Accurate Free Energy Estimations of Molecular Systems Via Flow-based Targeted Free Energy Perturbation

Anonymous authors
Paper under double-blind review

Abstract

The Targeted Free Energy Perturbation (TFEP) method aims to overcome the time-consuming and computer-intensive stratification process of standard methods for estimating the free energy difference between two states. To achieve this, TFEP uses a mapping function between the high-dimensional probability densities of these states. The bijectivity and invertibility of normalizing flow neural networks fulfill the requirements for serving as such a mapping function. Despite its theoretical potential for free energy calculations, TFEP has not yet been adopted in practice due to challenges in entropy correction, limitations in energy-based training, and mode collapse when learning density functions of larger systems with a high number of degrees of freedom. In this study, we expand flow-based TFEP to systems with variable number of atoms in the two states of consideration by exploring the theoretical basis of entropic contributions of dummy atoms, and validate our reasoning with analytical derivations for a model system containing coupled particles. We also extend the TFEP framework to handle systems of hybrid topology, propose auxiliary additions to improve the TFEP architecture, and demonstrate accurate predictions of relative free energy differences for large molecular systems. Our results provide the first practical application of the fast and accurate deep learning-based TFEP method for biomolecules and introduce it as a viable free energy estimation method within the context of drug design.

1 Introduction

Free energy calculation methods such as Free Energy Perturbation (FEP) (Zwanzig, 1954) are valuable in the field of Computer-aided Drug Design (CADD) to evaluate binding affinities between candidate compounds and receptors, or other biomolecular interactions (Brown et al., 2010; Pohorille et al., 2010; De Vivo et al., 2016). Although useful in cases when experimental testing is unfeasible, high computational cost remains the current drawback for otherwise reliable and accurate in silico methods. For example, FEP calculations for the free energy difference between two thermodynamic states requires a multitude of stratified molecular dynamics (MD) simulations so that the distributions of the explored configuration space by these intermediates sufficiently overlap to achieve convergence (Section A.1).

Targeted Free Energy Perturbation (TFEP) has been presented as a potential alternative method for free energy calculation by applying a generalized FEP identity (Equation 1) whereby an invertible, high-dimensional mapping function is used to transform a distribution of one system or state to another (Jarzynski, 2002; Hahn & Then, 2009). The invertible and bijective normalizing flow neural network has been suggested as a solution to overcoming the difficulty of formulating the required complex mapping functions in order to achieve overlap of configuration space distributions, and its application for TFEP has previously been demonstrated on a growing soft sphere solute in a solvated box (Wirnsberger et al., 2020).

Despite early success, several challenges remained that hindered immediate application of the TFEP method to larger systems with many more degrees of freedom (DOFs) which follow more complex density functions, such as biomolecules. In our study, we expand on the flow-based TFEP method by addressing the following points:

- The bijectivity of flow-based generative neural networks must be upheld by keeping the DOF consistent throughout, a critical point considering many biomolecular free energy studies are performed to compare systems of different sizes. We convert systems to hybrid topologies where deleted atoms are replaced by dummy atoms and inserted atoms are generated by geometric proposal engines.

- The use of dummy atoms in flow-based TFEP for single free energy difference calculations (ΔF), as opposed to dual free energy differences ($\Delta\Delta F$), leads to inaccuracies due to their entropic contributions. We first delineate this effect through analytical derivations using the coupled-particle toy system. We then propose two different methods of circumventing free energy estimation variance due to the presence of dummy atoms. The first method is incorporation of an auxiliary flow model to the TFEP flow architecture. The second method uses the thermodynamic cycle for relative free energy difference calculation to cancel out entropic effects resulting from dummy atoms.

- The difficulty in training a flow-based neural network rises concomitantly with increases in the DOF. A secondary benefit of adding the auxiliary flow is that it approximates a complex target probability density function which can then be set as prior for the free energy difference estimating bijector. The similarity between the prior and target densities reduces the scale and distance of transformation necessary by flow and impedes mode-seeking behavior that often leads to incorrect learning.

- We use our extended TFEP method to demonstrate successful prediction of free energy differences for two different tasks commonly studied in chemical biology and drug discovery: Computation of hydration free energy and free energy differences of protein stability due to mutagenesis. The results are comparable in accuracy to established computational methods at significantly reduced computational costs due to circumvention of long simulations.

2 RELATED WORKS

2.1 TARGETED FREE ENERGY PERTURBATIONS

TFEP is an elegant approach to free energy difference estimation devised by Jarzynski (2002), extending Zwanzig's FEP identity function (Zwanzig, 1954). For two thermodynamic states A and B, the relationship between the true free energy difference ΔF of the two states can be recovered using a generalized estimator Φ instead of potential energy differences ΔU:

$$e^{-\beta\Delta F} = \mathbb{E}_A\left[e^{-\beta\Phi_{A\to A'}}\right], \tag{1}$$

where β denotes the thermodynamic beta and A' the new proposal distribution so that for configurations $\mathbf{x} \sim A$, the mapping is $M(\mathbf{x}) \sim A'$. In the forward transfer mapping A to B, expectations are taken with respect to the equilibrium density of A, $\rho_A \propto e^{-\beta U_A}$ (similar for B when mapping the reverse from B to A).

The neural network perfoming the role of TFEP mapping function $M : A \to A'$ must be invertible so that $M^{-1} : B \to B'$. The generalized energy differences for forward and reverse directions are defined as the difference in potential energies between target and base, from where the log determinant of the Jacobian (J) associated with the respective map direction has been subtracted:

$$\Phi_F(\mathbf{x}) = U_B(M(\mathbf{x})) - U_A(\mathbf{x}) - \beta^{-1}\log|J_M(\mathbf{x})|$$
$$\Phi_R(\mathbf{x}) = U_A(M^{-1}(\mathbf{x}) - U_B(\mathbf{x}) - \beta^{-1}\log|J_{M^{-1}}(\mathbf{x})|. \tag{2}$$

Additional details are provided in Section A.2.

TFEP harnesses a path-independent mapping function to overlap the configuration space distribution between reference and target state. With a correctly formulated mapping function in place, convergence of the free energy differences is instantaneous and can significantly accelerate free energy predictions compared to conventional methods dependent on simulations which require a lengthy sequence of energy calculations per stratification step. The bottleneck thus far has been the difficulty in formulating such a complex mapping function, for which normalizing flow neural networks have recently been presented as a solution Wirnsberger et al. (2020); Rizzi et al. (2021); Falkner et al. (2022). Normalizing flows rely on identical dimensionality between reference and target spaces for bijector transforms in order to perform exact density estimations, a point that is discussed further in Section 3.
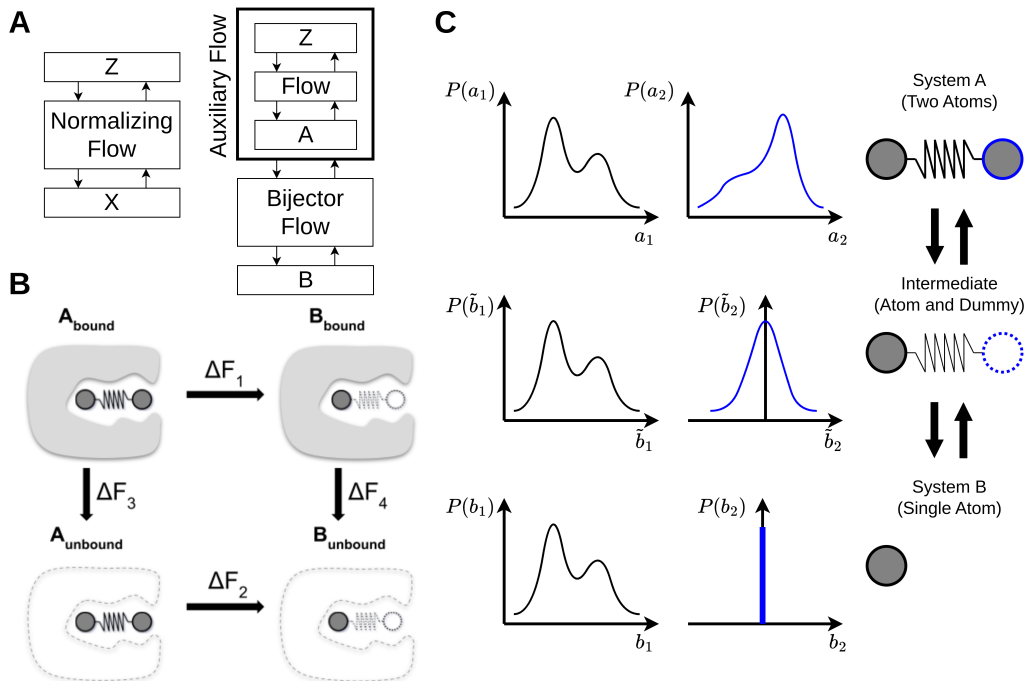
Figure 1: Schematics for the toy system example. (A) Comparison of a standard normalizing flow (left) and an auxiliary flow stacked on a bijector flow (right). Normalizing flows conventionally learn mapping between a data distribution $\hat{p}_X$ for data space X and a prior distribution p_Z (typically Gaussian) for reference space Z. (B) Schematic of the thermodynamic cycle for the coupled particle system. Dotted line particles represent dummy atoms. (C) Scheme representing the process of atom annihilation and its corresponding probability density. The probability density of the annihilated particle is transformed to a Gaussian distribution (intermediate state, with dummy atom). The theoretical mapping of this density to a Dirac delta function cannot be modeled by flow-based methods but is computed by analytical means.

2.2 Normalizing Flows and Boltzmann Generators

The architecture of the mapping function M is crucial for accurate free energy prediction using the TFEP method because convergence must be reached based on a finite number of samples obtained from the two end states A and B. Previous reports exhibit the effectiveness of normalizing flows (Papamakarios et al., 2019; Rezende et al., 2020) in free energy calculations (Ding & Zhang, 2021a;b; Noé et al., 2019), which are transformations of variables z from a known prior density such as a Normal distribution to those from a more complex distribution, $x = f(z)$. Boltzmann Generators are a type of flow-based generative model that learns such diffeomorphisms for molecular structures with many DOFs (Noé et al., 2019). Additional details are provided in Section A.3.

3 Entropic Contributions of Dummy Atoms

As reported by Wirnsberger et al. (2020), normalizing flows satisfy requirements for mapping function implementation in TFEP in that they are bijective, allow for efficient computation of the inverse and the Jacobian determinant, and are highly flexible. The significance of bijectivity is that although DOFs need to stay conserved for flow-based TFEP, the systems being compared do not always share the same number of atoms for many free energy questions posed in computational chemistry. To preserve the same dimensionality throughout the bijective transformation layers of normalizing flow, dummy atoms can be used as placeholders and the systems can be represented with a hybrid topology.

We designed a toy system where states A and B differ by a single particle to test our treatment of dummy atoms within the TFEP method, and validate that the differences in the Jacobian term are equivalent to the analytically derived entropy of a Normal distribution function. If a transfer mapping function $M : A \rightarrow A'$ with target B is optimally formulated such that $A' = B$, the Jacobian term from the TFEP generalized estimator $\log|J_M(\mathbf{x})|$ represents the entropy difference between A and B.

To numerically derive the entropic contribution by dummy atoms, we use a 2-dimensional coupled-harmonic particle system (Figure 1B, C), defined by the energy functions provided in Section A.4. These potential functions confine movement of the particles to a predefined range of coordinate space to emulate a physical harmonic bond term, and represent biological ligands in bound and unbound states. The objective is to analytically solve for the explicit entropy contribution to free energy difference predictions by the dummy particle that replaces the deleted atom.

The model we use for TFEP free energy estimation is a combination of two stacked flows as shown in Figure 1A. We construct an auxiliary flow as an RQNSF-based BG (Durkan et al. (2019a), Section A.7) of 4 affine coupling layers with alternating even and odd binary masking using the nflows library (Durkan et al., 2020). Supposing we perform a transfer mapping of B to A, this auxiliary flow is foremost trained as a density estimator for A (120 epochs, 4096 batch size, $5e^{-4}$ learning rate) by maximum likelihood. The prior is set as a Normal distribution function and the target as A.

The auxiliary flow is stacked on top of a second bijector flow, which is the mapping function between A and B and responsible for the free energy difference estimation between the two systems. The bijector is constructed and trained similarly to the auxiliary flow, the only difference being that the prior is set as the estimated density of the auxiliary flow (the learned A), and the target is B. Training is conducted via maximum likelihood so that samples from B are transformed such that the generated distribution B' has maximized overlap in the domain of the distribution of the auxiliary flow-estimated A. Finally, the TFEP loss function is used to evaluate the transformation.

The advantage of using a dual flow is that the learned prior and target of the bijector occupy distributions that are more similar than a normalizing flow with a noise prior, heightening the efficiency and accuracy of training (Figure S2). Invertible transformations between similar distributions are more straightforward and easier to find because they preserve the general structure of the distributions and can map similar regions to each other. This method avoids increased complexity in the transformation that can lead to difficulty in optimization, risk of overfitting, intractability in computation, and burdens in memory requirements.

In contrast, training a bijector alone with no auxiliary density estimate would be driven by minimization of Kullback-Leibler (KL) divergence, which leads to mode-seeking behavior. For larger systems that follow highly complex and multimodal density functions, this behavior leads to difficulties in convergence and ultimately inaccurate free energy difference predictions.

Table 1: Entropy Effects on Single Free Energy Differences

	Ground truth		Flow	
	Bound	Unbound	Bound	Unbound
ΔF	-1.853	-0.470	-3.300	-1.950
$\Delta\Delta F$	-1.383		-1.350	

As shown in Table 1, we obtained single free energy differences (ΔF) for atom deletion between prior (A, coupled two-particle system) and target (B, single-particle system) using the dual flow method with TFEP. In the bound state, the difference in ΔF between ground truth and flow is 1.447, and that for unbound state is 1.480. These values are indeed comparable with the known entropy of a Normal distribution, $S = \ln\sqrt{2\pi} + \frac{1}{2} = 1.4189$. We are therefore able to demonstrate with our coupled particle deletion example that a dummy atom defined by a Gaussian will contribute entropically to single free energy difference estimations, and it is crucial to take this into consideration when using the TFEP method for free energy studies of molecular systems with changes in DOFs.

4 CONSTRUCTION OF HYBRID TOPOLOGY

Relative free energy difference studies in drug discovery often involve end states that have different numbers of atoms, or DOFs. To preserve the bijective nature of normalizing flow for the TFEP method and simultaneously allow the method to be applicable for systems of different size and topology, we implement a hybrid topology approach. Hybrid topologies fuse two systems and map their corresponding atoms to each other (Figure 2A). In this section, we describe the particle types in hybrid topology, the setup for traditional relative free energy calculations by FEP approach (RFEP), and how it is integrated in our proposed deep learning model.

Particles in a hybrid topology are assigned to one of four groups: environment, core, unique old, and unique new. Suppose we are estimating free energy differences between Molecule A and Molecule B. All atoms of common residues in A and B are environment atoms, while common atoms on the differing residue are designated core atoms (e.g. the C-alpha atom of the mutated residue in the Trpcage W6F study, Section 7). When considering a transformation in the direction from A to B, unique atoms for A would be considered unique old and those for B would be considered unique new.

RFEP can be divided into three stages: First, a hybrid 2D topology is generated from single topologies of Molecules A and B. Second, the topology and coordinates of Molecule A are used as input to initiate a hybrid 3D system. Third, atoms are generated using a geometric proposal engine (Additional details, Section A.6).

For our study, we follow strategies implemented in the Perses framework (Rufa et al., 2022) that use a variant of maximum common substructure algorithm from OEChem TK (OpenEye Scientific Software, Santa Fe, NM) and the force constant and equilibrium state of dummy atoms.
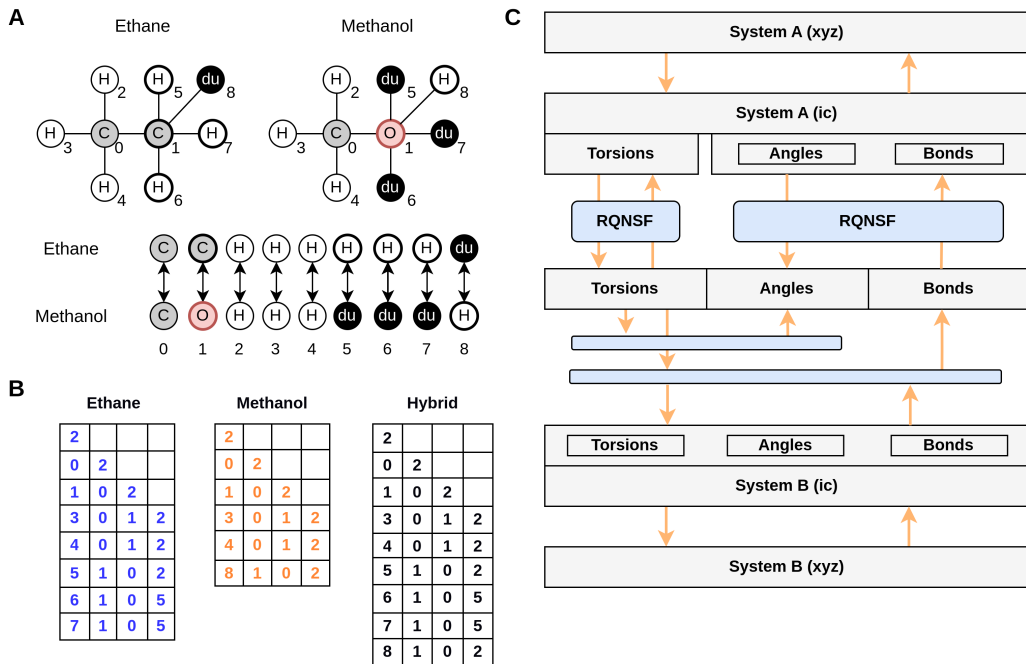

Figure 2: Flow-based TFEP model implementations. (A) Hybrid topology for ethane and methanol with dummy atoms (du) inserted for consistent DOFs in bijective transformation via flow models. Environment atoms are numbered 0, 2-4, core atoms are numbered 1, and unique atoms are numbered 5-8 and marked with a bold outline. Black bidirectional arrows indicate the mapping of each atom to its corresponding atom in the other system. (B) Z-matrices for ethane, methanol, and the hybrid systems. Atom numbering identical to (A). (C) Schematic summarizing the flow architecture that transforms between base System A and target System B. RQNSF coupling layers are represented as blue blocks, Cartesian coordinates are labeled 'xyz', and internal coordinates are 'ic'.

5   ARCHITECTURE AND METHOD DESIGN

Variations of normalizing flow networks have been published, where some are permutation equivariant but lacking rotation-translation equivariance (Wirnsberger et al., 2020) and others exhibit E(n) equivariance at the expense of learning instability (Satorras et al., 2021). We base our implementation (Fig. 2C) on the BG concept (Noé et al., 2019) with the following minor adjustments:

(i) We opt for a rotation-translation invariant solution by leveraging the space of internal coordinates as a product of hypertori and compact intervals, similar to previous literature (Noé et al., 2019; Köhler et al., 2021; Invernizzi et al., 2022), and bookend the bijector flow with layers that transform Cartesian to internal coordinates or vice versa. These are non-trainable layers that transform Euclidean coordinates $\boldsymbol{x} \in \mathbb{R}^{n \times 3}$ into bonds $\boldsymbol{d} \in [a_1, b_1] \times \ldots \times [a_{n-1}, b_{n-1}]$, angles $\boldsymbol{\alpha} \in [0, \pi]^{n-2}$, and torsions $\boldsymbol{\tau} \in \mathbb{T}^{n-3}$. The internal coordinate transformation is performed by rational quadratic spline flows (Durkan et al., 2019a). The model learns a joint distribution $p(\boldsymbol{d}, \boldsymbol{\alpha}, \boldsymbol{\tau})$ on the topological space $\mathcal{X}_{\mathrm{IC}} := \mathbb{I}^{2n-3} \times \mathbb{T}^{n-3}$, where the closed unit interval is $\mathbb{I} = [0, 1]$.

(ii) We set priors as nontrivial probability distributions that are sampled via MD. MD simulations are performed using the hybrid topology for each end state (lambda 0 and 1) in Cartesian coordinates from which we obtain a representative set of configurations for training in the forward (A to B) or reverse direction (B to A). Due to the similarity between the prior and the target, the learning by energy scheme (Noé et al. (2019), Section A.3) is much smoother and we are able to train at fast speed by using a simple flow architecture with only a few coupling layers.

Normalizing flows are capable of tractable transformations, but they are not necessarily expressive enough to capture any and all complex mapping from a naive prior to a high-dimensional, complex target distribution. Training a BG in practice may also be difficult because energy-based training is characterized by mode-seeking behavior and struggles to converge reliably when the target distribution is multimodal. Our method overcomes this by designing a mapping between similar density functions, which makes the transform more straightforward and enables the model to train quickly and accurately for molecular systems.

5.1   CANCELLATION OF DUMMY ENTROPY

For end states A and B that represent the initial Molecule A (λ=0, where λ represents an alchemical coupling parameter) and final Molecule B (λ=1), we perform simulations using a hybrid topology as described in Section 4, where dummy atoms are either turned on or annihilated during the transformation depending on which end state molecule they represent. Dummy atoms are connected to the common core through alchemical bonded force field terms, which can skew energies calculated for the end states if they are placed with disregard to physical context. Additionally, the DOFs of dummy atoms are included in the mapping between A and B and contribute entropically to the free energy difference.

The first method of balancing these energetic and entropic contributions by implementing an auxiliary flow as a density estimator for the free energy predicting bijector has been previously demonstrated using a toy system in Section 3. As a second approach, we take advantage of the dummy atoms decoupling from the core environment with regards to their nonbonded interactions in their annihilated state. This causes the configurational samples of the dummy atoms and their energetic evaluation to be independent of the core and environment. Moreover, these independent energetic and entropic contributions of the dummy atoms cancel each other out in parallel legs of thermodynamic cycles. Dummy atoms in the solvated and vacuum states cancel each other out for solvation free energy calculations, those in folded and unfolded states do so for mutation free energy calculations, and those in bound and unbound complexes also for relative free energies of binding. As a result, it is unnecessary to correct for the energetic and entropic contributions of annihilated dummy atoms for single free energy differences (ΔF) of each transformation leg separately. Instead, we directly compute the dual free energy difference ($\Delta \Delta F$) from the full thermodynamic cycle.
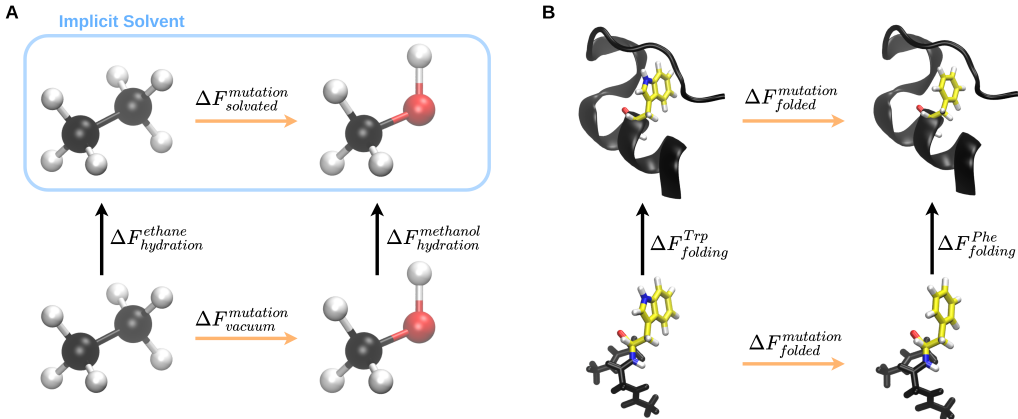
Figure 3: Thermodynamic cycles for application example systems. (A) Thermodynamic cycle for mutation between ethane and methanol. (B) Thermodynamic cycle for W6F residue mutation in Trp-cage miniprotein. Legs of the thermodynamic cycle where the flow-based TFEP mapping function is used are marked orange.

6   APPLICATION ON SMALL MOLECULE MUTATION

The energy difference when a molecule transitions from vacuum to a solvated state at constant temperature and pressure is represented by solvation free energy, $\Delta F^{\text{solvation}}$ (or hydration free energy in cases where the solvent is water). Solvation free energy difference estimations are pertinent to computational chemistry and drug design and traditionally require stratified, alchemical methods (Section A.8). Commonly, a series of nonphysical intermediates parameterized by the alchemical coupling parameter λ are simulated to ultimately transfer a solute from gas phase to solvent or vice versa.

We test our TFEP method on prediction of hydration free energy difference between ethane and methanol. From the thermodynamic cycle in Figure 3A, we see that $\Delta\Delta F_{hydration} = \Delta F_{hydration}^{methanol} - \Delta F_{hydration}^{ethane} = \Delta F_{solvated}^{mutation} - \Delta F_{vacuum}^{mutation}$. In this section, we investigate two pathways. The first is the solvation pathway adapted from Duarte Ramos Matos et al. (2017), where ethane and methanol are respectively flow transformed from water to vacuum or vice versa. The energy functions that calculate potential energy for each end state include terms and parameters for either vacuum or implicit solvent.

Table 2: Free energy difference estimation from solvation legs.

	Experimental	Vac $\rightarrow$ Water	Water $\rightarrow$ Vac	Gromacs TI
Ethane	1.83	1.89	-1.91	2.46
Methanol	-5.10	-4.75	4.71	-3.49
Hydration $\Delta\Delta F$ (kcal/mol)	-6.93	-6.64		-5.95
Dehydration $\Delta\Delta F$ (kcal/mol)			6.62	

For the second mutation pathway experiment, we use the hybrid z-matrix topology scheme constructed by merging the z-matrices of ethane and methanol as shown in Figure 2B. We compare the flow-based free energy estimations (NF) with reference methods thermodynamic integration (SOMD-TI) and Multiple Bennett Acceptance Ratio (SOMD-MBAR) (Section A.8) following alchemical free energy calculation molecular dynamics simulations run by the OpenMM-interfacing SOMD engine Loeffler et al. (2015); Hedges et al. (2019). NF results are comparable to other methods as shown in Table 3. The loss profile of the transformation of ethane to methanol in solvated and vacuum states and the solvation free energy is given in Figure S4.

Table 3: Comparison with alternative free energy calculation methods.

	NF	SOMD-TI	SOMD-MBAR
Ethane $\rightarrow$ Methanol (kcal/mol)	-5.75	-6.02	-6.16

7 APPLICATION ON PROTEIN RESIDUE MUTATION

Trp-cage or TC5b (Neidigh et al., 2002) is a 20-residue (NLYIQWLKDGGPSSGRPPPS) miniprotein with a two-state folding mechanism that has been featured extensively in energy and conformation change studies (Simmerling et al., 2002; Snow et al., 2002; Chowdhury et al., 2003; Nikiforovich et al., 2003; Pitera & Swope, 2003; Zhou, 2003; Ding & Dokholyan, 2005; Linhananta et al., 2005; Sidky et al., 2019; Juraszek & Bolhuis, 2006). The key stabilizing components are the six residues making up the buried hydrophobic core (TYR3, TRP6, LEU7, GLY11, PRO12, PRO19), four prolines that minimize ΔS_U, and a salt bridge interaction.

Many single-site mutation studies have been performed on Trp-cage, both empirically and computationally. Among these we use W6F, which reportedly leads to completely unfolding in water with destabilization of $\Delta\Delta G_F = 12.5 \pm 0.6$ kJ/mol (Barua et al., 2008).

Tripeptides and mutation structure files were created and prepared using Schrödinger Maestro (wildtype PDB ID: 1L2Y). Explicitly solvated structures were prepared using the pdb4amber tool from the AmberTools MD package (Case et al., 2020) and MD simulations were run using OpenMM (Eastman et al., 2017). Frames from a 40 ns simulation were collected at equal intervals and used as starting states for multiple short (1 ns) simulations. All spawned simulations were concatenated as a single 40 ns trajectory of 400,000 frames to be used as training input structures to ensure statistically converged results (Genheden & Ryde, 2010). For all simulations, nonbonded cutoff was 0.9 nm, temperature 300 K, timestep 1fs, and a Langevin dynamics integrator was used with 1 ps^{-1} friction coefficient. Solvent molecules were removed and potential energy was calculated implicitly prior to network training (Paschek et al. (2007), Section 8).

We obtained flow-based TFEP free energy difference estimations for W6F mutation (NF) and compared the results with predictions from a non-equilibrium switching method implemented by open source software pmx (Seeliger & De Groot, 2010; Gapsys et al., 2015). We also compared estimations from equilibrium replica-exchange via lambda hopping (FEP-REMD) as implemented by Schrödinger (Wang et al., 2015).

Table 4: Results for Trp-cage W6F mutation free energy estimation.

	Experimental	NF	pmx	FEP-REMD (MBAR)
W6F (kcal/mol)	2.98	3.18	3.48	3.19

As shown in Table 4, results for the flow-based TFEP method are comparable with both experimental and computational free energy difference values.

8 CONCLUSION AND DISCUSSION

In this work, we introduce a range of contributions to the concept of flow-based free energy difference estimations and related physical problems. The objective of the proposed approach is to provide an alternative computational method of free energy estimation that performs at accelerated speeds compared to conventional methods by eliminating the need for a multitude of computationally expensive, stratified MD simulations.

To summarize, we first acknowledge dummy atoms are necessary for preserving bijector dimensionality and these dummies should not influence the free energy calculations. We then analytically derive the Jacobian contribution in the TFEP generalized estimator by these dummy particles using a toy system.

In the next section, we propose integration of a hybrid topology scheme to the TFEP method as an efficient way of preserving dimensionality for molecular systems in normalizing flows. We also discriminate our method from normalizing flow-based BGs (Noé et al., 2019; Invernizzi et al., 2022; Mahmoud et al., 2022) by mapping between similar prior and target instead of between a noise prior and complex target, enhancing efficiency and the capacity to train larger systems.

We also provide two approaches to ensure that deep-TFEP free energy difference estimations do not suffer from variance due to dummy atoms. The first is an architectural solution, stacking an auxiliary flow density estimator on the bijector as a prior. The second is by taking advantage of double free energy difference settings, under assumption that free energy differences for parallel legs of the thermodynamic cycle will contain the same entropic contributions that cancel out. A caveat to note is that this assumption may not hold true in all cases Boresch & Karplus (1996); Fleck et al. (2021), indicating a need for further study.

Regarding training data for our deep generative models, we used Generalized Born (GB)-solvated data samples. Although better agreement can be achieved with experiments using explicit solvent, the choice for implicit solvation was deliberate due to challenges in modelling ensemble distributions that included explicit solvent molecules. Furthermore, inclusion of waters greatly increases the system size (DOFs) and also introduces permutation symmetry. Future studies on this topic may attempt a combination of promising developments in flow-based generative models for large systems with permutational symmetry (Noé et al., 2019; Wirnsberger et al., 2020; Köhler et al., 2023) with the current TFEP approach. Another possible direction for improving the current approach is to enhance physical accuracy by integrating deep learning potentials that implicitly account for explicit water (Chen et al., 2021; Mahmoud et al., 2020; Ghanbarpour et al., 2020).

REFERENCES

Bipasha Barua, Jasper C Lin, Victoria D Williams, Phillip Kummler, Jonathan W Neidigh, and Niels H Andersen. The trp-cage: optimizing the stability of a globular miniprotein. *Protein Engineering, Design & Selection*, 21(3):171–185, 2008.

Stefan Boresch and Martin Karplus. The jacobian factor in free energy simulations. *The Journal of chemical physics*, 105(12):5145–5154, 1996.

Scott Brown, Michael Shirts, and David Mobley. *Free-energy calculations in structure-based drug design*, pp. 61–86. Cambridge University Press, 05 2010. ISBN 978-0-521-88723-6.

David A Case, Kellon Belfon, Ido Ben-Shalom, Scott R Brozell, David S Cerutti, Thomas EI Cheatham, Vinícius WD Cruzeiro, Tom A Darden, Robert E Duke, George Giambasu, et al. Amber 2020: University of california. *San Francisco*, 2020.

Yaoyi Chen, Andreas Krämer, Nicholas E Charron, Brooke E Husic, Cecilia Clementi, and Frank Noé. Machine learning implicit solvation for molecular dynamics. *The Journal of Chemical Physics*, 155(8):084101, 2021.

Shibasish Chowdhury, Mathew C Lee, Guoming Xiong, and Yong Duan. Ab initio folding simulation of the trp-cage mini-protein approaches nmr resolution. *Journal of molecular biology*, 327 (3):711–717, 2003.

Marco De Vivo, Matteo Masetti, Giovanni Bottegoni, and Andrea Cavalli. Role of molecular dynamics and related methods in drug discovery. *Journal of medicinal chemistry*, 59(9):4035–4061, 2016.

Feng Ding and Nikolay V Dokholyan. Simple but predictive protein models. *Trends in biotechnology*, 23(9):450–455, 2005.

Xinqiang Ding and Bin Zhang. Computing absolute free energy with deep generative models. *Biophysical Journal*, 120(3):195a, 2021a.

Xinqiang Ding and Bin Zhang. Deepbar: a fast and exact method for binding free energy computation. *The journal of physical chemistry letters*, 12(10):2509–2515, 2021b.

Guilherme Duarte Ramos Matos, Daisy Y Kyu, Hannes H Loeffler, John D Chodera, Michael R Shirts, and David L Mobley. Approaches for calculating solvation free energies and enthalpies demonstrated with an update of the freesolv database. *Journal of Chemical & Engineering Data*, 62(5):1559–1569, 2017.

Conor Durkan, Artur Bekasov, Iain Murray, and George Papamakarios. Neural spline flows, 2019a. URL https://arxiv.org/abs/1906.04032.

Conor Durkan, Artur Bekasov, Iain Murray, and George Papamakarios. Neural spline flows. *Advances in neural information processing systems*, 32, 2019b.

Conor Durkan, Artur Bekasov, Iain Murray, and George Papamakarios. nflows: normalizing flows in PyTorch, November 2020. URL https://doi.org/10.5281/zenodo.4296287.

Peter Eastman, Jason Swails, John D Chodera, Robert T McGibbon, Yutong Zhao, Kyle A Beauchamp, Lee-Ping Wang, Andrew C Simmonett, Matthew P Harrigan, Chaya D Stern, et al. Openmm 7: Rapid development of high performance algorithms for molecular dynamics. *PLoS computational biology*, 13(7):e1005659, 2017.

Sebastian Falkner, Alessandro Coretti, Salvatore Romano, Phillip Geissler, and Christoph Dellago. Conditioning normalizing flows for rare event sampling. *arXiv preprint arXiv:2207.14530*, 2022.

Markus Fleck, Marcus Wieder, and Stefan Boresch. Dummy atoms in alchemical free energy calculations. *Journal of Chemical Theory and Computation*, 17(7):4403–4419, 2021.

Vytautas Gapsys, Servaas Michielssens, Daniel Seeliger, and Bert L De Groot. pmx: Automated protein structure and topology generation for alchemical perturbations, 2015.

Samuel Genheden and Ulf Ryde. How to obtain statistically converged mm/gbsa results. *Journal of computational chemistry*, 31(4):837–846, 2010.

Ahmadreza Ghanbarpour, Amr H Mahmoud, and Markus A Lill. Instantaneous generation of protein hydration properties from static structures. *Communications Chemistry*, 3(1):188, 2020.

Aljoscha M Hahn and H Then. Using bijective maps to improve free-energy estimates. *Physical Review E*, 79(1):011113, 2009.

Lester Hedges, Antonia Mey, Charles Laughton, Francesco Gervasio, Adrian Mulholland, Christopher Woods, and Julien Michel. Biosimspace: An interoperable python framework for biomolecular simulation. *Journal of Open Source Software*, 4(43), 2019.

Michele Invernizzi, Andreas Krämer, Cecilia Clementi, and Frank Noé. Skipping the replica exchange ladder with normalizing flows. *The Journal of Physical Chemistry Letters*, 13:11643–11649, 2022.

Christopher Jarzynski. Targeted free energy perturbation. *Physical Review E*, 65(4):046122, 2002.

Jarek Juraszek and Peter G Bolhuis. Sampling the multiple folding mechanisms of trp-cage in explicit solvent. *Proceedings of the National Academy of Sciences*, 103(43):15859–15864, 2006.

Jonas Köhler, Andreas Krämer, and Frank Noé. Smooth normalizing flows, 2021. URL https://arxiv.org/abs/2110.00351.

Jonas Köhler, Michele Invernizzi, Pim de Haan, and Frank Noé. Rigid body flows for sampling molecular crystal structures. *arXiv preprint arXiv:2301.11355*, 2023.

Apichart Linhananta, Jesse Boer, and Ian MacKay. The equilibrium properties and folding kinetics of an all-atom g ō model of the trp-cage. *The Journal of chemical physics*, 122(11):114901, 2005.

Hannes H Loeffler, Julien Michel, and Christopher Woods. Fesetup: automating setup for alchemical free energy simulations, 2015.

Amr H Mahmoud, Matthew R Masters, Ying Yang, and Markus A Lill. Elucidating the multiple roles of hydration for accurate protein-ligand binding prediction via deep learning. *Communications Chemistry*, 3(1):19, 2020.

Amr H Mahmoud, Matthew Masters, Soo Jung Lee, and Markus A Lill. Accurate sampling of macromolecular conformations using adaptive deep learning and coarse-grained representation. *Journal of Chemical Information and Modeling*, 62(7):1602–1617, 2022.

Jonathan W Neidigh, R Matthew Fesinmeyer, and Niels H Andersen. Designing a 20-residue protein. *Nature structural biology*, 9(6):425–430, 2002.

Gregory V Nikiforovich, Niels H Andersen, R Matthew Fesinmeyer, and Carl Frieden. Possible locally driven folding pathways of tc5b, a 20-residue protein. *Proteins: Structure, Function, and Bioinformatics*, 52(2):292–302, 2003.

Frank Noé, Simon Olsson, Jonas Köhler, and Hao Wu. Boltzmann generators: Sampling equilibrium states of many-body systems with deep learning. *Science*, 365(6457):eaaw1147, 2019. doi: 10.1126/science.aaw1147. URL https://www.science.org/doi/abs/10.1126/science.aaw1147.

George Papamakarios, Eric Nalisnick, Danilo Jimenez Rezende, Shakir Mohamed, and Balaji Lakshminarayanan. Normalizing flows for probabilistic modeling and inference, 2019. URL https://arxiv.org/abs/1912.02762.

Dietmar Paschek, Hugh Nymeyer, and Angel E García. Replica exchange simulation of reversible folding/unfolding of the trp-cage miniprotein in explicit solvent: On the structure and possible role of internal water. *Journal of structural biology*, 157(3):524–533, 2007.

Jed W Pitera and William Swope. Understanding folding and design: Replica-exchange simulations of "trp-cage" miniproteins. *Proceedings of the National Academy of Sciences*, 100(13):7587–7592, 2003.

Andrew Pohorille, Christopher Jarzynski, and Christophe Chipot. Good practices in free-energy calculations. *The Journal of Physical Chemistry B*, 114(32):10235–10253, 2010.

Danilo Jimenez Rezende, George Papamakarios, Sébastien Racanière, Michael S. Albergo, Gurtej Kanwar, Phiala E. Shanahan, and Kyle Cranmer. Normalizing flows on tori and spheres, 2020. URL https://arxiv.org/abs/2002.02428.

Andrea Rizzi, Paolo Carloni, and Michele Parrinello. Targeted free energy perturbation revisited: Accurate free energies from mapped reference potentials. *The Journal of Physical Chemistry Letters*, 12(39):9449–9454, 2021.

Dominic A. Rufa, Ivy Zhang, Hannah E. Bruce Macdonald, Patrick B. Grinaway, Iván Pulido, Mike M. Henry, Jaime Rodríguez-Guerra, Matt Wittmann, Steven K. Albanese, William G. Glass, Ana Silveira, David Schaller, Levi N. Naden, and John D. Chodera. Perses, March 2022. URL https://doi.org/10.5281/zenodo.6328265. Please cite this software using these metadata.

Victor Garcia Satorras, Emiel Hoogeboom, Fabian B. Fuchs, Ingmar Posner, and Max Welling. E(n) equivariant normalizing flows, 2021. URL https://arxiv.org/abs/2105.09016.

Schrödinger. Schrödinger release (2018) 2: Maestro, version 11.8., schrödinger, LLC, new york, NY, 2021., 2021.

Daniel Seeliger and Bert L De Groot. Protein thermostability calculations using alchemical free energy simulations. *Biophysical journal*, 98(10):2309–2316, 2010.

Hythem Sidky, Wei Chen, and Andrew L Ferguson. High-resolution markov state models for the dynamics of trp-cage miniprotein constructed over slow folding modes identified by state-free reversible vampnets. *The Journal of Physical Chemistry B*, 123(38):7999–8009, 2019.

Carlos Simmerling, Bentley Strockbine, and Adrian E Roitberg. All-atom structure prediction and folding simulations of a stable protein. *Journal of the American Chemical Society*, 124(38): 11258–11259, 2002.

Christopher D Snow, Houbi Nguyen, Vijay S Pande, and Martin Gruebele. Absolute comparison of simulated and experimental protein-folding dynamics. *nature*, 420(6911):102–106, 2002.

Lingle Wang, Yujie Wu, Yuqing Deng, Byungchan Kim, Levi Pierce, Goran Krilov, Dmitry Lupyan, Shaughnessy Robinson, Markus K Dahlgren, Jeremy Greenwood, et al. 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. *Journal of the American Chemical Society*, 137 (7):2695–2703, 2015.

Peter Wirnsberger, Andrew J Ballard, George Papamakarios, Stuart Abercrombie, Sébastien Racanière, Alexander Pritzel, Danilo Jimenez Rezende, and Charles Blundell. Targeted free energy estimation via learned mappings. *The Journal of Chemical Physics*, 153(14):144112, 2020.

Ruhong Zhou. Trp-cage: folding free energy landscape in explicit water. *Proceedings of the National Academy of Sciences*, 100(23):13280–13285, 2003.

Robert W Zwanzig. High-temperature equation of state by a perturbation method. i. nonpolar gases. *The Journal of Chemical Physics*, 22(8):1420–1426, 1954.

A APPENDIX

A.1 FREE ENERGY PERTURBATION

The pairwise free energy difference for the transfer between neighboring intermediate states are individually calculated and summed to obtain the overall free energy change:

$$
\begin{aligned}
\Delta F &= F_B - F_A \\
&= (F_B - F_N) + (F_N - F_{N-1}) + ... + (F_1 - F_A) \\
&= -k_B T \sum_{i=0}^{N} \ln \left\langle \exp \left(-\frac{U_{i+1} - U_i}{k_B T} \right) \right\rangle_i
\end{aligned}
$$

The flow-based TFEP method allows us to bypass the computational expense of simulating numerous intermediate states i and accelerate computation time without compromising the accuracy of free energy difference estimation.

A.2 DEEP MAPPING IN TARGETED FREE ENERGY PERTURBATION

The relationship between the Kullback-Leibler Divergence loss function and the generalized estimator has been reformulated by Wirnsberger et al. (2020):

$$
D_{KL} [\rho_{A'} || \rho_B] = \beta(\mathbb{E}_A[\Phi_F] - \Delta F).
$$

When both Φ_F and Φ_R are available, the Bennett Acceptance Ratio (BAR) can be used to predict statistically accurate free energy:

$$
\mathbb{E}_A[f(\beta(\Phi_F - \Delta F))] = \mathbb{E}_B[f(\beta(\Phi_R - \Delta F))],
$$

While not directly related to the free energy calculations, the TFEP loss function has also been recently used to develop LREX to skip the replica-exchange ladder (Invernizzi et al., 2022).

A.3 BOLTZMANN GENERATORS

The multimodal complexity of energy function-dependent Boltzmann distributions requires a generative model that is capable of non-parametric density estimations. Flow-based Boltzmann Generators (BGs) have been reported to successfully obtain configuration samples of small molecular

structures (up to 892 atoms) from the Boltzmann distribution $e^{-U(\mathbf{x})}$ following a training protocol based on valid reference structures from molecular dynamics (MD) and the energy function $U(\mathbf{x})$ Noé et al. (2019). BGs learn invertible coordinate transformations of variables $\mathbf{x} \sim p_X(\mathbf{x})$, configuration states which have high Boltzmann probability, from the latent space variables $\mathbf{z} \sim p_Z(\mathbf{z})$. The latent space distribution follows a simple Gaussian for easy backward propagation derivative calculations, same as other NFs (Figure S1, left). Each latent variable $\mathbf{z}$ from this fixed and prescribed probability distribution function belongs to a unique conditional distribution that is learnable by parameters θ. The task of the BG network is therefore to learn θ for the transformations:

$$\mathbf{x} = M(\mathbf{z}; \theta), \quad \mathbf{z} = M^{-1}(\mathbf{x}; \theta), \tag{3}$$

BGs are tailored for generating states of molecular systems because not only are the models trained using energetically and structurally valid states obtained through MD simulations (*example-based training*), they are also trained on the energy function so that the target distribution $p_X(\mathbf{x}) \propto e^{-U(\mathbf{x})}$ (*energy-based training*). Once q has been learned and optimized, structures that do not violate the Boltzmann distribution can be generated from the trained model in one-shot.
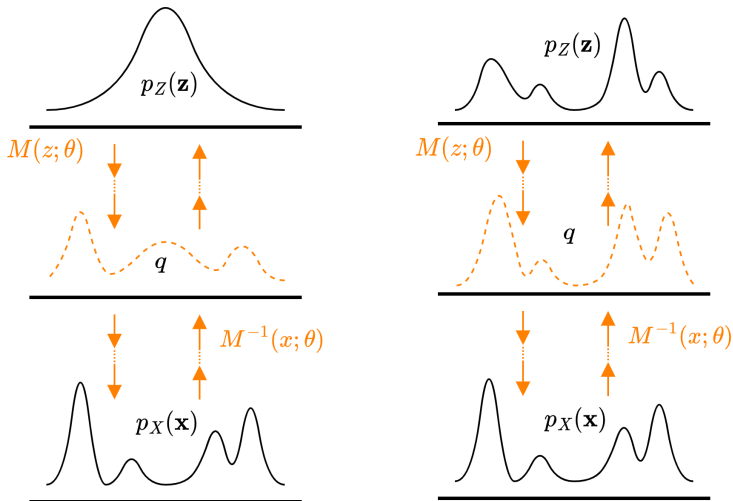

Figure S1: **Schematic for distribution learning in Boltzmann Generators.** The flow-based model learns the unique distribution for a protein system using a sequence of bijective, invertible functions, here depicted as dot-connected orange arrows. The exact prior and exact target are $p_Z(\mathbf{z})$ and $p_X(\mathbf{x}) \propto \exp(-u(\mathbf{x}))$. The BG-generated distribution is q, so that $q_X(\mathbf{x}) = M(\mathbf{z})$ and $q_Z(\mathbf{z}) = M^{-1}(\mathbf{x})$. The prior $p_Z(\mathbf{z})$ is typically a Gaussian for BGs (left). In our method, we set an auxiliary density estimator as prior instead of a Gaussian (right).

A.4 ENERGY FUNCTION DEFINITIONS FOR THE COUPLED PARTICLE SYSTEM.

Bound State

System A
$$V_1(x_1, x_2) = 0.5 \cdot k_1(x_1 - x_1^\circ)^2$$
$$V_2(x_1, x_2) = 0.5 \cdot k_2(x_2 - x_2^\circ)^2$$
$$V_3(x_1, x_2) = 0.5 \cdot \kappa \cdot 0.5(C\tanh(\alpha(x_1 - x_1^\circ)) + 1.0) \cdot ((x_2 - x_1) - d)^2$$
$$V = V_1 + V_2 + V_3$$

System B
$$V_1(x_1) = 0.5 \cdot k_1(x_1 - x_1^\circ)^2,$$

where $k_1 = 5.0, k_2 = 10.0, x_1^\circ = -2.0, x_2^\circ = 2.0, , d = 3.0, \kappa = 10.0, C = 0.5, \alpha = 2.0$.

Unbound State

System A
$$V_1(x_1) = \begin{cases} 0.5 \cdot k(x_1 - x_l)^2, & \text{if } x_1 < x_l \\ 0.5 \cdot k(x_1 - x_r)^2, & \text{if } x_1 > x_r \\ 0, & \text{otherwise} \end{cases}$$

$$V_2(x_2) = \begin{cases} 0.5 \cdot k(x_2 - x_l)^2, & \text{if } x_2 < x_l \\ 0.5 \cdot k(x_2 - x_r)^2, & \text{if } x_2 > x_r \\ 0, & \text{otherwise} \end{cases}$$

$$V_3(x_1, x_2) = 0.5 \cdot \kappa \cdot 0.5(C\tanh(\alpha(x_1 - x_1^\circ)) + 1.0) \cdot ((x_2 - x_1) - d)^2$$
$$V = V_1 + V_2 + V_3$$

System B
$$V_1(x_1) = \begin{cases} 0.5 \cdot k(x_1 - x_l)^2, & \text{if } x_1 < x_l \\ 0.5 \cdot k(x_1 - x_r)^2, & \text{if } x_1 > x_r \\ 0, & \text{otherwise,} \end{cases}$$

where $k = 10.0, x_1^\circ = -2.0, x_l = 3.0, x_r = 3.0, d = 3.0, \kappa = 10.0, C = 0.5, \alpha = 2.0$.
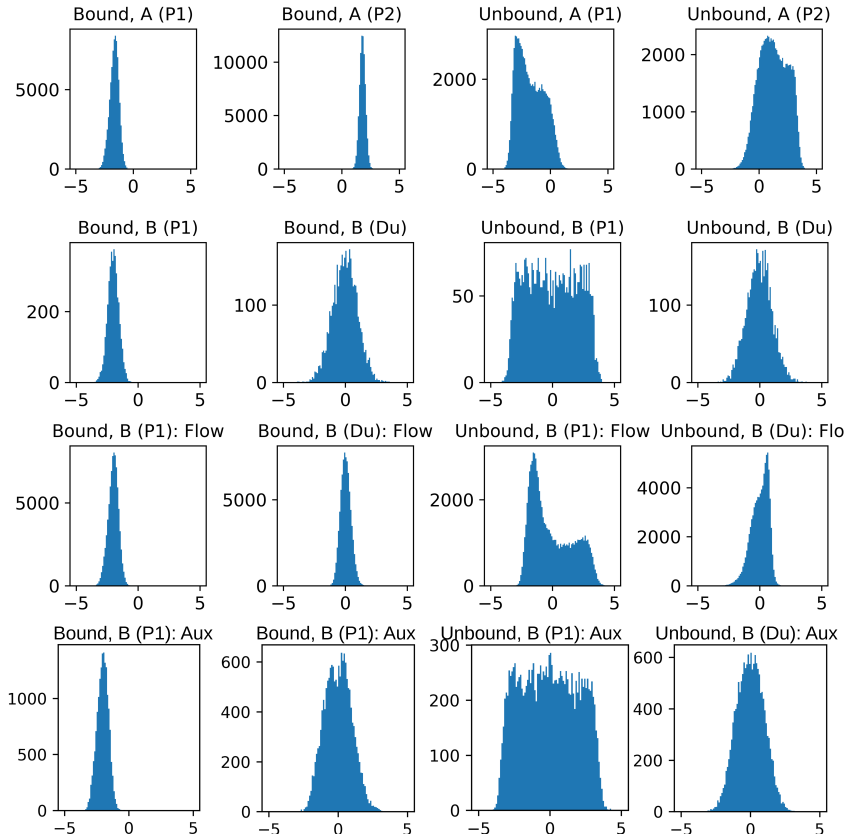
A.5   TRAINING RESULTS FOR DUAL FLOW METHOD.

Figure S2: Sampling histograms for the 2D coupled-particle system. System A is the two-particle system, System B is comprised of the first particle and an uncorrelated dummy atom that is defined by a Gaussian energy distribution with a mean of 0. The left two columns are for the bound state, the right two columns for the unbound state. The first row is the target System A, the second row target System B. The third row is the flow-generated distribution before using an auxiliary flow as prior. The final row shows correct training upon implementing the auxiliary flow. Improvements are especially more pronounced in histograms for the unbound state.

A.6   ADDITIONAL INFORMATION ON THE RFEP METHOD

Unique new atoms and their corresponding dummies receive coordinate placement by a reversible jump Monte Carlo geometry proposal engine to ensure generation of favorable molecular geometries that avoid steric clash or other errors in energy evaluation (Rufa et al., 2022). For the final stage, investigators have the option of using either equilibrium methods (e.g. replica exchange) or non-equilibrium methods (e.g. non-equilibrium switching) to estimate the free energy difference.

A.7   RATIONAL QUADRATIC NEURAL SPLINE FLOWS

Rational Quadratic Neural Spline Flows (RQNSFs) are a class of normalizing flow models that transform using rational quadratic spline functions (Equation 4), or continuous functions mapped to a bound region (i.e. $[-B, B]$) for each DOF. The monotonic increase of rational quadratic functions ensure bijectivity in transform. The continuous first derivatives are piece-wise defined on intervals splitting this region.

$$z = f(x) = z_0 + \frac{(z_1 - z_0)\left[s\xi^2 + d_0\xi\left(1 - \xi\right)\right]}{s + \left[d_1 + d_0 - 2s\right]\xi\left(1 - \xi\right)},\qquad(4)$$

We note in the above equation that $x_{0(1)}$, $z_{0(1)}$, and $d_{0(1)}$ are the locations and derivatives at the left (right) interval boundaries, and $\xi = (x - x_0)/(x_1 - x_0)$, $s = (z_1 - z_0)/(x_1 - x_0)$.

We implemented RQNSF Durkan et al. (2019b) coupling layers in the mapping transformations of our architecture (Figure S3) for a flow module expressive enough to capture the complex probability densities of systems that have high DOFs. Implementations are based on methods described in Mahmoud et al. (2022).
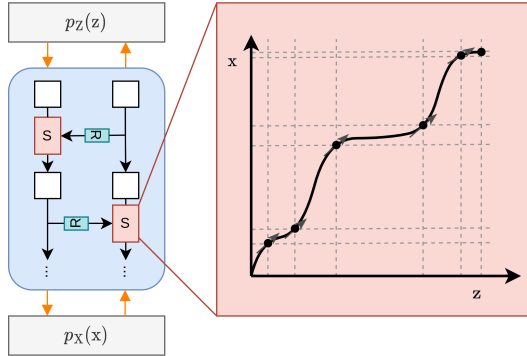

Figure S3: Scheme of a Rational Quadratic Spline function transformation. In our RQNSF implementation, a residual network is used to condition the spline coupling transformation.

A.8 FREE ENERGY ESTIMATION METHODS

In this section, we describe a few approaches to solvation free energy calculation for the sake of comparison with the TFEP approach. We define end states A and B with the respective Hamiltonians $\mathcal{H}_A(\mathbf{q}, \mathbf{p}; \lambda)$ and $\mathcal{H}_B(\mathbf{q}, \mathbf{p}; \lambda)$, where $\mathbf{q}$ and $\mathbf{p}$ represent positions and momenta of the system at given points in phase space and λ the nonphysical coupling parameter. We then obtain the λ-dependent Hamiltonian as:

$$\mathcal{H}(\mathbf{q}, \mathbf{p}; \lambda) = f(\lambda)\mathcal{H}_A(\mathbf{q}, \mathbf{p}; \lambda) + g(\lambda)\mathcal{H}_B(\mathbf{q}, \mathbf{p}; \lambda).$$

By convention, $\mathcal{H} = \mathcal{H}_A$ at $\lambda = 0$ and $\mathcal{H} = \mathcal{H}_B$ at $\lambda = 1$. The Hamiltonians are mixed using the functions $f(\lambda)$ and $g(\lambda)$. In the thermodynamic integration (TI) method, simulations are performed for a discrete set of λ values and subsequently the free energy difference between A and B can be estimated by solving for $\Delta F = \int_{\lambda=0}^{\lambda=1} \left\langle \frac{\partial \mathcal{H}}{\partial \lambda} \right\rangle_\lambda \mathrm{d}\lambda$ by numerical quadrature approach. In the exponential averaging (EXP) or free energy perturbation (FEP) method (Zwanzig, 1954), λ-dependent Hamiltonians are similarly used to to estimate $\Delta F = -\frac{1}{\beta} \ln \left\langle \mathrm{e}^{-\beta[\mathcal{H}_B(\mathbf{qp}; \lambda) - \mathcal{H}_A(\mathbf{q}, \mathbf{p}; \lambda)]} \right\rangle_A$

As a third alternative method, Bennett Acceptance Ratio (BAR) relies on bidirectional transfers to iteratively solve for the following:

$$\left\langle \frac{1}{1 + \frac{N_A}{N_B} \mathrm{e}^{\beta \Delta \mathcal{H}_{BA}(\mathbf{q}, \mathbf{p}) - \beta \Delta F}} \right\rangle_A = \left\langle \frac{1}{1 + \frac{N_B}{N_A} \mathrm{e}^{\beta \Delta \mathcal{H}_{AB}(\mathbf{q}, \mathbf{p}) + \beta \Delta F}} \right\rangle_B.$$

Here, the respective numbers of statistically independent samples from A and B are denoted as N_A and N_B and the Hamiltonian differences as $\Delta \mathcal{H}_{BA}(\mathbf{q}, \mathbf{p}) = \mathcal{H}_B(\mathbf{q}, \mathbf{p}) - \mathcal{H}_A(\mathbf{q}, \mathbf{p}) = -\Delta \mathcal{H}_{AB}(\mathbf{q}, \mathbf{p})$.

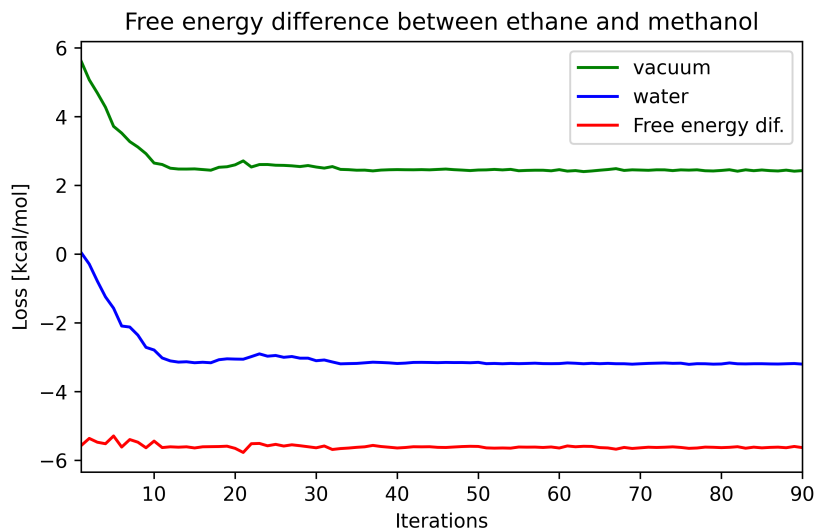## A.9 ETHANE TO METHANOL SOLVATION LEARNING CURVE

Figure S4: Learning curve and free energy difference between ethane and methanol in water and vacuum.