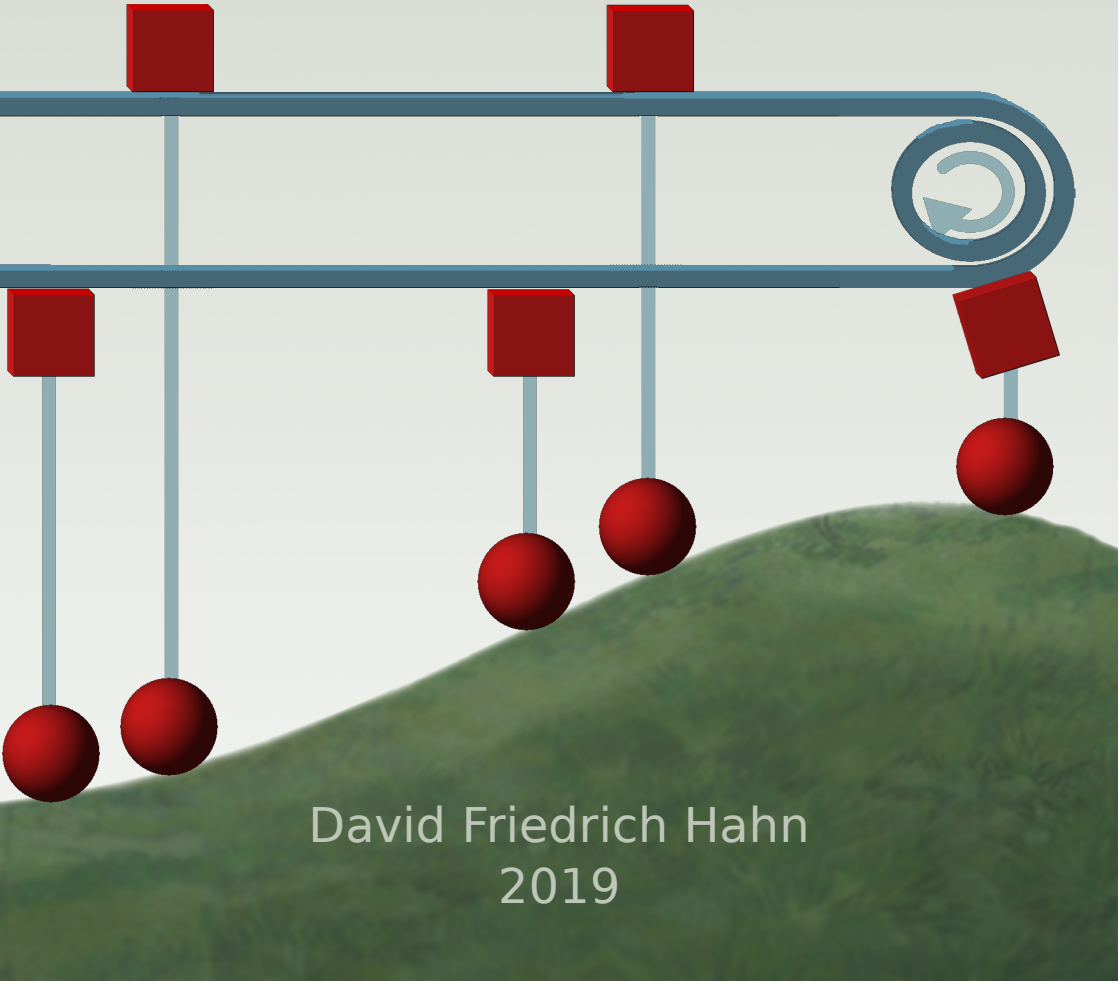
DISS. ETH NO. 25914

Development and Application of Free-energy Calculation Methods based on Molecular Dynamics Simulations



David Friedrich Hahn

2019

DISS. ETH NO. Free Energy Methods with Biomols

Super Title

A thesis submitted to attain the degree of

DOCTOR OF SCIENCES of ETH ZURICH

(Dr. sc. ETH Zurich)

presented by

Benjamin Joachim Ries

MSc. in Bioinformatics

born on 15.03.1991
citizen of Germany

accepted on the recommendation of

Prof. Dr. Sereina Riniker, examiner

Prof. Dr. Philippe Hünenberger, co-examiner
Prof. Dr. Niels Hansen, co-examiner

2019

Here a nice dedication

Acknowledgements

"Der Toaster"

Denker

This thesis would not have been possible without many people, who have supported me over the last years.

Contents

Summary

Zusammenfassung

Publications

The following publications are included in parts or in an extended version in this thesis. The other chapters are in preparation for publication.

Chapter 1

B. Ries, S. M. Linker, D. F. Hahn, G. König, S. Riniker *J. Chem. Inf. Model.* **2021**: Ensembler: A Simple Package for Fast Prototyping and Teaching Molecular Simulations

Chapter 2

B. Ries, K. Normak, R. G. Weiß, S. Rieder, C. Candide, G. König, S. Riniker *J. Comput. Aided Mol. Des.* **2021**, *submitted*: Relative Free-Energy Calculations for Scaffold Hopping-Type Transformations with an Automated RE-EDS Sampling Procedure

Chapter 3

B. Ries, S. Rieder, C. Rhiner, S. Riniker *xxxxx* **2021**, *in progress*: A Graph-Based Approach to the Restraint Problem in Dual Topology Approaches with RestraintMaker

Chapter 5

C. Comeau, B. Ries, T. Stadelmann, J. Tremblay, S. Poulet, U. Fröhlich, J. Côté, P. Boudreault, R. M. Derbali, P. Sarret, M. Grandbois, G. Leclair, S. Riniker, É. Marsault *J. Med. Chem.* **2021**: Modulation of the Passive Permeability of Semipeptidic Macrocycles: N- and C-Methylations Fine-Tune Conformation and Properties

CONTRIBUTIONS

Free Energies

G. König, N. Glaser, B. Schroeder, A. Kubincová, P. H. Hünenberger, S. Riniker *J. Chem. Inf. Model.* **2020**: An Alternative to Conventional -Intermediate States in Alchemical Free Energy Calculations: λ-Enveloping Distribution Sampling

G. König, B. Ries, P. H. Hünenberger, S. Riniker *J. Chem. Inf. Model.* **2021** *submitted*: Efficient Alchemical Intermediate States in Free Energy Calculations Using λ-EDS

E. P. Barros, B. Ries, L. Böselt, C. Champion, S. Riniker *Curr. Opin. Struct. Biol.* **2021** *submitted*: Recent Developments in Multiscale Free Energy Simulations

Cyclic Peptides

J. Witek, S. Wang, B. Schroeder, R. Lingwood, A. Dounas, H. Roth, M. Fouché, M. Blatter, O. Lemke, B. Keller, and S. Riniker *J. Chem. Inf. Model.* **2019** : Rationalization of the Membrane Permeability Differences in a Series of Analogue Cyclic Decapep-

tides

S. M. Linker, S. Wang, B. Ries, T. Stadelmann, S. Riniker *CHIMIA* **2021**: Passing the Barrier – How Computer Simulations Can Help to Understand and Improve the Passive Membrane Permeability of Cyclic Peptides

Machine Learning

R. G. Weiß, B. Ries, S. Wang, S. Riniker *J. Chem. Phys.* **2021**: Volume-scaled common nearest neighbor clustering algorithm with free-energy hierarchy

Other

M. T. Lehner, B. Ries, S. Rieder, S. Riniker *Zenodo* **2021**: riniker-lab/PyGromosTools: PyGromosTools_V2

B. Ries, L. A. Völker, R. Dubey, S. M. Linker *CHIMIA***2021**: A perspective on Virtual Events during the Corona Pandemic exemplified by the Career Track with IBM Research Europe - Zurich

1 Introduction

" hmm such a nice quote"

Author

1.1 HISTORY OF MOLECULAR MODELING

Since ancient times, humans have strived to understand their environment and the phenomena they observed. They have conceived models describing the composition of the surrounding matter and attempting to describe its behavior. The starting point was century-long disputes between representatives of atomic and continuum theories. The atomic theory was first proposed[?] in the 5^{th} century BC by Presocratics Leucippus and Democritus. This theory argued that matter is not infinitely divisible, but that there ultimately particles which are inalterable and indivisible: the atoms. It was Democritus, who claimed[? ?]

> "by convention sweet and by convention bitter, by convention hot, by convention cold, by convention color; but in reality atoms and void".

This groundbreaking statement was far ahead of its time and was mostly rejected, especially by Aristotle with his work *De caelo*, stating that matter is continuous and consists of five elements.[?] As the church adhered to the continuity theory of Aristotle, it was predominant in Europe, until the theory of Democritus experienced a revival with the mechanical atomism in the 17$^{\text{th}}$ century, promoted by philosophers like René Descartes, Pierre Gassendi and Robert Boyle.[?]

While Newton led the foundations of classical mechanics in his work *Philosophiae Naturalis Principia Mathematica*,[?] he also shared atomistic views as he stated that[?]

> "the least parts of bodies to be - all extended, and hard and impenetrable, and moveable, and endowed with their proper inertia".

Before, the atomists were uncertain about the laws governing the movements of atoms. With Newton's three laws of motion, the dynamics of atoms could in principle be determined. The limitation, however, was that the nature of forces between atoms needed to be known.[?]

The early mechanical atomism was then superseded by the atomic theory of John Dalton in the early 19$^{\text{th}}$ century.[?] For the first time, properties, such as the relative weight could be assigned to atoms. Chemical elements were already known, but now atoms could be ascribed to smallest unit of what was understood as an element. Likewise, chemical compounds were found to be defined by specific combinations of atoms.

The word "chemical structure" was coined in the mid 19$^{\text{th}}$ century by chemists like Archibald Scott Couper, Friedrich August Kekulé and Alexander Mikhailovich Butlerov. They proposed the first chemical structures, thereby developing important concepts

like valency, chemical bond and substituent.[?] In 1861, Johann Josef Loschmidt published a collection of 384 molecular structures, some examples of which are given in Figs. 1.1a and 1.1b. His graphical representations included the spatial extent of different atoms in surprisingly correct proportions considering the publication date. Additionally, he proposed the first (correct!) benzene structure, although credit is often incorrectly given to Kekulé. However, the latter invented the resonance formulas of benzene, which are showcased in Fig. 1.1c.[? ? ? ?]

The third spatial dimension came into play with August Wilhelm Hofmann, who created, using table crocket balls, three-dimensional models of methane (Fig. 1.1d), chloroform, and other small organic molecules.[?] Although the structures were not all in agreement with today's state of knowledge, he was the one who devised the color scheme for atoms (*e.g.* black for carbon, white for hydrogen) which is still in use today. In the following years such models were progressively refined, following both experimental and theoretical advances. Structural representations of larger and more complex (bio)molecules were created. Important milestones in this respect were the model of penicillin based on X-ray crystallographic data by Hodgkin et al.,[?] the model of protein α-helices by Pauling et al.,[?] and the famous DNA model by Crick and Watson.[?]

Finally, in 1958, the first (though still coarse) model of a complete protein was proposed by Kendrew et al.[?] (based on the work by Perutz). It was a model of myoglobin made out of wood, plasticine and paint, and is shown in Fig. 1.1e. A model of the structurally similar haemoglobin was created by Perutz et al. a few years later.[?]

The advent of digital computers in the 1940s opened up unprecedented possibilities for building molecular models. Com-

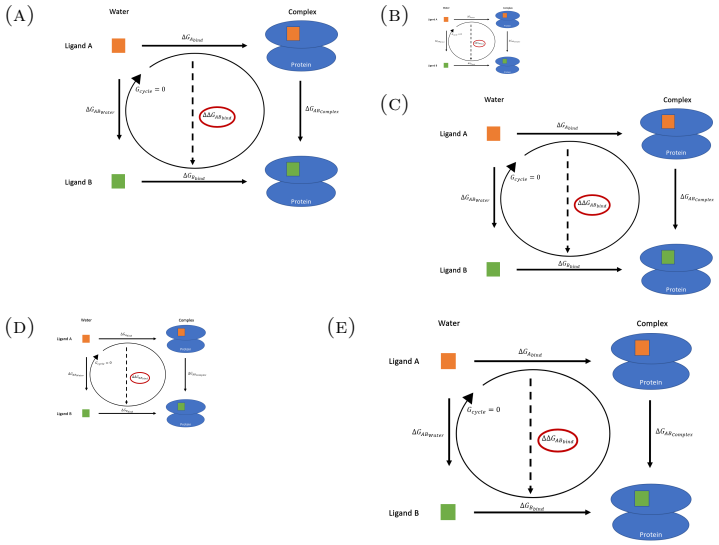


FIGURE 1.1: *Examples of early molecular models.* Graphs (a) and (b) show schemes copied from "Konstitutionsformeln der organischen Chemie in graphischer Darstellung" by J. Loschmidt from 1961, pp. 7 and 62 of Ref.[?] (a): Illustrations of the structures of methane, carbon dioxide, formaldehyde, methanol, formic acid and carbonic acid. (b): Structure of toluene with the first correct prediction of the benzene ring copied from Ref.[?] (c): Drawings of the two resonance formulas of benzene by Kekulé. Copied from (d): Photograph of a methane molecular model, created out of table crocket balls around 1860 by August Wilhelm von Hofmann, which is now part of the collection of the Royal Institution of London. (It is photographed by Henry Rzepa, with the kind permission of the Royal Institution of London, in whose collection the model resides. Wikimedia Commons, the free media repository. Accessed March 10, 2019. `https://commons.wikimedia.org/wiki/File:Molecular_Model_of_Methane_-Hofmann.jpg`, CC BY-SA 4.0) (e) shows the first model of a complete protein, created and published in 1958 by Kendrew et al.[?] (Science Museum Group. Kendrew's original model of the myoglobin molecule. 1975-533. Science Museum Group Collection Online. Accessed March 10, 2019. `https://collection.sciencemuseum.org.uk/objects/co13543.1.1`, CC BY-NC-SA)

puters were not only used to resolve the X-ray structures of the proteins mentioned above, but also to solve the equations of the physical models which were otherwise intractable. These included

both quantum-chemical as well as classical-mechanics simulations. After the first Monte Carlo sampling[?] and then molecular dynamics simulation[?] of hard disks in 2 dimensions in the 1950s, the first Lennard-Jones fluid was simulated in 1964.[?] Computer simulations added a new and essential aspect to the molecular models: the alteration of the structure with time, resulting in translations, vibrations, and conformational changes of molecules. The crucial roal of the atomic and molecular motions in defining material properties and - ultimately - life was famously phrased by Feynman who stated[?]

> "that all things are made of atoms, and that everything that living things do can be understood in terms of the jigglings and wigglings of atoms."

The new field of molecular simulation was born, contributing to the understanding of matter and life in all its details at the atomistic level.

Polyatomic molecules like water[?] and proteins[?] were simulated shortly thereafter. In the following decades, progress was made in developing force fields (interaction functions) and algortihms for reliable, efficient and accurate molecular dynamics (MD) simulations. Since the 1990s, MD has become an indispensable tool in nearly all fields of science, including physics, chemistry, biology, pharmaceutical and material science, and medicine. Due to the ever increasing computational power, larger systems become tractable, which in turn increases the scope of the applications. Apart from visualizing molecules for a better understanding, computations provide a mean for calculating properties, thus complementing and even, sometimes, substituting experiments.

1.2 CLASSICAL MECHANICS

The foundations of classical mechanics were laid by Newton,[?] and later generalized by Lagrange and Hamilton. While the laws of classical mechanics are not valid when dealing with high energies (*e.g.* high velocities, light-matter interactions) or low masses (*e.g.* electrons, other elementary particles), where quantum or/and relativistic mechanics is required, they can provide a sufficient accuracy for understanding macroscopic phenomena at a molecular level with comparatively low computational costs.

Classical systems are usually described by N point particles $i = 0, 1, \ldots, N-1$ with mass m_i in Cartesian space, *i.e.* using x, y and z coordinates. Each particle resides at a certain position $\mathbf{r}_i = \{x_i, y_i, z_i\}$ with a momentum $\mathbf{p}_i = m_i \mathbf{v}_i = m_i \{v_{x,i}, v_{y,i}, v_{z,i}\}$ where $\mathbf{v}$ is the velocity. The vectors $\mathbf{r}$ and $\mathbf{p}$ can be united in a position-momentum vector $\mathbf{x}_i = \{\mathbf{r}_i, \mathbf{p}_i\}$. The vectors $\mathbf{x}_i$ of all particles are the elements of the phase-space vector $\mathbf{x} = \{\mathbf{x}_0, \mathbf{x}_1, \ldots, \mathbf{x}_{N-1}\}$ which contains the information about the state of a system. The phase space is the associated $6N$-dimensional space, which is defined by all possible combinations of positions and momenta of all particles i.

The associated energy is given by the so-called Hamilonian $\mathcal{H}$, which is a function of the phase space vector $\mathbf{x}$, and can be separated into a kinetic-energy term $\mathcal{K}$ and a potential-energy term $\mathcal{V}$ of the system,

$$\mathcal{H}(\mathbf{x}) = \mathcal{K}(\mathbf{p}) + \mathcal{V}(\mathbf{r}). \tag{1.1}$$

The kinetic energy is given by

$$\mathcal{K}(\mathbf{p}) = \sum_i \frac{\mathbf{p}_i^2}{2m_i}, \tag{1.2}$$

which is the sum over the kinetic energies of all particles. The potential energy term is a mathematical expression for the interaction between the particles, and possibly with the environment. It can be given different functional forms, depending on the nature of the system and the level of approximation, which is typically chosen according to the desired level of accuracy.

The time evolution of the system in Hamiltonian mechanics is given by two equations. The first equation describes how the positions evolve in time,

$$\dot{\mathbf{r}}_i = \frac{\mathrm{d}\mathbf{r}_i}{\mathrm{d}t} = \frac{\partial \mathcal{H}}{\partial \mathbf{p}_i}. \tag{1.3}$$

By inserting the definition of the kinetic energy, Eq. 1.2, one obtains $\dot{\mathbf{r}}_i = \partial\mathcal{H}/\partial\mathbf{p}_i = \partial\mathcal{K}/\partial\mathbf{p}_i = \mathbf{p}_i/m_i = \mathbf{v}_i$, which tells us that the positions change in time according to the velocities of the particles. The second equation describes the time evolution of the momenta,

$$\dot{\mathbf{p}}_i = \frac{\mathrm{d}\mathbf{p}_i}{\mathrm{d}t} = -\frac{\partial \mathcal{H}}{\partial \mathbf{r}_i}. \tag{1.4}$$

The time derivative of the momentum of a particle i is equal to its acceleration a_i multiplied by its mass, $\dot{\mathbf{p}}_i = m_i\dot{\mathbf{v}}_i = m_i a_i$. Therefore the second equation is another formulation of Newton's second law $\mathbf{f}_i = m_i\mathbf{a}_i$, where $\mathbf{f}_i$ is the force, given by the negative gradient of the Hamiltonian function with respect to the position $\mathbf{f}_i = -\partial\mathcal{H}/\partial\mathbf{r}_i$.

It is important to note that the total energy of a system

governed by Eqs. 1.3 and 1.4 is constant, *i.e.* $\mathcal{H}$ in Hamiltonian mechanics is a constant of motion.

1.3 CLASSICAL STATISTICAL MECHANICS

Statistical mechanics[? ? ? ? ?] is a branch of physics which links the macrosopic properties to the microscopic behavior of a system. The microscopic behavior of the system is governed by the interactions of its particles, *i.e.* by the interaction function $\mathcal{V}$ in Eq. 1.1. In classical statistical mechanics, the interaction functions are formulated in the framework of classical mechanics and are part of the Hamiltonian of the system. Because of the immense size of macroscopic systems ($1\,\mathrm{mol} = 6.022 \cdot 10^{23}$ particles) and the non-trivial microscopic interactions between particles in real systems, the treatment even according to classical mechanics is generally not feasible analytically, except for the simplest systems (ideal gas, harmonic crystal). However, by statistically evaluating the microscopic details, we can still calculate the macroscopic properties of a system.

The statistical evaluation is performed on the basis of an ensemble, which is a collection of microscopic states (microstates) of the same system obeying the same microscopic interactions and fulfilling certain macroscopic boundary conditions. A macroscopic property of interest can then be calculated by averaging over the ensemble. According to the ergodic hypothesis,[?] this property can also be calculated by observing the evolution of one system over an infinite period of time. A microstate is defined by a point $\mathbf{x} = \{\mathbf{r}, \mathbf{p}\}$ in the phase space of the system. Typically, a system

treated by statistical mechanics has many degrees of freedom, *i.e.* a high-dimensional $\mathbf{x}$. An ensemble encompasses all microstates which obey some given macroscopic boundary conditions, which are usually three constant thermodynamic observables, among them at least one extensive quantity. Common ensembles are: (*i*) the microcanonical ensemble with constant number N of particles, volume V and energy E; (*ii*) the canonical ensemble with constant number N of particles, volume V and temperature T; (*iii*) the isothermal-isobaric (Gibbs) ensemble with constant number N of particles, pressure P and temperature T; (*iv*) the grand canoncial ensemble with constant chemical potential μ, volume V and temperature T. In the following, we consider only the microcanonical and canonical ensembles, but the discussion can be straightforwardly extended to other systems.

1.3.1 The microcanonical ensemble

The microcanonical ensemble is the simplest ensemble, corresponding to an isolated system with a constant number N of particles in a container of constant volume V at constant energy E.

To be part of the ensemble, the microstates defined by the phase space vector $\mathbf{x}$ have to fulfill the condition

$$\mathcal{H}(\mathbf{x}) = E. \tag{1.5}$$

All accessible microstates $\mathbf{x}$ of the ensemble lie on a constant-energy hypersurface in phase space and are equally probable. This is known as the assumption of equal a priori probability.

A measure of the amount of phase space available to the system is the partition function. The microcanonical partition function

Ω is calculated as an integral over phase space

$$\Omega(N, V, E) = \frac{E_0}{N! h^{3N}} \int \mathrm{d}\mathbf{x} \delta(\mathcal{H}(\mathbf{x}) - E), \tag{1.6}$$

where the prefactor accounts for particle indistinguishability, unit conversion to a dimensionless quantity, and Heisenberg's uncertainty relation. The partition function is a fundamental quantity in statistical mechanics. All thermodynamic properties of the system can be derived from it. A property related to the number of microstates and, therefore, the partition function is the entropy S, which is linked to Ω via the Boltzmann relation

$$S(N, V, E) = k_\mathrm{B} \ln \Omega(N, V, E) \ , \tag{1.7}$$

where the proportionality factor $k_\mathrm{B} = 8.3146\,\mathrm{J\,K^{-1}mol^{-1}}$ is the Boltzmann constant. For a microcanonical ensemble, the entropy indicates which macrostate is favored. Given two different macrostates, the system will always prefer the state with the higher entropy. A process to this state will always occur spontaneously. However, the entropy difference does not provide any information on the timescale of the corresponding process, which can be short or long due to kinetic hindrance. Other thermodynamic properties can be derived by taking the total differential of the entropy $S(N, V, E)$, known from phenomenological thermodynamics,

$$\begin{aligned} \mathrm{d}S &= \left(\frac{\partial S}{\partial E}\right)_{N,V} \mathrm{d}E + \left(\frac{\partial S}{\partial V}\right)_{N,E} \mathrm{d}V + \left(\frac{\partial S}{\partial N}\right)_{V,E} \mathrm{d}N \\ &= \frac{1}{T}\mathrm{d}E + \frac{P}{T}\mathrm{d}V - \frac{\mu}{T}\mathrm{d}N \ , \end{aligned} \tag{1.8}$$

where T is the temperature, P the pressure and μ the chemical

potential. If one equates the coefficients, *i.e.*

$$\begin{aligned}\frac{1}{T} &= \left(\frac{\partial S}{\partial E}\right)_{N,V},\\ \frac{P}{T} &= \left(\frac{\partial S}{\partial V}\right)_{N,E},\\ \frac{\mu}{T} &= \left(\frac{\partial S}{\partial N}\right)_{V,E},\end{aligned} \tag{1.9}$$

one can relate the thermodynamic variables T, P, and μ to $\Omega(N,V,E)$ by combining Eq. 1.7 with Eq. 1.9:

$$\begin{aligned}\beta &= \left(\frac{\partial \ln\Omega}{\partial E}\right)_{N,V},\\ \beta P &= \left(\frac{\partial \ln\Omega}{\partial V}\right)_{N,E}\\ \beta\mu &= \left(\frac{\partial \ln\Omega}{\partial N}\right)_{V,E}.\end{aligned} \tag{1.10}$$

The normalized probability $P(\mathbf{x})$ of finding a system in a given microstate $\mathbf{x}$ can also be related to the inverse partition function

$$P(\mathbf{x}) = \frac{E_0}{N!h^{3N}}\frac{\delta(\mathcal{H}(\mathbf{x})-E)}{\Omega(N,V,E)}. \tag{1.11}$$

The normalized probability defines the distribution of macrostates in equilibrium. Accordingly, any thermodynamic observable $Y(N,V,E)$ can be calculated by multiplying the instantaneous quantity $\mathcal{Y}(\mathbf{x})$ of a microstate with its probability and integrating over the whole phase space, *i.e.*

$$\begin{aligned}Y(N,V,E) &= \langle\mathcal{Y}(\mathbf{x})\rangle\\ &= \int \mathrm{d}\mathbf{x}\, P(\mathbf{x})\mathcal{Y}(\mathbf{x})\end{aligned} \tag{1.12}$$

where $\langle \dots \rangle$ denotes the ensemble average. Note that Eq. 1.12 is also valid for other ensembles, but the probability, given by Eq. 1.11 for a microcanonical ensemble, has to be adapted.

While being the simplest ensemble, the use of the microcanocial ensemble is limited, because most experiments are conducted under isothermal conditions.

1.3.2 The canonical ensemble

The canonical ensemble is an ensemble of systems with a constant number N particles in a container of constant volume V at temperature T, *i.e.* the systems are in contact with a heat bath allowing the exchange of energy (heat) in order to keep the temperature of the system constant.

The main difference to the microcanonical ensemble resides in the heat exchange with the environment. The total energy is not conserved and the microstates now obey a Boltzmann distribution, with probabilities proportional to $\exp(-\beta\mathcal{H}(\mathbf{x}))$, where $\beta = (k_\mathrm{B}T)^{-1}$ and k_B is the Boltzmann's constant,. The canonical partition function is an integral over the Boltzmann factor for all possible microstates

$$Q(N,V,T) = \frac{1}{h^{3N}N!}\int \mathrm{d}\mathbf{x}\exp(-\beta\mathcal{H}(\mathbf{x})). \tag{1.13}$$

The energy of the ensemble E is the ensemble average, given by

Eq. 1.12, of the Hamiltonian.

$$\begin{aligned} E(N,V,T) &= \langle \mathcal{H}(\mathbf{x}) \rangle. \\ &= \int \mathrm{d}\mathbf{x}\, P(\mathbf{x}) \mathcal{H}(\mathbf{x}) \\ &= -\frac{1}{Q(N,V,T)} \frac{\partial}{\partial \beta} Q(N,V,T) \\ &= -\frac{\partial}{\partial \beta} \ln Q(N,V,T). \end{aligned} \tag{1.14}$$

For a canonical ensemble, the entropy difference between two states does not tell anything about the spontaneity of a process, because the corresponding entropy change in the environment (resulting from the transfer of heat) has to be taken into account. Therefore it is convenient to define the Helmholtz free energy A as

$$A(N,V,T) = -\beta^{-1} \ln Q(N,V,T). \tag{1.15}$$

This quantity gives information about the spontaneity of a process. If the free-energy difference is negative, the process will occur. If it is zero, the two states are in equilibrium.

1.4 MOLECULAR DYNAMICS SIMULATION

Molecular dynamics (MD) simulations apply classical mechanics (Sect. 1.2) to molecular systems. Common applications include organic or biological molecules in solvent. In the following, the framework of MD is briefly introduced.

1.4.1 The System

A typical simulated system consists of one or more solute(s), *e.g.* an organic molecule, a protein, a lipid, a saccharide, a nucleic acid or a combination of those. The environment of this solute is either vacuum, implicit solvent (which models the average influence of solvent molecules *via* a continuous medium) or explicit solvent (solvent molecules are added to the system). The number of atoms can be on the order of several millions of atoms, which leads to system sizes on the order of $10 \times 10 \times 10\,\mathrm{nm}^3$. The times simulated are on the order of nanoseconds to milliseconds per day computer time, depending on the available computing resources, on the system size, on the employed approximations and on the algorithms used.

The atoms to be simulated are described as N point particles having masses $\{m_i\}$ and charges $\{q_i\}$. Often, a number of atoms are combined to one point particle, which is referred to as an united atom[?] (*e.g.* a methyl group) or a coarse grained bead[? ?] (*e.g.*, a whole ester functional group). The electrons and other elementary particles are not included explicitly. Only their average influence is modeled by means of the masses and charges. Therefore, MD simulations cannot describe electronically excited states, chemical reactions or light-matter interactions. In the case of explicit solvation, periodic boundary conditions are usually applied to avoid surface effects. The system is then modeled in a space-filling polyhedron, *e.g.*, a cube or a truncated octahedron, and an atom leaving the boundary on one side enters again on the opposite side. In other words, the system is allowed to interact with periodic copies of itself.

1.4.2 The Interaction Function (Force Field)

The potential energy $\mathcal{V}$ (Sect. 1.2) is a mathematical expression to describe the nature of the interactions within the system. In MD, this function is referred to as a force field. A force field consists of many terms, which can be classified as covalent and non-bonded terms.[?] Usually non-bonded terms are two-body terms, *i.e.* they include only the coordinates of two point particles. Some n-body terms with $n > 2$ can be considered in exceptional cases.

The covalent terms describe the interactions within one molecule up to third-neighbor interactions, *i.e.* between atoms separated by up to three chemical bonds. They usually include: (*i*) bond stretching terms, which are harmonic potentials controlling the distance between two connected atoms; (*ii*) bond-angle bending terms, which are harmonic potentials controlling the angle spanned by three atoms; (*iii*) torsional-dihedral terms, which are usually cosine series controlling the configuration of four consecutively bonded atoms; (*iv*) improper-dihedral terms, which are harmonic potentials controlling the configuration of four atoms, of which three are bonded to a central atom. Other terms (*e.g.* covalent cross-terms) are possible, but seldomly used.

Interactions between atoms separated by more than two bonds (including pairs of atoms in different molecules) are described by non-bonded interaction terms. These terms depend on the distance between two atoms. There are typically two different terms considered which describe the electrostatic and van-der-Waals interactions between the atoms. The electrostatic interactions are defined by a Coulomb potential having a $1/r$ dependence, where r is the distance of the atoms. These terms model the attraction of two atoms with partial charges of opposite signs, or the repulsion of atoms with partial charges of the same sign.

The van-der-Waals interactions are commonly modeled with a Lennard-Jones functional form, including the repulsion due to the Pauli exclusion principle for the electrons at short distances (r^{-12} dependence), and the attraction due to London dispersion forces (r^{-6} dependence) for longer distances. Because non-bonded interactions, especially the electrostatic interactions, decay very slowly, special care has to be taken for long-range non-bonded interactions. Otherwise one would need to simulate very large systems which is computationally expensive. The most prominent methods in this regard are the reaction field[?] (RF) and the particle-mesh Ewald[?] (PME) methods. In the RF method, a cutoff is applied which sets the interaction to zero at distances larger than the cutoff distance. To avoid the neglect of longer-range interactions, a reaction-field term is added, accounting for the effect of a dielectric environment beyond the cutoff sphere. Since the covalent terms take care of interactions between atoms separated by up to three bonds, the nonbonded interactions for first and second neighbors are usually excluded. Third-neighbor covalent interactions are usually scaled, or handled using special (reduced) non-bonded interaction parameters.

All force-field terms extensively make use of parameters like force constants, reference bond lengths, angles, torsions and improper dihedrals, partial charges and Lennard-Jones parameters. The number of parameters is on the order of number of degrees of freedom of the molecules considered. For example, a water (H_2O) molecule is in principle described by 10 parameters, including 4 Lennard-Jones parameters (2 for the oxygen, 2 for the hydrogen, the latter sometimes set to zero), 2 partial charges, and two reference values and two force constants for the bond lengths and angle, respectively. These parameters have to be fitted in order to reproduce experimental properties in an MD simulation. Initial

guesses of parameters like force constants (energetic information) are taken from experimental spectroscopic measurements. For reference structural information (bond lengths, angles, torsions and improper dihedrals), data from X-ray crystallography can be used. All experimentally derivable parameters can be refined or even completely replaced by quantum-mechanical (QM) estimates. However, the initial parameters taken from experiments or calculations have to be refined to make them compatible to each other and to the employed algorithms. Therefore, force-field parameterization is a time consuming and difficult task, which is still the subject of ongoing research.

1.4.3 Integration of the Equations of Motion

Given a force field for the interactions of the particles and after defining the Hamiltonian $\mathcal{H}$ (Eq. 1.1) of the system, the classical equations of motion in Eqs. 1.3 and 1.4 can be integrated. In view of the large size of the system ($6N$ dimensions), the analytical integration of the equations of motion is impossible and has to be performed numerically instead. One employs a finite time step Δt, which has to be short enough to avoid quadrature errors and long enough to provide computational efficiency and to avoid round-off errors. It is typically chosen on the order of 1/10 of the fastest timescale of the system, determined by the bond vibrations, on a femtosecond timescale. Different algorithms aiming at reducing the integration error have been developed. One example is the leap-frog algorithm[?] which evolves the positions $\mathbf{r}_i$ and velocities

$\mathbf{v}_i$ shifted by a half timestep

$$\mathbf{v}_i(t + \frac{\Delta t}{2}) = \mathbf{v}_i(t - \frac{\Delta t}{2}) + \frac{\mathbf{f}_i(\mathbf{r}(t))}{m_i}\Delta t \tag{1.16}$$

$$\mathbf{r}_i(t + \Delta t) = \mathbf{r}_i(t) + \mathbf{v}_i(t + \frac{\Delta t}{2})\Delta t. \tag{1.17}$$

This leads to a vanishing $(\Delta t)^2$ term and, therefore, to a reduced error on the order of $(\Delta t)^3$.

1.4.4 Thermostatting and Barostatting

The integration of the equations of motion leads to a microcanonical ensemble (Sect. 1.2). To reproduce experimental conditions, it is necessary to simulate at constant temperature and, possibly, pressure. The temperature and pressure is regulated by employing thermostats and barostats.

The instantaneous temperature, *i.e.* the temperature of a specific state $\mathbf{x}$ of a system, depends solely on the kinetic energy as defined by the momenta of the particles,

$$\begin{aligned} \mathcal{T} &= \frac{2\mathcal{K}}{\mathcal{N}_{\mathrm{dof}} k_{\mathrm{B}}}, \\ &= \sum_i \frac{\mathbf{p}_i^2}{m_i \mathcal{N}_{\mathrm{dof}} k_{\mathrm{B}}} \end{aligned} \tag{1.18}$$

where $N_{\mathrm{dof}} = 3N - 6$ is the number of degrees of freedom of the system with N particles. In principle one could just constrain the temperature to the desired target temperature by scaling the momenta of the particles at every integration step, but this leads to a constrained temperature, which is not correct for a canonical or Gibbs ensemble. Instead, the temperature distribution should be a Boltzmann distribution, where the instantaneous temperature is allowed to fluctuate. This temperature fluctuation is made

possible in the weak-coupling thermostat by Berendsen,[?] which rescales the velocities with the factor

$$\gamma = \left[1 + \frac{\Delta t}{\tau_{\mathrm{T}}}\left(\frac{T}{\mathcal{T}} - 1\right)\right]^{1/2}, \tag{1.19}$$

by employing a coupling time parameter τ_{T}. The temperature distribution is still not a Boltzmann distribution, which requires a more advanced velocity-scaling thermostat like the Nosé-Hoover[? ? ?] or Nosé-Hoover chain[?] thermostats. Another means of temperature control relies on stochastic approaches like the Andersen thermostat,[?] where the velocities of the particles are reset to a Maxwell-Boltzmann distribution at constant time periods. Other equations of motion such as the Langevin equation of motion are also able to generate constant-temperature ensembles. Here, the forces are attenuated by a friction force while additional stochastic forces ("random kicks") are applied to the particles. Pure Monte Carlo approaches with the Metropolis-Hastings acceptance criterion[?] also automatically generate an NVT ensemble.

The instantaneous pressure $\mathcal{P}$, on the other hand, depends on both the momenta and positions,

$$\mathcal{P} = \frac{1}{3V}\sum_i \frac{\mathbf{p}_i^2}{m_i} + \mathbf{r}_i \cdot f_i, \tag{1.20}$$

where the product of position and force, $\mathbf{r}_i \cdot \mathbf{f}_i$, is the virial, which accounts for the deviation from ideal-gas behavior. To simulate at the correct average pressure, the positions of the particles and including the box size are usually rescaled, *e.g.*, by the Berendsen weak-coupling barostat.[?]

1.4.5 Free-Energy Calculations

Many thermodynamic properties like the temperature, pressure, or the density can be extracted from a simulation by calculating the average over the corresponding instantaneous variable. For the calculation of free energies, however, it is not a viable alternative.[? ?] Although we can express Eq. 1.13 for the free energy as an ensemble average by noting that $Q(N,V,T) = \langle \exp(\beta \mathcal{H}(\mathbf{x})) \rangle$,

$$A(N,V,T) = -\beta^{-1} \ln \langle \exp(\beta \mathcal{H}(\mathbf{x})) \rangle \tag{1.21}$$

it is not possible to calculate free energies in this way. The ensemble average will not converge, because configurations $\mathbf{x}$ with high energies $\mathcal{H}$ are seldom visited in a simulation (low Boltzmann weight), but have the strongest impact on the ensemble average. Thus, it would be impossible to obtain a converged result. But we can take advantage of the fact that one is usually interested in free-energy differences between two states A and B instead of the absolute free energy, *i.e.*

$$\Delta A(N,V,T) = A_\mathrm{B} - A_\mathrm{A} \tag{1.22}$$

$$= -\beta^{-1} \ln \frac{Q_\mathrm{B}}{Q_\mathrm{A}} \tag{1.23}$$

where the subscript A or B denote a property pertaining to state A or B, respectively.

One distinguishes between thermodynamic, conformational and alchemical free-energy differences, depending on the end states considered. Thermodynamic free-energy differences consider end states which differ in a thermodynamic property like the temperature, pressure or number of molecules. Conformational free-energy differences consider end states which are different parts of phase

space. In other words, the free-energy change upon changing conformations is observed along one (or more) degree(s) of freedom of the system. Alchemical free-energy differences consider end states which differ in the Hamiltonian. An alchemical coordinate is added to the system, along which the Hamiltonian is changed, while the degrees of freedom (number of atoms) remain unchanged. Whereas the former free-energy differences are physical, the alchemical one is unphysical, and has no experimental counterpart. But in MD simulations, it often offers a sampling advantage to conduct alchemical calculations and compare the results of such simulations with experimental data *via* a thermodynamic cycle. An example is the free-energy of binding of a drug molecule to a receptor, which is depicted in Fig. 1.2. One can alchemically mutate the drug molecule to a dummy skeleton (non-interacting particles), once bound to the receptor and once unbound. The comparison of these two differences gives access to the binding free-energy at considerably lower computational cost than directly calculating the conformational free-energy difference.

FIGURE 1.2: *Schematic illustration of a thermodynamic cycle considering the binding of a drug molecule A (magenta) to a receptor (brown).* The upper horizontal processe defines the conformational change from the unbound receptor and drug molecule to the bound complex, which means the drug molecule is displaced from the bulk water into the binding pocket. The corresponding free-energy difference is $\Delta A_A^{\text{bound}-\text{unbound}}$. The lower horizontal processe defines the same conformational change for the dummy state D (non-interacting skeleton). The corresponding free-energy difference is $\Delta A_D^{\text{bound}-\text{unbound}}$ is zero, because the dummy D is not interacting. The vertical processes correspond to the alchemical transformation of molecule A to the dummy state D, once in the unbound state and once in the bound state. The total free-energy change around the cycle is zero. Therefore the free-energy difference of interest ($\Delta A_A^{\text{bound}-\text{unbound}}$) can be inferred from the two alchemical calculations, which can be performed at considerably lower computational cost than the direct conformational calculation.

For conformational free-energy calculations, the Hamiltonian

remains unchanged. Therefore, the ratio between the partition functions can be expressed as ratio of probabilities P_A and P_B of being in state A or B,

$$\begin{aligned}\Delta A(N,V,T) &= -\beta^{-1}\ln\frac{\int_\mathrm{B}\mathrm{d}\mathbf{x}\exp(-\beta\mathcal{H}(\mathbf{x}))}{\int_\mathrm{A}\mathrm{d}\mathbf{x}\exp(-\beta\mathcal{H}(\mathbf{x}))}\\ &= -\beta^{-1}\ln\frac{P_\mathrm{B}}{P_\mathrm{A}}.\end{aligned} \tag{1.24}$$

Note that the integral is not running over the whole phase space anymore, but is restricted to the different regions A and B of the phase space. The probabilities of being in state A and state B are easily calculated from a simulation and can then be used to determine the free-energy difference. This approach is frequently used and termed direct counting. However, it requires sufficient sampling and interconversion between the two states A and B, which is only possible if the free-energy difference is small enough and the states are not separated by a too high barrier.

In alchemical calculations, when $\mathcal{H}_\mathrm{A}$ is not equal to $\mathcal{H}_\mathrm{B}$, one applies the following transformation

$$\begin{aligned}&\Delta A(N,V,T)\\ &\quad= -\beta^{-1}\ln\frac{\int\mathrm{d}\mathbf{x}\exp(-\beta\mathcal{H}_\mathrm{B}(\mathbf{x}))}{\int\mathrm{d}\mathbf{x}\exp(-\beta\mathcal{H}_\mathrm{A}(\mathbf{x})))}\\ &\quad= -\beta^{-1}\ln\frac{\int\mathrm{d}\mathbf{x}\exp(-\beta(\mathcal{H}_\mathrm{B}(\mathbf{x})-\mathcal{H}_\mathrm{A}(\mathbf{x})))\exp(-\beta\mathcal{H}_\mathrm{A}(\mathbf{x}))}{\int\mathrm{d}\mathbf{x}\exp(-\beta\mathcal{H}_\mathrm{A}(\mathbf{x}))}\\ &\quad= -\beta^{-1}\ln\int\mathrm{d}\mathbf{x}P_\mathrm{A}\exp(-\beta(\mathcal{H}_\mathrm{B}(\mathbf{x})-\mathcal{H}_\mathrm{A}(\mathbf{x})))\\ &\quad= -\beta^{-1}\ln\langle\exp[-\beta(\mathcal{H}_\mathrm{B}(\mathbf{x})-\mathcal{H}_\mathrm{A}(\mathbf{x}))]\rangle_\mathrm{A}.\end{aligned} \tag{1.25}$$

This approach is called free-energy perturbation and Eq. 1.25 is

the Zwanzig formula.[?] In practice one simulates state A (with Hamiltonian $\mathcal{H}_\mathrm{A}$) and one infers the energy of state B from the configurations sampled at state A. This approach will only yield good results if there is enough phase-space overlap between states A and B, *i.e.* if the configurations sampled in A are also low-energy conformations for B.

If the phase-space overlap is not sufficient, a hybrid Hamiltonian can be constructed with a coupling parameter λ that defines a continuous transformation between the Hamiltonians of the physical end-states A and B. The hybrid Hamiltonian $\mathcal{H}(\mathbf{x};\lambda)$ must satisfy the boundary conditions $\mathcal{H}(\mathbf{x};0) = \mathcal{H}_A(\mathbf{x})$ and $\mathcal{H}(\mathbf{x};1) = \mathcal{H}_B(\mathbf{x})$. This enables splitting up the alchemical transformation into several parts by simulating at intermediate states along the coupling parameter and calculating the free-energy differences for only a part of the whole transformation. The phase-space overlap between two neighboring states is increased, which leads to better convergence. Finally the total free-energy difference can be obtained as a sum over the parts.

A different and frequently used approach is the thermodynamic integration method.[? ? ?] Here one integrates the mean force along the alchemical coupling parameter,

$$\Delta A(N,V,T) = \int_0^1 \frac{\partial A(\lambda)}{\partial \lambda} \mathrm{d}\lambda \ . \tag{1.26}$$

The derivative of the free energy with respect to the coupling

parameter is given by

$$\begin{aligned}\frac{\partial A(\lambda)}{\partial \lambda} &= \frac{\partial}{\partial \lambda}\left(-\beta^{-1}\ln\frac{1}{h^{3N}N!}\int \exp(\beta\mathcal{H}(\mathbf{x};\lambda))\mathrm{d}\mathbf{x}\right)\\ &= -\beta^{-1}\frac{\frac{\partial}{\partial\lambda}\int \exp(\beta\mathcal{H}(\mathbf{x};\lambda))\mathrm{d}\mathbf{x}}{\int \exp(\beta\mathcal{H}(\mathbf{x};\lambda))\mathrm{d}\mathbf{x}}\\ &= -\beta^{-1}\frac{\int \frac{\partial\mathcal{H}(\mathbf{x};\lambda)}{\partial\lambda}\exp(\beta\mathcal{H}(\mathbf{x};\lambda))\mathrm{d}\mathbf{x}}{\int \exp(\beta\mathcal{H}(\mathbf{x};\lambda))\mathrm{d}\mathbf{x}}\\ &= \left\langle\frac{\partial\mathcal{H}}{\partial\lambda}\right\rangle_{\lambda}, \end{aligned} \tag{1.27}$$

i.e. it is exactly the ensemble average of the Hamiltonian derivative with respect to λ. The free-energy difference is thus calculated by simulating at different λ-values, calculating the ensemble average of the Hamiltonian derivative at the respective λ values and finally integrating numerically to obtain the final free-energy difference.

1.5 AIM OF THIS THESIS

This thesis deals with methodological developments of free-energy calculations and their application in MD simulations. In Chapter **??**, a force-field for resorcin[4]arenes is presented, which is employed for the calculation of the free-energy difference between two distinct conformations, a closed VASE conformation with a cavity and an open KITE conformation with an expanded surface. To efficiently calculate the free-energy difference, a method called ball-and-stick local elevation umbrella sampling (B&S-LEUS) is used. Chapters **??**-**??** deal with the development and application of the so-called conveyor belt scheme. This scheme employs multiple coupled replicas which concertedly move on a forward-

turn-backward path, akin a conveyor belt, along a coordinate of interest. In Chapters **??** and **??**, the coordinate of interest is the alchemical coupling parameter λ, and the scheme is therefore termed conveyor belt thermodynamic integration (CBTI). CBTI is introduced and applied to the aqueous annihilation of methanol. In Chapter **??**, the performance is compared and tested on two other systems, namely to the alchemical mutations of parts of a tripeptide and a guanosine triphosphate. Finally, in Chapter **??**, the conveyor belt scheme is extended to conformational changes, now termed conveyor belt umbrella sampling (CBUS), and applied to the calculation of binding free energies between ions and crown ethers in various solvents.

2 Using simple Models to understand and develop Methodology - Ensembler

"Let us learn to dream, gentlemen, and then perhaps we shall learn the truth."

August Kekulé, 1865

Ensembler is a python package that allows for fast and easy access to the simulation of one and two-dimensional model systems. It enables method development using small test systems and to deepen the understanding of a broad spectrum of molecular dynamics (MD) methods, starting from basic techniques to enhanced sampling and free-energy calculations. The ease of installing and using the package increases shareability, comparability, and reproducibility of scientific code developments. Here, we provide a description of the implementation and usage of the package as well as an application example for free-energy calculation. The code of Ensembler is freely available on GitHub *https://github.com/rinikerlab/Ensembler*.

2.1 INTRODUCTION

Newly developed advanced simulation methods are routinely tested on simple one- and two-dimensional model systems. They provide valuable insights into the theory, conceptual advantages and limitations (for examples see e.g. Refs.[1–8]). While the results of new methods are published, the implementation details may not always be available or difficult to use with different computer infrastructure. As a result, sharing, reproducing, understanding, and comparing simulation methodologies is often cumbersome.[9] To address this issue, we have developed the Ensembler package, an easy-to-use, yet powerful platform that enables fast prototyping of new methods and comparison against existing techniques using one or two-dimensional test systems.

Ensembler is designed following the recommendations of Stodden *et al.*[10] for the enhanced reproducibility of computational methods, which includes making code publicly accessible, providing documentation, and using open licensing.[10] Furthermore, Ensembler uses state-of-the-art software engineering tools (i.e. git,[11] MolSSI cookie-cutter,[12] and binder[13]) to fulfill these recommendations and enable features like continuous integration and the transparent versioning of the code.

2.2 METHOD DEVELOPMENT

The lean Python3 code[14] of Ensembler allows for easy prototyping of new methods and comparison against a wide range of already implemented techniques. In contrast, the C/C++[15] code of traditional high-performance molecular dynamics (MD) packages (e.g.

Refs.[16–20]) is more efficient but also much more complex. The methods currently available in Ensembler are:

- *Model systems*: Harmonic oscillators as well as dihedral-angle, double-well, and Lennard-Jones potential-energy functions[21]

- *Sampling algorithms*: Conjugated gradient[22] for energy minimization, Metropolis Monte Carlo (MC),[23] leap-frog integration[24] for MD, and Langevin integration[25] for stochastic dynamics (SD)

- *Enhanced sampling techniques*: Umbrella sampling,[26] simulated tempering/temperature replica-exchange simulations,[27] local elevation/metadynamics,[1,2]

- *Free-energy methods*: Free-energy perturbation (FEP),[28] Bennett's acceptance ratio (BAR),[29] thermodynamic integration (TI),[30] enveloping distribution sampling (EDS),[3,31,32] λ-EDS,[5] replica-exchange EDS (RE-EDS),[33] and conveyor-belt TI[34]

2.3 TEACHING

Simple model systems can also be used for teaching MD concepts to students, as they allow to intuitively understand fundamental concepts.[35] Ensembler is well suited for didactic purposes because it is not only easy to use, but supports also a range of visualizations, i.e. interactive widgets, animations, and plots, which can be embedded in Jupyter notebooks.[36] Example Jupyter notebooks[36] are provided in the Ensembler GitHub repository.

2.4 THEORY

Ensembler is implemented in Python3[14] and available on GitHub[37] (*rinikerlab/Ensembler*). The repository is based on the template of the MolSSI cookie-cutter[12] and comprises a code folder, an example folder for tutorials, example models contained in the provided Jupyter notebooks,[36] an automatic pytest suite,[38] and the automatically generated sphinx[39] documentation. The code is continuously integrated via GitHub Actions,[40] providing information about code quality, test correctness, test coverage, and generation of an up-to-date documentation. Ensembler uses only open-source packages like the SciPy library[41–45] and Jupyter notebooks.[36] In the following, a user and a developer perspective are provided for the code structure.

2.4.1 User level

A simulation model in Ensembler consists of a potential class, a sampler class, and a system class wrapping the potential and the sampler (Figure **??**), and provides control over the simulation approach. Additionally, multiple condition classes can be added that directly influence the simulation (e.g. periodic boundary condition[46,47] or thermostatting[48]). After the construction of the system, the simulation can be started directly with the *simulate* function. The resulting trajectory is in the form of a Pandas data frame.[44] The trajectory is thus easily compatible with other packages like NumPy[42] or scikit-learn[49] and can be stored in different formats, e.g. as .csv or .hf5 file. The system itself can be stored directly via the save function using serialization of the object with the Python package pickle. In most cases, only a few

additional lines are needed to go from simple simulation technique to more advanced one, as shown below.

2.4.2 Developer level

The code of Ensembler is built on five interface-like base classes that allow extensive use of the inheritance concept and polymorphism[15] throughout the package. These fundamental classes are *potential*, *sampler*, *condition*, *system*, and *ensemble* (Figure **??**), which can be grouped into three layers. *Potential*, *sampler*, and *condition classes* form the primary layer, providing different techniques to be used as components in a simulation. *Potential classes* provide the potential-energy functions in a symbolic form using SymPy,[43] enabling automatic on-the-fly derivation and simplification of the potential-energy function. *Sampler classes* are used to explore the potential-energy function (e.g. conjugate gradient,[22] Metropolis MC,[23] or leap-frog[24] integration). A new method can easily be implemented by inheriting from the *sampler class* and overwriting a single function called *step*. Finally, *condition classes* provide additional functionalities such as thermostatting[48] and periodic boundary conditions[46,47]). New techniques can be implemented by inheriting the base *condition class* and overwriting the function *apply*. In the second layer, the first-layer components are wrapped into one *system class* that executes the simulation(s) and manages the input and output. An optional higher-order layer is available in form of the *ensemble class*, which allows the user to perform simulations with replica exchange.[27,33,50,51] If additional parameters are needed in a newly designed class, the constructor of the new child class can be adapted but must call the parent constructor.

2.5 COMPUTATIONAL DETAILS

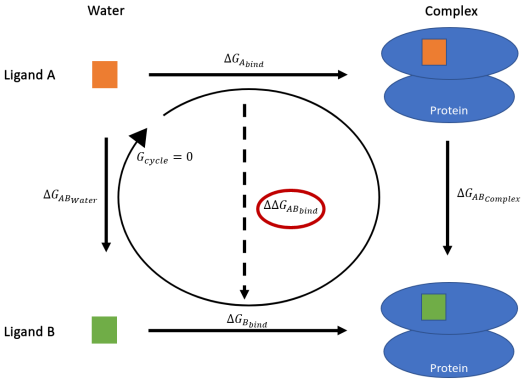
2.6 RESULTS AND DISCUSSION

2.7 APPLICATION EXAMPLE: SIMPLE SIMULATIONS

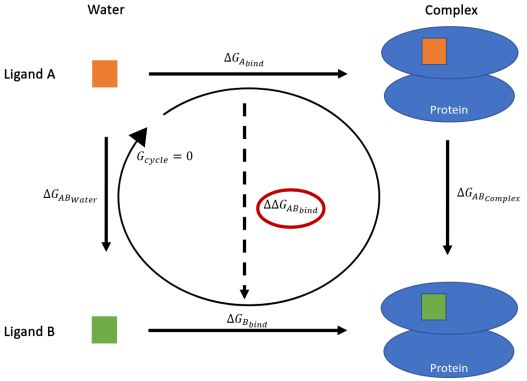
In the following, simple code examples are shown to introduce the usage of Ensembler. In addition, an application example is provided to illustrate the use of Ensembler for teaching about free-energy methods. The code for these examples can be found in the GitHub repository *https://github.com/rinikerlab/Ensembler/examples*.

In typical applications of Ensembler, the user selects a potential-energy function from the available ones. In the following example, a potential-energy function with four wells is selected and initialized with chosen parameters. To sample this four-well potential-energy function with stochastic dynamics (SD),[25] the sampling method is instantiated and passed to the *system class*, which controls the execution of the simulation. The simulation is performed by calling the function *simulate* with the desired number of simulation steps passed as parameter. Subsequently, the results can be visualized using the built-in visualization functions that are compatible with the *simulation class* of Ensembler. As can be seen in Figure 2.1a, the energy barriers between the different minima were not crossed during the chosen simulation length. To overcome the sampling issue, enhanced sampling techniques can be employed.[35] In this example, local elevation[1]/metadynamics[2] is used to overcome the energy barriers (Figure 2.1b). The method

(A) Standard Langevin Simulation



(B) Langevin Simulation with Local Elevation/Metadynamics



(C) Example Source Code

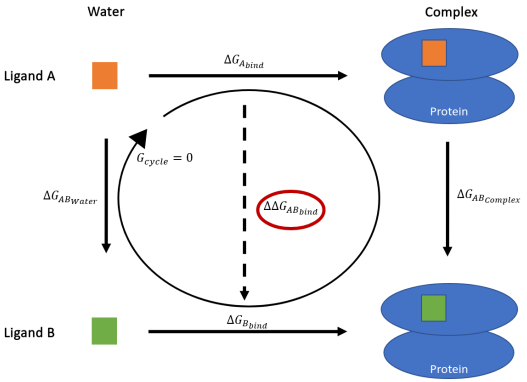


FIGURE 2.1: Langevin simulation of a four-well potential energy-function: Results when sampling (1000 steps) with the standard SD integrator (a) or with local elevation[1]/metadynamics[2] (b). The left

adds a time-dependent biasing potential to the system, i.e. it adds a Gaussian biasing potential to positions that were already visited such that they become energetically less favorable. This decreases the likelihood of visiting known positions again. The enhanced sampling technique can be applied by adding a single line of code compared to the previous simulation (Figure 2.1c).

2.8 APPLICATION EXAMPLE: FREE-ENERGY CALCULATION

Free-energy calculation is an important field in computational chemistry because free-energy differences govern the outcome of processes in nature, e.g. protein-ligand binding or polymer formation.[32,52–54] The calculation of alchemical free-energy differences with Ensembler is exemplified with a mutation of the equilibrium position of a one-dimensional harmonic oscillator (Figure 2.2a). This mutation corresponds to a change of a covalent bond type at the terminus of a linear molecule and can be calculated analytically (Table 2.1). In practical applications, however, it is usually not possible to calculate the free-energy difference analytically. In these cases, MD-based simulation techniques can be employed. In the following, the sampling of the two end states of the model system and the results of the free-energy calculation with different methods are discussed. For more details, we refer to the Jupyter notebook in the Ensembler GitHub repository.

A simple free-energy method is to simulate one end state and estimate the free-energy difference with the Zwanzig equation.[28]

The quality of the result depends on a sufficient phase-space overlap between the two end states.[55] Alternatively, one can simulate both end states separately and use BAR[29] (Figure 2.2a), yielding more converged results.[55] If the phase-space overlap between the two end states is not sufficient, more advanced sampling methods are necessary to obtain converged free-energy differences. One possibility to increase the phase-space overlap is to generate intermediate states as a linear combination of the two end states A and B with the coupling parameter λ, i.e. $H(\lambda) = (1-\lambda)H_A + \lambda H_B$, such that $H(\lambda = 0) = H_A$ and $H(\lambda = 1) = H_B$. The intermediate states are positioned at discrete λ-points between 0 and 1 (Figure 2.2b).[56,57] The free-energy difference can be estimated using FEP[28] or BAR[29] as the path over all intermediates, or with TI[30] as the integral along λ.

Another elegant free-energy method is EDS,[3,31] where a reference-state Hamiltonian H_r is sampled. H_r is constructed as a log-sum of the Hamiltonians of the two (or more) end states, guaranteeing the phase-space overlap of the reference state with all end states,

$$H_R = -\frac{1}{\beta s}\ln(e^{(-\beta s(H_A - E_A^R)} + e^{(-\beta s(H_B - E_B^R)}), \tag{2.1}$$

where $1/\beta = k_B T$, k_B being the Boltzmann constant and T the absolute temperature. H_R can be optimized for sampling using two kinds of parameters: The smoothing parameter s lowers the energy barriers between the end states, whereas the energy offsets E^R ensure equal weighting of all end states. In our example, both end states are sampled sufficiently during the EDS simulation with $s = 0.3$ and the energy offsets $E^R = [0, 0]$ (Figure 2.2c). Subsequently, the Zwanzig equation[28] is used to obtain the free-energy difference between the end states.[3,31] Recently, a hybrid form of EDS and λ-coupling was introduced, termed λ-EDS.[5] At

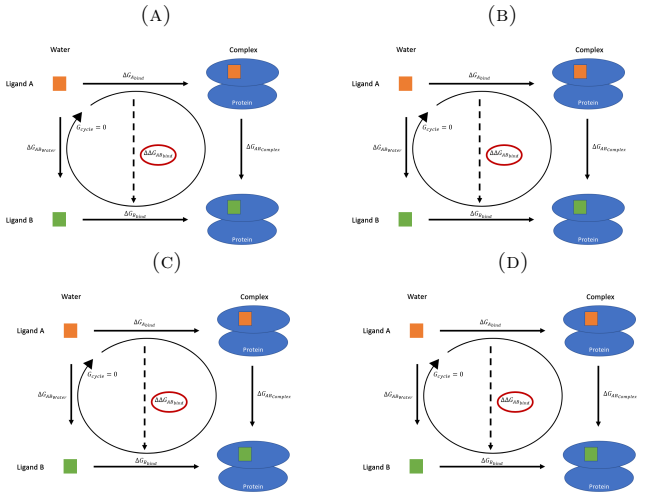


FIGURE 2.2: Illustration of different free-energy methods implemented in Ensembler. (a) For FEP[28] and BAR,[29] the two end states (grey and black) are sampled separately (green and blue). (b) To increase the phase-space overlap, the two end states can be coupled as a linear combination of their Hamiltonians using a coupling parameter λ. This allows the generation of intermediate states (grey to black) and sampling of those (colored points). (c) An alternative method is EDS,[3,31,32] where a reference-state Hamiltonian (blue line) is sampled (blue points), which envelopes the end states. By setting the reference-state parameters to $s = 0.3$ and energy offsets = [0,0], all relevant phase-space regions can be sampled. (d) A recently developed approach called λ-EDS[5] introduces a λ-dependence in the EDS method (blue, orange and green line). Colored points indicate sampling. The reference-state parameters were set to $s = 0.3$ and energy offsets = [0,0], and three different lambda values 0.25,0.5, 0.75 were chosen.

$\lambda = 0$ or 1, the H_R is equal to the Hamiltonians of the respective end states, while conventional EDS is recovered with λ=0.5 (except for an offset).[5] λ-EDS allows for a λ-weighting of the exponential terms in the EDS equation. In the example in Figure 2.2d, the same reference-state parameters were used as before.

All free-energy calculations discussed above were performed with Ensembler for a total of 10'000 Monte Carlo (MC)[23] steps, and each simulation was repeated five times. The simulation results listed in Table 2.1 show that larger errors are obtained without intermediate states due to insufficient phase-space overlap. Using ten λ–intermediate states together with TI gave the best result, however, this approach is also the computationally most expensive one (i.e. ten separate simulations). s EDS and λ-EDS, on the other hand, yielded also good results, while requiring only one simulation (given a set of suitable reference-state parameters). We refer to the Jupyter notebook in the Ensembler GitHub repository for the source code, more detailed information on these methods as well as additional methods like conveyor-belt TI[34] and RE-EDS,[33,58] which combine enhanced sampling and free-energy methods.

TABLE 2.1: Estimated free-energy difference for the model system shown in Figure 2.2. Sampling was performed with Monte Carlo (MC)[23] for 10'000 steps in each simulation. Each calculation was replicated five times and the averaged result is shown together with the standard deviation. The duration of the computations (without visualizations) was estimated directly in the Jupyter notebook and is given relative to the FEP simulation (absolute duration = 2.0 seconds). The performance was tested on a Lenovo Thinkpad T420s with an Intel i5-2520 (2.5 GHz) CPU and 8 GB RAM. The RAM usage for the full Jupyter notebook execution was in total 578 MB.

Method	Average ΔF $[k_BT]$	Deviation from analytical result $[k_BT]$	Speed (rel. Simulation
analytical	1.275	-	-
FEP[28]	6.579 ± 1.009	5.305	1.0
BAR[29]	2.437 ± 0.500	2.437	3.0
FEP 10-λ-points	1.406 ± 0.431	0.131	14.0
TI[30] 10-λ-points	1.242 ± 0.015	0.033	14.0
EDS[3,31,32]	0.958 ± 0.110	0.317	2.4
λ-EDS[5] $\lambda = 0.5$	0.987 ± 0.111	0.287	3.1

2.9 CONCLUSION

In this work, we introduced the Ensembler package as a tool to support teaching of MD simulations and free-energy techniques, and to enable rapid prototyping of new methods using 1D or 2D model systems. The package provides a large set of implemented methods for comparison. The open-source basis, the lean structure, and the simplicity of Python3 form a convenient and efficient framework. The code examples and application example for free-energy calculation highlight the ease of using Ensembler. With this, Ensembler can contribute to improving the shareability, comparability, and reproducibility for method development in our field.

3

Pushing the borders with the alchemical path free multistate method RE-EDS: Method Development, Application and Automatization

"Let us learn to dream, gentlemen, and then perhaps we shall learn the truth."

August Kekulé, 1865

The calculation of relative free-energy differences between different compounds plays an important role in drug design to identify potent binders for a given protein target. Most rigorous methods based on molecular dynamics (MD) simulations estimate the free-energy difference between pairs of ligands. Thus, the comparison of multiple ligands requires the construction of a "state graph", in which the compounds are connected by alchemical transformations. The computational cost can be optimized by reducing the state graph to

a minimal set of transformations. However, this may require individual adaptation of the sampling strategy if a transformation process does not converge in a given simulation time. In contrast, path-free methods like replica-exchange enveloping distribution sampling (RE-EDS) allow the sampling of multiple states within a single simulation without the pre-definition of alchemical transition paths. To optimize sampling and convergence, a set of RE-EDS parameters needs to be estimated in a pre-processing step. Here, we present an automated procedure for this step that determines all required parameters, improving the robustness and ease of use of the methodology. To illustrate the performance, the relative binding free energies are calculated for a series of checkpoint kinase 1 (CHK1) inhibitors containing challenging transformations in ring size, opening/closing, and extension, which reflect changes observed in scaffold hopping. The simulation of such transformations with RE-EDS can be conducted with conventional force fields and, in particular, omit the need for soft bond-stretching terms.

3.1 INTRODUCTION

Rigorous free-energy calculations using molecular dynamics (MD) simulations have become an important tool to estimate binding free energies of novel compounds for lead optimization in drug discovery.[53,54,59] Although computationally relatively expensive, these methods are needed to properly account for entropic contributions introduced by protein/ligand conformational changes, entropy-enthalpy compensation, and the desolvation of a ligand.[60]

Computational free energy calculations typically make use of thermodynamic cycles, i.e., the transitive difference relations of idealized states of the system of interest that are representable by a graph. For instance, to estimate the binding free energy of five compounds, a "state graph" can be constructed (Figure 3.1), where the nodes represent the end states and the edges the free-energy differences between them. Although not impossible,[61] the direct calculation of (absolute) binding free-energies (ΔG_i^{bind}) is generally very challenging to achieve computationally.[59] A simpler alternative is to calculate the alchemical free-energy differences between two compounds i and j in a given environment (ΔG_{ji}^{env}) and then compare the relative binding free energy $\Delta\Delta G_{ji}^{\text{bind}}$ with the difference of the ΔG_i^{bind} obtained from experiments,[62,63]

$$\Delta\Delta G_{ji}^{\text{bind}} = \Delta G_{ji}^{\text{protein}} - \Delta G_{ji}^{\text{water}} = \Delta G_j^{\text{bind}} - \Delta G_i^{\text{bind}} \quad (3.1)$$

3.1.1 Path Methods

Conventional free-energy methods such as thermodynamic integration (TI)[30] and free-energy perturbation (FEP)[28] introduce a coupling parameter λ to define a pathway from end state i ($\lambda = 0$) to end state j ($\lambda = 1$). In practice, simulations at discrete inter-

mediate λ-points are performed to obtain converged free-energy differences. If a (large) series of N compounds is investigated, the free-energy difference for all $(N(N-1))/2$ pairs of ligands would in principle have to be calculated. To reduce the computational cost, automatic schemes have been developed to identify the edges in the state graph (Figure 3.1) with the smallest perturbations such that all nodes (for a given environment) are connected.[64–66] It is thereby important to include some cycles as cycle closure is a frequently used measure to assess convergence. Nevertheless, manual optimizations may sometimes be required to determine the best sampling strategy.[67] Furthermore, calculating only a subset of the edges leads to a larger uncertainty in the estimated free-energy difference for pairs that are no longer directly connected. As $\Delta\Delta G_{ji}^{\text{bind}}$ values are often relatively small, the increased uncertainty may negatively impact the usefulness of such calculations in practical applications.

3.1.2 Pathless Methods

An attractive and more efficient alternative to path-dependent methods is to simulate a reference state, which includes all N end states simultaneously, without the specification of pathways (green rings in Figure 3.1). Such a reference state is provided by the enveloping distribution sampling (EDS)[3,68–70] method. The EDS reference state can be further tuned for optimal sampling with parameters. Note that cycle closure is guaranteed by definition in this approach. In order to enhance sampling further, combinations of EDS with enhanced sampling methods were developed such as replica-exchange EDS (RE-EDS)[71–73] and accelerated EDS.[74,75]

In this study, we present an improved automated workflow for RE-EDS simulations to calculate the relative binding free energies

of multiple ligands from a single simulation per environment. The robustness and versatility of the RE-EDS workflow are demonstrated on a series of five inhibitors of human checkpoint kinase 1 (CHK1).[76] These ligands were selected by Wang *et al.*[77] as a challenging benchmarking set for FEP calculations since the changes between these ligands exemplify different types of core-hopping transformations (i.e. ring size change, ring opening/closing, and ring extension). Special soft bond-stretching terms were developed to be able to handle these transformations.[77] In contrast to many other methods, no such special soft bonds are required with RE-EDS as we can use a "dual topology" approach[70] in a straightforward manner.

3.2 THEORY

3.2.1 Enveloping Distribution Sampling (EDS)

In EDS, free-energy differences between multiple end states are obtained by sampling a reference-state Hamiltonian, i.e. without the definition of specific alchemical paths.[3,68,70] Given N end states, the potential energy function V of the EDS reference state R is defined as,

$$V_R(\vec{r}; s, \vec{E}^R) = -\frac{1}{\beta s} \ln \left[\sum_{i=1}^{N} e^{-\beta s \left(V_i(\vec{r}) - E_i^R \right)} \right], \tag{3.2}$$

where $\beta = (k_\mathrm{B} T)^{-1}$ with k_B being the Boltzmann constant and T the absolute temperature. The smoothing parameter s and the energy offsets $\vec{E}^R$ were introduced to enable tuning of the reference state for optimal sampling of all end states.[3,68]

A smoothness parameter set to $s = 1.0$ gives a reference potential-energy landscape that contains all the relevant minima of the end states. However, these might be separated by high barriers. For $s < 1$, the energy barriers between different end states V_i are smoothed in the reference potential V_R, increasing the transition rates between the different minima (Figure 3.2a).[68] However, if s is chosen too small, V_R consists of a global unphysical minimum, which does not correspond to any of the end states. In the limit of $s \to 0$, all end states contribute equally to the potential-energy function of the reference state,[5] which can lead to unphysical deformations. The situation with a too small s has been termed "undersampling".[70]

The energy offsets $\vec{E}^R$ are used to ensure equal weighting of all end states V_i in V_R (Figure 3.2b). Note that the optimal values of s and $\vec{E}^R$ are not independent of each other (as can be seen in Eq. (3.2)).[68] Different schemes have been proposed to determine optimal reference-state parameters,[69,70,78] however, these are only applicable to systems with two end states.

The force on a particle k in the EDS reference state is calculated as,[68]

$$\vec{f}_k(t) = -\frac{\partial V_R(\vec{r}; s, \vec{E}^R)}{\partial \vec{r}_k} = \sum_{i=1}^{N} \frac{e^{-\beta s(V_i(\vec{r}) - E_i^R)}}{\sum_{j=1}^{N} e^{-\beta s(V_j(\vec{r}) - E_j^R)}} \left(-\frac{\partial V_i(\vec{r})}{\partial \vec{r}_k} \right) . \tag{3.3}$$

For s values close to one, the reference-state forces are dominated by the one end state, for which the current coordinates are most favourable, while the other end states give high energies and therefore contribute little (i.e. "dummy states"). For small s values (undersampling situation), all end states contribute effectively to the forces, resulting in the global unphysical minimum.

The free-energy difference between two end states A and B can be calculated by employing the Zwanzig equation twice forming a path via the reference state R,[3,28,68]

$$\begin{aligned}\Delta G_{\mathrm{BA}} &= \Delta G_{\mathrm{BR}} + \Delta G_{\mathrm{RA}} \\ &= -\frac{1}{\beta}\left(\ln\langle e^{-\beta(V_B - V_R)}\rangle_R - \ln\langle e^{-\beta(V_A - V_R)}\rangle_R\right) \quad (3.4) \\ &= -\frac{1}{\beta}\ln\frac{\langle e^{-\beta(V_B - V_R)}\rangle_R}{\langle e^{-\beta(V_A - V_R)}\rangle_R}. \quad (3.5)\end{aligned}$$

3.2.2 Replica-Exchange EDS (RE-EDS)

The recently introduced RE-EDS method[72,73] is a type of Hamiltonian replica exchange[50,80] with the smoothness parameter s as the exchange dimension ($1 >= s > 0$), which was inspired from constant pH simulations by Lee *et al.*[71,81] The approach is shown schematically in Figure 3.3. RE-EDS does not require a single (optimal) s-value. Instead enhanced sampling is achieved by exchanging between the replicas with different smoothness levels. This simplifies the parameter choice problem and thus, the method can be applied to systems with more than two end states.[72,73]

For the pairwise exchanges between neighboring replicas k and l, a Metropolis-Hastings criterion[23] is used,[50,72]

$$p_{k,l} = min\left(1, \exp\left[V_R(\vec{r}_k; s_l) - V_R(\vec{r}_l; s_k) - (V_R(\vec{r}_l; s_l) - V_R(\vec{r}_k; s_k)\right]\right), \quad (3.6)$$

where H_{R_k} and H_{R_l} are the reference-state Hamiltonians of the respective replicas, $\vec{r}_k$ and $\vec{r}_l$ are the current coordinates of the

replicas.

Replicas are placed between $s = 1.0$ and a lower bound of s, where the reference state is in undersampling. The replicas with low s values facilitate the transitions between the low-energy regions of the different end states. Especially for systems with slowly adapting environments (e.g. protein binding pockets), regions in s-space with very low acceptance probability can occur. Thus, to ensure sufficient exchanges between all pairs of replicas, a local variant of the round-trip time optimisation algorithm[82,83] was developed to optimally place the replicas in s-space.[73] It was found that a single set of energy offsets can be used for all replicas.[72] However, it is important that these energy offsets are chosen well to avoid "leakage" effects, resulting in one or more end states not being properly sampled. The final free-energy differences are estimated from the replica at $s = 1.0$, which represents the physical minima of the end states.

3.2.3 Automatic Parameter Optimization

To facilitate the determination of the energy offsets and s-parameter distribution, we have extended and further automatized the previous[73] RE-EDS workflow (Figure 3.4). The initial input for a system with N end states consists of a prepared EDS system (i.e. topology, perturbation topology, initial coordinates, and distance restraints), a list of energy offsets of length N with $E_i^R = 0;\ \forall\ i \in [1, ..., N]$, and a list of s-parameters, which are logarithmically distributed in the range $s_i \in [1, 10^{-5}]$. Typically, we use 21 initial s values.

The parameter exploration consists of two substeps: (i) determining the lower bound for the s-distribution, and (ii) obtaining optimized coordinates within the EDS set-up for each end state.

To enable sampling of all end states at $s = 1.0$, some replicas have to be in undersampling to facilitate transitions. However, for efficiency reasons (and numerical stability) the number of replicas M in undersampling should be small and the lowest s-value should be as high as possible. From a short simulation with the initial s-distribution between $[1, 10^{-5}]$, the highest smoothing parameter $s_{M_{\mathrm{us}}}$ at which undersampling still occurs is determined and used in the following as a lower bound for the s-distribution. The s-distribution for the next step is then defined by 21 replicas logarithmically distributed between $s = 1.0$ and the new lower bound.

Optimized coordinates for each end state in the EDS setup are obtained by short parallel simulations, where one end state in turn is favoured by setting an arbitrarily large energy offset for this state. The optimized coordinates allow the user to start RE-EDS simulations from different end states and are needed for the subsequent parameter optimization.

In the second step, the energy offsets and subsequently the optimal s-distribution are estimated. For the first substep, the previously developed parallel energy offset estimation (PEOE)[72] scheme is used. From a short simulation with the initial parameters, the energy offsets are extracted from the replicas k in the undersampling region using,[72]

$$E_i^R(new) = -\frac{1}{\beta} \ln \left\langle e^{-\beta\left(V_i(\vec{r}) - V_R(\vec{r}; s_k, \vec{E}^R(old))\right)} \right\rangle_{R(s_k, \vec{E}^R(old))}. \tag{3.7}$$

The s-distribution is optimized by minimizing the round-trip time τ using the multistate local round-trip time optimization (N-LRTO) algorithm.[73] The optimization is performed in an iterative manner with short simulations. New replicas in an iteration are

positioned by linear interpolation in the regions where exchange bottlenecks are detected, while the replicas of previous iteration remain fixed. The bottlenecks are identified for each end state separately (i.e. multistate). The number of replicas added can be chosen by the user. The iteration is stopped when the average round trip time $\overline{\tau}$ converges. The local variant of the optimization algorithm is needed for situations with severe bottlenecks with the initial logarithmic s-distribution (e.g. protein binding pockets). For systems with smaller perturbations, the global multistate variant (N-GRTO)[73] can be more efficient. In this study, we started with the same number of replicas as used for the PEOE scheme above (21 s-values), and added four replica positions per iteration in the N-LRTO algorithm.

After optimizing the RE-EDS parameters, the production run is performed for a chosen length. The free-energy differences are subsequently calculated using the replica at $s = 1.0$ with Eq. (3.5).

Starting State Mixing

The sampling in RE-EDS simulations can be further improved by using starting coordinates for the replicas corresponding to the different end states (i.e. replica 1 starts in a low-energy configuration for end state 1, replica 2 in a low-energy configuration for end state 2, etc.). This technical approach is called "starting state mixing" (SSM) in the following and is also used for Hamiltonian replica-exchange TI calculations (see e.g.[84,85]). The optimized coordinates obtained in the parameter exploration step can be used for SSM. We compare RE-EDS simulations with SSM and with a single set of starting coordinates (abbreviated as 1SS).

Analysis

Three types of metrics were used to quantify the sampling in RE-EDS simulations. The first metric determines for each end state i the sampling fraction as dominating state, i.e. f_i^{domin}. A dominating state is defined as the end state with the lowest potential energy in a frame. As can be seen in Eq. (3.3), dominating end states have the largest impact on the reference state sampling at a given time point. Optimal sampling in a RE-EDS system is achieved when all end states are sampled as dominating states to an equal extent at $s = 1.0$, i.e.

$$f_i^{\text{domin,ideal}} = \frac{1}{N} \ , \forall \ i \ \in \ \{1, ..., N\} \tag{3.8}$$

The second metric is the sampling fraction of "physical occurrence" of an end state i, i.e. f_i^{occur}. As a result of phase-space overlap with the current dominating end state, other end states in the EDS system might be sampled simultaneously. An end state is counted as "occurred" when its potential energy is below the threshold T_i^{phys} at a time point t. These thresholds are estimated during the second substep of the parameter exploration phase. If end states show no phase-space overlap, f_i^{occur} will be (nearly) the same as f_i^{domin}.

Undersampling is detected with a third metric using the thresholds T_i^{us}. These thresholds are determined in the first substep of the parameter exploration phase from the simulation with the lowest s-value. If all end states have a potential energy below their respective T_i^{us}, the current frame is characterized as undersampling.[72]

3.3 COMPUTATIONAL DETAILS

3.3.1 Model System

To showcase the performance of RE-EDS, a system of five inhibitors (L1, L17, L19, L20 and L21) of checkpoint kinase 1 (CHK1) taken from Ref.[76] was chosen (Figure 3.5). The numbering of the compounds is according to Ref.[76] The same system was studied in Ref.[77] as part of a series of scaffold hopping systems. Although the five ligands share a common substructure, they were considered to exemplify different types of core-hopping transformations (i.e. ring size change, ring opening/closing, ring extension) and R-group modifications.[77]

For the protein, the GROMOS 54A7 force field[86] was used. For the ligands, topologies were generated using the parametrization by the ATB server[87] as an initial guess. The bonded terms were manually harmonized and adjusted to match the parameterization of similar functional groups in the GROMOS 54A7 force field. Partial charges were generated with our previous machine learning approach[88] ($\epsilon = 4$) and manually arranged into charge groups. The input files can be retrieved from:
https://github.com/rinikerlab/reeds/tree/main/examples/systems.

3.3.2 System Preparation

The crystal structure of CHK1 in complex with ligand L1 (PDB ID:3U9N) was used as starting structure. The initial coordinates for ligands L17, L19, L20, L21 were generated with the `ConstrainedEmbed()` functionality in the RDKit,[89] where the common part was kept fixed in the crystal conformation. The coordinates of each ligand and those of the protein were subse-

quently energy minimized in vacuum using the steepest descent[90] approach implemented in the GROMOS software package.[91]

A "dual topology" approach was used for the RE-EDS simulations, i.e. each ligand is present in the system separately.[70] Thus, each end state comprises of one active ligand and $N-1$ inactive (dummy) ligands. To avoid spatial drifting of the dummy ligands, eight distance restraints per ligand pair were defined within the common substructure (Figure 3.5) to connect all ligands in a ring with the help of the restraintmaker program (https://github.com/rinikerlab/restraintmaker) (order: -L1-L17-L19-L20-L21-). The reference distance was set to 0.0 nm and the force constant to 1000 kJ mol^{-1} nm^{-2}. The combined topology file was generated with the program `prep_eds` in the GROMOS++[92] package. The EDS system was solvated in a cubic box of single-point-charge (SPC)[93] water (resulting in 1'848 solvent molecules for the ligands in water and 15'639 solvent molecules for the protein-ligands complex). An energy minimization was carried out with the steepest descent algorithm,[90] where all solute atoms were position restrained with a force constant of 25′000 kJ mol^{-1} nm^{-2}.

3.3.3 Simulation Details

All simulations were performed with the GROMOS software package[91] (freely available on http://www.gromos.net). The equilibrations and production runs were carried out under isothermal-isobaric (NPT) conditions using the leap-frog integration algorithm[94] and a time step of 2 fs. Bond lengths were constrained with SHAKE[95] using a tolerance of 10^{-4}. The nonbonded contributions were calculated with a twin-range scheme using a short-range cutoff of 0.8 nm and a long-range cutoff of 1.4 nm. The electrostatic nonbonded contributions beyond the long-range cutoff

were calculated with the reaction-field[96] approach and a dielectric permittivity of 66.7[97] for water. The temperature was kept constant at 300 K using the weak coupling scheme[98] and a coupling time of 0.1 ps^{-1}. The pressure was kept at 1.031 bar (1 atm) with the same type of algorithm and a coupling time of 0.5 ps^{-1} and an isothermal compressibility of $4.575 \cdot 10^{-4}$ (kJ mol^{-1} $nm^{-3})^{-1}$. Rotation and translation of the center of mass of the simulation box were removed every 2 ps. Energies were written to file every 20 steps and coordinates every 5′000 steps. In the RE-EDS simulations, replica exchanges was attempted every 20 steps.

3.3.4 RE-EDS Workflow

The Python code to manage the RE-EDS workflow, including the analysis steps, can be retrieved from: https://github.com/riniker-lab/reeds. The workflow starts with the energy-minimized coordinates of the EDS system (all N ligands plus environment, dominating end state is L20) into the parameter exploration step, which is used as equilibration phase. A RE-EDS simulation of 0.2 ns length was performed with 21 logarithmically distributed replicas between $s = 1.0$ and 10^{-5} and all energy offsets set to zero. The thresholds T_i^{us} were estimated from replicas with very low s-values. Undersampling was observed when each end state occurred with a fraction $f_i^{\mathrm{occur,us}} \geq 0.75$ during the simulation period. To be conservative, the lower bound of the s-parameters for the following steps was set to the s-value two levels below the highest replica with undersampling.

To optimize the coordinates of the system for each end state, an EDS simulation of 0.2 ns length was performed for each end state i with $s = 1.0$ and $E_i^R = 500$ kJ mol^{-1} while the energy offsets of all other end states were set to -500 kJ mol^{-1}. L20 was

the initial dominating end state in the starting configuration. The coordinates were considered to be optimized when the desired end state was constantly sampled as the dominating state in the last 30 % of the simulation.

To determine the energy offsets, a 1.2 ns RE-EDS simulation was carried out with 21 logarithmically distributed replicas between $s = 1.0$ and the lower bound (determined above). The first 0.4 ns of the simulation were discarded as equilibration. This simulation was performed in two manners: (i) using the final coordinates from the lower-bound determination as starting configuration for all replicas (1SS approach), or (ii) using the different optimized coordinates from the previous substep for the replicas in an alternating way (SSM approach). For the PEOE[72] scheme, the following parameters were used: fraction $f_i^{\mathrm{us}} \geq 0.9$ and the potential thresholds determined in the lower bound exploration T_i^{us}.

The iterative optimization of the s-distribution with the N-LRTO[73] algorithm was started with the energy offsets and the final coordinates of the previous substep. Four replicas were added per iteration. The simulation length of the first iteration was 0.4 ns, and subsequently increased by 0.4 ns at each iteration until a maximum length of 1.2 ns was reached. The s-distribution was considered converged if all end states were sampled as dominating states during the RE-EDS simulation at $s = 1.0$ and the improvement of the round-trip time was below $\Delta\tau < 5$ ps.

The production run with constant reference-state parameters was performed for 4 ns.

3.3.5 Simulation of Single States

The input coordinates for the simulations of the individual end states were extracted from the RE-EDS starting coordinates and subsequently energy minimized. Next, a production run of 4 ns was performed.

3.3.6 Analysis

Free-energy differences were calculated with the program `dfmult` from the *GROMOS++*[92] package. Statistical analysis and handling of the workflow steps are based on the Python packages pandas,[44] Matplotlib,[45] NumPy,[42] SciPy,[41] and PyGromosTools.[99]

3.4 RESULTS AND DISCUSSION

The chosen model system of five inhibitors of CHK1 kinase exemplifies different core-hopping transformations (i.e. ring size change, ring opening/closing, ring extension) and R-group modifications,[77] increasing the complexity compared to the systems previously studied with RE-EDS. Furthermore, the performance can be directly compared to the results obtained with FEP+ and OPLS3 in Ref.[77] as well as with QligFEP results in Ref.[67]

3.4.1 Parameter Exploration and Parameter Optimization

The RE-EDS workflow was started by estimating the lower bound for the s-distribution. Using the above mentioned undersampling criterion (see Methods section), a lower bound of $s = 0.003$ was

determined. Optimized coordinates were obtained for all five ligands, as verified by comparing the potential-energy distribution from the EDS simulation with the one extracted from a standard MD simulation of the respective ligand (Figure S1 in the Supporting Information). From these same steps, the potential-energy thresholds for the occurrence sampling (T_i^{phys}) and undersampling (T_i^{us}) were estimated.

The energy offsets $\vec{E}^R$ were estimated from a short RE-EDS simulation with the PEOE[72] scheme and are listed in Table 3.1. For $s = 1.0$, the energy offsets should ideally be equal to the free energy of the corresponding state (i.e. $\Delta E_{ji}^R = \Delta G_{ji}$) such that the partition function of the reference state is the sum of the partition functions of the end states.[68] Therefore, the comparison between the relative estimated energy offsets in water and in complex ($\Delta\Delta E_{ji}^R = \Delta E_{ji,\text{complex}}^R - \Delta E_{ji,\text{water}}^R$) and the relative binding free energy $\Delta\Delta G_{ji}^{\text{bind}}$ can be used to (roughly) assess the quality of the estimated energy offsets. As shown in Figure S2 in the Supporting Information, the energy offsets estimated from the SSM simulations are in better agreement with the experimental relative binding free energies than those estimated from the 1SS simulations.

TABLE 3.1: Energy offsets $\vec{E}^R$ estimated from a short RE-EDS simulation using the PEOE[72] scheme. The errors indicate the standard deviation over the different replicas in undersampling. All energy offsets were calculated relative to ligand L1. The starting coordinates were selected following the 1SS or the SSM approach (see Theory and Methods sections).

Ligand	RE-EDS 1SS		RE-EDS SSM	
	Water [kJ mol^{-1}]	Complex [kJ mol^{-1}]	Water [kJ mol^{-1}]	Complex [kJ mol^{-1}]
L1	0.0	0.0	0.0	0.0
L17	15.9 ± 4.9	14.1 ± 1.9	12.9 ± 2.3	19.1 ± 3.2
L19	9.6 ± 5.3	-5.8 ± 0.5	-5.4 ± 4.7	-2.3 ± 3.1
L20	-52.4 ± 4.4	-48.5 ± 3.0	-49.7 ± 8.8	-55.0 ± 1.5
L21	-69.9 ± 1.8	-70.0 ± 8.8	-72.5 ± 6.0	-72.9 ± 3.0

The optimization of the s-distribution was performed with the N-LRTO[73] algorithm, thereby minimizing the average round-trip time $\overline{\tau}$ in the replica graph. In the first iteration, the number of total round trips is relatively small and the average round-trip time is large for all simulations (Figure 3.6). The number of round trips is smaller in the complex than in water due to a more pronounced gap region.[73] Already after the second iteration, the round-trip time is generally reduced. An exception was observed for RE-EDS 1SS in the complex, where no round trips occurred during the second iteration. The improvement of the $\overline{\tau}$ over the iterations can also be seen in Figure S4 in the Supporting Information. As can be seen in the third row of Figure 3.6, the optimization algorithm increases the density of the replicas around $s = 0.041$, where the major gap region lies.

In addition, we monitored the relative sampling of the end states at $s = 1.0$ during the iterations. Ideally, each end state should be sampled equally in an optimized RE-EDS simulation (see Eq. (3.8)). The last row in Figure 3.6 shows f_i^{domin} as a function of the iteration. For all end states, the sampling fraction converges (slowly) towards the ideal value. The s-optimization for the ligands in water converged after the fourth iteration with $\Delta\overline{\tau} < 10\ ps$ (Figure S4 in the Supporting Information). This resulted in the final 36 replicas. For the protein-ligands complex, the optimization converged after the fifth iteration, resulting in 41 replicas. The average round-trip time after convergence was $\overline{\tau} = 9.6 \pm 0.9$ ps for all simulations.

3.4.2 Free-Energy Calculation

After successfully optimizing the RE-EDS parameters, the production runs were performed for 4 ns. Both in water and in complex,

the potential-energy distributions of the end states match generally well the corresponding distributions from the standard MD simulations of the single end states (Figure 3.7). The analysis of the dominating end states at $s = 1.0$ shows that L19 is generally oversampled, while L20 and/or L21 are less sampled than expected (Figure S5 in the Supporting Information). At $s = 1.0$, f_i^{occur} and f_i^{domin} are very similar, which indicates sampling of clearly separated states in those simulations.

From the replica at $s = 1.0$, the free-energy differences were calculated using Eq. (3.5) and the resulting $\Delta\Delta G_{ji}^{\mathrm{bind}}$ were compared with the experimental results taken from Ref.[76] The results are shown graphically in Figure 3.8 and numerically in Table 4.2. The individual free-energy differences are given in Table S3 in the Supporting Information. The RMSE with RE-EDS 1SS is 7.3 kJ mol^{-1} and the MAE is 5.75 ± 4.4 kJ mol^{-1}. The main deviations stem from ligand L19 in the RE-EDS 1SS approach.

The performance was substantially improved using the SSM approach with RE-EDS, giving an RMSE of 2.5 kJ mol^{-1} and an MAE of 2.1 ± 1.3 kJ mol^{-1}. Only one value (L21-L17) deviates more than 4.184 kJ mol^{-1} (i.e. 1 kcal mol^{-1}) from experiment. The Spearman correlation coefficient for RE-EDS 1SS is $r^2_{\mathrm{Spearman}} = 0.87$ and for RE-EDS SSM $r^2_{\mathrm{Spearman}} = 0.84$.

Next, we assessed the convergence of the ΔG_{ji} values as a function of simulation time (Figure S7 in the Supporting Information). For the RE-EDS 1SS approach, all free-energy differences appeared converged after 1.48 ns in water and after 1.04 ns in the complex. For the RE-EDS SSM approach, convergence was observed after 0.52 ns in water and after 0.88 ns in the complex. These findings indicate that the use of different starting configurations representing all end states enhances sampling further and reduces the simulation time needed to obtain converged results.

By applying the RE-EDS methodology to the same system of five CHK1 inhibitors as studied by Wang *et. al.*[77] and later on also Jespers *et al.*,[67] a direct comparison with FEP+ and QligFEP is possible (Table 4.2). Note that the quality metrics were calculated over all possible pairs of ligands, not only those directly calculated by FEP+ and QligFEP. For FEP+, we obtained an RMSE of 2.4 kJ mol^{-1} and an MAE of 1.8 ± 1.2 kJ mol^{-1} with a Spearman correlation coefficient of $r^2_{\text{Spearman}} = 0.67$. Including cycle closure correction (CC)[77] reduced the RMSE to 2.1 kJ mol^{-1} and the MAE to 1.9 ± 1.0 kJ mol^{-1}. The Spearman correlation coefficient increased to $r^2_{\text{Spearman}} = 0.73$. Jespers *et al.*[67] reported free-energy differences with QligFEP as an average over ten independent replicas, each with significantly less simulation time per λ-window than in Ref.[77] For QligFEP, an RMSE of 2.3 kJ mol^{-1}, an MAE of 2.0 ± 1.2 kJ mol^{-1}, and a Spearman coefficient of $r^2_{\text{Spearman}} = 0.61$ was obtained.

Overall, the performance of RE-EDS SSM is comparable with the pairwise methods. The results with FEP+ CC and QligFEP showed a slightly higher accuracy compared to experiment, likely due to the different force fields used. The Spearman correlation coefficient is higher for both RE-EDS approaches than with the pairwise approaches, indicating a good ranking of the ligands with the RE-EDS method. A strong correlation with experiment is of interest in drug design approaches, as the ranking of ligands in virtual screening is important to suggest the most promising drug candidates to be synthesized.

In terms of computational cost, the RE-EDS approach (with 4 ns per replica) resulted in about half the total simulation time (in ns) than reported for the FEP+ calculations in Ref.[77] A major advantage of the simultaneous simulation of multiple ligands in a single RE-EDS simulation is that all $N(N-1)/2$ transforma-

tions are sampled directly, leading to low statistical errors and removing the need of a state graph. This advantage increases with increasing number of ligands. The current workflow of RE-EDS uses a relatively large amount of simulation time for parameter optimization. Future work will focus on further optimization of the workflow to reduce the pre-processing time.

TABLE 3.2: Relative binding free energies $\Delta\Delta G_{ji}^{\text{bind}}$ from experiment and calculated with the RE-EDS 1SS and RE-EDS SSM approaches. For comparison, the results for FEP+ with and without cycle closure (CC) correction taken from Ref.[77] and the results for QligFEP taken from Ref.[67] are listed. The free-energy differences of directly simulated paths were used to infer not directly simulated free-energy differences (marked in bold). If multiple indirect paths were possible, their average was used. The errors for QligFEP were determined in Ref.[67] by calculating the standard deviation over ten replicas. For FEP+, the error of the results was taken from the used BAR[29] method and the FEP+ CC errors were obtained from the cycle closure analysis. For the RE-EDS approaches, the reported error is based on the statistical uncertainties of the ΔG_{ji}^{env} values estimated using Gaussian error approximation.[68]

Ligands i	Ligands j	Exp.[76] [kJ mol^{-1}]	FEP+[77] [kJ mol^{-1}]	FEP+ CC[77] [kJ mol^{-1}]	QligFEP[67] [kJ mol^{-1}]	RE-EDS 1SS [kJ mol^{-1}]	RE-EDS SSM [kJ mol^{-1}]
L17	L1	0.1	-3.6 ± 0.4	-2.9 ± 1.0	-1.6 ± 1.7	1.2 ± 0.3	3.4 ± 0.2
L19	L1	-4.8	-3.9 ± 0.3	-4.0 ± 0.6	-1.7 ± 2.0	-14.0 ± 0.3	-3.9 ± 0.3
L20	L1	-2.0	-2.5 ± 0.1	-3.1 ± 1.0	-1.3 ± 1.3	2.6 ± 0.3	-2.6 ± 0.4
L21	L1	-2.3	**-3.4 ± 0.7**	**-3.2 ± 1.3**	**-0.1 ± 3.5**	-1.7 ± 0.4	-3.6 ± 0.9
L19	L17	-4.9	-1.4 ± 0.3	-1.1 ± 1.0	**0.1 ± 2.6**	-15.2 ± 0.2	-7.3 ± 0.2
L20	L17	-2.1	0.3 ± 0.4	-0.1 ± 0.8	-1.3 ± 2.3	1.4 ± 0.3	-6.0 ± 0.4
L21	L17	-2.4	-1.1 ± 0.4	-0.9 ± 0.9	**0.7 ± 2.6**	-2.9 ± 0.4	-7.0 ± 0.9
L20	L19	2.8	**0.8 ± 0.6**	**0.1 ± 1.3**	**-0.4 ± 3.7**	16.6 ± 0.4	1.3 ± 0.4
L21	L19	2.5	-0.1 ± 0.6	0.6 ± 0.1	0.6 ± 4.9	12.3 ± 0.5	0.3 ± 0.9
L21	L20	-0.3	-0.3 ± 0.8	-0.6 ± 0.8	0.6 ± 1.1	-4.3 ± 0.5	-1.0 ± 0.9
RMSE			2.4	2.1	2.3	7.3	2.5
MAE			1.8 ± 1.2	1.9 ± 1.0	2.0 ± 1.2	5.8 ±4.4	2.1 ± 1.3
r^2_{Spearman}			0.67	0.73	0.61	0.87	0.84

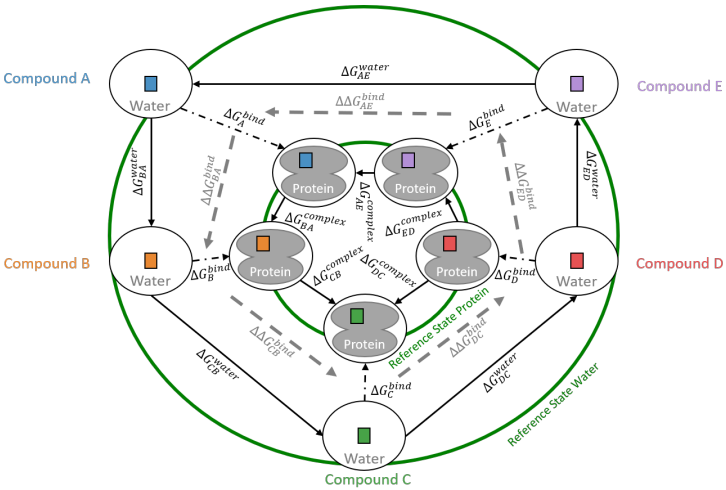
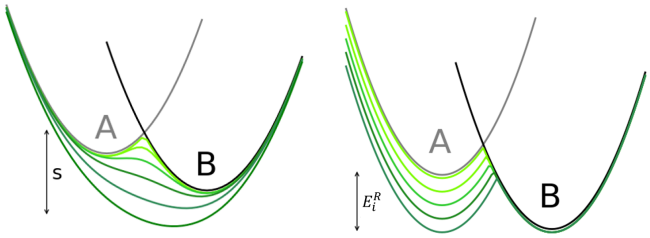


FIGURE 3.1: State graph to calculate relative binding free energies, where the nodes represent specific compounds A - E in a particular environment (water/protein). The connecting (directed) edges describe the transformations from one end state to another. The dashed-dotted arrows denote the direct calculation of the (absolute) binding free energy of compound i to the protein, ΔG_i^{bind}, whereas solid arrows indicate alchemical transformations between compound i to compound j in a given environment. From the resulting $\Delta G_{ji}^{\text{env}}$, $\Delta\Delta G_{ji}^{\text{bind}}$ can be calculated and compared with the value obtained from the difference of the experimentally determined ΔG_i^{bind} (gray dashed arrows). In pathway-dependent methods, each edge between two end states is calculated separately. With (RE-)EDS, all end states in a given environment can be considered simultaneously in a single simulation of a reference state (green circles).



(A) Effect of s on V_R (B) Effect of E_i^R on V_R

FIGURE 3.2: Schematic illustration of the effect of the two types of EDS reference-state parameters. (a) The smoothing parameter s decreases the barriers between the end states. If s is too small, an "undersampling" situation occurs with a global unphysical minimum. (b) The energy offsets $\vec{E}^R$ provide equal weighting to all end states in the EDS reference state. The figure was generated with Ensembler.[79]

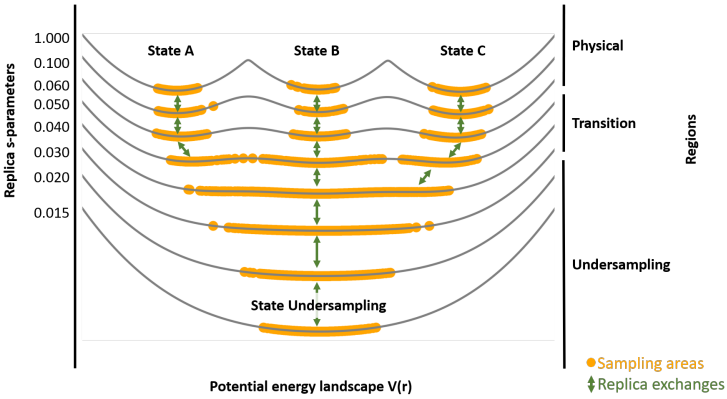


FIGURE 3.3: Schematic illustration of RE-EDS with three harmonic oscillators as end states (A, B, and C). Each replica differs by the s-parameter, generating reference states with a different degree of smoothness. Sampling of each replica is denoted with orange dots. Exchanges between the replicas are indicated with green arrows. The replica graph shows three regions: a "physical" region where s is close to 1, a transition region, and the "undersampling" region when s approaches zero. The figure was generated with Ensembler.[79]

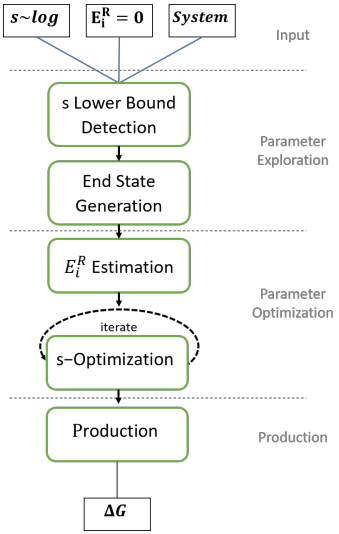


FIGURE 3.4: The RE-EDS workflow can be split into four steps: (1) Input stage with energy offsets set to $E_i^R = 0$ and a set of s-parameters logarithmically distributed between 1 and 10^{-5}; (2) Parameter exploration to determine the lower bound for s and to obtain equilibrated coordinates for each end state; (3) Parameter optimization to determine the energy offsets with the PEOE scheme[72] and the optimized s-distribution with the N-LRTO algorithm[73] ; (4) Production run and calculation of the free-energy differences.

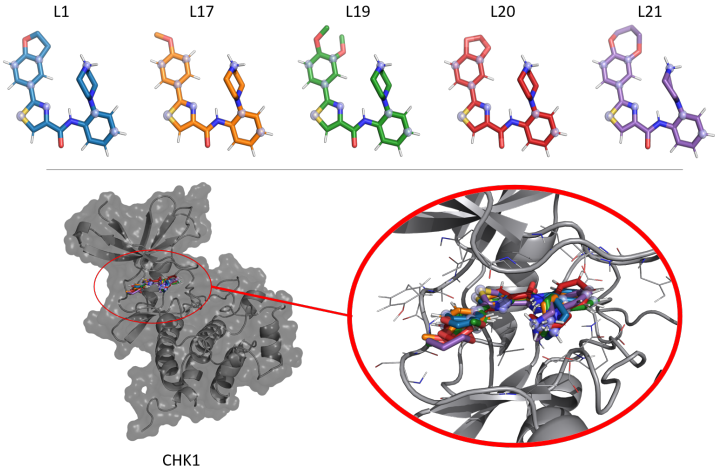


FIGURE 3.5: (Top): 3D depiction of the five CHK1 inhibitors L1, L17, L19, L20, and L21 (numbering according to Ref.[76]). The selected locations of the distance restraints are indicated by the silver spheres. (Bottom): CHK1 protein in complex with the ligand bundle (PDB ID:3U9N).

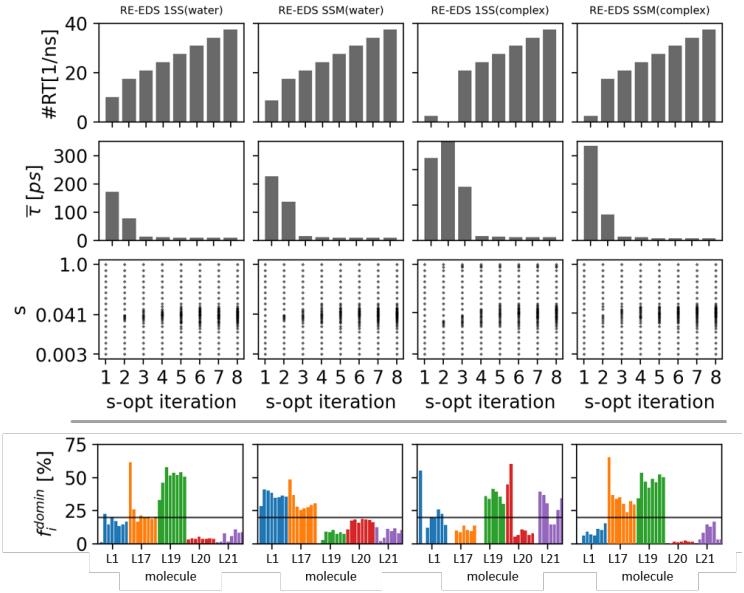


FIGURE 3.6: Optimization of the s-distribution with the N-LRTO[73] algorithm over eight iterations. The measured quality criteria were the number of round trips (1. row), the average round-trip time $\overline{\tau}$ (2. row), the placement of the replicas in s-space (3. row), and the sampling fractions of dominating states f_i^{domin} (4. row).

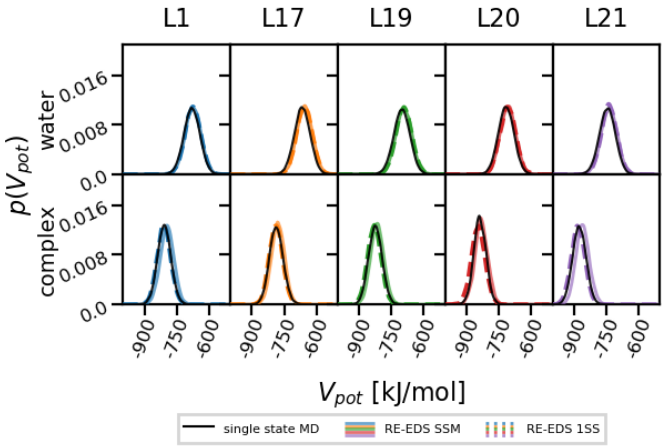


FIGURE 3.7: Comparison of the Boltzmann reweighted potential-energy distributions obtained from standard MD simulations of a given end state (black) and from the RE-EDS production runs (colored).

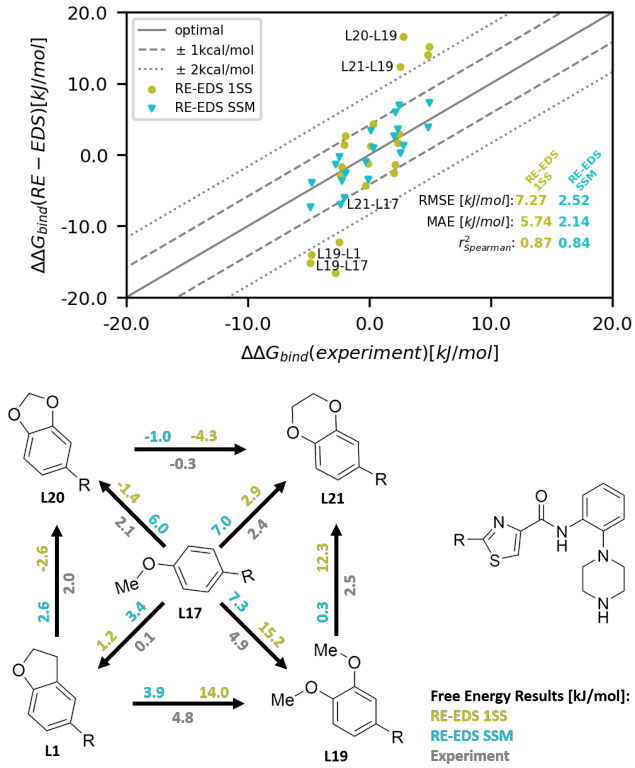


FIGURE 3.8: Free-energy differences estimated from the production run of 4 ns length. (Top): Comparison between the experimental and calculated $\Delta\Delta G_{ji}^{\text{bind}}$ using RE-EDS 1SS and RE-EDS SSM. (Bottom): Graphical representation of the $\Delta\Delta G_{ji}^{\text{bind}}$ results with structures, inspired by the one in Ref.[77]

3.5 CONCLUSION

This study reports the recent developments for the multistate free-energy method RE-EDS, which omits the definition of alchemical transition paths. The automatic workflow for RE-EDS was improved in robustness, and was applied to estimate the relative binding free energies of five CHK1 inhibitors containing typical core-hopping transformations. This system was investigated previously with FEP+ and QligFEP, allowing for a direct comparison of RE-EDS with state-of-the-art pairwise free-energy methods. Using different starting configurations representing all end states (SSM approach) in the parameter optimization of the RE-EDS workflow improved the sampling, convergence, and the accuracy of the resulting free-energy differences. The performance of RE-EDS SSM was found to be comparable with FEP+ and QligFEP, and shows that RE-EDS with a "dual topology" approach can be readily applied to challenging ligand transformations like ring size change, ring opening/closing, and ring extension.

In terms of computational efficiency, the total production run time with RE-EDS (4 ns per replica) was about half of that reported for FEP+ with this system. For RE-EDS, the simulation time could have been reduced further as the free-energy differences were found to be converged already after about 1 ns. As multiple ligands are simulated simultaneously in a single RE-EDS simulation, this sampling enhancement will increase with increasing number of ligands. However, the pre-processing phase in the RE-EDS workflow currently uses a relatively large amount of simulation time. Making these steps more efficient will be addressed in future work.

The Python code for the RE-EDS workflow is provided on

Github
https://github.com/rinikerlab/reeds and can be used with the current version of GROMOS, freely available from http://www.gromos.net.

Appendix 3.A PARAMETER EXPLORATION

A fast transition of the initial dominating end state to the desired dominating end state was observed by monitoring the dominating end state over time. The transition occurred latest after 0.5 ns, and the system remained in the biased end state for the rest of the simulation time. In both water and complex simulations, the desired end state was sampled about 99% of the simulation time with the exception of L19 in water (Table 3.A.3). To inspect if the optimized state simulations' results sufficiently represent the target states, a comparison between the target state obtained potential energy distributions in the eds simulations with MD simulations consisting of only the target state was conducted (Figure 3.A.9).

TABLE 3.A.3: Fraction of the simulation time (in %) that the desired end state was sampled as the dominating state during the EDS simulation to optimize the coordinates for a desired end state.

Ligand	Water	Complex
L1	99.84	99.97
L17	99.99	99.97
L19	36.07	99.98
L20	99.99	100
L21	100	99.97

TABLE 3.A.4: Potential thresholds for occurrence sampling (T_i^{phys}) and undersampling (T_i^{us}) determined during the parameter exploration (in kJ mol^{-1}).

Ligand	Water		Complex	
	T^{phys}	T^{us}	T^{phys}	T^{us}
L1	-582.96	-436.05	-737.37	-516.41
L17	-572.41	-419.16	-717.95	-492.83
L19	-579.13	-415.91	-738.95	-483.78
L20	-636.00	-492.75	-759.01	-549.35
L21	-656.22	-488.43	-805.30	-539.78

Appendix 3.B ENERGY OFFSET ESTIMATION

The relative energy offsets $\Delta\Delta E_{ji}^R$ are compared with the experimental relative binding free energies $\Delta\Delta G_{ji}^{\text{bind}}$ in Figure 3.B.10. The root mean squared error (RMSE) between $\Delta\Delta E_{ji}^R$ obtained with RE-EDS 1SS and $\Delta\Delta G_{ji}^{\text{bind}}$ is 12.6 kJ mol^{-1}. Outliers are mainly related to L19. With the RE-EDS SSM approach, the RMSE was reduced to 7.0 kJ mol^{-1}. No clear outliers were observed in this case. Thus, the use of the SSM approach is recommended for RE-EDS simulations.

The energy offsets obtained with the SSM approach are shown as a function of the replica (s-value) in Figure 3.B.11.

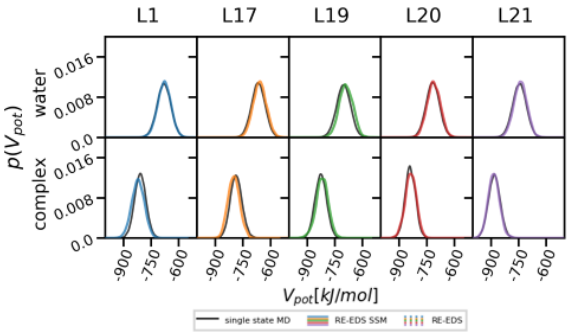


FIGURE 3.A.9: Comparison of the potential-energy distribution obtained from a standard MD simulation of a given end state (black) and from an EDS simulation with the given end state favoured (colored) from the first step of the RE-EDS workflow.

Appendix 3.C OPTIMIZATION OF THE S-DISTRIBUTION

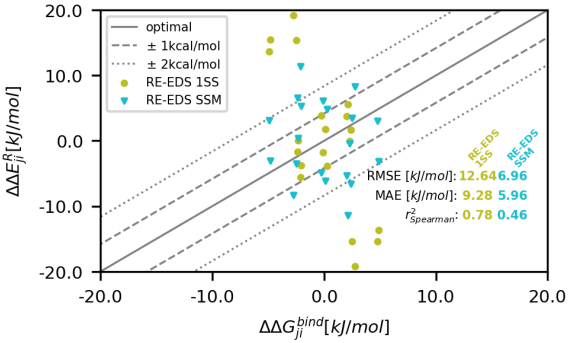


FIGURE 3.B.10: Comparison of the relative energy offsets $\Delta\Delta E^R_{ji}$ in water and complex with the experimental relative binding free energies $\Delta\Delta G^{\text{bind}}_{ji}$. The energy offsets were estimated from RE-EDS simulations using the 1SS (green) or SSM (blue) approach to select the starting configurations of the replicas.

Appendix 3.D FREE-ENERGY CALCULATION

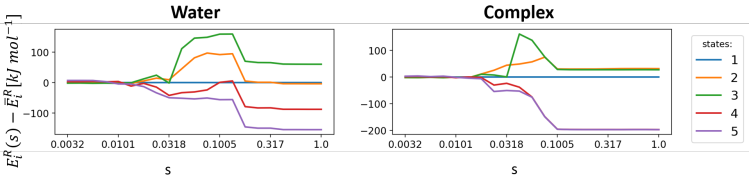


FIGURE 3.B.11: Energy offsets (relative to the average energy offset $\overline{E}_i^R$) estimated from the simulation with the SSM approach in water (left) and in complex (right) as a function of the replica (s-value).

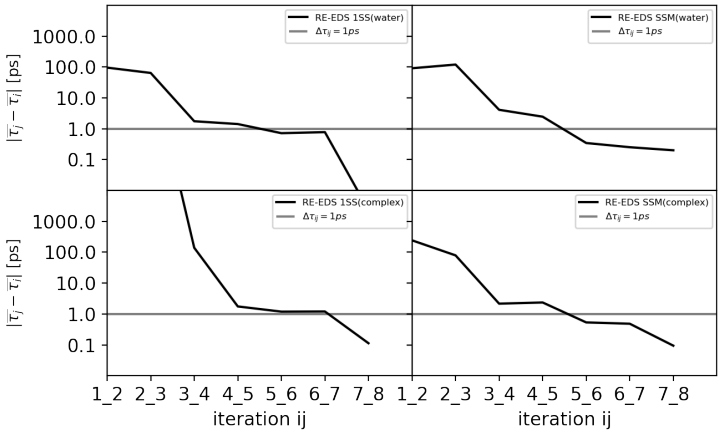


FIGURE 3.C.12: Improvement of the average round-trip time between iterations i and j ($\overline{\tau}_j - \overline{\tau}_i$) on a logarithmic scale. An s-distribution was considered converged when ($\overline{\tau}_j - \overline{\tau}_i$) reached 1 ps.

TABLE 3.D.5: Free-energy differences in water and in complex calculated from the production run of 4 ns of length with the RE-EDS 1SS and RE-EDS SSM approaches.

Ligand		RE-EDS 1SS		RE-EDS SS	
I	J	water [kJ mol^{-1}]	complex [kJ mol^{-1}]	water [kJ mol^{-1}]	comp
L17	L1	15.22 ± 0.15	16.44 ± 0.22	-13.61 ± 0.11	1
L19	L1	6.73 ± 0.13	-7.24 ± 0.16	-1.95 ± 0.18	-
L20	L1	-48.59 ± 0.22	-45.99 ± 0.21	-48.09 ± 0.13	-4
L21	L1	-67.83 ± 0.326	-69.48 ± 0.20	-69.48 ± 0.16	-7
L19	L17	-8.49 ± 0.12	-23.68 ± 0.25	-15.56 ± 0.18	-2
L20	L17	-63.81 ± 0.22	-62.42 ± 0.19	-61.69 ± 0.14	6
L21	L17	-83.05 ± 0.31	-85.92 ± 0.29	-83.09 ± 0.17	-8
L20	L19	-55.32 ± 0.21	38.75 ± 0.33	-46.13 ± 0.20	-4
L21	L19	-74.56 ± 0.31	-62.24 ± 0.29	-67.52 ± 0.21	-6
L21	L20	-19.24 ± 0.35	-23.50 ± 0.31	-21.39 ± 0.18	-2

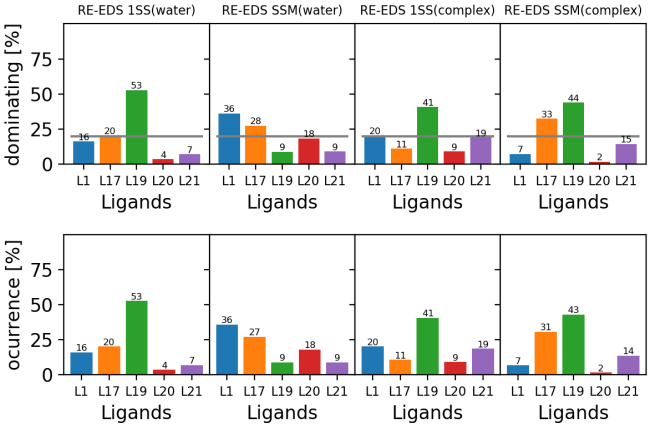


FIGURE 3.D.13: Sampling of the end states in the final production run at replica $s = 1.0$. Sampling was assessed by monitoring the dominant end state (top panels) and by counting all end states a potential energy below T_i^{phys} (see Table 3.A.4) (bottom panels). Ideally, the sampling fraction as dominating end state should be $1/N$ (Eq. (8) in the main text) for all end states, indicated as a black horizontal line.

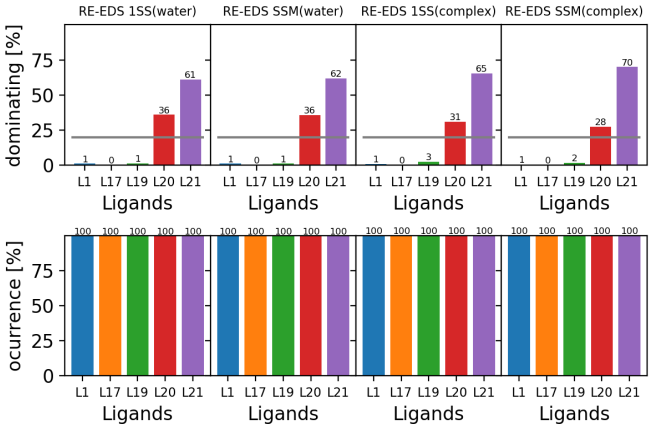


FIGURE 3.D.14: Sampling of the end states in the final production run at the undersampling replica position $s = 0.0032$. Sampling was assessed by monitoring the dominant end state (top panels) and by counting all end states a potential energy below T_i^{phys} (see Table 3.A.4) (bottom panels). Ideally, the sampling fraction as dominating end state should be $1/N$ (Eq. (8) in the main text) for all end states, indicated as a black horizontal line.

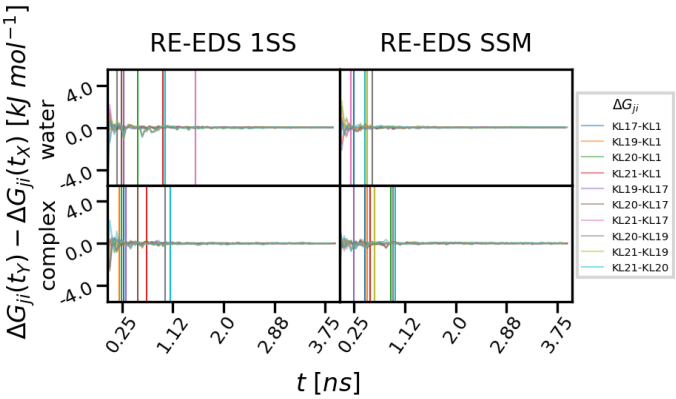


FIGURE 3.D.15: Convergence analysis of the RE-EDS production runs (total 4 ns): Free-energy differences relative to the final free-energy results after 4 ns are plotted as a function of the simulation time. The vertical lines indicate when a particular ΔG_{ji} value was found to be converged (deviation below 0.25 kJ mol^{-1}).

4 Dual Topology and the Challenge how to pick the restraints - Restraintmaker

"Let us learn to dream, gentlemen, and then perhaps we shall learn the truth."

August Kekulé, 1865

dual top fun

4.1 INTRODUCTION

Relative Alchemical Free Energy (RAFE) Calculations on an atomistic level are on the verge of becoming a common tool in virtual high-throughput screening (vHTS).[59] A lot of investment in the past and current approaches can be observed into the improvement of method robustness and usability by adding criteria, that support judging the outcome of RAFE estimations and automatization of the approaches

The challenges to RAFE calculations are increasing by adding more and more complex transitions from one to another state, leading to the so called scaffold hopping.

In practice there can be three topology types differentiated: single topology, hybrid topology, and dual topology.

In the following we will describe the topology types with an alchemical transition from one ligand to another one (state A and state B).

The single topology approach contains exactly the same set of atom coordinates for both states and switches them from one atom type to another. Differences in the molecular graphs can be realized with using dummy atom types. soft-bonds - FEP+[77? ?]

"Two main approaches exist for calculating ΔΔGsite, the free energy of transforming A into B in the binding site: the single topology approach and the dual topologies approach. In the single topology approach, there is one ligand molecule in all simulations, which morphs from being in state A (having a particular chemical structure and interaction parameters), to a different state B, where the structure and interaction parameters are different.4–13 In some transformations, there could be atoms on A that have no direct analog on B."[?]

The hybrid approach diverges from the single topology approach by adding atoms for the substituent if they are not the same ....

"In the dual topologies approach, the computational morphing process carries along two ligand molecules at the same time throughout all simulations.14–16 Using this method, the simulations begin with both a "real" ligand A and a "dummy" ligand B, and over the course of the alchemical transformation, A becomes the dummy ligand while B becomes the real ligand. "?

The Dual-Topology approach is adding two independent sets of atom-coordinates for molecules differing in the states. This approach allows the largest possible . There are different ways how the state molecules are kept in a similar coordinate space, in order to guarantee efficient sampling. One way is to define harmonic

For this approach, there are sub variants for how the QligFEP[67]

4.2 THEORY

The target of the here defined algorithm is, to find a good placement for distance restraints between two molecules m_i and m_j for a Dual topology approach.

The following assumptions are made towards this process: , therefore only rigid areas of the moelcules can be selected like atom-rings.

The goal is reached greedily

In order to reach the target of the algorithm, a set of operations befor the algorithm are required in an applied case.

4.3 COMPUTATIONAL DETAILS

here is fun!

4.4 RESULTS AND DISCUSSION

Pairwise

We could calculate the free energy with for the M030- centered graph. the free energy is good!

Table 4.1: Pairwise Hydration Free Energies with one base molecule

Ligands i	Ligands j	Experiment[76] [kJ mol^{-1}]	ATB [kJ mol^{-1}]	TI [kJ mol^{-1}]
_O6T	M030	18.53	26.22±	18.56±
_O70	M030	8.28	14.43	8.28
_O71	M030	16.60	22.38	16.60
_P8I	M030	9.33	15.34	9.33
6J29	M030	47.77	48.23	47.77
6KET	M030	30.55	36.09	30.55
8018	M030	10.37	22.83	10.37
E1VB	M030	-1.73	5.20	-1.73
F313	M030	25.61	30.18	25.61
G078	M030	5.92	6.45	5.92
G277	M030	14.75	16.98	14.75
M030	M097	-16.58	-18.60	-16.58
M030	M218	-20.46	-24.90	-20.46
M030	S002	-10.43	-14.90	-10.43
M030	TVVS	-23.69	-26.10	-25.69
RMSE			6.71	4.43
MAE			5.46	3.34
r^2_{Pearson}			0.97	0.98
r^2_{Spearman}			0.94	0.96

Multistate

Some information

6Ligands

Table 4.2: Multistate Hydration Free Energies for the first set containing six ligands

Ligands i	Ligands j	Experiment [76] [kJ mol^{-1}]	ATB [kJ mol^{-1}]	TI [kJ mol^{-1}]	RE-EDS [kJ mol^{-1}]
_O6T	6KET	-9.75	-9.78	-11.27	-10.65
_O6T	F313	7.90	-3.39	-7.09	-6.40
_O6T	G277	-1.90	9.24	3.62	6.37
_O6T	M030	18.53	26.22	18.56	18.27
_O6T	M097	1.76	7.62	2.21	2.83
6KET	F313	0.75	5.91	4.72	4.25
6KET	G277	7.85	19.11	14.98	17.02
6KET	M030	28.28	36.09	30.55	28.93
6KET	M097	11.51	17.49	12.98	13.49
F313	G277	7.10	13.20	10.94	12.77
F313	M030	27.53	30.18	25.61	24.68
F313	M097	10.76	11.58	9.17	9.24
G277	M030	20.43	16.98	14.75	11.90
G277	M097	3.66	-1.62	-1.69	-3.54
M030	M097	-16.77	-18.60	-16.58	-15.44
RMSE			6.75	5.25	6.03
MAE			5.76	3.73	4.48
r^2_{Pearson}			0.91	0.91	0.88
r^2_{Spearman}			0.84	0.84	0.81

4.5 CONCLUSION

nice work!

5 Machine learning Methods with Free Energy Calculations

"Let us learn to dream, gentlemen, and then perhaps we shall learn the truth."

August Kekulé, 1865

Let's see if I have something here

5.1 INTRODUCTION

5.2 THEORY

5.3 COMPUTATIONAL DETAILS

5.4 RESULTS AND DISCUSSION

5.5 CONCLUSION

6

Modulation of the Passive Permeability of Semipeptidic Macrocycles: N- and C-Methylations Fine-Tune Conformation and Properties

"Let us learn to dream, gentlemen, and then perhaps we shall learn the truth."

August Kekulé, 1865

Incorporating small modifications to peptidic macrocycles can have a major influence on their properties. For instance, N-methylation has been shown to impact permeability. A better understanding of the relationship between permeability and structure is of key importance as peptidic drugs are often associated with unfavorable pharmacokinetic profiles. Starting from a semipeptidic macrocycle backbone composed of a tripeptide tethered head-to-tail with an alkyl linker, we investigated two small changes: peptide-to-peptoid

substitution and various methyl placements on the nonpeptidic linker. Implementing these changes in parallel, we created a collection of 36 compounds. Their permeability was then assessed in parallel artificial membrane permeability assay (PAMPA) and Caco-2 assays. Our results show a systematic improvement in permeability associated with one peptoid position in the cycle, while the influence of methyl substitution varies on a case-by-case basis. Using a combination of molecular dynamics simulations and NMR measurements, we offer hypotheses to explain such behavior.

6.1 INTRODUCTION

Macrocycles have recently gathered increasing levels of interest in medicinal chemistry.(1−6) Their unique combination of conformationally constrained structure and high level of structural information allows for the design of large, organized structures suitable to interact with extended and featureless binding sites such as those found in protein–protein interactions.(7−10) Most Food and Drug Administration (FDA)-approved macrocyclic drugs belong to natural products (e.g., erythromycin, tacrolimus) or peptides (e.g., sandostatin, eptifibatin).(11) Peptidic or semipeptidic scaffolds bridge the gap between small molecules and biologics, allying synthetic ease and broad choice of natural and non-natural amino acids required for rapid and thorough pharmacophoric exploration. The main challenge with peptides resides in their physicochemical and pharmacokinetics-absorption, distribution, metabolism, and excretion (PK-ADME) properties. While cyclic peptides are typically more stable to proteases compared to their linear counterparts, their high polarity often translates into low bioavailability.(12,13) Nonetheless, some cyclic peptides cross cell membranes.(12,14,15) Developing tools and knowledge to optimize and better predict their structure–permeability relationship is therefore a requirement for the field. Such quest found inspiration in studies of the natural cycloundecapeptide cyclosporine A, which is administered orally. One prominent structural feature of this natural macrocycle is its high number of N-methylated residues (7 out of 11) and its dynamic structural adaptation to its environment (also known as (aka) chameleonic properties).(16−18) The effect of N-methylation on permeability of cyclic hexa- and heptapeptides has been systematically investigated since the number

and position of N-methylations may be beneficial or detrimental for permeability.(15,19−23) Less explored are the N-alkylated glycines—aka peptoids—in which side chain has been moved from the carbon to the amide nitrogen.(24) Similarly to N-methylation, this modification removes one H-bond donor, yet it also removes one stereogenic center and induces glycine-like secondary structures. The peptoid amide also facilitates cis–trans isomerization compared to the corresponding N-methylation.(25) Synthetically, the inclusion of peptoids is also compatible with solid-phase protocols and allows for an almost unlimited variety of side chains, where virtually any primary amine can be used.(26) More recently, the impact of the dynamics of macrocycles in response to their environment, which can range from polar in water, nonhomogeneous in the presence of its target, to lipophilic in the membrane, has been appreciated.(17,18,27,28) A powerful tool to modulate the properties of peptidic macrocycles is the inclusion of a nonpeptidic tether unit.(29−31) This tether can serve multiple purposes: in the context of a target interacting with a specific sequence, various tethers can be screened without modifying the peptide recognition sequence, while providing a simple handle for modulating affinity and PK properties. Small modifications in size, shape, or functional groups on the tether can dramatically influence on this kind of constrained system.(32) Additionally, a tether may facilitate macrocyclization, which can be challenging synthetically.(6) The relationship between structure and permeability is known to be elusive for this class of compounds, with small structural modifications often yielding permeability cliffs.(14,19,21,23,31,33−36) To support our efforts in this direction(31) and pinpoint the effects of conformational modulation on permeability, we synthesized a library of closely related compounds based on chemotype A composed of a tripeptide tethered head-to-tail with a nonpeptidic

linker (Figure 1). Two classes of modifications were implemented on chemotype A: single peptoid replacement (B, Figure 1) or regio- and stereocontrolled linker C-methylation (C, Figure 1). All of the possible combinations of these variations were generated, providing a total of $4 \times 9 = 36$ compounds with identical molecular weights (except for non-methylated tether derivatives), nearly identical sequence and identical ring sizes, leaving as little room as possible for confounding factors. The passive permeability of the resulting macrocycles was measured in the parallel artificial membrane permeability assay (PAMPA) and their cellular permeability in the Caco-2 assay.(37) We then selected two pairs of diastereomers that differ only by their stereochemistry of the tether methyl group yet either differ greatly in passive permeability or not, and performed molecular dynamics (MD) simulations coupled with solution NMR to rationalize the origin of these differences.(38)

6.2 THEORY

6.3 COMPUTATIONAL DETAILS

6.4 RESULTS AND DISCUSSION

As for the influence of the methyl group, no obvious trend could be observed. Yet, there are some significant differences between pairs of epimers. Also, the sole exception to the increase in permeability with Nleu was found in compound Nleu-5S, while its epimer Nleu-5R is the most permeable of the series. We decided to investigate this central observation by analyzing the associated

conformations in greater detail using a combination of structural data (NMR measurements) and computer-simulated trajectories (molecular dynamics simulations). A "permeability cliff" such as that observed between Nleu-5R and Nleu-5S (Figure 6) is also observed, to a lesser extent, between some other pairs of epimers (e.g., Nala-4R vs 4S with $-\log(\mathrm{Pe}) = 7.63$ and 6.63, Nphe-2R vs 2S with 7.16 and 6.31, and Nphe-3R vs 3S with 7.49 and 6.49). To obtain a better understanding of the underlying conformational changes, extensive MD simulations of the Nleu-5R and Nleu-5S macrocycles (the most distant epimers in terms of permeability) were performed in both polar and apolar environments (i.e., water and chloroform). The starting conformations used for simulations showed similar distributions in terms of hydrogen bonds (H-bonds) and backbone torsional angles (Tables S7 and S8 in the SI). For each molecule, approximately 50Figure 6

Figure 6. Selected cyclic peptides studied with experimental NMR analysis and molecular dynamics (MD) simulations. The cumulative 25 s simulation data for each peptide and solvent were clustered separately based on the backbone dihedrals and the polar atom distances. The resulting clusters could be structurally classified depending on the conformation of the peptoid bond (i.e., cis or trans; see Tables S9 and S10 in the SI). The cis–trans isomerization represents a very slow process in the simulations, which occurred only rarely (Table S11 in the SI). Due to the low number of transitions, the process could not be modeled robustly. Therefore, the clusters with the cis- and trans-peptoid bond are analyzed separately in the following. The NMR experiments in chloroform-d showed that the four compounds adopt at least two different conformations in solution. The major conformer was identified with all amides in trans conformation (Table S1 in the SI). It was not possible to assign the minor conformers due to signal over-

lap and low intensity. In the case of Nleu-5R and Nleu-5S, a third conformer could be identified based on exchange spectroscopy (EXSY) cross-peaks in the nuclear Overhauser enhancement spectroscopy (NOESY) spectrum, which is barely detectable in the 1H spectrum. The corresponding conformer ratios are listed in Table 1. The results from the MD simulations are compared to the NMR data of the major conformer (i.e., 3JHN–H coupling constants and nuclear Overhauser effect (NOE)-derived distances, given in Tables S2–S6 in the SI) to validate the simulation results. Table 1. Ratios of Conformer Population Observed in NMR Spectra (CDCl3) compound ratio Nleu-2R 100:8 Nleu-2S 100:3 Nleu-5R 100:4:0 Nleu-5S 100:16:1 The clusters with all amides in trans conformation are in good agreement with the 3JHN–H coupling constants (Figure 7), whereas the clusters containing the cis-peptoid bond deviate significantly from the experimental values. For Nleu-2R, the 3JHN–H coupling analysis is missing as we could not determine the 3JHN–H couplings reliably due to line broadening in the spectrum. The NOE upper distance bounds are also generally reproduced in these clusters (Figures S5–S9 in the SI). Based on these findings, we focus the analysis in the following on those clusters, which have a reasonable agreement with the NMR data (i.e., clusters 1 and 4 for Nleu-5R, clusters 1, 5, and 6 for Nleu-5S, cluster 1 for Nleu-2R, and clusters 1 and 2 for Nleu-2S). Figure 7

Figure 7. Root-mean-square deviation (RMSD, in hertz) between 3JHN–H coupling constants in chloroform from NMR measurements and from MD simulations. Clusters with the peptoid bond in trans conformation are shown in green. A necessary condition for good membrane permeability is the adoption of conformations that shield polar groups optimally from the apolar environment.(45−47) Therefore, we first analyzed the hydrogen-

bonding patterns in the clusters in chloroform. For the peptides in this study, a maximum number of two H-bonds can be formed in a conformation due to ring strain. As can be seen in Table 2, the percentage of sampled conformations with two H-bonds differs significantly between Nleu-5R (30Table 2. Percentage of Sampled Conformations with Zero, One, or Two Hydrogen Bonds in Chloroforma number of hydrogen bonds 0 1 2 Nleu-5R (Nleu-5S (Nleu-2R (Nleu-2S (a

Analysis was restricted to the clusters with the trans-peptoid bond. For a given molecule in an apolar environment, having access to conformations in which polar groups are shielded—such as by H-bonding—should be energetically favorable. To assess this effect, we extracted the potential energy of the peptides (i.e., intramolecular and peptide-solvent contributions) from the trajectories. The normality of each potential-energy distribution was confirmed by the Shapiro–Wilk test(48) (Table S12 in the SI). The Fisher t-test(49) was employed to determine if the means of the distributions differ statistically significantly ($p < 0.05$). This was found to be the case for each pair of distributions (Table S13 in the SI). On average, the potential energy of Nleu-5R is 9 kJ/mol lower (i.e., more favorable) in chloroform compared to Nleu-5S, whereas the difference in the average potential energy between Nleu-2R and Nleu-2S is 6 kJ/mol. In many studies in the literature, it was found that the three-dimensional (3D) polar surface area (3D-PSA) is a good measure for the degree of polar shielding in conformations.(31,45,50,51) However, for the present set of four peptides, no correlation was observed between the 3D-PSA and the potential energy (Figure S10 in the SI). The ring strain in the relatively small backbone cycle of the peptides affects the geometry of the intramolecular H-bonds, which is likely not reflected appropriately in the 3D-PSA

calculation. In summary, the ranking Nleu-5R < Nleu-2S < Nleu-2R < Nleu-5S, which was found in terms of both hydrogen-bonding patterns and potential energies, matches well with the experimental permeability data. The findings described above indicate that the change in stereochemistry of the methyl group in position 5 between Nleu-5R and Nleu-5S leads to different conformational behavior. A detailed analysis of the H-bonds showed that only Nleu-5S forms a H-bond between Ala-O and the tether-NH with an occurrence of 24Figure 8

Figure 8. Snapshots of Nleu-5R (A) and Nleu-5S (B) from MD simulations in chloroform. Hydrogen bonds are shown with their percentage of the absolute occurrence in chloroform in the trans-peptoid clusters. Pictures were generated with PyMol.(52) Table 3. Hydrogen Bond Occurrence in Percentage for the Sampled Conformations in Chloroforma H-bond Nleu-2R (Nleu-O tether-NH 74 37 28 33 Ala-O tether-NH <1 <1 <1 24 Phe-O Ala-NH <1 35 57 <1 Ala-O Phe-NH 27 25 36 17 a

Analysis was restricted to the clusters with the trans-peptoid bond. Next, we analyzed the torsional-angle distributions in the backbone ring of the peptides. The change in stereochemistry of the methyl group at position 5 leads to a shift in the torsional-angle distributions of the tether units for Nleu-5S compared to Nleu-5R (Figure 9A). This shift results in a bent conformation of the ring (Figure 9B), which allows only one H-bond to form between Ala-O and tether-NH (Figure 10). There is also a shift in the backbone torsional-angle distributions between Nleu-2R and Nleu-2S, however, to a much smaller extent (Figure S11 in the SI). Figure 9

Figure 9. (A) Torsional-angle distributions of the tether in Nleu-5R (blue) and Nleu-5S (orange) in chloroform. The analysis was restricted to the clusters with the trans-peptoid bond.

(B) Torsional angles of the tether (shown in cyan and orange) corresponding to the peaks of the distributions. Pictures were generated with PyMol.(52) The change in the stereocenter also affects the 1-angle of the phenylalanine residue as the tether conformation hinders the rotation around this torsion due to a steric clash with the carbonyl group that is facing out of the backbone ring (Figure 10). Figure 10

Figure 10. (A) Torsional-angle distributions of the 1 torsional angle of the phenylalanine residue in Nleu-5R (blue) and Nleu-5S (orange) in chloroform. Analysis was restricted to the clusters with the trans-peptoid bond. (B) 1 torsional angle of the phenylalanine residue (shown in purple) corresponding to the peaks of the distributions. The backbone carbonyl interferes with the rotation around this torsion is highlighted with a red circle. Pictures were generated with PyMol.(52) The results for the simulations in water are given in the SI (Tables S10–S15). The analysis of the hydrogen-bonding patterns in water showed that Nleu-5R has a higher percentage (about 10The findings, taken together, suggest that the permeability cliff observed between Nleu-5R and Nleu-5S is related to their propensity for conformations with a maximized number of intramolecular H-bonds in the apolar environment. Their ability to adopt such conformations is in turn affected by the stereochemistry of the methyl group at position 5 in the tether as it determines the preferred torsional angles of the tether.

6.5 CONCLUSION

A total of 42 macrocycles were synthesized and their permeability assessed in the PAMPA and Caco-2 assays. The combination of these data, NMR measurements, and molecular dynamics simulations allows us to draw some conclusions that are hopefully applicable to other systems. The systematic higher permeability of macrocycles bearing an Nleu peptoid is striking and well above statistical significance. Our experiments suggest this effect is due to the removal of this specific H-bond donor, thus working similarly to the more widely used N-methylation strategy. This systematic effect shows that "masking" H-bond donors should be considered early in the design of cyclic peptides. Possibly, it is a matter of finding the right one(s), i.e., those that allow for the most favorable H-bonding patterns in the rest of the macrocycle. The methyl position on the tether had little effect in most cases, with a few notable exceptions. Nala-2S has the lowest passive permeability, while its epimer is average. Conversely, Nleu-5R is the most permeable compound from our initial library, while its counterpart Nleu-5S is the exception among the Nleu compounds for its low permeability. A detailed analysis of torsion angles points once more at intramolecular H-bonds. Nleu-5R and Nleu-5S have different intramolecular H-bonding patterns. It seems likely that the 2 and 5 positions have the highest potential to introduce significant conformational changes due to their proximity to H-bond partners (the tether's carbonyl and nitrogen, respectively). These positions might also have more impact due to the flexible nature of the tether we used, as they are close to the sp2-like amides. It is also noteworthy that a simple inversion of stereochemistry was shown to exert long-distance influence,

modifying the phenylalanine's rotation. Altogether, this study sheds light on the relationship between structure and permeability in this class of compounds. The two seemingly very different substitutions we explored were both found to affect permeability through a change in the intramolecular H-bonding pattern.

7 Outlook

“ tata ”

tutu

7.1 IMPROVEMENTS FOR RE-EDS

Bibliography

[1] Thomas Huber, Andrew E. Torda, and Wilfred F. van Gunsteren. Local elevation: A method for improving the searching properties of molecular dynamics simulation. *J. Comput. Aided Mol. Des.*, 8:695–708, 1994.

[2] Alessandro Laio and Michele Parrinello. Escaping free-energy minima. *Proceed. Natl. Acad. Sci. U.S.A.*, 20:12562–12566, 2002.

[3] Clara D. Christ and Wilfred F. van Gunsteren. Enveloping distribution sampling: A method to calculate free energy differences from a single simulation. *J. Chem. Phys.*, 126:184110, 2007.

[4] Gerhard König and Stefan Boresch. Non-boltzmann sampling and bennett's acceptance ratio method: How to profit from bending the rules. *J. Comput. Chem.*, 32:1082–1090, 2012.

[5] Gerhard König, Nina Glaser, Benjamin Schroeder, Philippe Henry Hünenberger, and Sereina Riniker. An alternative to conventional λ-intermediate states in alchemical free energy calculations: λ-enveloping distribution sampling. *J. Chem. Inf. Model.*, 60:5407—5423, 2020.

[6] Serena Donnini, R. Thomas Ullmann, Gerrit Groenhof, and Helmut Grubmüller. Charge-neutral constant ph molecular dynamics simulations using a parsimonious proton buffer. *J. Chem. Theory Comput.*, 12:1040–1051, 2016.

[7] R. Gregor Weiß, Piotr Setny, and Joachim Dzubiella. Solvent fluctuations induce non-markovian kinetics in hydrophobic pocket-ligand binding. *J. Phys. Chem. B*, 120:8127–8136, 2016.

[8] Oliver Lemke and Bettina G. Keller. Common nearest neighbor clustering – a benchmark. *Algorithms*, 11:19, 2018.

[9] Roger D. Peng. Reproducible research in computational science. *Science*, 334:1226–1228, 2011.

[10] Victoria Stodden, Marcia McNutt, David H. Bailey, Ewa Deelman, Yolanda Gil, Brooks Hanson, Michael A. Heroux, John P.A. Ioannidis, and Michela Taufer. Enhancing reproducibility for computational methods. *Science*, 354(6317):1240–1241, 2016.

[11] Scott Chacon and Ben Straub. *Pro git*. Springer Nature, 2014.

[12] Levi N. Naden and Daniel G. A. Smith. Cookiecutter for computational molecular sciences (cms) python packages, 2018.

[13] Project Jupyter, Matthias Bussonnier, Jessica Forde, Jeremy Freeman, Brian Granger, Tim Head, Kyle Kelley, Gladys Nalvarte, Andrew Osheroff, M Pacer, Yuvi Panda, Fernando Perez, Benjamin Ragan-kelley, and Carol Willing. Binder 2.0 – reproducible, interactive, sharable environments for science at scale. *Proc. of the 17th python in Science Conf.*, pages 113–120, 2018.

[14] Guido Van Rossum and Fred L. Drake. Python 3 reference manual, 2009.

[15] Bjarne Stroustrup. *The C++ Programming Language*. Addison-Wesley, 4 edition, 2013.

[16] H J C Berendsen, D van der Spoel, and R van Drunen. Gromacs: A message-passing parallel molecular dynamics implementation. *Comput. Phys. Commun.*, 91:43–56, 1995.

[17] Erik Lindahl, Berk Hess, and David van der Spoel. Gromacs 3.0: A package for molecular simulation and trajectory analysis. *J. Mol. Model.*, 7:306–317, 2001.

[18] David van der Spoel, Erik Lindahl, Berk Hess, Gerrit Groenhof, Alan E. Mark, and Herman J.C. Berendsen. Gromacs: Fast, flexible, and free. *J. Comput. Chem.*, 26:1701–1718, 2005.

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

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

[21] J. E. Jones. On the determination of molecular fields. i. from the variation of the viscosity of a gas with temperature. *Proc. Royal Soc. London*, 106(Ser. A):441–462, 1924.

[22] Magnus R Hestenes and Eduard Stiefel. Methods of conjugate gradients for solving linear systems. *J. Res. Natl. Bur. Stand.*, 49:409–436, 1952.

[23] W. K. Hastings. Monte carlo sampling methods using markov chains and their applications. *Biometrika*, 57:97–109, 1970.

[24] W. F. van Gunsteren and H. J.C. Berendsen. A leap-frog algorithm for stochastic dynamics. *Mol. Sim.*, 1:173–185, 1988.

[25] Axel Brünger, Charles L. Brooks, and Martin Karplus. Stochastic boundary conditions for molecular dynamics simulations of st2 water. *Chem. Phys. Lett.*, 105:495–500, 1984.

[26] G. M Torrie and J. P. Valleau. Nonphysical sampling distributions in monte carlo free-energy estimation: Umbrella sampling. *J. Comput. Phys.*, 23:187–199, 1977.

[27] Yuji Sugita and Yuko Okamoto. Replica-exchange molecular dynamics method for protein folding. *Chem. Phys. Lett.*, 314:141–151, 1999.

[28] Robert W Zwanzig. High-temperature equation of state by a pertubation method. i. nonpolar gases. *J. Chem. Phys.*, 22:1420–1426, 1954.

[29] Charles H Bennett. Efficient estimation of free energy differences from monte carlo data. *J. Comput. Phys.*, 22:245–268, 1976.

[30] John G Kirkwood. Statistical mechanics of fluid mixtures. *J. Chem. Phys.*, 3:300–313, 1935.

[31] Clara D. Christ and Wilfred F. van Gunsteren. Multiple free energies from a single simulation: Extending enveloping distribution sampling to nonoverlapping phase-space distributions. *J. Chem. Phys.*, 128:174112, 2008.

[32] Clara D Christ, Alan E Mark, and Wilfred F van Gunsteren. Basic ingredients of free energy calculations: A review. *J. Comput. Chem.*, 31:1569–1582, 2009.

[33] Dominik Sidler, Arthur Schwaninger, and Sereina Riniker. Replica exchange enveloping distribution sampling (re-eds): A robust method to estimate multiple free-energy differences from a single simulation. *J. Chem. Phys.*, 145:154114, 2016.

[34] David F Hahn and Philippe H Hünenberger. Alchemical free-energy calculations by multiple-replica -dynamics: The conveyor belt thermodynamic integration scheme. *J. Chem. Theory Comput.*, 15:2392–2419, 2019.

[35] Andrew Pohorille, Christopher Jarzynski, and Christophe Chipot. Good practices in free-energy calculations. *J. Phys. Chem. B*, 114:10235–10253, 2010.

[36] Thomas Kluyver, Benjamin Ragan-kelley, Fernando Pérez, Brian Granger, Matthias Bussonnier, Jonathan Frederic, Kyle Kelley, Jessica Hamrick, Jason Grout, Sylvain Corlay, Paul Ivanov, Damián Avila, Safia Abdalla, Carol Willing, and Jupyter Development Team. Jupyter notebooks —- a publishing format for reproducible computational workflows. *ELPUB*, pages 87–90, 2016.

[37] Github. Github, 2020.

[38] Holger Krekel, Bruno Oliveira, Ronny Pfannschmidt, Floris Bruynooghe, Brianna Laugher, Florian Bruhin, and Et Al. Pytest: Helps you write better programs, 2004.

[39] Georg Brandl. Sphinx documentation tool, 2008.

[40] Github. Github actions, 2007.

[41] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R.J. Nelson, Eric Jones, Robert Kern, Eric Larson, C. J. Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, Aditya Vijaykumar, Alessandro Pietro Bardelli, Alex Rothberg, Andreas Hilboll, Andreas Kloeckner, Anthony Scopatz, Antony Lee, Ariel Rokem, C. Nathan Woods, Chad Fulton, Charles Masson, Christian Häggström, Clark Fitzgerald, David A. Nicholson, David R. Hagen, Dmitrii V. Pasechnik, Emanuele Olivetti, Eric Martin, Eric Wieser, Fabrice Silva, Felix Lenders, Florian Wilhelm, G. Young, Gavin A. Price, Gert Ludwig Ingold, Gregory E. Allen, Gregory R. Lee, Hervé Audren, Irvin Probst, Jörg P. Dietrich, Jacob Silterra, James T. Webber, Janko Slavič, Joel Nothman, Johannes Buchner, Johannes Kulick, Johannes L. Schönberger, José Vinícius de Miranda Cardoso, Joscha Reimer, Joseph Harrington, Juan Luis Cano Rodríguez, Juan Nunez-Iglesias, Justin Kuczynski, Kevin Tritz, Martin Thoma,

Matthew Newville, Matthias Kümmerer, Maximilian Bolingbroke, Michael Tartre, Mikhail Pak, Nathaniel J. Smith, Nikolai Nowaczyk, Nikolay Shebanov, Oleksandr Pavlyk, Per A. Brodtkorb, Perry Lee, Robert T. McGibbon, Roman Feldbauer, Sam Lewis, Sam Tygier, Scott Sievert, Sebastiano Vigna, Stefan Peterson, Surhud More, Tadeusz Pudlik, Takuya Oshima, Thomas J. Pingel, Thomas P. Robitaille, Thomas Spura, Thouis R. Jones, Tim Cera, Tim Leslie, Tiziano Zito, Tom Krauss, Utkarsh Upadhyay, Yaroslav O. Halchenko, and Yoshiki Vázquez-Baeza. Scipy 1.0: Fundamental algorithms for scientific computing in python. *Nature Methods*, 17:261–272, 2020.

[42] Stéfan Van Der Walt, S. Chris Colbert, and Gaël Varoquaux. The numpy array: A structure for efficient numerical computation. *Comput. Sci. Eng.*, 13:22–30, 2011.

[43] Aaron Meurer, Christopher P Smith, Mateusz Paprocki, Ond\vrej Čertík, Sergey B Kirpichev, Matthew Rocklin, AMiT Kumar, Sergiu Ivanov, Jason K Moore, Sartaj Singh, Thilina Rathnayake, Sean Vig, Brian E Granger, Richard P Muller, Francesco Bonazzi, Harsh Gupta, Shivam Vats, Fredrik Johansson, Fabian Pedregosa, Matthew J Curry, Andy R Terrel, Štěpán Roučka, Ashutosh Saboo, Isuru Fernando, Sumith Kulal, Robert Cimrman, and Anthony Scopatz. Sympy: Symbolic computing in python. *PeerJ Comput. Sci.*, 3:e103, 2017.

[44] Wes McKinney. Data structures for statistical computing in python. *Proc. of the 9th Python in Science Conf.*, 445:51–56, 2010.

[45] John D. Hunter. Matplotlib: A 2d graphics environment. *Comput. Sci. Eng.*, 9:99–104, 2007.

[46] T. E. Cheatham, J. L. Miller, T. Fox, T. A. Darden, and P. A. Kollman. Molecular dynamics simulations on solvated biomolecular systems: The particle mesh ewald method leads to stable trajectories of dna, rna, and proteins. *J. Am. Chem. Soc.*, 117:4193–4194, 1995.

[47] Andrew R. Leach. *Molecular Modelling – Principles and Applications*. Pearson Education Limited, 2001.

[48] Hans C. Andersen. Molecular dynamics simulations at constant pressure and/or temperature. *J. Chem. Phys.*, 72(4):2384–2393, 1980.

[49] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.*, 12:2825–2830, 2011.

[50] Yuji Sugita, Akio Kitao, and Yuko Okamoto. Multidimensional replica-exchange method for free-energy calculations. *J. Chem. Phys.*, 113:6042–6051, 2000.

[51] Masataka Yamauchi and Hisashi Okumura. Development of isothermal-isobaric replica-permutation method for molecular dynamics and monte carlo simulations and its application to reveal temperature and pressure dependence of folded, misfolded, and unfolded states of chignolin. *J. Chem. Phys.*, 147:184107, 2017.

[52] Niels Hansen and Wilfred F. van Gunsteren. Practical aspects of free-energy calculations: A review. *J. Chem. Theory Comput.*, 10:2632–2647, 2014.

[53] Zoe Cournia, Bryce K. Allen, Thijs Beuming, David A. Pearlman, Brian K. Radak, and Woody Sherman. Rigorous free energy simulations in virtual screening. *J. Chem. Inf. Model.*, 2020.

[54] Kira A. Armacost, Sereina Riniker, and Zoe Cournia. Novel directions in free energy methods and applications. *J. Chem. Inf. Model.*, 60:1–5, 2020.

[55] Gerhard König, Bernard R. Brooks, Walter Thiel, and Darrin M. York. On the convergence of multi-scale free energy simulations. *Mol. Sim.*, 44:1062–1081, 2018.

[56] J. P. Valleau and D. N. Card. Monte carlo estimation of the free energy by multistage sampling. *J. Chem. Phys.*, 57:5457–5462, 1972.

[57] T. P. Straatsma and J. A. McCammon. Multiconfiguration thermodynamic integration. *J. Chem. Phys.*, 95:1175–1188, 1991.

[58] Dominik Sidler, Michael Cristòfol-Clough, and Sereina Riniker. Efficient round-trip time optimization for replica-exchange enveloping distribution sampling (re-eds). *J. Chem. Theory Comput.*, 13:3020–3030, 2017.

[59] Zoe Cournia, Bryce Allen, and Woody Sherman. Relative binding free energy calculations in drug discovery: Recent advances and practical considerations. *J. Chem. Inf. Model.*, 57:2911–2937, 2017.

[60] John D. Chodera and David L. Mobley. Entropy-enthalpy compensation: Role and ramifications in biomolecular ligand recognition and design. *Annual Rev. Biophys.*, 42:121–142, 2013.

[61] Matteo Aldeghi, Alexander Heifetz, Michael J. Bodkin, Stefan Knapp, and Philip C. Biggin. Accurate calculation of the absolute free energy of binding for drug molecules. *Chem. Sci.*, 7:207–218, 2016.

[62] William L. Jorgensen, J. Kathleen Buckner, Stephane Boudon, and Julian Tirado-Rives. Efficient computation of absolute free energies of binding by computer simulations. application to the methane dimer in water. *J. Chem. Phys.*, 89(6):3742–3746, 1988.

[63] Kenneth M. Merz. Carbon dioxide binding to human carbonic anhydrase ii. *J. Am. Chem. Soc.*, 113:406–411, 1991.

[64] S. Liu, Y. Wu, T. Lin, R. Abel, J. P. Redmann, C. M. Summa, V. R. Jaber, N. M. Lim, and D. L. Mobley. Lead optimization mapper: Automating free energy calculations for lead optimization. *J. Comput. Aided Mol. Des.*, 27:755–770, 2013.

[65] L. Wang, Y. Wu, Y. Deng, B. Kim, L. Pierce, G. Krilov, D. Lupyan, S. Robinson, M. K. Dahlgren, J. Greenwood, D. L. Romero, C. Masse, J. L. Knight, T. Steinbrecher, T. Beuming, W. Damm, E. Harder, W. Sherman, M. Brewer, R. Wester, M. Murcko, L. Frye, R. Farid, T. Lin, D. L. Mobley, W. L. Jorgensen, B. J. Berne, R. A. Friesner, and R. Abel. 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, 2015.

[66] Q. Yang, W. Burchett, G. S. Steeno, S. Liu, M. Yang, D. L. Mobley, and X. Hou. Optimal designs for pairwise calculation: An application to free energy perturbation in minimizing prediction variability. *J. Comput. Chem.*, 41:247–257, 2020.

[67] Willem Jespers, Mauricio Esguerra, Johan Åqvist, and Hugo Gutiérrez-De-Terán. Qligfep: An automated workflow for small molecule free energy calculations in q. *J. Cheminf.*, 11:26, 2019.

[68] Clara D. Christ and Wilfred F. van Gunsteren. Multiple free energies from a single simulation: Extending enveloping distribution sampling to nonoverlapping phase-space distributions. *J. Chem. Phys.*, 128:174112, 2008.

[69] Clara D. Christ and Wilfred F. van Gunsteren. Simple, efficient, and reliable computation of multiple free energy differences from a single simulation: A reference hamiltonian parameter update scheme for enveloping distribution sampling (eds). *J. Chem. Theory Comput.*, 5(2):276–286, 2009.

[70] Sereina Riniker, Clara D. Christ, Niels Hansen, Alan E. Mark, Pramod C. Nair, and Wilfred F. van Gunsteren. Comparison of enveloping distribution sampling and thermodynamic integration to calculate binding free energies of phenylethanolamine n-methyltransferase inhibitors. *J. Chem. Phys.*, 135:24105, 2011.

[71] Juyong Lee, Benjamin T Miller, Ana Damjanović, and Bernard R Brooks. Constant ph molecular dynamics in ex-

plicit solvent with enveloping distribution sampling and hamiltonian exchange. *J. Chem. Theory Comput.*, 10:2738–2750, 2014.

[72] Dominik Sidler, Arthur Schwaninger, and Sereina Riniker. Replica exchange enveloping distribution sampling (re-eds): A robust method to estimate multiple free-energy differences from a single simulation. *J. Chem. Phys.*, 145:154114, 2016.

[73] Dominik Sidler, Michael Cristòfol-Clough, and Sereina Riniker. Efficient round-trip time optimization for replica-exchange enveloping distribution sampling (re-eds). *J. Chem. Theory Comput.*, 13:3020–3030, 2017.

[74] Jan Walther Perthold and Chris Oostenbrink. Accelerated enveloping distribution sampling: Enabling sampling of multiple end states while preserving local energy minima. *J. Phys. Chem. B*, 122:5030–5037, 2018.

[75] Jan Walther Perthold, Drazen Petrov, and Chris Oostenbrink. Toward automated free energy calculation with accelerated enveloping distribution sampling (a-eds). *J. Chem. Inf. Model.*, 60:5395–5406, 2020.

[76] Xiaohua Huang, Cliff C Cheng, Thierry O Fischmann, José S Duca, Xianshu Yang, Matthew Richards, and Gerald W Shipps. Discovery of a novel series of chk1 kinase inhibitors with a distinctive hinge binding mode. *ACS Med. Chem. Lett.*, 3:123–128, 2012.

[77] Lingle Wang, Yuqing Deng, Yujie Wu, Byungchan Kim, David N LeBard, Dan Wandschneider, Mike Beachy, Richard A Friesner, and Robert Abel. Accurate modeling of

scaffold hopping transformations in drug discovery. *J. Chem. Theory Comput.*, 13:42–54, 2017.

[78] N. Hansen, J. Dolenc, M. Knecht, S. Riniker, and W. F. van Gunsteren. Assessment of enveloping distribution sampling to calculate relative free enthalpies of binding for eight netropsin–dna duplex complexes in aqueous solution. *J. Comput. Chem.*, 33:640–651, 2012.

[79] Benjamin Ries, Stephanie M. Linker, David F. Hahn, Gerhard König, and Sereina Riniker. Ensembler: A simple package for fast prototyping and teaching molecular simulations. *J. Chem. Inf. Model.*, 61:560–564, 2021.

[80] Ulrich H E Hansmann. Parallel tempering algorithm for conformational studies of biological molecules. *Chem. Phys. Lett.*, 281:140–150, 1997.

[81] Juyong Lee, Benjamin T Miller, Ana Damjanović, and Bernard R Brooks. Enhancing constant-ph simulation in explicit solvent with a two-dimensional replica exchange method. *J. Chem. Theory Comput.*, 11:2560–2574, 2015.

[82] Helmut G Katzgraber, Simon Trebst, David A Huse, and Matthias Troyer. Feedback-optimized parallel tempering monte carlo. *J. Stat. Mech.*, page P03018, 2006.

[83] W. Nadler, J. H. Meinke, and U. H. Hansmann. Folding proteins by first-passage-times-optimized replica. *Exchange. Phys. Rev.*, 8:061905, 2008.

[84] Michael M. H. Graf, Manuela Maurer, and Chris Oostenbrink. Free-energy calculations of residue mutations in a tripeptide using various methods to overcome inefficient sampling. *J. Comp. Chem.*, 37:2597–2605, 2016.

[85] David F. Hahn, Gerhard König, and Philippe H. Hünenberger. Overcoming orthogonal barriers in alchemical free energy calculations: On the relative merits of λ-variations, λ-extrapolations, and biasing. *J. Chem. Theory Comput.*, 16:1630–1645, 2020.

[86] N. Schmid, A. P. Eichenberger, A. Choutko, S. Riniker, M. Winger, A. E. Mark, and W. F. van Gunsteren. Definition and testing of the GROMOS force-field versions: 54A7 and 54B7. *Eur. Biophys. J.*, 40:843–856, 2011.

[87] Alpeshkumar K. Malde, Le Zuo, Matthew Breeze, Martin Stroet, David Poger, Pramod C. Nair, Chris Oostenbrink, and Alan E. Mark. An automated force field topology builder (atb) and repository: Version 1.0. *J. Chem. Theory Comput.*, 7:4026–4037, 2011.

[88] Patrick Bleiziffer, Kay Schaller, and Sereina Riniker. Machine learning of partial charges derived from high-quality quantum-mechanical calculations. *J. Chem. Inf. Model.*, 58:579–590, 2018.

[89] RDKit: Cheminformatics and machine learning software, 2021. Accessed March 2021.

[90] S. Ruder. An overview of gradient descent optimization algorithms. *arXiv preprint*, page arXiv:1609.04747, 2016.

[91] N. Schmid, C. D. Christ, M. Christen, A. P. Eichenberger, and W. F. van Gunsteren. Architecture, implementation and parallelization of the GROMOS software for biomolecular simulation. *Comp. Phys. Comm.*, 183:890–903, 2012.

[92] A. P. Eichenberger, J. R. Allison, J. Dolenc, D. P. Geerke, B. A. C. Horta, K. Meier, C. Oostenbrink, N. Schmid,

D. Steiner, D. Wang, and W. F. van Gunsteren. The GROMOS++ software for the analysis of biomolecular simulation trajectories. *J. Chem. Theory Comput.*, 7:3379–3390, 2011.

[93] H. J. C. Berendsen, J. P. M. Postma, W. F. van Gunsteren, and J. Hermans. *Interaction Models for Water in Relation to Protein Hydration*, pages 331–342. Reidel, Dordrecht, The Netherlands, 1981.

[94] R. W. Hockney. The potential calculation and some applications. *Methods Comput. Phys.*, pages 136–210, 1970.

[95] J.-P. Ryckaert, G. Ciccotti, and H. J. C. Berendsen. Numerical integration of the cartesian equations of motion of a system with constraints: Molecular dynamics of n-alkanes. *J. Comput. Phys.*, 23:327–341, 1977.

[96] I. Tironi, R. Sperb, P. E. Smith, and W. F. van Gunsteren. A generalized reaction field method for molecular dynamics simulations. *J. Chem. Phys.*, 102:5451–5459, 1995.

[97] Alice Glättli, Xavier Daura, and Wilfred F. van Gunsteren. Derivation of an improved simple point charge model for liquid water: Spc/a and spc/l. *J. Chem. Phys.*, 116:9811–9828, 2002.

[98] H. J. C. Berendsen, J. P. M. Postma, W. F. van Gunsteren, A. DiNola, and J. R. Haak. Molecular dynamics with coupling to an external bath. *J. Chem. Phys.*, 81:3684–3690, 1984.

[99] Benjamin Ries and Marc Thierry Lehner. rinikerlab/pygromostools: Pygromostools_v1, Mar 2021.

Curriculum Vitæ

Benjamin Joachim Ries

15.03.1991
Ettlingen, Germany
German citizen

Education

2017 – 2021	PhD, ETH Zürich
2012 – 2014	Bioinformatics MSc., Universität Tübingen
2009 – 2013	Biochemistry BSc., Universität Tübingen
2002 – 2011	secondary school examinations (Abitur), Albertus Magnus Gymnasium, Ettlingen, Germany

Experience

2019	Summerschool, Universita della Svizzera Italiana, Switzerland: Effective High-Performance Computing & Data Analytics with GPUs
2017	Summerschool, University of Jyväskylä, Finland: Measuring and Modelling Proton Equilibria in Complex Macromolecular Systems
2016 – 2017	Erasmus+ internship, Uppsala Universitet, Sweden
2013 – 2014	Lab Assistant, MPI, Developmental Biology, Tübingen

Teaching Assistant

2018	Physical Chemistry I: Thermodynamics (spring semester), F. Merkt
2018, 2019	Algorithms and Programming in C++ (autumn semester), S. Riniker
2019, 2020	Physical Chemistry Practicum for Biology and Pharmacy Students: Molecular Dynamics (spring semester), E. Meister
2021	Statistical Physics and Computer Simulation for CSE (spring semester), P. Hünenberger and S. Riniker

Commitment

2019-2021	young swiss chemical society (youngSCS), ETH Representative (2019-2020), President(2020-2021)
2012-2017	juniorGBM Tübingen, president(2016)