, Chekler E, Czodrowski P, Dorsch D, Eguida MKI, Follows B, Fuchß T, Grädler U, Gunera J, Johnson T, Lebrun CJ, Karra S, Klein M, Knehans T, et al. Large-Scale Assessment of Binding Free Energy Calculations in Active Drug Discovery Projects. *J Chem Inf Model*. 2020 Aug; 60(11):5457–5474. doi: [10.1021/acs.jcim.0c00900](https://doi.org/10.1021/acs.jcim.0c00900).
- [37] **Schöller A**, Woodcock HL, Boresch S. Exploring Routes to Enhance the Calculation of Free Energy Differences via Non-Equilibrium Work SQM/MM Switching Simulations Using Hybrid Charge Intermediates between MM and SQM Levels of Theory or Non-Linear Switching Schemes. *Molecules*. 2023 5; 28(10):4006. doi: [10.3390/molecules28104006](https://doi.org/10.3390/molecules28104006).
- [38] **Shirts MR**, Chodera JD. Statistically optimal analysis of samples from multiple equilibrium states. *J Chem Phys*. 2008 Sep; 129(12):124105. doi: [10.1063/1.2978177](https://doi.org/10.1063/1.2978177).
- [39] **Smith JS**, Isayev O, Roitberg AE. ANI-1, A data set of 20 million calculated off-equilibrium conformations for organic molecules. *Scientific Data* 2017 4:1. 2017 12; 4(1):1–8. doi: [10.1038/sdata.2017.193](https://doi.org/10.1038/sdata.2017.193).
- [40] **Tkaczyk S**, Karwounopoulos J, Schöller A, Woodcock HL, Langer T, Boresch S, Wieder M. Using Neural Network Potentials to efficiently calculate indirect free energy estimates. *ChemRxiv*. 2023 5; Cambridge(Cambridge Open Engage). doi: [10.26434/chemrxiv-2023-qq206](https://doi.org/10.26434/chemrxiv-2023-qq206).
- [41] **Valiev M**, Bylaska EJ, Dupuis M, Tratnyek PG. Combined Quantum Mechanical and Molecular Mechanics Studies of the Electron-Transfer Reactions Involving Carbon Tetrachloride in Solution. *J Phys Chem A*. 2008 3; 112(12):2713–2720. doi: [10.1021/jp7104709](https://doi.org/10.1021/jp7104709).

- [42] **Vanommeslaeghe K**, MacKerell AD. Automation of the CHARMM general force field (CGenFF) I: Bond perception and atom typing. *J Chem Inf Model*. 2012 Dec; 52(12):3144–3154. doi: 10.1021/Ci300363C.
- [43] **Vanommeslaeghe K**, Raman EP, MacKerell AD. Automation of the CHARMM General Force Field (CGenFF) II: Assignment of Bonded Parameters and Partial Atomic Charges. *J Chem Inf Model*. 2012 Dec; 52(12):3155–3168. doi: 10.1021/ci3003649.
- [44] **Wang M**, Mei Y, Ryde U. Predicting Relative Binding Affinity Using Nonequilibrium QM/MM Simulations. *Journal of Chemical Theory and Computation*. 2018; 14(12):6613–6622. <https://doi.org/10.1021/acs.jctc.8b00685>, doi: 10.1021/acs.jctc.8b00685.
- [45] **Wieder M**, Fleck M, Braunsfeld B, Boresch S. Alchemical free energy simulations without speed limits. A generic framework to calculate free energy differences independent of the underlying molecular dynamics program. *J Comput Chem*. 2022 Jun; 43(17):1151–1160. doi: 10.1002/jcc.26877.
- [46] **Zheng L**, Chen M, Yang W. Random walk in orthogonal space to achieve efficient free-energy simulation of complex systems. *Proc Natl Acad Sci U S A*. 2008 12; 105(51):20227–20232. doi: 10.1073/pnas.0810631106.
- [47] **Zinovjev K**. Electrostatic Embedding of Machine Learning Potentials. *J Chem Theory Comput*. 2023 3; 19(6):1888–1897. doi: 10.1021/acs.jctc.2c00914.

3 Discussion

3.1 Summary

This thesis describes several enhancements to the original `transformato` implementation.⁶³ First (Section 2.1), we extended `transformato` to be able to calculate RBFE and obtained promising results when comparing to experimental results as well as to other computational studies.^{30,72–74} For setting up the systems, we utilized the CHARMM-GUI web interface.^{64,65} Building on CHARMM-GUI’s tremendous capabilities means that FES carried out by `transformato` are not limited to protein–ligand systems. Within the limits of the CC/SAI approach to carry out the alchemical transformation, one can use `transformato` to set up RBFE calculations for any system CHARMM-GUI can handle. In particular, one could carry out RBFE calculations involving membrane proteins without any change to the `transformato` source code.

Next (Section 2.2), we extended `transformato` to the calculation of ASFE. As a practical application, we calculated ASFES for almost all molecules in the FreeSolv database.⁷⁷ Overall, the agreement with the experimental results is good, demonstrating the quality of the CGenFF force field. However, we encountered functional groups, where the CGenFF force field performed poorly. E.g., for amines, it exhibited suboptimal performance, suggesting potential areas for force field refinement. Using the CHARMM-GUI web interface to set up hundreds of solvated systems would have been cumbersome and would have put an unnecessary load on the server. Therefore, we developed a small Python tool, called `macha`, which can be run locally and served as a replacement for CHARMM-GUI (for more information on `macha` see Section 3.2). It bears pointing out that while `transformato` profits from CHARMM-GUI’s capabilities, it does not depend on it. If a user prefers a different setup tool, combining it with `transformato` would require only minor adaptations. A particularity of the SAI approach is its dependence on the number of atoms that need to be "annihilated"; i.e., the computational cost of an annihilation calculation when using `transformato` depends on the number of atoms of the solute. Calculating ASFES of small molecules requires fewer intermediate states (and less computational time) than those of larger molecules. Since the calculations are trivial to parallelize, this does not necessarily increase the wall time of the full ASFE calculation. In addition, the larger computational effort is spent only when it is needed, rather than the "one-size-fits-all" approach used in traditional approaches (i.e., using a fixed number of λ states independent of the solute size).

When comparing the ASFE results obtained with CGenFF, we observed compounds

that performed particularly poorly compared to the experimental values. For that reason, we employed an indirect pathway to correct the intramolecular force field description with a more accurate ML method (Section 2.3). Since the end states in `transformato` are physical molecules without any dummy atoms, end state correction to a higher level can be applied straightforwardly. An automated workflow was developed and applied to calculate end state corrections for poorly performing compounds. The starting points were ASFes obtained with both the CGenFF and the OpenFF⁸¹ force fields. In a few instances, the end state corrections notably improved the alignment with experimental data. However, overall, we observed only a slight improvement in RMSE, MAE, and correlation coefficients when considering all molecules.

3.2 Further/ongoing enhancements of `transformato`

In addition to the main results reported in Chapter 2, several smaller side projects aimed at enhancing the usability and utility of `transformato` have been carried out.

Accessibility and Usability: As mentioned, `transformato` is an open-source project, with its source code publicly available on GitHub¹. To increase usability, the package is additionally available on Conda-Forge. We have also created, and are continuing to improve, the documentation² to assist non-expert users in utilizing `transformato` for setting up FES.

Introducing `macha`: While having the full support of CHARMM-GUI for generating initial structures that can be directly used with `transformato` is beneficial, it can become impractical when dealing with extensive data sets of hundreds of compounds (e.g., the FreeSolv database as described in Section 2.2). To address this, we developed a small Python utility, called `macha`³, to create solvated systems. `macha` uses established Python libraries, like `Open Babel`⁸² and `parmed`⁸³ accompanied by CHARMM scripts to create systems from PDB files or SMILES strings. With the help of CHARMM scripts in analogy to the ones provided by CHARMM-GUI,^{64,65} the systems were solvated and preprocessed using CHARMM. The output is used as starting structures for the respective end states. The utility provides experimental support for carrying out analogous setup steps in the context of RBFE calculations.

¹<https://github.com/wiederm/transformato>

²<https://wiederm.github.io/transformato/>

³<https://github.com/akaupang/macha>

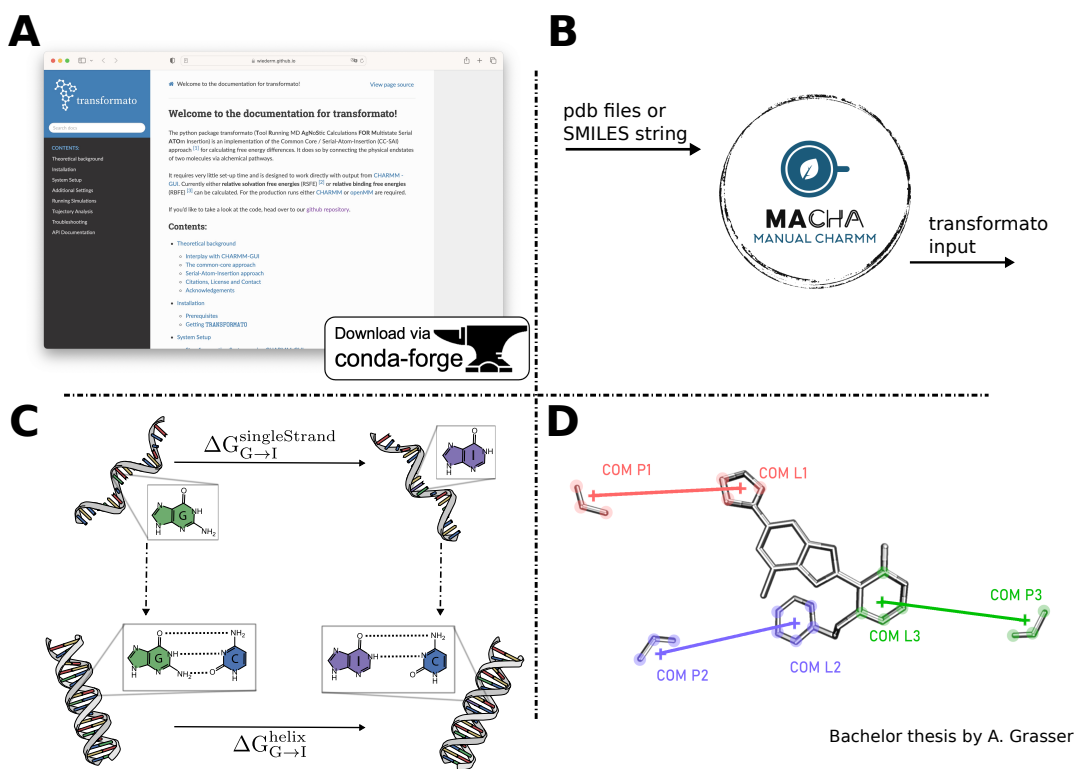


Figure 3.1 Additional enhancements have been introduced to make `transformato` more accessible and versatile: **A)** A comprehensive documentation was written to assist non-expert users, along with the convenience of downloading `transformato` from Conda-Forge into any Conda environment. **B)** In addition, a separate Python tool, called `macha`, is available to facilitate the automated generation of initial input for `transformato`. **C)** `transformato` has been expanded to calculate binding free energies between a pair of nucleotides in RNA double helices. **D)** We have added an option for restraining the ligand to a particular binding site.

Auxiliary Restraints for RBFE: If the CC in a RBFE calculation is considerably smaller than either of the end states, there is the distinct possibility that the CC might move out of the binding site or adopt a pose which is different from that of either end state. Although we have so far not observed such behavior, we want to be prepared for such scenarios. In the course of a bachelor thesis by Alexander Grasser, it is now possible to restrain a ligand to its original starting position as an additional option within the `transformato` framework. The option is already part of the master branch of `transformato`, but has not been tested extensively yet.

Binding free energy differences in RNA double strands: In the course of a collaboration (Prof. Ivo Hofacker, Institute of Theoretical Chemistry, University of Vienna) it became of interest to calculate RBFEs between two nucleotides in an RNA double helix. The necessary steps are analogous to protein–ligand RBFE simulations, the differences being that the single strand solvated in a waterbox corresponds to the solvated ligand in standard RBFE simulations, and the double helix to the ligand bound to the protein. By altering a nucleobase, we can determine the specific nucleobase’s free energy difference between its unbound state (single strand) and its bound state (double strand). The en-

hancement for calculating RBFs for duplex systems is currently available as a separate branch and will be integrated into `transformato` soon.⁴

Support for Lennard-Jones Particle Mesh Ewald method: As discussed briefly in the Introduction, LJ interactions are truncated after a certain distance. If done so, long-range dispersion is neglected, which can affect the prediction of FES. In the context of ASFE calculations, it is possible to use an isotropic, analytical term to calculate the dispersion correction, the so-called long-range correction (LRC), in a post-processing step. This was done for the results reported in Section 2.2. Recently, MD engines like GROMACS,⁸⁴ CHARMM,⁶⁸ and OpenMM⁶⁹ have implemented the Lennard-Jones Particle Mesh Ewald method (LJPME).⁸⁵ Here, the PME method is used to compute long-range interaction for Coulomb and LJ interactions directly and, aside from periodicity, is free from imposing a cut-off. Since the workflow for `transformato` is totally independent of the MD run, it is trivial to use LJPME to compute LJ interactions during the OpenMM simulation of the individual intermediate states. We compared results from ASFE calculations for 21 molecules of our validation set (see Section 2.2) calculated with `transformato` once with the analytical post-processing and once when using LJPME. As one sees in Figure 3.2 results obtained with LRC and LJPME align almost perfectly (RMSE: 0.11 kcal/mol, MAE 0.07 kcal/mol).

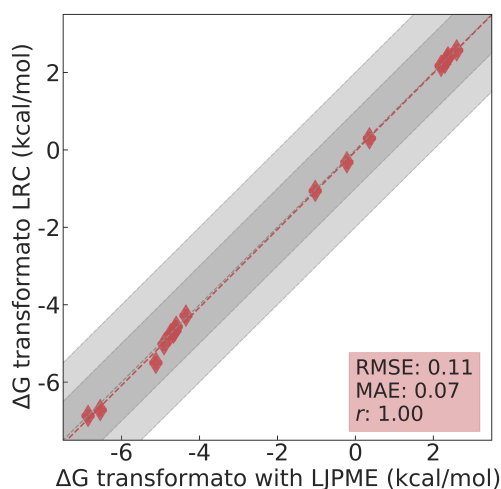


Figure 3.2 ASFEs for the subset of 21 molecules used to validate the `transformato` approach to compute ASFEs in Section 2.2 (see Fig. S2 in Appendix B). ASFEs calculated including LJPME (x-axis) are plotted against ASFEs obtained after applying a post-processing LRC (y-axis).

⁴<https://github.com/wiederm/transformato/pull/98>

3.3 Concluding discussion and outlook

With `transformato`, an open-source Python tool was introduced that compartmentalizes the setup of the alchemical transformation, carrying out the MD simulations for sampling, and post-processing to compute the free energy differences of interest. By using the SAI approach, no special purpose code in the MD program is required, making `transformato` effectively independent of the underlying MD engine. Specifically, we demonstrated that CHARMM as well as OpenMM can be used for the underlying MD simulation (Section 2.1). It would be straightforward to combine `transformato` with MD programs that use CHARMM-style topology, parameter, and PSF formats, e.g., NAMD.⁸⁶ Adding support for programs using a different file format for this information (e.g., AMBER or GROMACS), would require either conversion tools or I/O routines for the respective formats, but no fundamental changes in `transformato`’s workflow and program logic.

In this work, I described how the initial version of `transformato`, tested by computing RSFE,⁶³ was enhanced to be able to calculate numerous free energy differences in a (mostly) automated manner. The setup of RBEF calculations is analogous to that of RSFE. The handling of the alchemical mutation is identical, only the environments are different (ligand and complex in RBEF in contrast to ligand and gas phase in RSFE). Thus, only adaptations to handle the complex correctly were needed (Section 2.1).

To gain insights into the accuracy of the CGenFF force field, ASFE calculations were incorporated into `transformato`. Additionally, `macha` was developed to facilitate the creation of the initial configurations needed as the input for `transformato`. Analyzing the ASFE for 621 compounds in the FreeSolv database and comparing the results to experimental data allowed us to identify individual molecules and groups of compounds where the force field exhibited suboptimal performance (Section 2.2).

Several recent developments further highlight the flexibility of `transformato` and its underlying approach. As just mentioned in Section 3.2, it was straightforward to adapt `transformato`’s CC/SAI approach to calculate RBEF in RNA strands. If the underlying MD program supports HMR, the increased integration time step of up to 4 fs accelerates the FES considerably (Section 2.1). A novel treatment of LJ interactions, LJPME, also can be used after trivial changes to `transformato`, provided the MD program used supports the method.

Obviously, `transformato` does have drawbacks and limitations. Like many other FES tools, it does not support scaffold hopping. The CC/SAI approach works best for a congeneric series of ligands. When ligands become too dissimilar, the CC is small, and many steps are needed to reach it. This is not only computationally inefficient, but a small CC may also not remain stable in the binding site, making the calculations unreliable. Currently, `transformato` does not support any enhanced sampling method. In principle, adding replica exchange steps between the otherwise independent MD simulations at regular intervals would be straightforward, but the necessary code has to

be written. Also, only force fields in CHARMM format, such as CGenFF, can be used easily. AMBER force fields, such as GAFF, either would have to be converted, or explicit support (reading and writing of, as well as rewriting in the native file format) would have to be programmed.

Finally, we explored the integration of a machine learning potential and an indirect free energy approach to reduce the reliance on force fields and to improve the quality of solvation free energy predictions (Section 2.3). The coupling of `transformato` with OpenMM’s MM/ML interface made this straightforward, and the calculations are fast. Disappointingly, however, no statistically significant improvement was observed. The limiting factor seems to be the coupling between MM and ML. It corresponds to what is called mechanical embedding in QM/MM. In this approach, solute-solvent interactions remain classical. Improving the quality of the predicted ASFEs most likely requires moving beyond mechanical embedding. In fact, it is tempting to describe the solute and its surrounding water entirely using the ANI-2x (or another ML) potential. Initial tests suggest that the computational cost to simulate small solute-solvent boxes fully at the ML level of theory is affordable. This would make it possible to compute ASFEs either directly at the high level of theory or to switch the solute-solvent system *completely* from the MM to the ML level of theory.

Bibliography

- [1] Z. Cournia et al., in *Free Energy Methods in Drug Discovery: Current State and Future Directions*, American Chemical Society, 2021, pp. 1–38.
- [2] D. H. De Jong et al., *J. Comput. Chem.*, 2011, **32**, 1919–1928.
- [3] V. Ovchinnikov, K. Nam and M. Karplus, *J. Phys. Chem. B*, 2016, **120**, 8457–8472.
- [4] C. E. M. Schindler et al., *J. Chem. Inf. Model.*, 2020, **60**, 5457–5474.
- [5] J. D. Chodera et al., *Curr. Opin. Struct. Biol.*, 2011, **21**, 150–160.
- [6] Z. Cournia, B. Allen and W. Sherman, *J. Chem. Inf. Model.*, 2017, **57**, 2911–2937.
- [7] S. Lifson and A. Waeshel, *J. Chem. Phys.*, 1968, **49**, 5116–5129.
- [8] J. Huang et al., *Nat. Methods*, 2016, **14**, 71–73.
- [9] N. Schmid et al., *Eur. Biophys. J.*, 2011, **40**, 843–856.
- [10] C. Tian et al., *J. Chem. Theory Comput.*, 2020, **16**, 528–552.
- [11] A. D. Mackerell, M. Feig and C. L. Brooks, *J. Comput. Chem.*, 2004, **25**, 1400–1415.
- [12] P. P. Ewald, *Ann. Phys.*, 1921, **369**, 253–287.
- [13] T. Darden, D. York and L. Pedersen, *J. Chem. Phys.*, 1993, **98**, 10089–10092.
- [14] U. Essmann et al., *J. Chem. Phys.*, 1995, **103**, 8577.
- [15] A. Pohorille, C. Jarzynski and C. Chipot, *J. Phys. Chem. B*, 2010, **114**, 10235–10253.
- [16] R. W. Zwanzig, E. G. Rauh and R. J. Thorn, *J. Chem. Phys.*, 1954, **22**, 1420–1426.
- [17] C. H. Bennett, *J. Comput. Phys.*, 1976, **22**, 245–268.
- [18] M. R. Shirts and J. D. Chodera, *J. Chem. Phys.*, 2008, **129**, 124105.
- [19] J. G. Kirkwood, *J. Chem. Phys.*, 1935, **3**, 300–313.
- [20] S. Bruckner and S. Boresch, *J. Comput. Chem.*, 2011, **32**, 1320–1333.
- [21] M. Fleck, M. Wieder and S. Boresch, *J. Chem. Theory Comput.*, 2021, **17**, 4403–4419.
- [22] P. V. Klimovich, M. R. Shirts and D. L. Mobley, *J. Comput.-Aided Mol. Des.*, 2015, **29**, 397–411.
- [23] A. Villa and A. E. Mark, *J. Comput. Chem.*, 2002, **23**, 548–553.

- [24] M. R. Shirts, J. W. Pitner, W. C. Swope and V. S. Pande, *J. Chem. Phys.*, 2003, **119**, 5740–5761.
- [25] Y. Deng and B. Roux, *J. Phys. Chem. B*, 2004, **108**, 16567–16576.
- [26] A. S. J. S. Mey et al., *LiveCoMS*, 2020, **2**, 18378.
- [27] W. L. Jorgensen and C. Ravimohan, *J. Chem. Phys.*, 1985, **83**, 3050–3054.
- [28] M. M. Ghahremanpour, J. Tirado-Rives and W. L. Jorgensen, *J. Phys. Chem. B*, 2022, **126**, 5896–5907.
- [29] B. L. Tembre and J. A. McCammon, *Computers & Chemistry*, 1984, **8**, 281–283.
- [30] L. Wang et al., *J. Am. Chem. Soc.*, 2015, **137**, 2695–2703.
- [31] T. Simonson, *Mol. Phys.*, 1993, **80**, 441–447.
- [32] T. C. Beutler et al., *Chemical Physics Letters*, 1994, **222**, 529–539.
- [33] T. Steinbrecher, D. L. Mobley and D. A. Case, *J. Chem. Phys.*, 2007, **127**, 214108.
- [34] M. Zacharias, T. P. Straatsma and J. A. McCammon, *J. Chem. Phys.*, 1994, **100**, 9025–9031.
- [35] T. Steinbrecher, I. Joung and D. A. Case, *J. Comput. Chem.*, 2011, **32**, 3253–3263.
- [36] T. S. Lee et al., *J. Chem. Theory Comput.*, 2020, **16**, 5512–5525.
- [37] S. Boresch and S. Bruckner, *J. Comput. Chem.*, 2011, **32**, 2449–2458.
- [38] D. A. Kofke and P. T. Cummings, *Mol. Phys.*, 1997, **92**, 973–996.
- [39] A. Warshel and M. Levitt, *J. Mol. Biol.*, 1976, **103**, 227–249.
- [40] V. Luzhkov and A. Warshel, *J. Comput. Chem.*, 1992, **13**, 199–213.
- [41] X. Lu et al., *Mol. Simul.*, 2016, **42**, 1056–1078.
- [42] M. J. Field, P. A. Bash and M. Karplus, *J. Comput. Chem.*, 1990, **11**, 700–733.
- [43] Q. Cui and M. Elstner, *Chall. Adv. Comput. Che.*, 2009, **7**, 173–196.
- [44] M. R. Reddy, U. C. Singh and M. D. Erion, *J. Am. Chem. Soc.*, 2011, **133**, 8059–8061.
- [45] M. Valiev, E. J. Bylaska, M. Dupuis and P. G. Tratnyek, *J. Phys. Chem. A*, 2008, **112**, 2713–2720.
- [46] L. Zheng, M. Chen and W. Yang, *Proc. Natl. Acad. Sci. U. S. A.*, 2008, **105**, 20227–20232.
- [47] J. Heimdal and U. Ryde, *Phys. Chem. Chem. Phys.*, 2012, **14**, 12592–12604.
- [48] Q. Cui, T. Pal and L. Xie, *J. Phys. Chem. B*, 2021, **125**, 689–702.
- [49] T. J. Giese and D. M. York, *J. Chem. Theory Comput.*, 2019, **15**, 5543–5562.
- [50] F. L. Kearns, P. S. Hudson, H. L. Woodcock and S. Boresch, *J. Comput. Chem.*, 2017, **38**, 1376–1388.

- [51] P. S. Hudson, H. L. Woodcock and S. Boresch, *J. Phys. Chem. Lett.*, 2015, **6**, 4850–4856.
- [52] C. Cave-Ayland, C. K. Skylaris and J. W. Essex, *J. Phys. Chem. B*, 2015, **119**, 1017–1025.
- [53] R. H. Wood, W. C. Mühlbauer and P. T. Thompson, *J. Phys. Chem.*, 1991, **95**, 6670.
- [54] M. Kulichenko et al., *J. Phys. Chem. Lett.*, 2021, **12**, 6227–6243.
- [55] J. S. Smith, O. Isayev and A. E. Roitberg, *Scientific Data* 2017 4:1, 2017, **4**, 1–8.
- [56] C. Devereux et al., *J. Chem. Theory Comput.*, 2020, **16**, 4192–4202.
- [57] S. Park, H. Han, H. Kim and S. Choi, *Chemistry – An Asian Journal*, 2022, **17**, e202200203.
- [58] F. Han et al., *J. Chem. Inf. Model.*, 2023, **63**, 6608–6618.
- [59] D. A. Rufa et al., *bioRxiv*, 2020, DOI: 10.1101/2020.07.29.227959.
- [60] E. Akkus, O. Tayfuroglu, M. Yildiz and A. Kocak, *J. Chem. Inf. Model.*, 2022, **62**, 4095–4106.
- [61] S. Tkaczyk et al., *ChemRxiv*, 2023, **Cambridge**, DOI: 10.26434/chemrxiv-2023-qq206.
- [62] R. Galvelis et al., *J. Chem. Inf. Model.*, 2023, **63**, 5701–5708.
- [63] M. Wieder, M. Fleck, B. Braunsfeld and S. Boresch, *J. Comput. Chem.*, 2022, **43**, 1151–1160.
- [64] S. Jo, T. Kim, V. G. Iyer and W. Im, *J. Comput. Chem.*, 2008, **29**, 1859–1865.
- [65] J. Lee et al., *J. Chem. Theory Comput.*, 2016, **12**, 405–413.
- [66] T. Kluyver et al., *International Conference on Electronic Publishing*, 2016, 87–90.
- [67] B. R. Brooks et al., *J. Comput. Chem.*, 1983, **4**, 187–217.
- [68] B. R. Brooks et al., *J. Comput. Chem.*, 2009, **30**, 1545–1614.
- [69] P. Eastman et al., *PLOS Comput. Biol.*, 2017, **13**, 1005659.
- [70] P. Eastman et al., 2023.
- [71] Y. Hu et al., *J. Comput. Aided Mol. Des.*, 2016, **30**, 533–539.
- [72] L. F. Song et al., *J. Chem. Inf. Model.*, 2019, **59**, 3128–3135.
- [73] V. Gapsys, S. Michielssens, D. Seeliger and B. L. de Groot, *J. Comput. Chem.*, 2015, **36**, 348–354.
- [74] X. He et al., *ACS Omega*, 2020, **5**, 4611–4619.
- [75] C. W. Hopkins, S. Le Grand, R. C. Walker and A. E. Roitberg, *J. Chem. Theory Comput.*, 2015, **11**, 1864–1874.
- [76] H. Zhang et al., *J. Chem. Inf. Model*, 2021, **61**, 56.

- [77] D. L. Mobley and J. P. Guthrie, *J. Comput. Aided Mol. Des.*, 2014, **28**, 711–720.
- [78] K. Vanommeslaeghe and A. D. MacKerell, *J. Chem. Inf. Model.*, 2012, **52**, 3144–3154.
- [79] K. Vanommeslaeghe, E. P. Raman and A. D. MacKerell, *J. Chem. Inf. Model.*, 2012, **52**, 3155–3168.
- [80] J. Wang et al., *J. Comput. Chem.*, 2004, **25**, 1157–1174.
- [81] S. Boothroyd et al., *J. Chem. Theory Comput.*, 2023, **19**, 3251–3275.
- [82] N. M. O’Boyle et al., *J. Cheminf.*, 2011, **3**, 1–14.
- [83] M. R. Shirts et al., *J. Comput. Aid. Mol. Des.*, 2017, **31**, 147–161.
- [84] D. Van Der Spoel et al., *J. Comput. Chem.*, 2005, **26**, 1701–1718.
- [85] C. L. Wennberg, T. Murtola, B. Hess and E. Lindahl, *J. Chem. Theory Comput.*, 2013, **9**, 3527–3537.
- [86] J. C. Phillips et al., *J. Chem. Phys.*, 2020, **153**, 44130.

Appendix A:

Supplementary information for

Relative binding free energy calculations with transformato: A molecular dynamics engine-independent tool

Front. Mol. Biosci. 9:954638 (2022)

Authors: Johannes Karwounopoulos, Marcus Wieder and Stefan Boresch

Contributions: Conceptualization: JK, MW, and SB; methodology: JK, MW, and SB; software: JK and MW; investigation: JK, MW, and SB; writing: JK, MW, and SB; supervision: MW and SB

Supplementary Material

This document contains supplementary figures and tables. In addition, a zip-archive is available, containing our raw results in machine-readable form, as well as the Jupyter notebook with which additional statistical analyses were carried out.

Contents of the zip-archive:

- `summary_ddG.csv`
- `summary_dG.csv`
- `RBFE_transformato_workbook.ipynb`

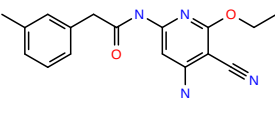
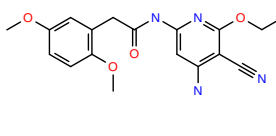
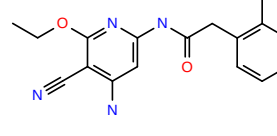
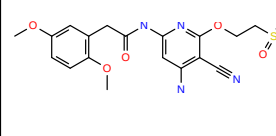
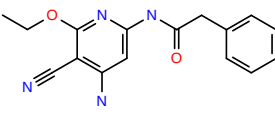
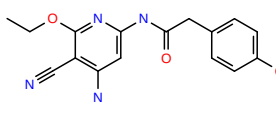
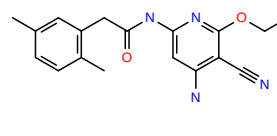
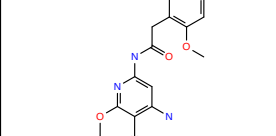
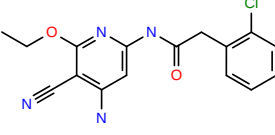
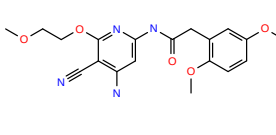
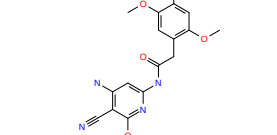
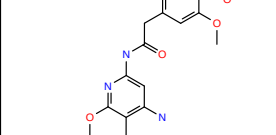
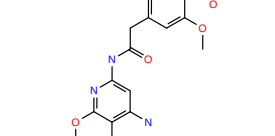
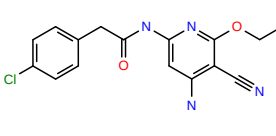
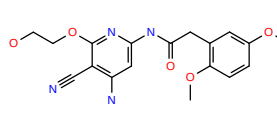
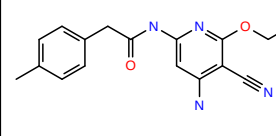
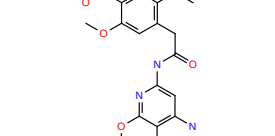
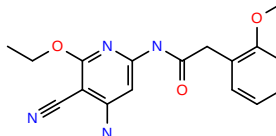
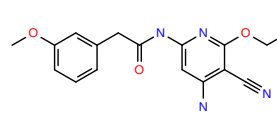
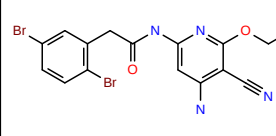
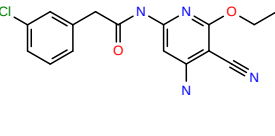
			
title 18629-1	title 18634-1	title 18628-1	title 18660-1
			
title 18624-1	title 18633-1	title 18635-1	title 17124-1
			
title 18625-1	title 18659-1	title 18637-1	title 18638-1
			
title 18652-1	title 18627-1	title 18658-1	title 18630-1
			
title 18639-1	title 18631-1	title 18632-1	title 18636-1
			
title 18626-1			

Figure S1. Structures for the ligands investigated for the JNK1 system with their corresponding abbreviations (Wang et al., 2015). Different CCs were used depending on the mutation studied.

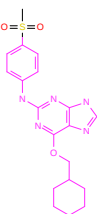
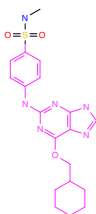
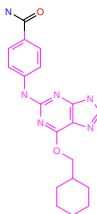
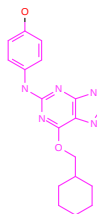
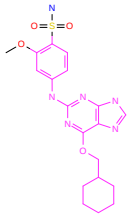
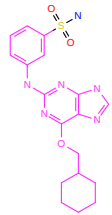
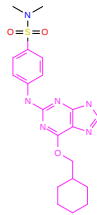
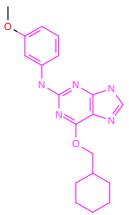
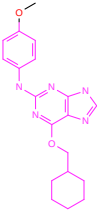
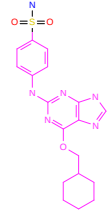
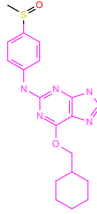
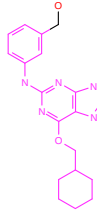
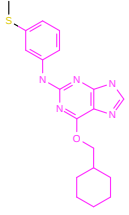
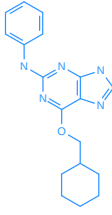
			
title 30	title 28	title 1oiy	title 1oi9
			
title 32	title 1oiu	title 29	title 21
			
title 26	title 1h1s	title 31	title 20
			
title 22	title 1h1q		

Figure S2. Structures for the ligands investigated for the CDK2 system with their corresponding abbreviations (Wang et al., 2015). Ligand 1h1q, highlighted in blue, is the reference structure for the consensus common core (CC) used in this work.

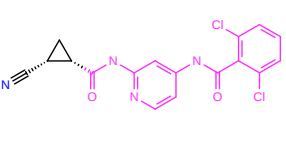
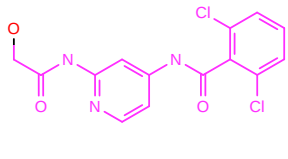
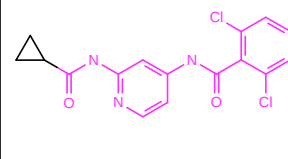
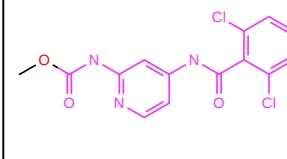
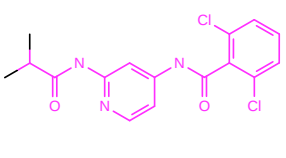
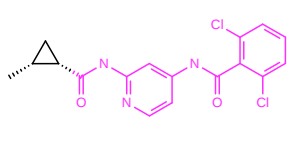
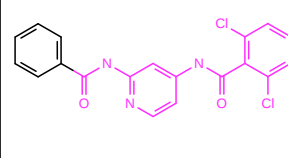
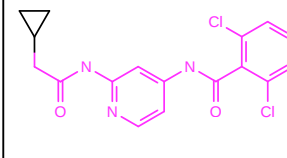
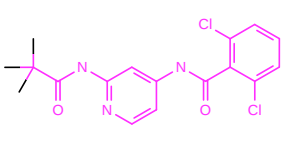
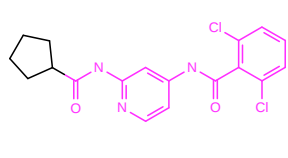
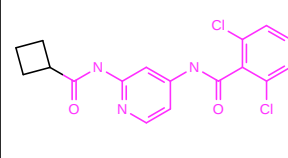
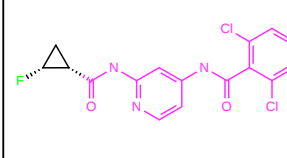
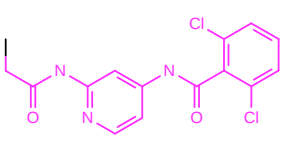
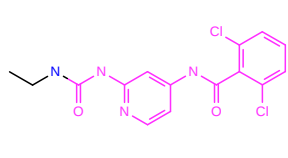
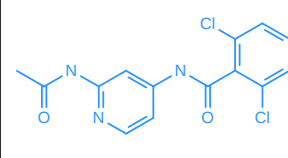
			
title jmc_30	title ejm_50	title ejm_46	title ejm_55
			
title ejm_43	title jmc_28	title ejm_49	title ejm_45
			
title ejm_44	title ejm_48	title ejm_47	title jmc_23
			
title ejm_42	title ejm_54	title ejm_31	

Figure S3. Structures for the ligands investigated for the TYK2 system with their corresponding abbreviations (Wang et al., 2015). Ligand ejm_31, highlighted in blue, is the reference structure for the consensus common core (CC) used in this work.

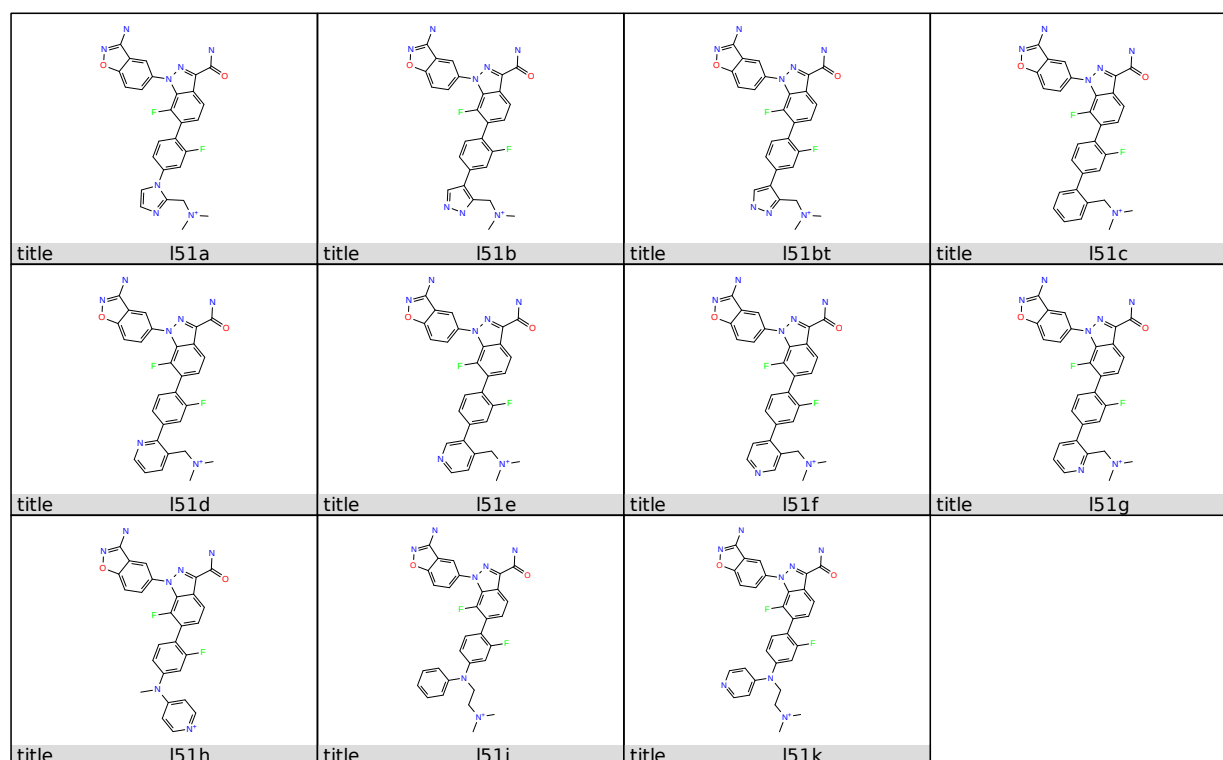


Figure S4. Structures for the ligands investigated for the XFa system with their corresponding abbreviations (Homeyer and Gohlke, 2013). Different CCs were used depending on the mutation studied.

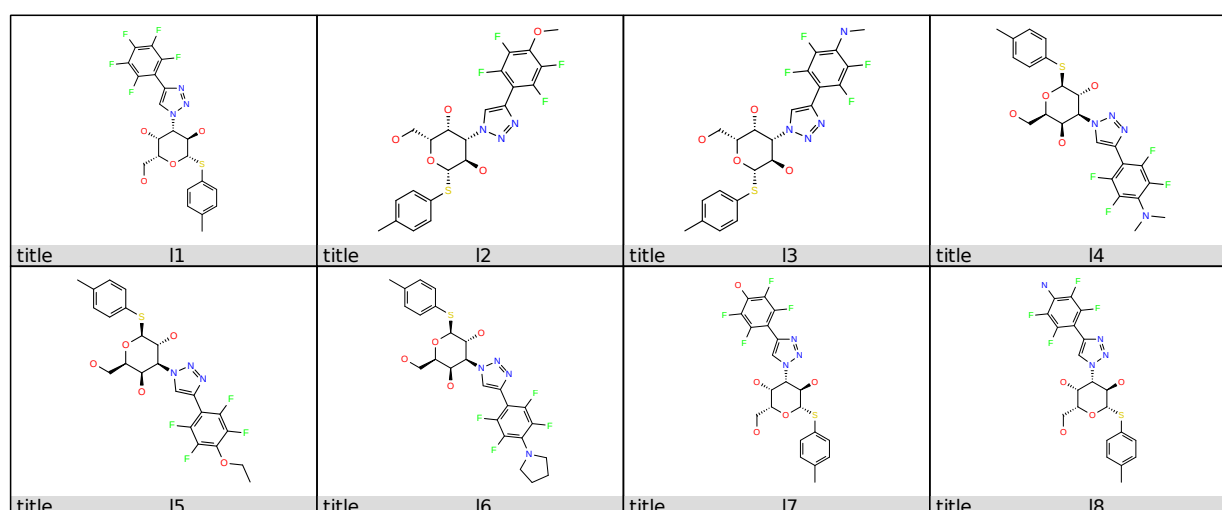


Figure S5. Structures for the ligands investigated for the Galectin-3 system with their corresponding abbreviations (Gapsys et al., 2020). Different CCs were used depending on the mutation studied.

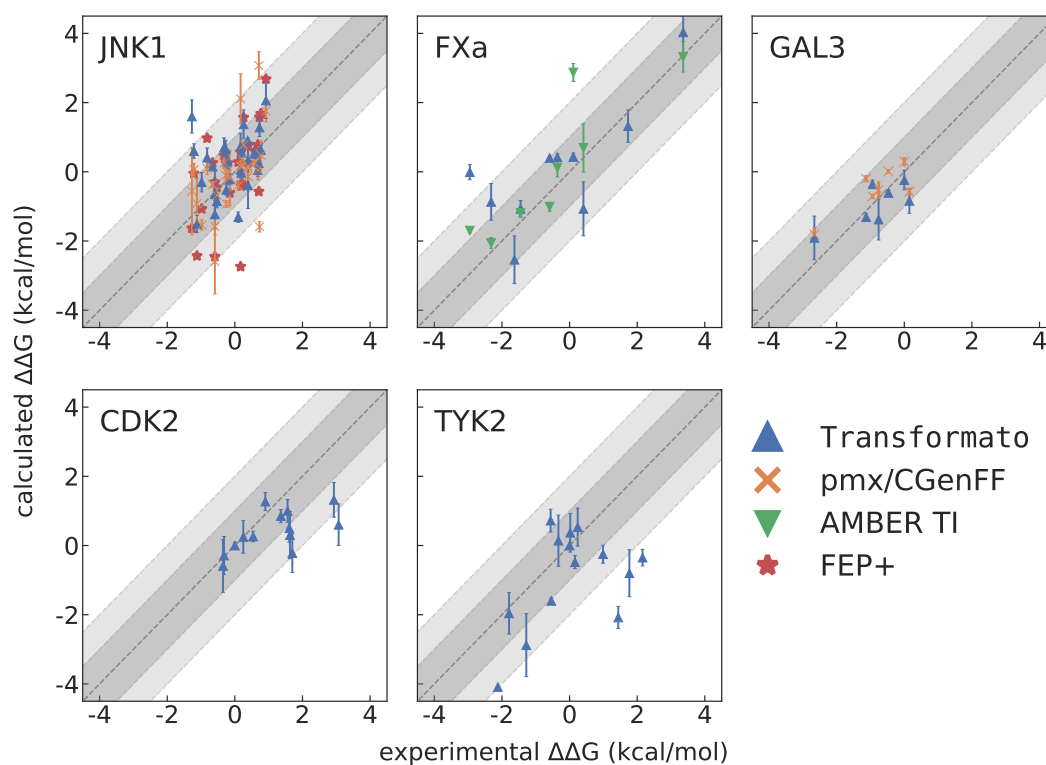


Figure S6. $\Delta\Delta G_{L_1 \rightarrow L_2}^{bind}$ obtained with Transformato (blue triangles) compared to results obtained by pmx/CGenFF for JNK1, GAL3 (orange crosses), by FEP+ for JNK1 (red stars), and by AMBER TI for FXa (green triangles). The corresponding RMSE and MAE values can be found in Table 2a of the main manuscript.

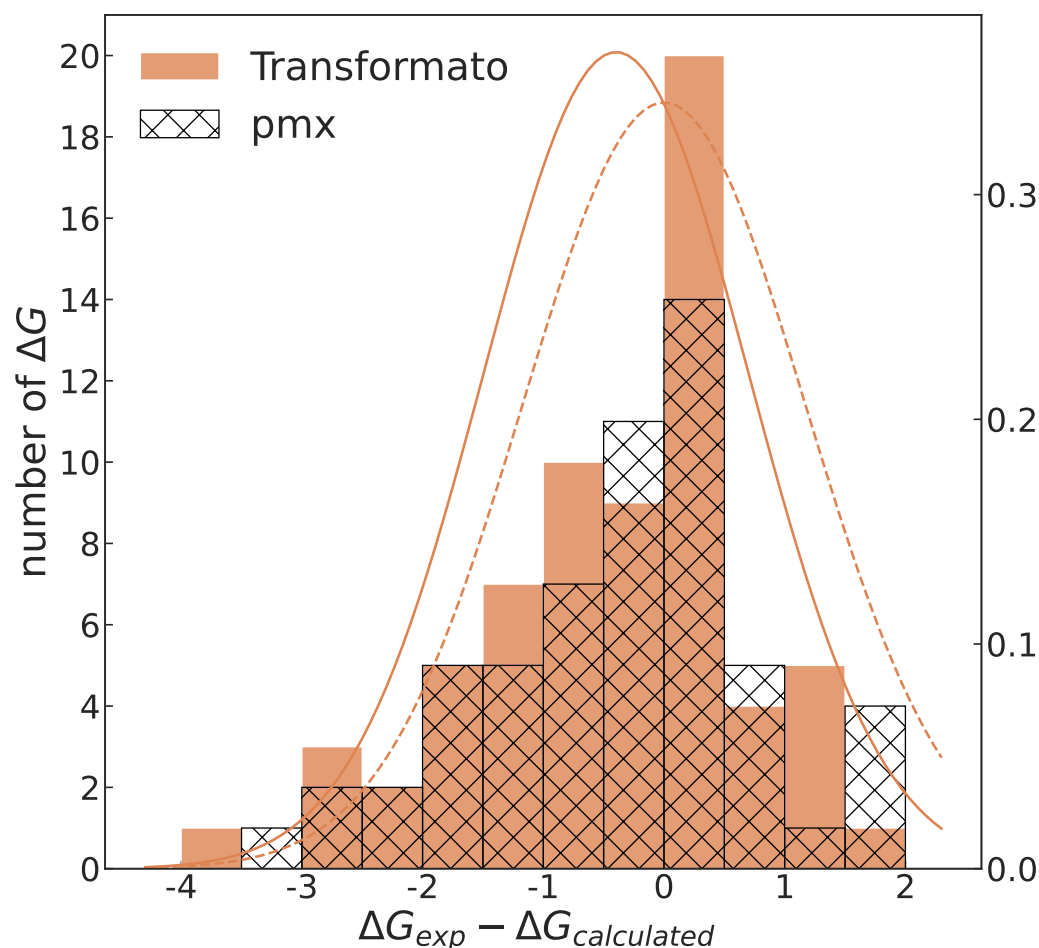


Figure S7. Histogram of deviations between the computed binding free energies ΔG^{bind} obtained by Transformato (orange) and pmx/CGenFF (black, hatched histogram) and experiment. The data obtained with Transformato are superposed with two Gaussian distributions. Solid orange line: $\mu = 0$, $\sigma = 1.17$ (1.17 kcal/mol is the RMSE of the predicted binding free energies for Transformato). Dashed orange line: Best fit Gaussian $\mu = -0.40$, $\sigma = 1.09$. For clarity, the analogous graphs for pmx/CGenFF are not shown; the parameters for the fitted Gaussian is $\mu = -0.41$, $\sigma = 1.15$

Table S1. Comparison of experimental ΔG values with calculated ΔG values for pmx/CGenFF and Transformato (TF) for the JNK1 system. The computed $\Delta\Delta G$ values for pmx/CGenFF were taken from Gapsys et al. (2020)

	exp. ΔG^a	calc. ΔG (pmx/CGenFF)	UE (pmx/CGenFF)	calc. ΔG TF	UE TF	Calculation of ΔG
18628	-8.70	-8.70 $\pm$ 0.38	0.00	-8.81 $\pm$ 0.10	0.11	$\Delta G(18626) + \Delta\Delta G(18626 \rightarrow 18628)$
18624	-8.49	-8.76 $\pm$ 0.03	0.27	-7.97 $\pm$ 0.07	0.52	$\Delta G(18626) + \Delta\Delta G(18626 \rightarrow 18624)$
18639	-9.74	-8.86 $\pm$ 0.10	0.88	-9.85 $\pm$ 0.15	0.11	$\Delta G(18634) - \Delta\Delta G(18639 \rightarrow 18634)$
18660	-8.70	-6.76 $\pm$ 0.72	1.94	-8.14 $\pm$ 0.38	0.56	$\Delta G(18626) + \Delta\Delta G(18626 \rightarrow 18660)$
18630	-9.14	-8.97 $\pm$ 0.16	0.17	-8.21 $\pm$ 0.25	0.93	$\Delta G(18626) + \Delta\Delta G(18626 \rightarrow 18630)$
18632	-9.08	-8.93 $\pm$ 0.09	0.15	-8.56 $\pm$ 0.23	0.52	$\Delta G(18626) + \Delta\Delta G(18626 \rightarrow 18632)$
18636	-7.51	-7.22 $\pm$ 0.15	0.29	-7.66 $\pm$ 0.28	0.15	$\Delta G(18624) - \Delta\Delta G(18636 \rightarrow 18624)$
18652	-10.68	-11.08 $\pm$ 1.32	0.40	-8.42 $\pm$ 0.78	2.26	$\Delta G(18631) + \Delta\Delta G(18631 \rightarrow 18652)$
17124	-9.68	-10.04 $\pm$ 0.75	0.36	-11.07 $\pm$ 0.38	1.39	$\Delta G(18634) - \Delta\Delta G(17124 \rightarrow 18634)$
18635	-7.29	-8.82 $\pm$ 0.24	1.53	-8.56 $\pm$ 0.33	1.27	$\Delta G(18624) - \Delta\Delta G(18635 \rightarrow 18624)$
18627	-8.48	-8.71 $\pm$ 0.09	0.23	-8.55 $\pm$ 0.33	0.07	$\Delta G(18626) + \Delta\Delta G(18626 \rightarrow 18627)$
18637	-10.14	-9.45 $\pm$ 0.81	0.70	-10.95 $\pm$ 0.32	0.46	$\Delta G(18634) + \Delta\Delta G(18634 \rightarrow 18637)$
18634	-9.99	-9.79 $\pm$ 0.73	0.20	-10.38 $\pm$ 0.25	0.39	$\Delta G(18626) + \Delta\Delta G(18626 \rightarrow 18634)$
18629	-8.67	-8.82 $\pm$ 0.77	0.15	-8.56 $\pm$ 0.43	0.11	$\Delta G(18624) - \Delta\Delta G(18635 \rightarrow 18624)$
18631	-9.41	-10.52 $\pm$ 0.40	1.11	-10.02 $\pm$ 0.61	0.61	$\Delta G(18624) - \Delta\Delta G(18631 \rightarrow 18624)$
18633	-9.17	-8.81 $\pm$ 0.29	0.36	-8.02 $\pm$ 0.38	1.15	$\Delta G(18624) - \Delta\Delta G(18633 \rightarrow 18624)$
18658	-9.70	-9.53 $\pm$ 0.76	0.17	-9.46 $\pm$ 0.73	0.24	$\Delta G(18638) + \Delta\Delta G(18638 \rightarrow 18658)$
18638	-10.09	-9.39 $\pm$ 0.74	0.70	-9.07 $\pm$ 0.29	1.02	$\Delta G(18634) - \Delta\Delta G(18638 \rightarrow 18634)$
18625	-8.11	-8.39 $\pm$ 0.42	0.28	-8.24 $\pm$ 0.13	0.13	$\Delta G(18626) + \Delta\Delta G(18626 \rightarrow 18625)$
18659	-9.47	-11.46 $\pm$ 0.94	1.99	-9.52 $\pm$ 0.71	0.05	$\Delta G(18626) + \Delta\Delta G(18626 \rightarrow 18659)$
18626	-8.87	-8.87	0.00	-8.87	0.00	experimental value

^a as reported by Wang et al. (2015)

Table S2. Comparison of experimental ΔG values with calculated ΔG values for pmx/CGenFF and Transformato (TF) for the XF_a system. The computed $\Delta \Delta G$ values for pmx/CGenFF were taken from Hu et al. (2016). The unsigned error (UE) is given for all values besides the reference structure.

	exp. ΔG^a	ΔG (pmx/CGenFF)	UE (pmx/CGenFF)	ΔG (TF)	UE (TF)	Calculation of ΔG
151a	-10.71	-11.23 $\pm$ 0.57	0.81	-11.52 $\pm$ 0.53	0.81	$\Delta G(151g) + \Delta \Delta G(151g \rightarrow 151a)$
151bt	-12.15	-11.66 $\pm$ 0.75	0.27	-12.48 $\pm$ 0.54	0.33	$\Delta G(151a) + \Delta \Delta G(151a \rightarrow 151bt)$
151c	-11.11	-11.11	0.00	-11.11	0.00	experimental value
151d	-7.75	-7.79 $\pm$ 0.44	0.04	-7.08 $\pm$ 0.66	0.67	$\Delta G(151c) + \Delta \Delta G(151c \rightarrow 151d)$
151e	-9.38	-10.07 $\pm$ 0.20	0.68	-9.79 $\pm$ 0.47	0.41	$\Delta G(151c) + \Delta \Delta G(151c \rightarrow 151e)$
151f	-11.70	-12.12 $\pm$ 0.13	0.42	-10.71 $\pm$ 0.03	0.99	$\Delta G(151c) + \Delta \Delta G(151c \rightarrow 151f)$
151g	-11.47	-10.99 $\pm$ 0.27	0.48	-10.69 $\pm$ 0.10	0.78	$\Delta G(151c) + \Delta \Delta G(151c \rightarrow 151g)$
151h	-11.51	-11.80 $\pm$ 0.70	0.29	-10.04 $\pm$ 0.78	1.47	$\Delta G(151c) - \Delta \Delta G(151h \rightarrow 151c)$
151i	-8.48	-7.23 $\pm$ 0.06	1.25	-9.60 $\pm$ 0.22	1.12	$\Delta G(151k) - \Delta \Delta G(151i \rightarrow 151k)$
151k	-11.41	-8.94 $\pm$ 0.26	2.48	-9.61 $\pm$ 0.11	1.80	$\Delta G(151h) + \Delta \Delta G(151h \rightarrow 151k)$

^a as reported by Wang et al. (2015)

Table S3. Comparison of experimental ΔG values with calculated ΔG values for pmx/CGenFF and Transformato (TF) for the Galectin-3 system. The computed $\Delta \Delta G$ values for pmx/CGenFF were taken from Hu et al. (2016). The unsigned error (UE) is given for all values besides the reference structure.

	exp. ΔG^a	ΔG (pmx/CGenFF)	UE (pmx/CGenFF)	ΔG (TF)	UE (TF)	Calculation of ΔG
11	-7.50	-7.28 $\pm$ 0.13	0.22	-8.65 $\pm$ 0.38	1.15	$\Delta G(17) + \Delta \Delta G(17 \rightarrow 11)$
12	-6.51	-6.51	0.00	-6.51	0.00	experimental value
13	-6.51	-6.80 $\pm$ 0.12	0.29	-6.27 $\pm$ 0.28	0.24	$\Delta G(12) - \Delta \Delta G(13 \rightarrow 12)$
14	-6.03	-6.81 $\pm$ 0.13	0.77	-5.66 $\pm$ 0.40	0.37	$\Delta G(13) - \Delta \Delta G(14 \rightarrow 13)$
15	-5.56	-5.80 $\pm$ 0.06	0.23	-6.14 $\pm$ 0.12	0.58	$\Delta G(12) - \Delta \Delta G(15 \rightarrow 12)$
16	-4.83	-5.47 $\pm$ 0.23	0.64	-6.74 $\pm$ 0.74	1.91	$\Delta G(11) - \Delta \Delta G(16 \rightarrow 11)$
17	-6.36	-7.08 $\pm$ 0.09	0.71	-7.35 $\pm$ 0.36	0.98	$\Delta G(12) + \Delta \Delta G(12 \rightarrow 17)$
18	-6.80	-7.34 $\pm$ 0.27	0.54	-7.04 $\pm$ 0.71	0.24	$\Delta G(14) + \Delta \Delta G(14 \rightarrow 18)$

^a taken from Peterson (2018)

Table S4. Comparison of experimental ΔG values with the corresponding calculated ΔG value for the CDK2 system. The formula how ΔG was derived from the $\Delta\Delta G$ values as reported by Gapsys et al. (2020) by pmx/CGenFF is given in the last column. The unsigned error (UE) is given for all values besides the reference structure.

exp.	ΔG^a	ΔG (pmx/CGenFF)	UE (pmx/CGenFF)	Calculation of ΔG
20	-8.72	-8.55 $\pm$ 0.07	0.17	$\Delta G(\text{lh1q}) - \Delta\Delta G(20 \rightarrow \text{lh1q})$
21	-7.83	-8.35 $\pm$ 0.59	0.52	$\Delta G(17) + \Delta\Delta G(17 \rightarrow 21)$
22	-7.86	-7.73 $\pm$ 0.56	0.13	$\Delta G(17) + \Delta\Delta G(17 \rightarrow 22)$
26	-8.43	-8.62 $\pm$ 0.22	0.19	$\Delta G(\text{lh1q}) - \Delta\Delta G(26 \rightarrow \text{lh1q})$
28	-11.11	-9.48 $\pm$ 0.41	1.63	$\Delta G(26) - \Delta\Delta G(28 \rightarrow 26)$
29	-9.88	-8.42 $\pm$ 0.25	1.46	$(\Delta G(26) - \Delta\Delta G(29 \rightarrow 26)) + (\Delta G(\text{loi9}) + \Delta\Delta G(\text{loi9} \rightarrow 29)) / 2$
30	-9.81	-9.69 $\pm$ 0.26	0.12	$(\Delta G(26) - \Delta\Delta G(30 \rightarrow 26)) + (\Delta G(31) + \Delta\Delta G(30 \rightarrow 31)) / 2$
31	-9.54	-8.56 $\pm$ 0.52	0.98	$\Delta G(\text{loi9}) + \Delta\Delta G(\text{loi9} \rightarrow 31)$
lh1q	-8.18	-8.18	0.00	experimental value
lh1s	-11.25	-9.17 $\pm$ 0.27	2.08	$(\Delta G(26) - \Delta\Delta G(\text{lh1s} \rightarrow 26)) + (\Delta G(\text{loi9}) + \Delta\Delta G(\text{lh1s} \rightarrow \text{loi9})) / 2$
loi9	-9.74	-8.57 $\pm$ 0.23	1.17	$(\Delta G(26) + \Delta\Delta G(26 \rightarrow \text{loi9})) + (\Delta G(\text{loi9}) + \Delta\Delta G(\text{loi9} \rightarrow \text{loi9})) / 2$
loi9	-9.08	-11.06 $\pm$ 0.35	1.98	$\Delta G(\text{lh1q}) - \Delta\Delta G(\text{loi9} \rightarrow \text{lh1q})$
loi9	-9.79	-8.66 $\pm$ 0.49	1.13	$\Delta G(\text{lh1q}) - \Delta\Delta G(\text{loi9} \rightarrow \text{lh1q})$

^a as reported by Wang et al. (2015)**Table S5.** Comparison of experimental ΔG values with the corresponding calculated ΔG value for the TYK2 system. The formula how ΔG was derived from the $\Delta\Delta G$ values as reported by Gapsys et al. (2020) by pmx/CGenFF is given in the last column. The unsigned error (UE) is given for all values besides the reference structure.

exp.	ΔG^a	ΔG (pmx/CGenFF)	UE (pmx/CGenFF)	Calculation of ΔG
ejm_31	-9.54	-9.54	0.00	experimental value
ejm_42	-9.78	-8.37 $\pm$ 0.27	1.41	$(\Delta G(\text{ejm48}) - \Delta\Delta G(\text{ejm42} \rightarrow \text{ejm48})) + (\Delta G(\text{ejm45}) + \Delta\Delta G(\text{ejm45} \rightarrow \text{ejm42})) / 2$
ejm_43	-8.26	-6.40 $\pm$ 0.61	1.86	$\Delta G(\text{ejm31}) + \Delta\Delta G(\text{ejm31} \rightarrow \text{ejm43})$
ejm_44	-7.42	-5.65 $\pm$ 0.45	1.77	$(\Delta G(\text{ejm42}) - \Delta\Delta G(\text{ejm44} \rightarrow \text{ejm42})) + (\Delta G(\text{ejm55}) - \Delta\Delta G(\text{ejm44} \rightarrow \text{ejm55})) / 2$
ejm_45	-9.56	-8.68 $\pm$ 0.40	0.88	$\Delta G(\text{ejm31}) + \Delta\Delta G(\text{ejm31} \rightarrow \text{ejm45})$
ejm_46	-11.31	-8.85 $\pm$ 0.07	2.46	$\Delta G(\text{ejm31}) + \Delta\Delta G(\text{ejm31} \rightarrow \text{ejm46})$
ejm_47	-9.70	-6.71 $\pm$ 0.22	2.99	$\Delta G(\text{ejm31}) - \Delta\Delta G(\text{ejm47} \rightarrow \text{ejm31})$
ejm_48	-9.00	-8.12 $\pm$ 0.20	0.88	$\Delta G(\text{ejm31}) + \Delta\Delta G(\text{ejm31} \rightarrow \text{ejm48})$
ejm_49	-7.75	-6.89 $\pm$ 0.17	0.86	$\Delta G(\text{ejm31}) - \Delta\Delta G(\text{ejm49} \rightarrow \text{ejm31})$
ejm_50	-8.98	-8.56 $\pm$ 0.31	0.42	$\Delta G(\text{ejm49}) + \Delta\Delta G(\text{ejm50})$
ejm_54	-10.53	-9.37 $\pm$ 0.33	1.16	$(\Delta G(\text{ejm42}) + \Delta\Delta G(\text{ejm42} \rightarrow \text{ejm54})) + (\Delta G(\text{ejm55}) - \Delta\Delta G(\text{ejm55} - \text{ejm54})) / 2$
ejm_55	-9.21	-10.77 $\pm$ 1.17	1.56	$(\Delta G(\text{ejm43}) - \Delta\Delta G(\text{ejm43} \rightarrow \text{ejm55})) + (\Delta G(\text{ejm47}) - \Delta\Delta G(\text{ejm47} - \text{ejm55})) / 2$
jmc_23	-11.70	-8.80 $\pm$ 0.13	2.90	$\Delta G(\text{ejm46}) - \Delta\Delta G(\text{jmc23} \rightarrow \text{ejm46})$
jmc_28	-10.98	-7.63 $\pm$ 0.31	3.35	$\Delta G(\text{ejm31}) + \Delta\Delta G(\text{ejm31} \rightarrow \text{jmc28})$
jmc_30	-10.94	-9.23 $\pm$ 0.56	1.71	$\Delta G(\text{jmc28}) + \Delta\Delta G(\text{jmc28} \rightarrow \text{jmc30})$

^a as reported by Wang et al. (2015)

Table S6. Comparison of experimental ΔG values with the corresponding calculated ΔG value for the CDK2 system. The formula how ΔG was derived from the $\Delta\Delta G$ values as reported by *Transformato* is given in the last column. The unsigned error (UE) is given for all values besides the reference structure.

	exp. ΔG^a	ΔG (TF)	UE (TF)	Calculation of ΔG
20	-8.72	-8.48 ± 0.15	0.24	$\Delta G(1h1q) - \Delta\Delta G(20 \rightarrow 1h1q)$
21	-7.83	-7.63 ± 0.76	0.20	$\Delta G(1h1q) - \Delta\Delta G(21 \rightarrow 1h1q)$
22	-7.86	-7.93 ± 0.55	0.07	$\Delta G(1h1q) - \Delta\Delta G(22 \rightarrow 1h1q)$
26	-8.43	-8.43 ± 0.47	0.00	$\Delta G(1h1q) - \Delta\Delta G(26 \rightarrow 1h1q)$
28	-11.11	-9.50 ± 0.50	1.61	$\Delta G(1h1q) - \Delta\Delta G(28 \rightarrow 1h1q)$
29	-9.88	-7.96 ± 0.56	1.92	$\Delta G(1h1q) - \Delta\Delta G(29 \rightarrow 1h1q)$
30	-9.81	-8.48 ± 0.68	1.33	$\Delta G(1h1q) - \Delta\Delta G(30 \rightarrow 1h1q)$
31	-9.54	-9.03 ± 0.19	0.51	$\Delta G(1h1q) - \Delta\Delta G(31 \rightarrow 1h1q)$
1h1q	-8.18	-8.18	0.00	experimental value
1h1s	-11.25	-8.78 ± 0.60	2.47	$\Delta G(1h1q) - \Delta\Delta G(1h1s \rightarrow 1h1q)$
loi9	-9.74	-9.18 ± 0.33	0.56	$\Delta G(1h1q) - \Delta\Delta G(loi9 \rightarrow 1h1q)$
loiu	-9.08	-9.49 ± 0.26	0.41	$\Delta G(1h1q) - \Delta\Delta G(loiu \rightarrow 1h1q)$
loiy	-9.79	-8.68 ± 0.41	1.11	$\Delta G(1h1q) - \Delta\Delta G(loiy \rightarrow 1h1q)$

^a as reported by Wang et al. (2015)

Table S7. Experimental ΔG values with the corresponding calculated ΔG for the TYK2 system. The formula how ΔG was derived from $\Delta\Delta G$ by *Transformato* is given in the last column. The unsigned error (UE) is given for all values besides the reference structure.

	exp. ΔG^a	ΔG (TF)	UE (TF)	Calculation of ΔG
ejm_31	-9.54	-9.54	0.00	experimental value
ejm_42	-9.78	-10.07 ± 0.26	0.29	$\Delta G(ejm31) - \Delta G(ejm42 \rightarrow ejm31)$
ejm_43	-8.26	-6.66 ± 0.18	1.60	$\Delta G(ejm31) - \Delta G(ejm43 \rightarrow ejm31)$
ejm_44	-7.42	-5.45 ± 0.55	1.97	$\Delta G(ejm31) - \Delta G(ejm44 \rightarrow ejm31)$
ejm_45	-9.56	-9.91 ± 0.50	0.35	$\Delta G(ejm31) - \Delta G(ejm45 \rightarrow ejm31)$
ejm_46	-11.31	-8.74 ± 0.29	2.57	$\Delta G(ejm31) - \Delta G(ejm46 \rightarrow ejm31)$
ejm_47	-9.70	-9.06 ± 0.23	0.64	$\Delta G(ejm31) - \Delta G(ejm47 \rightarrow ejm31)$
ejm_48	-9.00	-7.94 ± 0.40	1.06	$\Delta G(ejm31) - \Delta G(ejm48 \rightarrow ejm31)$
ejm_49	-7.75	-7.58 ± 0.68	0.17	$\Delta G(ejm31) - \Delta G(ejm49 \rightarrow ejm31)$
ejm_50	-8.98	-10.26 ± 0.25	1.28	$\Delta G(ejm31) - \Delta G(ejm50 \rightarrow ejm31)$
ejm_54	-10.53	-9.29 ± 0.46	1.24	$\Delta G(ejm31) - \Delta G(ejm54 \rightarrow ejm31)$
ejm_55	-9.21	-9.68 ± 0.74	0.47	$\Delta G(ejm31) - \Delta G(ejm55 \rightarrow ejm31)$
jmc_23	-11.70	-9.19 ± 0.24	2.51	$\Delta G(ejm31) - \Delta G(jmc23 \rightarrow ejm31)$
jmc_28	-10.98	-7.46 ± 0.32	3.52	$\Delta G(ejm31) - \Delta G(jmc28 \rightarrow ejm31)$
jmc_30	-10.94	-8.04 ± 0.67	2.90	$\Delta G(ejm31) - \Delta G(jmc30 \rightarrow ejm31)$

^a as reported by Wang et al. (2015)

Table S8. Distribution of absolute errors of calculated binding affinities with respect to experiment $|\Delta G_{i,exp}^{bind} - \Delta G_i^{bind}|$: Obs % is the percentage of deviations falling within the specified absolute error range. Anticip % is the percentage of deviations expected given an underlying Gaussian error distributions with RMSE of 1.17 kcal/mol (cf. Figure S7, dashed orange line).

absolute error (kcal/mol)	anticip %	obs %
<0.5	33.08	43.28
<1.0	60.72	64.18
<1.5	80.01	82.09
<2.0	91.26	91.04
<2.5	96.73	94.03
>2.5	3.26	5.97

REFERENCES

- Gapsys, V., Pérez-Benito, L., Aldeghi, M., Seeliger, D., van Vlijmen, H., Tresadern, G., et al. (2020). Large scale relative protein ligand binding affinities using non-equilibrium alchemy. *Chem Sci* 11, 1140–1152. doi:10.1039/c9sc03754c
- Homeyer, N. and Gohlke, H. (2013). FEW: A workflow tool for free energy calculations of ligand binding. *J Comput Chem* 34, 965–973. doi:10.1002/jcc.23218
- Hu, Y., Sherborne, B., Lee, T. S., Case, D. A., York, D. M., and Guo, Z. (2016). The importance of protonation and tautomerization in relative binding affinity prediction: a comparison of AMBER TI and Schrödinger FEP. *J. Comput. Aided Mol. Des.* 30, 533–539. doi:10.1007/s10822-016-9920-5
- Peterson, K. (2018). *Molecular basis for galectin-ligand interactions Design, synthesis and analysis of ligands*. Ph.D. thesis, Lund University, Faculty of Science, Department of Chemistry, Centre for Analysis and Synthesis
- Wang, L., Wu, Y., Deng, Y., Kim, B., Pierce, L., Krilov, G., et al. (2015). Accurate and reliable prediction of relative ligand binding potency in prospective drug discovery by way of a modern free-energy calculation protocol and force field. *J. Am. Chem. Soc.* 137, 2695–2703. doi:10.1021/ja512751q

Appendix B:

Supplementary information for

Calculations of Absolute Solvation Free Energies with `transformato`— Application to the FreeSolv Database Using the CGenFF Force Field

J. Chem. Theory Comput. 19, 5988-59998 (2023)

Authors: Johannes Karwounopoulos, Åsmund Kaupang, Marcus Wieder and Stefan Boresch

Contributions: Conceptualization: JK, MW, and SB; methodology: JK, AK, MW, and SB; software: JK and AK; investigation: JK, and SB; writing: JK, AK, MW, and SB; supervision: MW and SB

Calculations of absolute solvation free energies with transformato— application to the FreeSolv database using the CGenFF force field

Johannes Karwounopoulos,^{†,‡} Åsmund Kaupang,[¶] Marcus Wieder,[§] and Stefan
Boresch^{*,†}

¹ [†]*Faculty of Chemistry, Institute of Computational Biological Chemistry, University of
Vienna, Vienna, Austria*

[‡]*Vienna Doctoral School of Chemistry (DoSChem), University of Vienna, Vienna, Austria*

[¶]*Department of Pharmacy, Section for Pharmaceutical Chemistry, University of Oslo,
0316 Oslo, Norway*

[§]*Department of Pharmaceutical Sciences, Pharmaceutical Chemistry Division, University
of Vienna, Althanstrasse 14, 1090 Vienna, Austria*

E-mail: stefan@mdy.univie.ac.at

2 Bootstrapping statistics

3 For all statistical measures (root mean squared error (RMSE), mean absolute error (MAE),
4 Pearson correlation (r), and Spearman's rank correlation reported in this work, we report
5 error estimates obtained by bootstrapping. Values were selected randomly with replacement
6 from the original dataset 1,000 times, and all statistical measures were computed for each
7 random selection. The resulting resampled metrics were stored in a list. Subsequently, the
8 95% confidence interval was computed by determining the 2.5th and 97.5th percentiles of the
9 resampled metrics. These were used as upper and lower bounds for the statistical measures.

10 Potential-based switching functions in OpenMM and 11 CHARMM

12 Default switching function in OpenMM

13 The standard switching function of OpenMM referred to as OMMvswi in this manuscript is
14 defined as follows *:

$$S(x) = 1 - 6x^5 + 15x^4 - 10x^3 \quad (1)$$

15 where $x = \frac{(r - r_{\text{switch}})}{(r_{\text{cutoff}} - r_{\text{switch}})}$. It decreases from 1 at $r = r_{\text{switch}}$ to 0 at $r = r_{\text{cutoff}}$.

16 Switching function in CHARMM

17 Since it is somewhat hidden in a relatively old publication, we also give the equations for
18 the vswitch function of CHARMM as described originally by ?:

$$S(x, x_{\text{on}}, x_{\text{off}}) = \frac{(x_{\text{off}} - x)^2(x_{\text{off}} + 2x - 3x_{\text{on}})}{(x_{\text{off}} - x_{\text{on}})^3} \quad (2)$$

*http://docs.openmm.org/7.7.0/userguide/theory/02_standard_forces.html

19 **Note** that this function is used as $S(\mathbf{r}^2, \mathbf{r}_{\text{on}}^2, \mathbf{r}_{\text{off}}^2)$, where r is the distance between the two
20 particles, r_{off} the cutoff distance, and r_{on} the distance where the switching region starts. The
21 function decreases from 1 at $r = r_{\text{on}}$ to 0 at $r = r_{\text{off}}$.

22 Figures

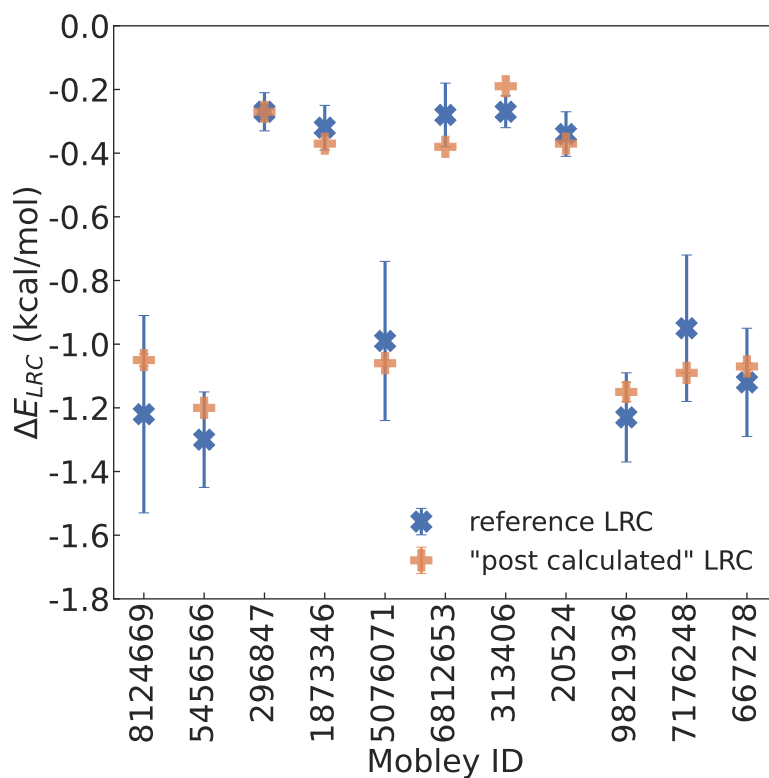


Figure S1: Comparison of two approaches to compute the Lennard-Jones long-range correction (LRC). For all compounds studied in this work, the LRC was estimated in a post-processing step ("post-calculated LRC", orange crosses); cf. main manuscript. For eleven compounds, the LRC was also calculated more correctly as follows: For each of the molecules, the ASFE was recomputed from scratch with OpenMM's LRC set to active during all MD simulations, thus including the LRC to the virial. The difference between the ASFE including the full isotropic LRC computed in this manner and the ASFE without LRC is referred to as "reference LRC" and plotted with a blue x. As one can see, the two LRCs agree reasonably well in all cases.

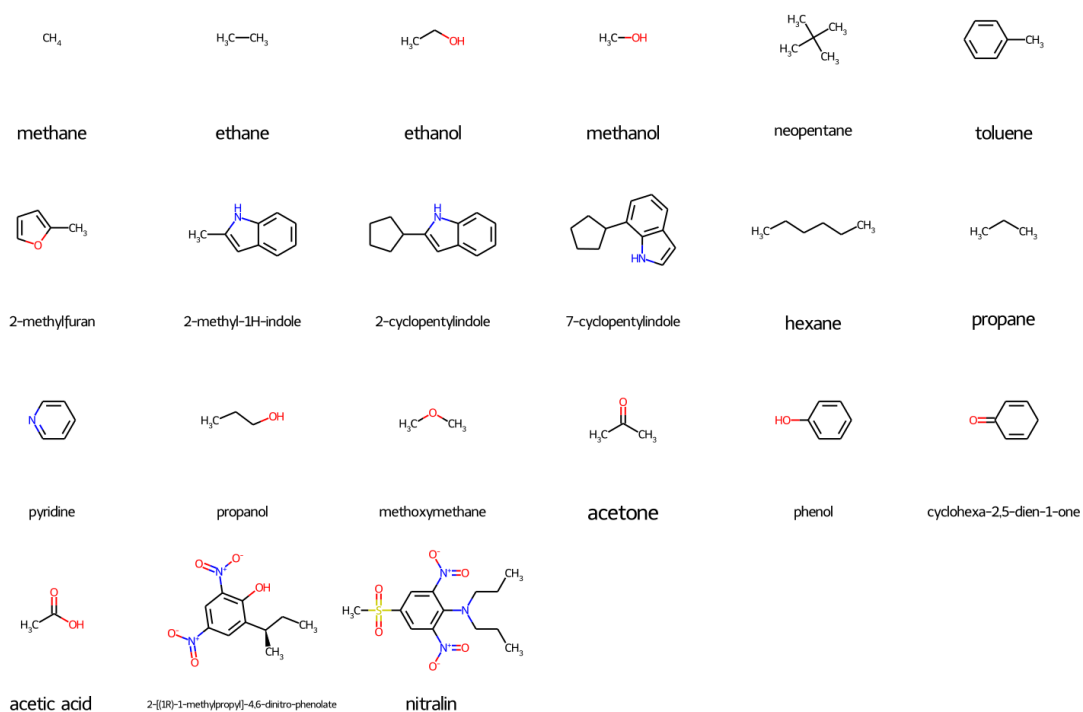


Figure S2: Molecules for which ASFEs with comparable simulation setups and force field parameters have been reported in the literature.^{1,2} This set was augmented by three additional molecules (last row) taken from the FreeSolv database: acetic acid (Mobley ID: 2725215), 2-[(1R)-1-methylpropyl]-4,6-dinitro-phenolate (Mobley ID: 3274817) and nitralin (Mobley ID: 725215).

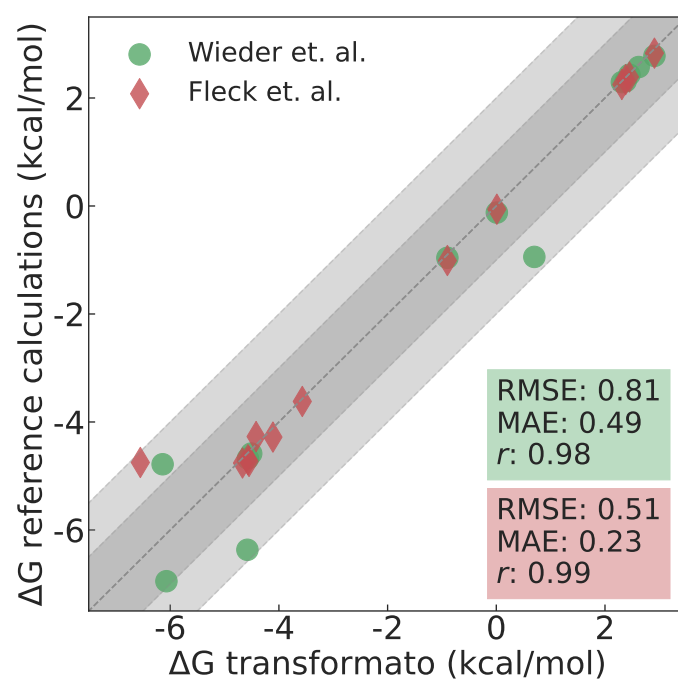


Figure S3: The initial comparison between results from **transformato** (TF) and reference results from the literature^{1,2} *before* ensuring that identical force field parameters were used.

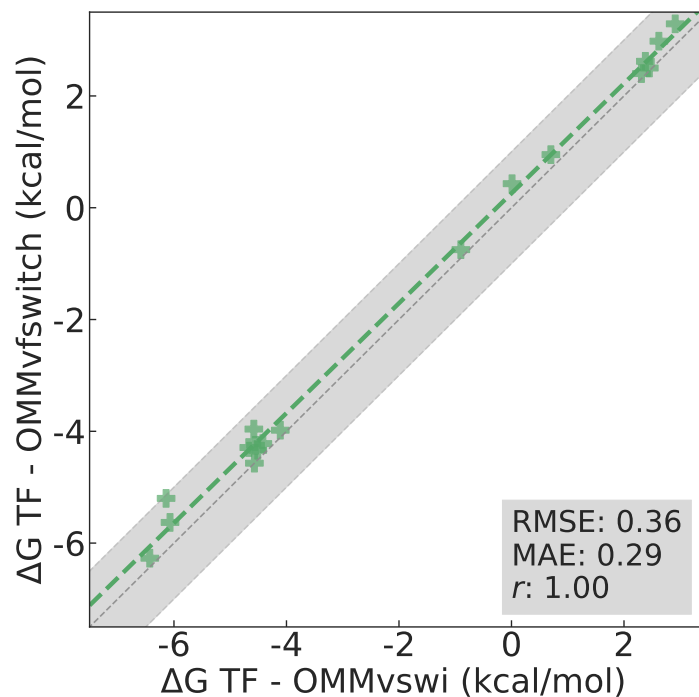


Figure S4: Comparison of ASFEs calculated with **transformato** (TF) when using two different switching functions for the Lennard-Jones interactions (OMMvswi and OMMvswitch); cf. the main manuscript. Values for all compounds in Fig. S2 – including the ones for which no experimental values are available – are shown. The green dashed line represents the linear regression line $y = 0.98x + 0.25$.

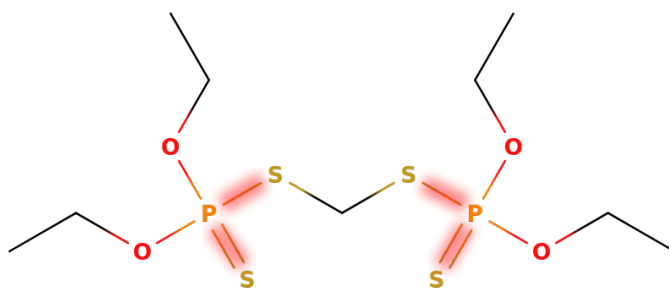


Figure S5: One of the 11 organophosphorodithioates for which we report no ASFEs (Mobley ID: 7754849). The phosphorus-sulfur bonds for which **cgenff** provided dubious force field parameters are highlighted in red. The Mobley IDs of the other compounds with the same motif are: 1770205, 1922649, 1849020, 2518989, 1849020, 2518989, 5393242, 8916409, 9281946, 7326706.

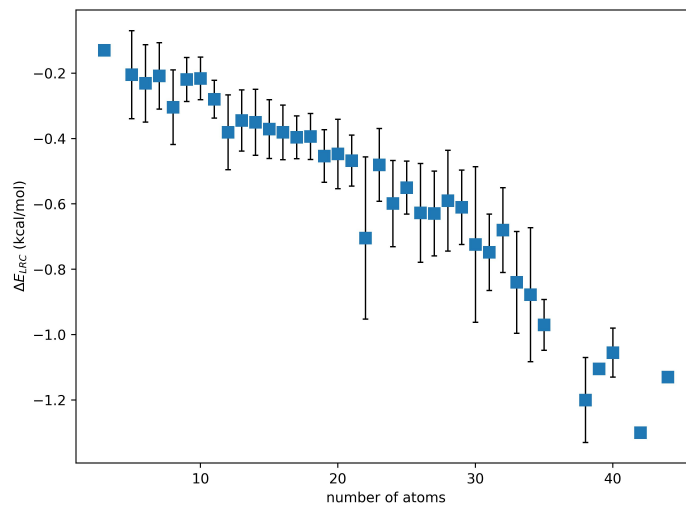


Figure S6: The average long-range correction ΔE_{LRC} as a function of the number of atoms per molecule, including hydrogens, are shown as blue squares. The spread of the correction for all molecules consisting of the same number of atoms are shown as error bars. If there are less than 3 molecules with the same number of atoms, no error bars are shown.

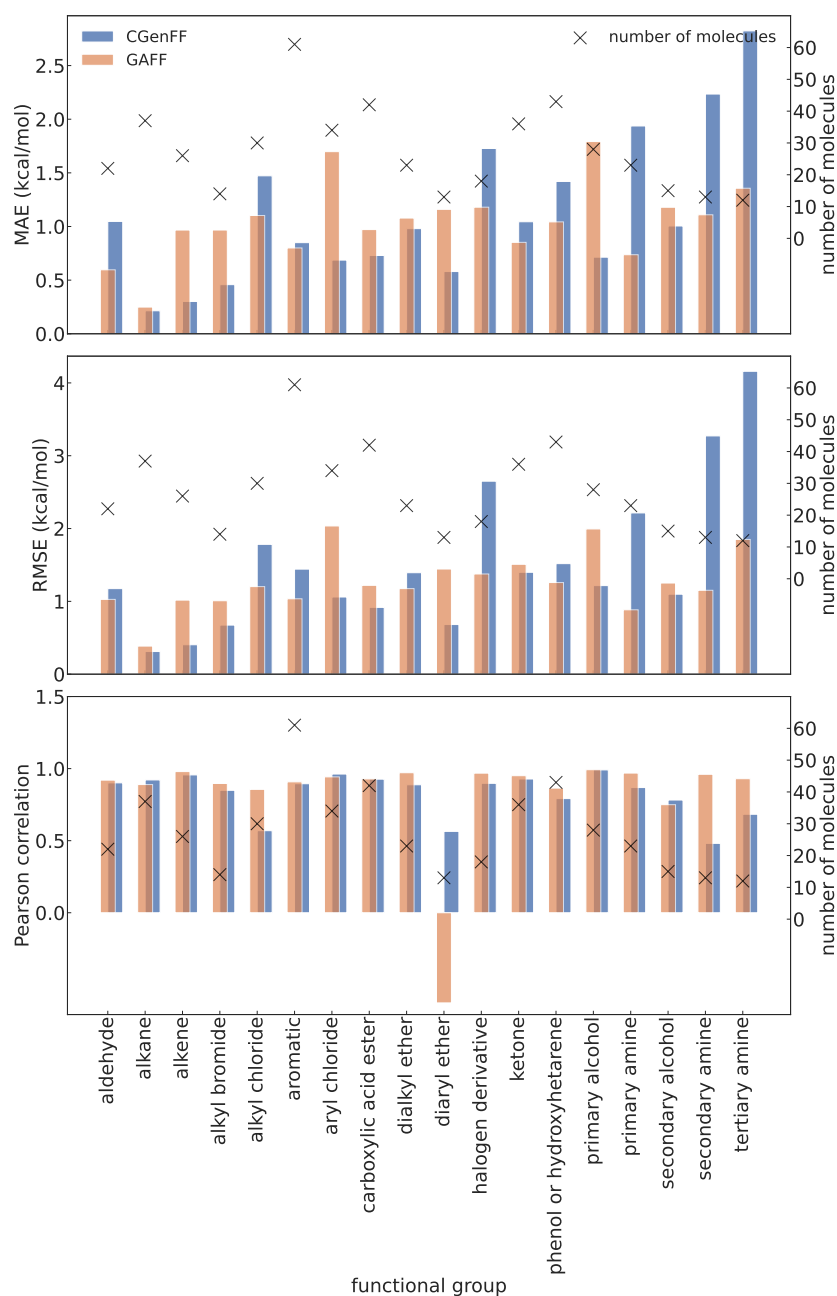


Figure S7: Expanded statistical measures for the results reported in this study (blue) and by Mobley and Guthrie³ (orange). From top to bottom: mean absolute errors (MAEs), root-mean-squared errors (RMSEs) and Pearson correlations. The molecules are classified according to their primary functional group. In each plot, the crosses, corresponding to the second y-axis on the right, indicate the number of molecules belonging to each group.

Tables

Table S1: All molecules for which the `cgenff` program failed to provide parameters.

Mobley ID	smiles	IUPAC name
2146331	<chem>C=O</chem>	formaldehyde
6359135	<chem>C(Cl)(Cl)(Cl)Cl</chem>	carbon tetrachloride
7578802	<chem>C(Br)(Br)Br</chem>	bromoform
676247	<chem>C(F)(F)(F)F</chem>	tetrafluoromethane
8311321	<chem>C(F)(F)Cl</chem>	chloro-difluoro-methane
5631798	<chem>N</chem>	ammonia
7732703	<chem>C(F)(F)(F)Br</chem>	bromo-trifluoro-methane
2996632	<chem>C(Cl)(Cl)Cl</chem>	chloroform
9571888	<chem>C1[C@@H]2[C@H](COS(=O)O1)[C@@]3(C(=C([C@]2(C3(Cl)Cl)Cl)Cl)Cl)Cl</chem>	endosulfan alpha

Table S2: Molecules, for which the standard deviation (std) of the ASFE (dG TF) was > 0.6 kcal/mol when using the default simulation length per state of 5 ns. Increasing the simulation length per state to 10 ns (dG TF (10 ns)) reduced the standard deviation (std) to < 1 kcal/mol in all cases.

Mobley ID	exp	dG TF (5 ns)	std	dG TF (10 ns)	std
2613240	-5.33	-1.85	1.64	-4.63	0.66
5200358	-9.84	-10.32	1.51	-9.70	0.82
3265457	-7.78	-8.66	1.28	-9.61	0.45
6935906	-2.49	-3.07	1.16	-3.14	0.58
4587267	-23.62	-17.05	1.01	-17.43	0.51
6688723	-6.44	-7.08	0.98	-6.99	0.50
3515580	-5.94	-3.66	0.84	-3.3	0.52
8705848	-3.22	0.14	0.75	0.23	0.3
8916409	-8.15	-16.58	0.6	-15.8	0.7

24 References

- 25 (1) Wieder, M.; Fleck, M.; Braunsfeld, B.; Boresch, S. Alchemical free energy simulations
26 without speed limits. A generic framework to calculate free energy differences inde-
27 pendent of the underlying molecular dynamics program. *J. Comput. Chem.* **2022**, *43*,
28 1151–1160.
- 29 (2) Fleck, M.; Wieder, M.; Boresch, S. Dummy Atoms in Alchemical Free Energy Calcula-
30 tions. *J. Chem. Theory Comput.* **2021**, *17*, 4403–4419.
- 31 (3) Mobley, D. L.; Guthrie, J. P. FreeSolv: a database of experimental and calculated hy-
32 dration free energies, with input files. *J. Comput. Aided Mol. Des.* **2014**, *28*, 711–720.

Appendix C:

Supplementary information for

Insights and Challenges in Correcting Force Field Based Solvation Free Energies Using A Neural Network Potential

ChemRxiv. (2023)

Authors: Johannes Karwounopoulos, Zhiyi Wu, Sara Tkaczyk, Shuzhe Wang, Adam Baskerville, Kavindri Ranasinghe, Thierry Langer, Geoffrey Wood, Marcus Wieder, Stefan Boresch

Contributions: Conceptualization: JK, MW, and SB; methodology: JK, ZW, MW, and SB; software: JK, ZW and MW; investigation: JK, ZW, MW and SB; writing: JK, ZW, MW, and SB; supervision: MW and SB

Supplementary Information: Insights and Challenges in Correcting Force Field Based Solvation Free Energies Using A Neural Network Potential

Johannes Karwounopoulos ^{1,2}, Zhiyi Wu ⁵, Sara Tkaczyk ^{3,4}, Shuzhe Wang ⁵, Adam Baskerville ⁵, Kavindri Ranasinghe ⁵, Thierry Langer ³, Geoffrey Wood ⁵, Marcus Wieder ^{5, 6*}, Stefan Boresch ^{1*}

***For correspondence:**

stefan@mdy.univie.ac.at (SB);
marcus.wieder@gmail.com (MW)

¹Faculty of Chemistry, Institute of Computational Biological Chemistry, University Vienna, Währingerstr. 17, 1090 Vienna, Austria; ²Vienna Doctoral School of Chemistry (DoSChem), University of Vienna, Währingerstr. 42, 1090 Vienna, Austria; ³Department of Pharmaceutical Sciences, Pharmaceutical Chemistry Division, University of Vienna, Josef-Holaubek-Platz 2, 1090 Vienna, Austria; ⁴Vienna Doctoral School of Pharmaceutical, Nutritional and Sport Sciences (PhaNuSpo), University of Vienna, Josef-Holaubek-Platz 2, 1090 Vienna, Austria; ⁵Exscientia plc, Schroedinger Building, Oxford, United Kingdom; ⁶Open Molecular Software Foundation, Davis, California 95616, United States

Additional details for the EXS and UVIE protocols

Protocol UVIE— CGenFF as the classical force field

ASFE calculations with `transformato`

Here, a short summary of the protocol used in our previous work is given; for the full details, see [11]. For each calculation, we generated the solvated system using the SMILES provided by the FreeSolv database with the Python extension of Open Babel (Pybel) [14]. Missing solute parameters were generated with a stand-alone version of `cgenff` (v2.5.1), based on version 4.6 of the CHARMM general force field (CGenFF). [7, 17–20] The solutes were placed in cubic simulation boxes with a side length of ≥ 26 Å, which is sufficiently large to be commensurate with the default CHARMM cut-off of 12 Å. These steps were automated with a Python package `macha`,¹ which uses CHARMM scripts generated by CHARMM-GUI [9, 12] as templates. When used to compute ASFES, `transformato` provides an automated workflow to set up the serial atom insertion (SAI) approach [2]. Since electrostatic and Lennard-Jones intramolecular interactions are annihilated, the gas phase correction step (see Figure 1 of the main manuscript) must restore both intramolecular interactions. Each ASFE calculation was repeated four times with different initial random velocities. The standard deviation was calculated from these four runs.

¹<https://github.com/akaupang/macha>

Treatment of virtual particles on chlorine atoms

Recent versions of the CGenFF force field [18, 19] use virtual particles to better describe the halogen bond, a specific non-covalent interaction between a halogen atom and another electronegative atom, driven by a localized positive electrostatic region known as the "sigma hole" [7]. While these virtual particles can be handled straightforwardly in the `transformato` workflow, they must not be present during the NEQ switching simulations from the MM to the ANI-2x level of theory. For the 24 molecules that contain a chlorine atom, we, therefore, had to ensure the closing of the alchemical cycle by introducing a suitable intermediate state. We calculated the free energy difference between the original state (fully charged virtual particle attached to each chlorine present) and an intermediate state, where the virtual particles are present, but their charges have been transferred to the corresponding chlorine atom. In this intermediate state, the massless virtual particles do not contribute to the system's free energy. Hence, the virtual particles can be safely removed when carrying out the endstate correction from MM to ANI-2x.

Nonequilibrium switching simulations

Setup of the forward (MM→ML/MM) switching simulations: Four independent 5 ns MD simulations were carried out, using the Langevin integrator with a target temperature of 303K. Pooling the trajectories of the 4 simulations of the physical endstate for each of the systems, we obtain a 20 ns trajectory containing 40,000 coordinate sets from which the forward switching protocol (MM → ML/MM) was initialized.

Setup of the backward (ML/MM→MM) switching simulations: At the ML/MM endstate, a single 10 ns trajectory, containing 10,000 coordinate frames, was generated for each system (simulation parameters as described previously and in the main text).

Switching simulations: From the pool of initial configurations, 300 coordinate sets were selected randomly (with replacement) as the starting point of the forward and backward NEQ switching simulations, both in the gas phase and in solution.

Protocol EXS— OpenFF as the classical force field

ASFE with openmmtools

Molecules were parametrized using Biosimspace [8] and the OpenFF-2.0 force field [1]. The solvated systems were created by solvating the ligand in a $40\text{\AA}^3$ cubic water box. In solution, electrostatic interactions were computed using PME [6] with a short-range cutoff of 10 Å; no cutoff was used in the gas phase. Bond lengths involving hydrogens were constrained to their parameter value in all simulations. OpenMM 8.0 [5] was used to perform all simulations. Before starting an MD simulation, both the gas phase and solvated structure were minimized. Simulations were performed using the LangevinMiddleIntegrator implemented in openmmtools [10] with a target temperature of 300 K and a friction coefficient of 1/ps, using a time step of 2 fs. Each system in

the gas phase was equilibrated in NVT for 1 ns. Each solvated system was equilibrated for 1 ns in the NPT ensemble with a Monte Carlo barostat at a target pressure of 1 atm. The ASFE calculation consisted of two phases: In solution, interactions were decoupled/annihilated using 26 λ -windows. First, intra- and intermolecular electrostatic interactions were scaled to zero using an equidistant λ -spacing of 0.1 (i.e., electrostatic interactions were annihilated in 11 steps). Then, the van der Waals interactions were decoupled with a non-equidistant scheduling (using 0.05 spacing until $\lambda = 0.5$, then continuing with a 0.1 spacing). Soft core potentials were used as described in Equation 13 of Ref. [15] Because of the mixed annihilation/decoupling scheme, the gas phase corrections only had to restore intramolecular electrostatic interactions. Each λ -window was simulated for 10 ns. Errors were estimated from the single repeat equilibrium simulations via the bootstrapping functionality of MBAR as implemented in `pymbar` [16].

Nonequilibrium switching simulations

Simulations of the physical endstates were repeated, treating the solute as fully flexible (i.e., only water molecules were kept rigid by constraints). During these simulations, 5,000 frames were saved. From these coordinates, 300 were selected randomly (with replacement) and used to initialize 5 ps forward switching simulations (MM $\rightarrow$ ML/MM).

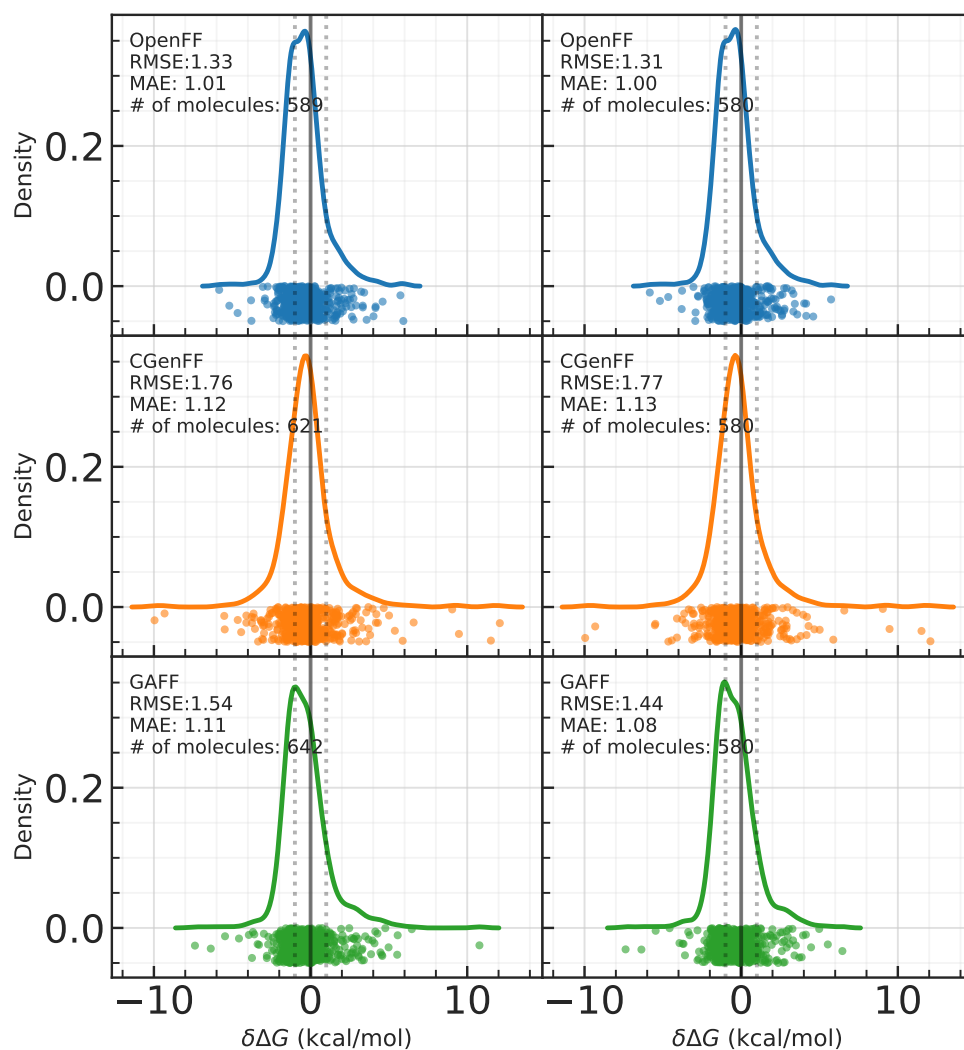


Figure S1. Kernel Density Estimation (KDE) of $\delta\Delta G = \Delta G_{exp} - \Delta G_{calc}$, the error in calculated ASFES compared with the experimental data, using three classical force fields. Top: OpenFF (this work); middle: CGenFF [11]; bottom: GAFF [4, 13]. The plots on the left are for all available data for the respective force field (OpenFF: 589, CGenFF: 621, GAFF: 642). Plots on the right display the results for the 580 molecules for which calculated ASFE values are available across all three force fields.

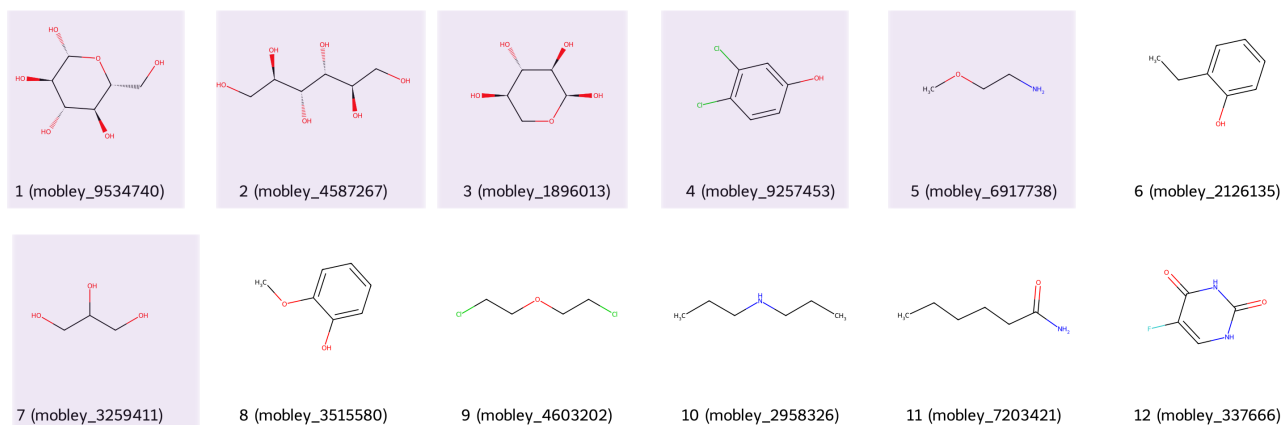


Figure S2. Structure of outliers depicted in the left panel of Figure 5 (protocol UVIE), connecting the numbering of Figure 5 with the FreeSolv ID. Molecules that were outliers in both UVIE and EXS are drawn on a shaded background.

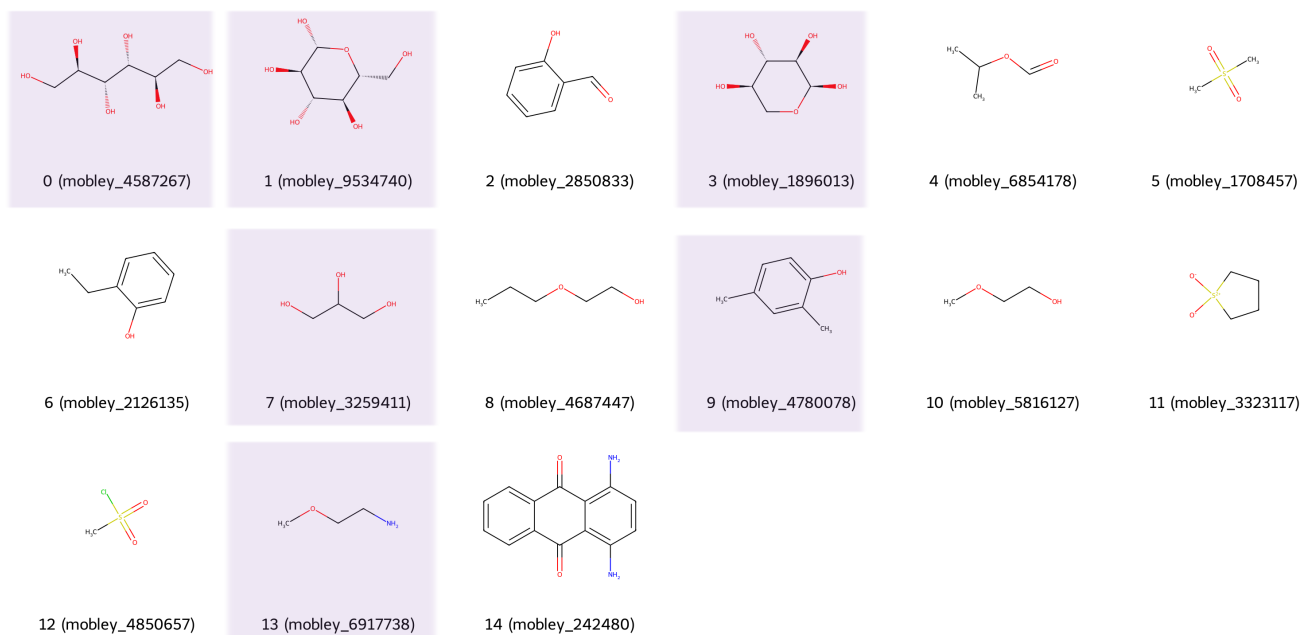


Figure S3. Structure of outliers depicted in the right panel of Figure 5 (protocol EXS), connecting the numbering of Figure 5 with the FreeSolv ID. Molecules that were outliers in both UVIE and EXS are drawn on a shaded background.

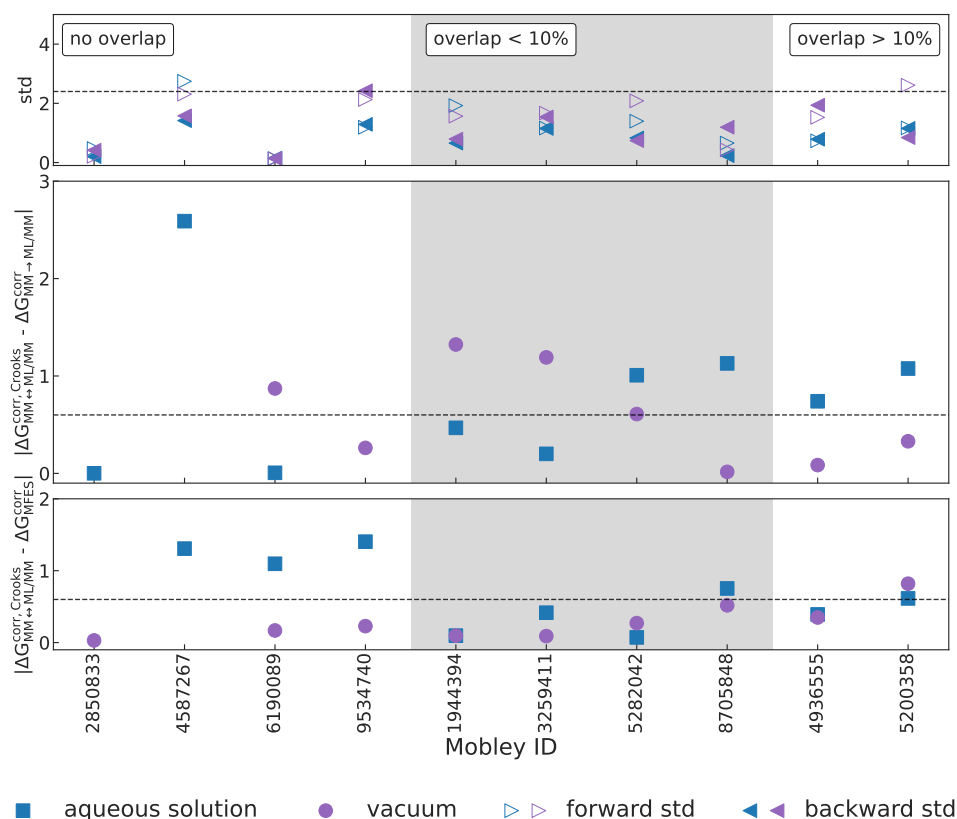


Figure S4. Several characteristics of the 10 compounds out of the 156 molecule subset, for which the MM→ML/MM correction differed by more than 1 kT when computed by Jarzynski's and Crooks' equation, either in vacuum (purple symbols) or in the aqueous environment (blue symbols), or both. The entries are sorted according to the overlap of their forward and backward work distributions: those with no overlap in either environment are presented on the left, compounds with less than 10% overlap in at least one environment are displayed in the middle gray area, and compounds with more than 10% overlap are shown on the right. All free energy and work related quantities are in kcal/mol.

Top panel: Standard deviation of the work values for the forward (▷) and backward switches (◁) in aqueous solution (blue) and vacuum (purple). The dashed line is drawn at 2.4 kcal/mol (4 kT) [3]. Middle panel: Absolute value of the deviation between the unidirectional Jarzynski and bidirectional Crooks results, $|\Delta G_{MM \leftrightarrow ML/MM}^{corr, Crooks} - \Delta G_{MM \leftrightarrow ML/MM}^{corr}|$. Bottom panel: Absolute deviation between values calculated from NEQ simulations using the two-sided Crook's equation ($\Delta G_{MM \leftrightarrow ML/MM}^{corr, Crooks}$) and values calculated by MFES (ΔG_{MFES}^{corr}). Any missing entries in the middle and bottom panels indicate values that are off-scale, i.e., > 3 kcal/mol (middle panel) or > 2 kcal/mol (bottom panel).

References

- [1] **Boothroyd S**, Behara PK, Madin OC, Hahn DF, Jang H, Gapsys V, Wagner JR, Horton JT, Dotson DL, Thompson MW, Maat J, Gokey T, Wang LP, Cole DJ, Gilson MK, Chodera JD, Bayly CI, Shirts MR, Mobley DL. Development and Benchmarking of Open Force Field 2.0.0: The Sage Small Molecule Force Field. *J Chem Theory Comput.* 2023 6; 19(11):3251–3275. doi: 10.26434/chemrxiv-2022-n2z1c-v2.
- [2] **Boresch S**, Bruckner S. Avoiding the van der Waals endpoint problem using serial atomic insertion. *J Comput Chem.* 2011 Aug; 32(11):2449–2458. doi: 10.1002/jcc.21829.
- [3] **Boresch S**, Woodcock HL. Convergence of single-step free energy perturbation. *Mol Phys.* 2017 6; 115(9-12):1200–1213. doi: 10.1080/00268976.2016.1269960.
- [4] **Duarte Ramos Matos G**, Kyu DY, Loeffler HH, Chodera JD, Shirts MR, Mobley DL. Approaches for Calculating Solvation Free Energies and Enthalpies Demonstrated with an Update of the FreeSolv Database. *J Chem Eng Data.* 2017 5; 62(5):1559–1569. doi: 10.1101/104281.
- [5] **Eastman P**, Galvelis R, Peláez RP, Abreu CRA, Farr SE, Gallicchio E, Gorenko A, Henry MM, Hu F, Huang J, Krämer A, Michel J, Mitchell JA, Pande VS, Rodrigues JP, Rodriguez-Guerra J, Simmonett AC, Swails J, Zhang I, Chodera JD, et al. OpenMM 8: Molecular Dynamics Simulation with Machine Learning Potentials. *arxiv.* 2023 10; doi: 10.48550/ARXIV.2310.03121.
- [6] **Essmann U**, Perera L, Berkowitz ML, Darden T, Lee H, Pedersen LG. A smooth particle mesh Ewald method. *J Chem Phys.* 1995 Aug; 103(19):8577. doi: 10.1063/1.470117.
- [7] **Gutiérrez IS**, Lin FY, Vanommeslaeghe K, Lemkul JA, Armacost KA, Brooks CL, MacKerell AD. Parametrization of halogen bonds in the CHARMM general force field: Improved treatment of ligand–protein interactions. *Bioorg Med Chem.* 2016 Oct; 24(20):4812–4825. doi: 10.1016/j.bmc.2016.06.034.
- [8] **Hedges LO**, Mey ASjs, Laughton CA, Gervasio FL, Mulholland AJ, Woods CJ, Michel J. BioSimSpace: An interoperable Python framework for biomolecular simulation. *Journal of Open Source Software.* 2019 11; 4(43):1831. doi: 10.21105/JOSS.01831.
- [9] **Jo S**, Kim T, Iyer VG, Im W. CHARMM-GUI: A web-based graphical user interface for CHARMM. *J Comput Chem.* 2008 Aug; 29(11):1859–1865. doi: 10.1002/jcc.20945.
- [10] **John Chodera**, Andrea Rizzi, Levi Naden, Kyle Beauchamp, Patrick Grinaway, Josh Fass, Iván Pulido, Alex Wade, Mike Henry, Gregory A Ross, Andreas Krämer, Hannah Bruce Macdonald, jaimergp, Bas Rustenburg, David W H Swenson, Ivy Zhang, Andy Simmonett, Mark J Williamson, hb0402, Jake Fennick, et al., choderalab/openmmtools: 0.23.1. Zenodo; 2023. <https://zenodo.org/records/8102771>, doi: 10.5281/ZENODO.8102771.
- [11] **Karwounopoulos J**, Kaupang Å, Wieder M, Boresch S. Calculations of Absolute Solvation Free Energies with Transformato – Application to the FreeSolv Database Using the CGenFF Force Field. *J Chem Theory Comput.* 2023 9; 19(17):5988–5998. doi: 10.1021/acs.jctc.3c00691.
- [12] **Lee J**, Cheng X, Swails JM, Yeom MS, Eastman PK, Lemkul JA, Wei S, Buckner J, Jeong JC, Qi Y, Jo S, Pande VS, Case DA, Brooks CL, MacKerell AD, Klauda JB, Im W. CHARMM-GUI Input Generator for NAMD, GROMACS, AMBER, OpenMM, and CHARMM/OpenMM Simulations Using the CHARMM36 Additive Force Field. *J Chem Theory Comput.* 2016 1; 12(1):405–413. doi: 10.1021/acs.jctc.5b00935.

- [13] **Mobley DL**, Guthrie JP. FreeSolv: a database of experimental and calculated hydration free energies, with input files. *J Comput Aided Mol Des*. 2014 Jun; 28(7):711–720. doi: 10.1007/s10822-014-9747-x.
- [14] **O'Boyle NM**, Morley C, Hutchison GR. Pybel: A Python wrapper for the OpenBabel cheminformatics toolkit. *Chem Cent J*. 2008 3; 2(1):1–7. doi: 10.1186/1752-153x-2-5.
- [15] **Pham TT**, Shirts MR. Identifying low variance pathways for free energy calculations of molecular transformations in solution phase. *The Journal of Chemical Physics*. 2011 07; 135(3):034114. doi: [10.1063/1.3607597](https://doi.org/10.1063/1.3607597).
- [16] **Shirts MR**, Chodera JD. Statistically optimal analysis of samples from multiple equilibrium states. *J Chem Phys*. 2008 Sep; 129(12):124105. doi: [10.1063/1.2978177](https://doi.org/10.1063/1.2978177).
- [17] **Vanommeslaeghe K**, Hatcher E, Acharya C, Kundu S, Zhong S, Shim J, Darian E, Guvench O, Lopes P, Vorobyov I, Mackerell AD. CHARMM general force field: A force field for drug-like molecules compatible with the CHARMM all-atom additive biological force fields. *J Comput Chem*. 2010 Mar; 31(4):671–690. doi: [10.1002/JCC.21367](https://doi.org/10.1002/JCC.21367).
- [18] **Vanommeslaeghe K**, MacKerell AD. Automation of the CHARMM general force field (CGenFF) I: Bond perception and atom typing. *J Chem Inf Model*. 2012 Dec; 52(12):3144–3154. doi: 10.1021/CI300363C.
- [19] **Vanommeslaeghe K**, Raman EP, MacKerell AD. Automation of the CHARMM General Force Field (CGenFF) II: Assignment of Bonded Parameters and Partial Atomic Charges. *J Chem Inf Model*. 2012 Dec; 52(12):3155–3168. doi: 10.1021/ci3003649.
- [20] **Yu W**, He X, Vanommeslaeghe K, MacKerell AD. Extension of the CHARMM general force field to sulfonyl-containing compounds and its utility in biomolecular simulations. *J Comput Chem*. 2012 Dec; 33(31):2451–2468. doi: [10.1002/jcc.23067](https://doi.org/10.1002/jcc.23067).