Thesis for The Degree of Doctor of Philosophy

Deep generative models for molecular dynamics and design

Toward the acceleration of molecular property prediction and discovery with deep learning

Juan Viguera Diez

Department of Computer Science and Engineering
Chalmers University of Technology | University of Gothenburg
Gothenburg, Sweden, 2026

Deep generative models for molecular dynamics and design
Toward the acceleration of molecular property prediction and discovery with deep learning

Juan Viguera Diez

ISBN 978-91-8103-361-8
Doktorsavhandlingar vid Chalmers tekniska högskola, Ny serie nr 5818.
ISSN 0346-718X

Department of Computer Science and Engineering
Division of Data Science and AI
AI and Machine Learning in the Natural Sciences
Chalmers University of Technology | University of Gothenburg
SE-412 96 Göteborg,
Sweden
Phone: +46(0)31 772 1000

Printed by Chalmers Digitaltryck,
Gothenburg, Sweden 2026.

A mi familia

Deep generative models for molecular dynamics and design

Toward the acceleration of molecular property prediction and discovery with deep learning

Juan Viguera Diez

Department of Computer Science and Engineering
Chalmers University of Technology | University of Gothenburg

Abstract

This thesis explores how Deep Generative Models (DGMs) can accelerate molecular modeling tasks central to drug discovery by addressing conditional sampling problems. It consists of four studies, the three first focusing on molecular dynamics (MD), and the last on molecular design. The first paper introduces Surrogate Model-Assisted Molecular Dynamics (SMA-MD), which combines a DGM with statistical reweighting and short MD simulations to efficiently sample Boltzmann ensembles of small molecules, producing more diverse and lower-energy configurations than conventional simulations. The second paper presents Transferable Implicit Transfer Operators (TITO), a transferable generative surrogate that learns time-integrated molecular dynamics directly from data, enabling propagation at arbitrarily large time steps with up to four orders of magnitude acceleration while maintaining thermodynamic and kinetic fidelity. The third paper, Boltzmann Priors for Implicit Transfer Operator learning (BoPITO), introduces equilibrium-aware priors to surrogate models of MD, improving data efficiency and long-term dynamical accuracy. Finally, the fourth paper develops a reinforcement learning scheme to fine-tune graph-based DGMs for *de novo* molecular design, guiding models toward molecules with desired properties even when such examples are rare or absent in the training data. These contributions constitute important stepping stones towards the automation of the drug discovery process.

Keywords

Cheminformatics, machine learning, drug discovery, generative modeling, conformational sampling, Boltzmann generators, transfer operators, AI4Science

List of Publications

Appended publications

This thesis is based on the following publications:

[**Paper I**] **J. Viguera Diez**, S. Romeo Atance, O. Engkvist, S. Olsson,
Generation of conformational ensembles of small molecules via Surrogate Model-Assisted Molecular Dynamics
2024 Mach. Learn.: Sci. Technol. 5 025010.
www.doi.org/10.1088/2632-2153/ad3b64

[**Paper II**] **J. Viguera Diez**, M. Schreiner, S. Olsson,
Transferable Generative Models Bridge Femtosecond to Nanosecond Time-Step Molecular Dynamics
Submitted, under revision.
www.doi.org/10.48550/arXiv.2510.07589

[**Paper III**] **J. Viguera Diez**, M. Schreiner, O. Engkvist, S. Olsson,
Boltzmann priors for Implicit Transfer Operators
International Conference of Learning Representations 2025.
www.doi.org/10.48550/arXiv.2410.10605

[**Paper IV**] S. Romeo Atance*, **J. Viguera Diez***, O. Engkvist, S. Olsson, R. Mercado,
De Novo Drug Design Using Reinforcement Learning with Graph-Based Deep Generative Models
J. Chem. Inf. Model. 2022, 62, 20, 4863–4872.
www.doi.org/10.1021/acs.jcim.2c00838
* indicates equal contribution.

Other publications

The following publications were published during my PhD studies, or are currently in submission/under revision. However, they are not appended to this thesis, due to contents overlapping that of appended publications or contents not related to the thesis.

[**a**] **J. Viguera Diez**, S. Romeo Atance, O. Engkvist, R. Mercado, S. Olsson, *A transferable Boltzmann generator for small-molecules conformers ELLIS Machine Learning for Molecule Discovery Workshop (November 2021), 7.*

Summary of contributions

The contributions to the appended papers by the author of the thesis are listed below:

[**Paper I**] Co-designed the dataset and study, implemented and generated the dataset, implemented the proposed method, performed the experiments, analyzed the results, managed submission, contributed to writing the manuscript and responding to reviewer feedback.

[**Paper II**] Co-designed study, contributed to implementing the proposed method, performed the experiments, analyzed the results, contributed to writing the manuscript and responding to reviewer feedback.

[**Paper III**] Co-designed study, implemented the proposed method, performed the experiments, analyzed the results, managed submission, contributed to writing the manuscript and responding to reviewer feedback, and presented results at the corresponding conference.

[**Paper IV**] The two first authors contributed equally to this work. Contributed to designing the study, performing the experiments, analyzing the results, writing the manuscript, responding to reviewer feedback and managed submission.

Acknowledgment

First and foremost, I would like to thank my academic supervisor and mentor, Simon Olsson. Thank you for always putting your PhD students first, for your invaluable guidance, flexibility and for always supporting and trusting me. You are an inspiration and an example for me to follow. Secondly, I would like to thank my industrial advisor, Ola Engkvist. Thank you for always having time for me, for your support and advice. Thank you for giving that talk at Chalmers that sparked my interest in molecular machine learning. I thank both for creating exceptional environments for doing research, both at Chalmers and at AstraZeneca, where people like me can thrive.

I am grateful to my examiner Devdatt Dubhashi and co-supervisor Morteza Haghir Chehreghani, for their support and valuable feedback. I would like to thank Tristan Bereau for accepting to be the opponent for my PhD defense, and Antonia Mey, Erik Lindahl and Roberto Covino for accepting to be in my evaluation committee. I am grateful to Søren Hauberg for being a great discussion leader during my Licentiate defense. I thank Andrew Ferguson for his feedback and support on an early draft of paper II in this thesis.

I am thankful to all the people I have shared experiences with in this beautiful journey. First and foremost, I am grateful to my best friend and partner, Sara, who has not only emotionally supported me along the way but has also made key contributions to the research in this thesis. I could not be more grateful to share my life with you and I really enjoyed exploring scientific problems together. I thank all my colleagues, both at the Data Science and AI (DSAI) division at Chalmers and the Molecular AI team at AstraZeneca. I have learned so much from you and your experiences and I am really happy to have shared my PhD journey with you. I especially thank Rocío Mercado for her scientific contributions, invaluable support, advice, feedback and expertise during her time at AstraZeneca and now at Chalmers. Thanks everyone in the AIMLeNs and the AIME groups. I really enjoyed our lunches, group meetings and random science discussions. I especially thank Christopher Kolloff. I am really grateful for having shared a million experiences together during this time and I am very happy to call you a friend. Thanks Beatrice, Flemming, Mathias, Ross, Selma and Télio for making every day better. I am also grateful to the DSAI administrative staff, Andrea Brodén, Clara Oders, and Wolfgang Ahrendt, for keeping everything running smoothly. I thank Claes Andersson for giving me the freedom I needed to develop my teaching skills.

At Molecular AI, I would like to especially thank Jon Paul Janet and Alessandro Tibo for their mentorship, guidance and exciting scientific discussions. I really thank Samuel Genheden, Marco Klähn, Thomas Löhr and Annie Westerlund for their advice, for sharing their experiences with me and for being great colleagues and making everyday life much more fun. Thank you to the fantastic group of Molecular AI students I have shared my time with, I appreciated every moment shared with you.

I am deeply grateful to Grant Rotskoff, for hosting me during my reserch stay at Stanford and for his financial support. I thank everyone in the group for being so welcoming and supportive. I really enjoyed working and adventures with Frank and Sebastian. And I am really thankful to Nick, Emmit, Steven, and Jérémie for being great colleagues and helping me feel like home.

Life would just not be the same without friends. I feel immensely grateful to all of mine, you are my treasure. I thank Adrià, Alex, Berta, Elena, Emma, Enric, Felix, Ignacio, Ida, Isa, Javier, Kruthika, Marie, Marisi, Marvin, Mikaela, Mithuran, Moksha, Nikhil, Rahul, Robert, Sandra, Sara, Simone, Sofía, Vivek and Yashi. Thank you for every trip, hike, meal and experience together. Gothenburg feels like home because of you. I am really grateful to all my friends in the distance. Adrián, Ángel, David, Elena, Irene, Jonathan, Miguel, Pablo, Paula and Pedro: I miss you and enjoy every second we get to spend together. I thank Kamran Hussain for being a true friend when I needed it the most, even if we had known each other for such a short time.

Last, but certainly not least, I am forever thankful to all my family, especially, my parents Marta and Luis and my brother Rubén for their unconditional love and support. Thank you for always being there and believing in me. Thank you for all the effort you have dedicated to my education and for the freedom you have given me to be the person I want to be. Nothing of this would have been possible without you. ¡Gracias!

I am really looking forward to my next role at AstraZeneca. I am deeply grateful to everyone who gave me their confidence and made this possible.

This work was partially supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP), funded by the Knut and Alice Wallenberg Foundation. I am very grateful for the funding (including grants for international research visits) and the educational and networking opportunities provided by this exceptional program. Thanks to the Barbro Osher Endowment Foundation for supporting my research exchange. Computational resources were provided by an allocation on the Berzelius resource provided by the Knut and Alice Wallenberg Foundation at the National Supercomputer Centre hosted by the National Academic Infrastructure for Supercomputing in Sweden (NAISS), partially funded by the Swedish Research Council through grant agreement no. 2022-06725.

Contents

Part I

Introductory chapters

Chapter 1

Introduction

Drug discovery aims to identify and develop compounds capable of modulating biological targets, such as proteins or genes, for therapeutic benefit. This process is inherently complex, expensive, and time-consuming, demanding close collaboration across multiple scientific disciplines. Current estimates suggest that bringing a new drug to market requires, on average, more than a decade of research and an investment of approximately 2.6 billion dollars [2]. Consequently, developing methods that can accelerate and simplify drug discovery is of critical importance.

Machine Learning (ML) approaches offer promising tools to enhance and accelerate many stages of the drug discovery pipeline. By leveraging large-scale biological and genomic datasets, ML models can assist in the identification and validation of novel therapeutic targets [3], [4]. They can also support the design and optimization of drug candidates through generative modeling and molecular simulations [5], [6], [7], as well as improve the prediction of molecular properties and safety profiles [8], [9], [10]. Furthermore, advances in natural language processing and computer vision have enabled new applications in clinical development and experimental analysis [11], [12], [13].

Conditional sampling problems are ubiquitous across scientific disciplines, with applications in drug discovery [14], [15], materials science [16], [17], and machine learning [18], [19]. Given a condition C, the objective is to generate samples $\mathbf{x}$ from a conditional probability distribution $p(\mathbf{x} \mid C)$, which is often accessible only up to an normalizing constant or through computationally expensive numerical evaluations. Classical approaches, including simulation-based methods [20], [21], Markov Chain Monte Carlo algorithms [22], [23], [24], and evolutionary strategies [25], often perform poorly when applied to high-dimensional, metastable, or topologically complex distributions such as those encountered in molecular systems. Deep Generative Models (DGMs) provide a powerful alternative, learning flexible parameterizations of complex probability distributions directly from data. Unlike traditional sampling techniques, DGMs are topologically flexible and scale effectively to high-dimensional spaces, making them particularly well suited for modeling molecules.

This thesis explores four solutions based on DGMs to sample from chal-

lenging molecular distributions. First, we explore molecular dynamics where $\mathbf{x}$ represents the three-dimensional atomic coordinates of a system, and the condition C encodes the molecular identity along with temporal information. Second, we address molecular design, in which $\mathbf{x}$ denotes a molecular graph and C specifies a desired property profile or target set of physicochemical characteristics.

The first paper [26] introduces a new method for generating equilibrium ensembles of molecules that combines a DGM with statistical reweighting and short Molecular Dynamics (MD) simulations. The method, called Surrogate Model-Assisted Molecular Dynamics (SMA-MD), can produce more diverse and lower energy ensembles than conventional MD simulations, and can also estimate implicit solvation free energies. SMA-MD is demonstrated to be an efficient and transferable approach for sampling from the Boltzmann distribution of small molecular systems.

The second paper [27] introduces Transferable Implicit Transfer Operators (TITO), a DGM that learns the statistical behavior of molecular dynamics directly from simulation data. By modeling transition probability distributions across different molecular systems and timescales, TITO eliminates the need for explicit femtosecond-scale integration and enables the propagation of molecular trajectories at arbitrarily large time steps while maintaining both thermodynamic and kinetic fidelity. Trained on small molecules and peptides, TITO achieves up to four orders of magnitude acceleration in simulation throughput relative to conventional molecular dynamics, while generalizing to larger and chemically distinct systems unseen during training.

The third paper [28] introduces Boltzmann Priors for Implicit Transfer Operator learning (BoPITO), which combines ideas from the two preceding works. BoPITO integrates equilibrium knowledge into deep generative surrogates of molecular dynamics by leveraging pre-trained Boltzmann Generators as priors to guide the learning of transition densities. This approach introduces an inductive bias toward physically consistent long-term dynamics, improving data efficiency by an order of magnitude and enabling interpolation between models trained on biased off-equilibrium data and the equilibrium distribution. BoPITO is demonstrated across diverse systems, including small peptides and fast-folding proteins, showcasing its ability to recover accurate long-timescale behavior even in limited-data regimes.

The fourth paper [29] presents a novel training scheme to fine-tune graph-based DGMs for *de novo* molecular design. This scheme guides the model to generate molecules with desired properties, even when they are rare or absent in the training data. We use a graph neural network to model the action probability distributions for building molecular graphs, and introduce a memory-aware loss function to speed up and stabilize learning. We demonstrate the effectiveness of this approach on several design tasks, especially for generating molecules with predicted dopamine receptor type D2 activity.

Finally, we discuss how these methods could impact drug design pipelines in the future and what important problems towards the automation of drug design remain unsolved.

Chapter 2

Drug discovery and molecular distributions

2.1 The drug discovery process

Drug discovery seeks to develop new medications by leveraging knowledge of biological targets implicated in disease. A biological target is typically a molecule—such as a protein or a gene—that plays a central role in a metabolic or signaling pathway associated with a pathological condition. The overarching goal of drug discovery is to design molecules that interact with such targets and modulate their activity in a way that yields therapeutic benefit [30] (Figure 2.1). This endeavor unfolds through a sequence of interdependent stages, each addressing a distinct aspect of the translation from biological insight to effective medication.

The process begins with target identification and validation. Target identification [31] involves selecting a biological entity whose modulation is expected to influence disease progression. This step commonly relies on high-throughput experimental techniques such as genomics [32] and proteomics [33], which characterize the structure and function of genes and proteins at scale. Target validation then seeks to establish a causal link between the target and the disease phenotype, often through perturbation experiments such as gene knockouts [34], where specific genes are selectively inactivated to assess their functional relevance.

Following target validation, drug design focuses on lead discovery and optimization. Lead discovery aims to identify candidate compounds with measurable activity against the target, typically through the exploration of large chemical spaces. Virtual screening [35] plays a central role in this stage by prioritizing existing or enumerated molecules according to their predicted binding affinity and biological activity. In contrast, *de novo* molecular design [36] seeks to generate novel chemical structures from scratch, optimizing them directly toward predefined physicochemical or pharmacological objectives. Once promising leads are identified, lead optimization iteratively refines their properties—including

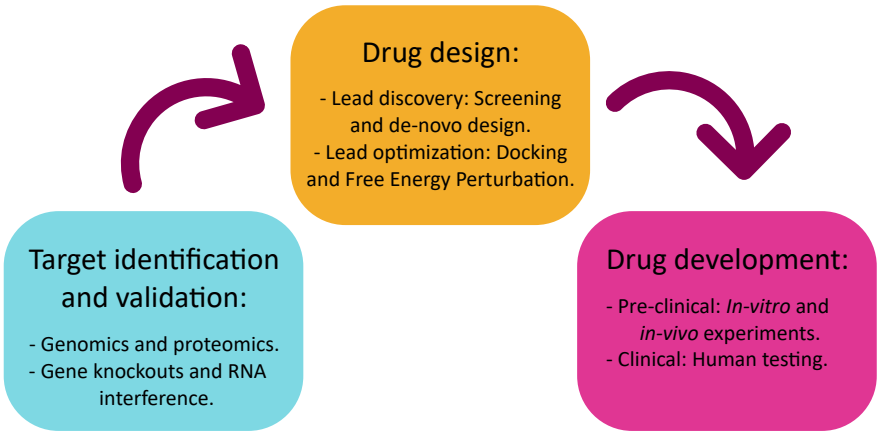


Figure 2.1: The drug discovery process: stages and key techniques

potency, selectivity, solubility, stability, and toxicity—to improve their suitability for downstream development. Quantitative Structure–Activity Relationship (QSAR) modeling [37], [38] plays a central role in this stage by relating molecular structure to biological activity through predictive models.

The final phase, drug development, encompasses the steps required to transform optimized lead compounds into approved therapeutics. This phase includes preclinical studies to assess safety and efficacy in animal models, followed by clinical trials to evaluate therapeutic effects in humans. Successful candidates then undergo regulatory review by agencies such as the European Medicines Agency or the Food and Drug Administration before entering the market, where post-approval monitoring ensures long-term safety and quality.

2.2 Molecular probability distributions in drug design

Drug design is the second stage of the drug discovery pipeline and involves iterative cycles of hypothesis generation and experimental validation. A typical question might be: how does property A change when substructure B is modified in a given molecule? This process is often conceptualized through the Design–Make–Test–Analyze (DMTA) cycle [39], [40], illustrated in Figure 2.2.

The DMTA cycle describes the iterative feedback loop through which new molecules are conceived, synthesized, evaluated, and refined. In the design stage, chemical structures expected to meet certain pharmacological or physicochemical objectives are proposed. These molecules are then made—either synthesized experimentally or generated computationally—and subsequently tested through *in vitro*, *in vivo*, or *in silico* assays to evaluate their relevant properties, such as potency, selectivity, or solubility. Here, *in vitro* refers to experiments conducted outside a living organism, typically in controlled laboratory environments such as cell cultures or biochemical assays; *in vivo*

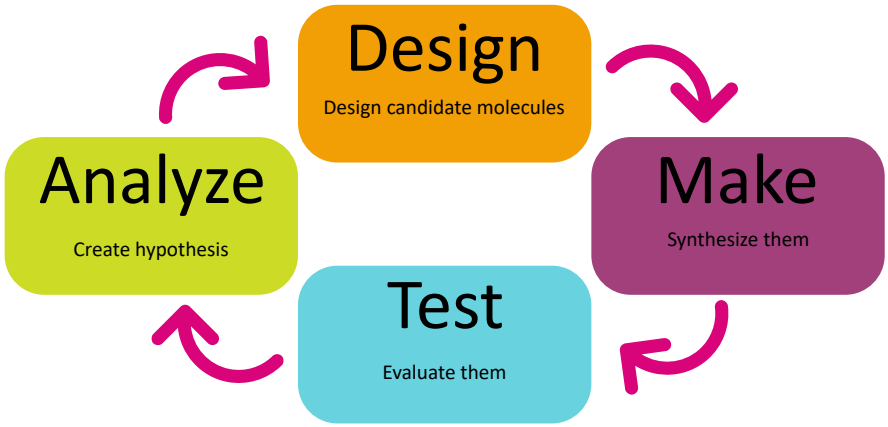


Figure 2.2: The Design–Make–Test–Analyze (DMTA) cycle — a conceptual model for iterative drug design.

denotes studies performed within living organisms to assess physiological efficacy and toxicity; and *in silico* encompasses computational simulations and predictive modeling approaches used to estimate molecular behavior and properties. Finally, the results are analyzed to extract insights that inform the next design iteration. This closed-loop process underpins modern drug discovery but is often slow and resource-intensive, motivating the need for computational methods that can predict outcomes and guide molecular optimization before synthesis.

This thesis focuses on developing methods to accelerate computational drug design by targeting two fundamental challenges: (1) accurately predicting molecular properties, and (2) generating molecular structures that satisfy specified property profiles. Both challenges can be formulated as conditional sampling problems.

Accurate prediction of molecular properties fundamentally depends on understanding how atoms and molecules evolve over time under the influence of interatomic forces. This process can be formally expressed as estimating the probability distribution of molecular states $\mathbf{x}_{\Delta t}$ after some lag time Δt given an initial configuration $\mathbf{x}_0$ and set of thermodynamic or simulation conditions $\boldsymbol{\xi}$,

$$p(\mathbf{x}_{\Delta t} \mid \mathbf{x}_0, \Delta t, \boldsymbol{\xi}). \tag{2.1}$$

This transition probability distribution encodes the stochastic evolution of the molecular system. Molecular property estimation then reduces to computing ensemble averages with respect to this distribution.

Molecular Dynamics (MD) provides a computational framework for approximating these transition probabilities by numerically integrating the equations of motion derived from a potential energy function, typically parameterized by classical force fields or obtained from quantum mechanical calculations. Starting from given initial positions and momenta, MD evolves the system according

to its underlying equations of motion, accounting for solvent effects and thermal fluctuations. When combined with appropriate thermostats and barostats, these simulations generate trajectories that sample well-defined statistical ensembles corresponding to real experimental conditions (e.g., constant-temperature or constant-pressure). Thus, MD offers a principled means of generating statistical realizations of the transition probability distribution through numerical integration, serving as a cornerstone for molecular property prediction.

Many relevant experimentally measurable molecular properties—such as the dissociation rate of a ligand from its receptor or the folding time of a protein—are directly determined by the statistics of these transition probabilities. Accurate estimation of such kinetic and thermodynamic quantities requires adequate sampling of the molecular ensemble over relevant timescales, which may range from microseconds to seconds. However, conventional MD simulations are constrained by the necessity of femtosecond-level integration steps, rendering long-timescale sampling computationally infeasible. Bridging the gap between the timescales accessible to MD and those relevant to experiments—commonly referred to as *the sampling problem*—remains a central challenge. Consequently, the development of methods that accelerate sampling from the transition probability distribution is essential for achieving accurate and efficient molecular property prediction.

Conversely, generating molecular structures that exhibit desired physicochemical or biological properties constitutes an inverse design problem. Such inverse problems can also be formulated as conditional sampling tasks, where the objective is to draw molecular samples that satisfy prescribed property constraints. Formally, this can be expressed as sampling from the property-conditioned molecular graph distribution,

$$p(G \mid \mathbf{y}), \tag{2.2}$$

where G denotes a molecular graph, $\mathbf{y}$ represents the target property specification, and the probability distribution p assigns higher likelihood to molecular graphs that fulfill the desired criteria.

In the context of drug discovery, $\mathbf{y}$ may encode multiple optimization objectives—such as binding affinity to a target receptor, membrane permeability, solubility, selectivity, or metabolic stability—that collectively determine a compound's efficacy, safety, and pharmacokinetic profile. Designing molecules that simultaneously satisfy several of these often conflicting objectives is a highly nontrivial task, as the mapping from molecular structure to property is typically nonlinear, high-dimensional, and discontinuous due to the discrete nature of chemical space. Moreover, the vastness of chemical space—estimated to contain more than 10^{33} synthetically accessible small molecules [41]—renders exhaustive search infeasible, highlighting the need for alternative strategies.

Therefore, efficiently sampling both molecular dynamics transition probability distributions and property-conditioned molecular distributions represents a central challenge in molecular science. Improved sampling strategies promise not only more accurate molecular property prediction but also more effective exploration and optimization of chemical space, ultimately facilitating the

rational design of drug-like compounds and expediting the discovery of new therapeutics.

Chapter 3

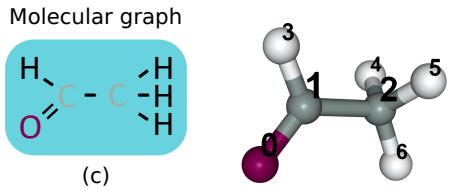
Molecular representations

To predict molecular properties and design new molecules computationally, we first need an appropriate representation of molecules within the computer. In this work, we represent molecules using the positions of their atomic nuclei. Molecules are inherently dynamic and can adopt a range of three-dimensional structures known as conformations. The collection of conformations adopted in equilibrium, referred to as equilibrium conformational ensemble, encodes important information about a molecule's structural diversity and the relative populations of its various states. However, generating representative conformational ensembles remains a significant challenge because conformational transitions often occur over long time and length scales [42].

In certain contexts, conformational ensembles can be approximated by considering only the dominant local maxima of the equilibrium distribution. These representative structures are known as conformers. Among them, the conformer with the lowest potential energy is referred to as the ground state and is often the most statistically representative. While conformers provide a simplified description of molecular structure, they offer a less complete characterization than full conformational ensembles.

Both conformational ensembles and individual conformers can be encoded as collections of three-dimensional structures. A straightforward representation uses the Cartesian coordinates (CCs) of all atoms, as illustrated in Figure 3.1 (a). Although simple, this representation is sensitive to global rotations and translations, which are typically irrelevant to molecular properties. To address this limitation, internal coordinates (ICs) are often employed instead. ICs describe molecular geometry in terms of bond lengths, bond angles, and dihedral angles, as shown in Figure 3.1 (b), and can be converted to Cartesian coordinates. A complete specification of a molecular structure in internal coordinates, together with atom identities and reference atoms, is commonly referred to as a Z-matrix.

For applications where detailed three-dimensional information is unnecessary, simpler representations may be preferable. In particular, a molecule can be described by its chemical graph, which encodes only topological connectivity via nodes representing atoms and edges representing chemical bonds, as



Cartesian coordinates

Atom	Type	Coordinates (Å)		
0	O	0.00	0.00	0.00
1	C	1.21	0.00	0.00
2	C	2.07	1.19	0.00
3	H	1.70	-0.97	0.00
4	H	2.70	1.14	0.98
5	H	2.76	1.13	-0.89
6	H	1.53	2.13	-0.02

(a)

Internal coordinates

Reference	d (Å)	α (rad)	τ (rad)
-	-	-	-
[0,-,-]	1.21	-	-
[1,0,-]	1.47	2.20	-
[1,0,2]	1.08	2.04	3.14
[2,1,0]	1.13	1.86	2.08
[2,1,0]	1.13	1.89	-2.16
[2,1,0]	1.10	2.00	-0.02

(b)

Figure 3.1: Molecular representations for acetaldehyde: Cartesian Coordinates (a), Internal Coordinates (b), consisting of distances (d), angles (α), and dihedral angles (τ), molecular graph (c), and SMILES strings (d).

depicted in Figure 3.1 (c). This graph-based information can be serialized into a compact textual format using the Simplified Molecular-Input Line-Entry System (SMILES) [43], illustrated in Figure 3.1 (d). SMILES strings are widely used for storing and exchanging molecular structures across databases and software platforms. While a molecular graph can be reconstructed from a SMILES string, multiple distinct SMILES representations may correspond to the same graph, which can lead to inconsistencies in generative or predictive models when different strings are used to represent the same molecule.

The choice of molecular representation ultimately depends on the task of interest and the available computational resources. For example, applications requiring accurate predictions of molecular interactions or biological activity often benefit from representations that capture three-dimensional geometry. In contrast, large-scale virtual screening campaigns may rely on graph-based representations to balance expressiveness and computational cost. More expressive representations generally enable more accurate modeling at the expense of increased computational demands. A comparative summary of the properties of different molecular representations is provided in Table 3.1.

3.1 Neural molecular representations

Modern machine learning methods for molecular modeling depend critically on how molecules are represented. Two primary paradigms have emerged for neural molecular representations: sequence-based and graph-based approaches.

One the one hand, sequence-based representations describe molecules using textual encodings such as SMILES. This formulation allows the application of Natural Language Processing (NLP) architectures , including recurrent neural networks [44], [45] and transformers [46], [47], to learn the syntax and

	Advantages	**Drawbacks**
Cartesian Coordinates	Simple	Sensitive to roto-translations
Internal Coordinates	Invariant to roto-translations	Requires specification of reference atoms
Graph	Unique	Lack of 3-dimensional information
SMILES	Simple and lightweight	Not unique. Lack of 3-dimensional information.

Table 3.1: Advantages and drawbacks of different molecular representations: Cartesian Coordinates (CCs), Internal Coordinates (ICs), chemical graphs, and Simplified Molecular-Input Line-Entry System (SMILES) string.

semantics of chemical structures in a manner analogous to language modeling. SMILES-based methods have shown success in molecular property prediction and *de novo* molecule generation [48], [49]. However, they struggle to capture molecular symmetries, are sensitive to the specific SMILES enumeration used, and do not naturally encode geometric or topological information.

On the other hand, graph-based methods represent molecules as a collection of atoms (nodes) connected through bonds (edges). This representation aligns directly with the relational nature of chemistry and enables the use of Graph Neural Networks (GNNs) [50], [51]. GNNs update atomic features by aggregating information from neighboring atoms through message passing:

$$x_i^{(l+1)} = \phi\left(x_i^{(l)}, \bigoplus_{j \in \mathcal{N}(i)} \psi\left(x_i^{(l)}, x_j^{(l)}\right)\right), \tag{3.1}$$

where $x_i^{(l)}$ is the feature vector of atom i at layer l, $\mathcal{N}(i)$ denotes its neighbors, ψ computes pairwise messages, $\bigoplus$ is a permutation-invariant aggregation function (e.g., sum or mean), and ϕ updates the atomic representation. This design ensures invariance to the ordering of atoms and provides a natural framework for incorporating other molecular symmetries.

Symmetries play a central role in the design of neural architectures for chemistry [52], [53], [54], [55]. For example, if we were to predict the forces on the atoms of an isolated molecule, it is desirable for the prediction to rotate along with the input molecule. However, if we were to predict its strain energy, the prediction should not change with global roto-translations of the input. We say that the forces model should be equivariant w.r.t. rotations while the energy model should be invariant. In general, given a model m and a transformation t, we say that m is equivariant w.r.t. t if

$$m(t(\mathbf{x})) = t(m(\mathbf{x})), \tag{3.2}$$

and invariant if

$$m(t(\mathbf{x})) = m(\mathbf{x}). \tag{3.3}$$

Note that the transformation acting before and after the model need not be identical, as it may operate on different spaces

Incorporating these symmetries not only enhances physical consistency but also reduces data requirements and constrains the hypothesis space of learnable models [53], [56]. In molecular systems, the most relevant symmetry groups are permutation, rotation, and translation. The permutation group is the largest of the three, and GNNs are inherently permutation-invariant. Additional symmetry properties can be achieved through careful design of the message-passing and update functions. For instance, defining messages based solely on interatomic distances yields rotation-invariant models, whereas representing messages in terms of relative position vectors enables rotationally equivariant outputs (Figure 3.2).

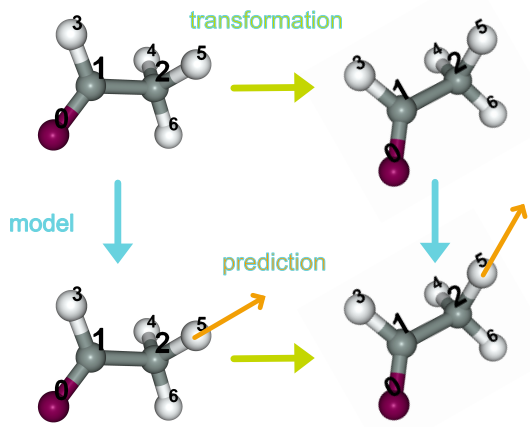


Figure 3.2: Model equivariant to rotations: applying a rotation to the input molecule before or after model evaluation leads to equivalently rotated outputs.

Chapter 4

Molecular dynamics

Molecules are inherently dynamic, and their properties depend on the ensemble of conformations they can adopt. Molecular dynamics (MD) is a computational technique that simulates the time evolution of molecular systems under specified thermodynamic conditions by numerically integrating their equations of motion. In this thesis, we use classical MD simulations within the canonical ensemble (constant number of particles, volume, and temperature). Under these conditions, the equations of motion govern the evolution of atomic positions and momenta according to the laws of classical mechanics, with interatomic forces derived from a potential energy surface defined by a molecular force field. To maintain a constant temperature, the system is coupled to a heat bath, which introduces random forces, resulting in stochastic dynamics.

4.1 Langevin dynamics

A widely used formulation of molecular dynamics, particularly when coupling to a thermal bath is desired, is *Langevin dynamics* [57]. The Langevin equation introduces both frictional and stochastic forces that mimic the interaction of the system with its environment. Let $\mathbf{x} \in \Omega$ and $\mathbf{p} \in \mathbb{R}^{3N}$ denote the positions and momenta of N particles, respectively. The Langevin dynamics stochastic differential equation (SDE) is given by

$$\begin{aligned} \mathrm{d}\mathbf{x} &= \frac{\mathbf{p}}{m}\,\mathrm{d}t, \\ \mathrm{d}\mathbf{p} &= -\nabla_{\mathbf{x}} U(\mathbf{x})\,\mathrm{d}t - \frac{\gamma}{m}\mathbf{p}\,\mathrm{d}t + \sqrt{2\gamma k_B T}\,\mathrm{d}\mathbf{W}, \end{aligned} \tag{4.1}$$

where $\nabla\mathbf{x}$ denotes the gradient w.r.t. to the positions, $U(\mathbf{x})$ is the potential energy, γ is the friction coefficient, m is the mass, k_B is the Boltzmann constant, T is the temperature, and $\mathbf{W}$ is a standard Wiener process [58] representing Gaussian thermal noise.

In the high-friction limit, inertial effects become negligible compared to frictional damping. This regime leads to the *overdamped* approximation [59], in which the momentum variables relax much faster than the positions and can be

effectively eliminated. The resulting dynamics evolve directly in configuration space according to

$$\gamma \, \mathrm{d}\mathbf{x} = -\nabla_{\mathbf{x}} U(\mathbf{x}) \, \mathrm{d}t + \sqrt{2\gamma k_B T} \, \mathrm{d}\mathbf{W}. \tag{4.2}$$

4.2 Fokker–Planck equations

The Langevin SDE defines a continuous-time Markov process on phase space $(\mathbf{x}, \mathbf{p})$. The evolution of the corresponding probability density $p(\mathbf{x}, \mathbf{p}, t)$ is governed by a *Fokker–Planck* equation [60], [61],

$$\frac{\partial p(\mathbf{x}, \mathbf{p}, t)}{\partial t} = -\mathcal{L} p(\mathbf{x}, \mathbf{p}, t), \tag{4.3}$$

where $\mathcal{L}$ is the Fokker-Planck operator. $\mathcal{L}$ can be derived for a given SDE by assuming that the *continuity equation*,

$$\frac{\partial p}{\partial t} + \nabla_{\mathbf{x}} \cdot \left(p \, \frac{\mathbf{p}}{m} \right) = 0, \tag{4.4}$$

holds.

For Langevin dynamics, $\mathcal{L}$ takes the form

$$\mathcal{L} p = \underbrace{\tfrac{\mathbf{p}}{m} \cdot \nabla_{\mathbf{x}} p + \nabla_{\mathbf{p}} \cdot [(\nabla_{\mathbf{x}} U(\mathbf{x}) + \tfrac{\gamma}{m} \mathbf{p}) p]}_{\text{Transport}} + \underbrace{\gamma k_B T \nabla_{\mathbf{p}}^2 p}_{\text{Diffusion}}, \tag{4.5}$$

which is commonly referred to as *Kramers' equation* [62].

Often, we are primarily interested in the time evolution and equilibrium statistics of the positional coordinates $\mathbf{x}$, rather than the full phase-space dynamics. The marginal configurational distribution is obtained by integrating out the momentum variables:

$$p(\mathbf{x}, t) = \int p(\mathbf{x}, \mathbf{p}, t) \, d\mathbf{p}. \tag{4.6}$$

The corresponding Fokker–Planck equation for the overdamped Langevin dynamics governs the time evolution of the configuration-space probability density $p(\mathbf{x}, t)$,

$$\frac{\partial p(\mathbf{x}, t)}{\partial t} = \nabla_{\mathbf{x}} \cdot \left[\underbrace{\frac{1}{\gamma} \nabla_{\mathbf{x}} U(\mathbf{x}) \, p(\mathbf{x}, t)}_{\text{Transport}} + \underbrace{\frac{k_B T}{\gamma} \nabla_{\mathbf{x}}^2 p(\mathbf{x}, t)}_{\text{Diffusion}} \right]. \tag{4.7}$$

which is known as the *Smoluchowski equation* [63]. Note that canceling the diffusion term, setting $\gamma = 1$ and identifying $-\nabla_{\mathbf{x}} U(\mathbf{x})$ with the velocity vector field recovers the continuity equation, Eq. 4.4.

4.3 Boltzmann (equilibrium) distribution

Assuming ergodic dynamics, i.e., the configurations space does not have two or more subsets that are dynamically disconnected, the stationary solution of Kramers' equation corresponds to the *Boltzmann distribution* [64]:

$$\mu(\mathbf{x}, \mathbf{p}) = Z^{-1} \exp[-\beta H(\mathbf{x}, \mathbf{p})] \,, \tag{4.8}$$

where $H(\mathbf{x}, \mathbf{p}) = \frac{\mathbf{p}^2}{2m} + U(\mathbf{x})$ is the Hamiltonian, $\beta = (k_B T)^{-1}$ is the inverse temperature, and Z is the partition function, which is only tractable in simple examples.

This stationary solution represents a balance between the dissipative effects of friction and the randomizing influence of thermal noise. In this state, probability fluxes in phase space cancel out, and Langevin dynamics can be interpreted as a stochastic process whose long-time behavior samples configurations from the canonical ensemble.

Because the Hamiltonian is separable in $\mathbf{x}$ and $\mathbf{p}$, the equilibrium distribution factorizes as

$$\mu(\mathbf{x}, \mathbf{p}) \propto \mu_{\mathbf{x}}(\mathbf{x})\, \mu_{\mathbf{p}}(\mathbf{p}), \tag{4.9}$$

where the momentum distribution $\mu_{\mathbf{p}}(\mathbf{p})$ follows a *Maxwell–Boltzmann distribution* [65],

$$\mu_{\mathbf{p}}(\mathbf{p}) = \left(\frac{\beta}{2\pi m}\right)^{3N/2} \exp\left[-\beta \frac{\mathbf{p}^2}{2m}\right], \tag{4.10}$$

and the configurational marginal $\mu_{\mathbf{x}}$ distribution is given by

$$\mu_{\mathbf{x}}(\mathbf{x}) = Z_x^{-1} \exp[-\beta U(\mathbf{x})], \tag{4.11}$$

with $Z_x = \int \exp[-\beta U(\mathbf{x})]\, \mathrm{d}\mathbf{x}$. Notably, $\mu_{\mathbf{x}}$ is also the stationary solution of Smoluchowski equation.

This marginal distribution over $\mathbf{x}$ corresponds to the canonical ensemble in configuration space. The potential energy $U(\mathbf{x})$ defines an energy landscape whose local minima correspond to metastable conformations, while thermal fluctuations allow transitions between them.

4.4 Transfer operators

Transfer operators provide a complementary perspective on the time evolution of probability densities. Assuming Markovian dynamics, the transition density $p(\mathbf{x}_\tau \mid \mathbf{x}_0, \tau)$ is well defined. The transition density denotes the probability of observing the system in state $\mathbf{x}_\tau$ after a lag time τ, given that it was in state $\mathbf{x}_0$ at time zero. The *Perron–Frobenius transfer operator*[1] $\mathcal{P}^\tau$ [66], [67] maps

[1]Technical note: There exist multiple formulations of transfer operators. Here, for clarity of presentation, we refer to the Perron–Frobenius operator as the transfer operator. In some contexts, including in Paper 3, the μ-weighted density propagator is also referred to as the transfer operator.

an initial probability density p_0 to the propagated density $p(\mathbf{x}, \tau)$ according to

$$p(\mathbf{x}, \tau) = (\mathcal{P}^\tau p_0)(\mathbf{x}) = \int p(\mathbf{x} \mid \mathbf{x}_0, \tau)\, p_0(\mathbf{x}_0)\, \mathrm{d}\mathbf{x}_0. \tag{4.12}$$

Thus, $\mathcal{P}^\tau$ acts as a linear operator that transports probability densities forward in time by integrating over all possible transitions between states.

The transfer operator can equivalently be expressed in differential form through its infinitesimal generator, given by the Fokker–Planck operator introduced in Eq. 4.3. Then, the finite-time propagator is recovered through the exponential relation

$$\mathcal{P}^\tau = e^{\tau \mathcal{L}}. \tag{4.13}$$

Hence, while the Fokker–Planck operator governs the instantaneous rate of change of probability densities, the transfer operator describes their evolution over a finite time horizon.

The transfer operator can also be written in spectral form [68]. Assuming detailed balance is satisfied, $\mathcal{P}^\tau$ admits an eigendecomposition of the form

$$\mathcal{P}^\tau p_0 = \sum_i \lambda_i \, \langle p_0, \psi_i \rangle \, \phi_i, \tag{4.14}$$

where $\{\lambda_i\}$ are the eigenvalues satisfying

$$1 = \lambda_1 > \lambda_2 \geq \lambda_3 \geq \cdots \geq -1,$$

and $\{\phi_i, \psi_i\}$ denote the corresponding right and left eigenfunctions, respectively. Since the transfer and Fokker–Planck operators are connected via an exponential relation (Eq. 4.13), the two operators share the same eigenfunctions. Their eigenvalues are also connected exponentially according to

$$\tau_i = -\frac{\tau}{\ln |\lambda_i|}. \tag{4.15}$$

where τ_i are the eigenvalues of the Fokker–Planck operator, corresponding to the characteristic relaxation rates of the underlying stochastic dynamics. Transfer operator eigenvalues close to one correspond to large relaxation times, indicating slow dynamical processes that dominate the long-term kinetics of the system.

The leading eigenfunction ϕ_1 corresponds to the stationary (equilibrium) density, since $\mathcal{P}^t \phi_1 = \phi_1 = \mu$. Moreover, the subdominant eigenfunctions $\phi_2, \phi_3, \ldots$ describe slowly relaxing modes of the dynamics, such as transitions between metastable conformations or large-scale collective rearrangements in molecular systems.

Finally, the action of the transfer operator is additive. If the transfer operator at lag time τ is known, then operators at integer multiples of this lag time, $\Delta t = N\tau$, can be obtained through sequential applications of $\mathcal{P}^\tau$,

$$\mathcal{P}^{N\tau} p_0 = (\mathcal{P}^\tau)^N p_0 = \sum_i \lambda_i^N \, \langle p_0, \psi_i \rangle \, \phi_i, \tag{4.16}$$

where the eigenvalues are raised to the Nth power, reflecting the exponential decay of the corresponding dynamical modes over time.

Chapter 5

Molecular properties

Predicting the properties of molecules is an essential part of drug design since it allows for the identification of promising candidates and reduces the cost and time of experimental testing [69], [70]. In principle, accurate property prediction requires access to representative conformational ensembles, which can be generated using MD simulations. However, some applications may allow compromising performance in favor of reduced runtime and graph-based methods may be preferred.

Graph-based models for property prediction take the molecular graph (or SMILES) as input and generate predictions as outputs. Diverse models can be used in this context such as logistic regression, support vector machines, random forests or neural networks [71]. One particularly important type of models are QSAR models [8], which are used to predict the activity of drug candidates w.r.t. a given target. QSAR models can be classified into regression models, which predict continuous activity values, or classification models, which predict categorical activity values [9]. These models tend to be lightweight and fast and therefore are often used for building scoring functions guiding generative models.

In contrast, ensemble-based methods rely on inferring molecular properties from a representative conformational ensemble, $\{\mathbf{x}\}$. Given independent and identically distributed (i.i.d.) conformations sampled from the Boltzmann distribution or the transition probability distribution, a molecular property (*observables*), O can be computed via Monte Carlo estimation [72],

1. **Stationary observables**:

$$O_f = \mathbb{E}_{\mathbf{x}\sim\mu}\left[f(\mathbf{x})\right]. \tag{5.1}$$

2. **Dynamic observables / Time correlation functions**:

$$O_{f(0),g(\Delta t)} = \mathbb{E}_{\mathbf{x}_0}\left[\mathbb{E}_{\mathbf{x}_{\Delta t}\sim p(\mathbf{x}_{\Delta t}|\mathbf{x}_0,\Delta t)}\left[f(\mathbf{x}_0)\cdot g(\mathbf{x}_{\Delta t})\right]\right], \tag{5.2}$$

Here, f and $g : \Omega \to \mathbb{R}^L$ denote observable or *forward model* functions that map microscopic configurations to measurable quantities. These functions

capture physical properties such as interatomic distances, binding states, or order parameters, leading to macroscopic observables such as binding affinities, reaction rates, or conformational transition timescales. In conclusion, accurate prediction of molecular properties fundamentally depends on the ability to sample from both the Boltzmann distribution and the corresponding transition density, with molecular dynamics remaining the principal computational tool for generating such samples.

5.1 Limitations of MD for predicting molecular properties

Despite its theoretical rigor, molecular dynamics (MD) faces significant challenges when used for molecular property prediction. These limitations arise from both analytical intractability and computational inefficiency inherent to high-dimensional stochastic dynamics.

Analytical solutions of the underlying transfer operator are only available for simple systems, such as free diffusers or harmonic oscillators. For realistic molecular systems with rugged, high-dimensional energy landscapes, closed-form solutions of the transfer operator are generally intractable. Consequently, the propagation of probability densities and the computation of ensemble averages must rely on numerical simulations.

While the Boltzmann distribution provides a formal expression for the stationary density, the partition function is typically intractable. Even when unnormalized analytical expressions for $\mu(\mathbf{x})$ are available, generating representative samples remains difficult because molecular systems tend to spend most of their time in low-energy regions, leading to inefficient exploration of the configurational space.

In practice, representative ensembles are obtained through numerical integration of MD SDEs, such as Eqs. 4.1 and 4.2. MD simulations must employ extremely small integration time steps (on the order of 1—2 femtoseconds) to accurately resolve the fastest degrees of freedom in the system, such as bond vibrations or angle oscillations, and preserve numerical stability. This constraint leads to the need for an enormous number of integration steps to access biologically relevant timescales, which often span from microseconds to milliseconds. Furthermore, the number of integration steps needed to obtain representative sampling grows rapidly with system dimensionality and with the metastability of the potential energy surface. This makes MD particularly inefficient for systems exhibiting rare transitions between long-lived states, such as protein folding, conformational rearrangements, or ligand binding and unbinding events.

To illustrate these challenges, consider the overdamped Langevin dynamics (Eq. 4.2). For an infinitesimally small time step τ, the conditional probability distribution of the particle position admits a closed-form Gaussian expression,

$$p(\mathbf{x}_\tau \mid \mathbf{x}_0, \tau) = \mathcal{N}\big(\mathbf{x}_\tau \mid \mathbf{x}_0 - \nabla U(\mathbf{x}_0)\gamma^{-1}\tau,\, 2k_B T\gamma^{-1}\tau\big)\,. \tag{5.3}$$

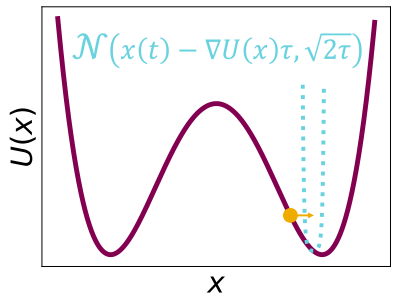


Figure 5.1: MD require many simulation steps to generate representative ensembles. Because integration times need to be small for stable simulation, the next step distribution is shallow. All constants but infinitesimally time increment, τ, are set to 1 for clarity of presentation.

This expression illustrates that transitions between wells correspond to rare events, since crossing the energy barrier requires a fluctuation far in the tail of the transition probability density. As a result, the system can remain trapped in a single well for long periods, leading to slow mixing and biased estimates of equilibrium ensemble properties (Fig. 5.1).

Alternative numerical approaches such as Markov Chain Monte Carlo (MCMC) [22], [23], [24] and enhanced sampling techniques [73] have been developed to accelerate convergence toward the Boltzmann distribution. MCMC algorithms can in principle sample from arbitrary distributions without explicitly integrating MD equations. However, they often suffer from low acceptance rates in high-dimensional or rugged energy landscapes, leading to poor exploration and long autocorrelation times—limitations closely related to those encountered in MD. Enhanced sampling methods, such as replica-based approaches [74], [75], [76], conformational flooding [77], meta-dynamics [78], and umbrella sampling [79], aim to overcome energy barriers by biasing the sampling process or introducing auxiliary variables. While these approaches can significantly improve mixing efficiency, they generally require extensive system-specific parameter tuning, careful bias design, or prior knowledge of slow collective variables. Moreover, because these techniques artificially alter the underlying dynamics, they do not provide physically meaningful information about kinetics or transition pathways, restricting their applicability when dynamical observables are of interest.

These limitations highlight the fundamental trade-off in molecular simulation: while MD provides a principled route to sampling equilibrium and dynamical distributions, its computational cost scales poorly with system complexity. This motivates the development of alternative or complementary density estimators—such as deep generative models—that can learn to approximate or accelerate the sampling of molecular ensembles.

Chapter 6

Traditional approaches to molecular generation

Traditionally, the design of new molecular structures satisfying specific physicochemical or biological requirements relied on a combination of human expertise, heuristic rules, and computational search methods. These approaches aimed to explore chemical space in a directed manner by using known molecular scaffolds, reaction templates, and property predictors to guide the search toward compounds with desirable features. While these strategies achieved significant successes, they were limited by combinatorial complexity, lack of diversity, and dependence on prior chemical intuition.

One of the earliest systematic approaches was *combinatorial chemistry* [80], in which large libraries of compounds were generated by combining sets of chemical building blocks according to predefined reaction schemes. This enabled rapid synthesis and screening of extensive molecular collections but provided little control over the resulting property distributions. Combinatorial chemistry was often coupled with *high-throughput screening* (HTS) [81], where physical or virtual assays were used to identify molecules showing desired biological activity. Although HTS revolutionized early-stage drug discovery, it remained costly and inefficient, as only a small fraction of tested compounds exhibited the intended activity or acceptable pharmacokinetic properties.

To improve the efficiency of this process, *structure–activity relationship* (SAR) [82] and *quantitative structure–activity relationship* (QSAR) [8], [9] models were developed. These models established empirical mappings between molecular descriptors and target properties, allowing the prediction of biological or physicochemical behaviors for untested compounds. Despite their usefulness, these approaches depended on manually designed molecular modifications guided by chemical expertise and iterative trial-and-error.

Another important paradigm was *fragment-based drug design* (FBDD), which aimed to assemble small, experimentally validated molecular fragments into larger and more potent ligands. This allowed for a more controlled exploration of chemical space by combining fragments that were already known to bind to specific regions of a target protein. Similarly, *pharmacophore*

modeling [83] abstracted the essential chemical features responsible for biological activity—such as hydrogen-bond donors, acceptors, or hydrophobic centers—and used these abstractions as templates to identify or design new molecules with comparable functionality.

Further important contributions were *virtual screening* and *evolutionary algorithms*. In virtual screening [84], large molecular databases were computationally evaluated against scoring functions that approximated binding affinity or other desired properties. Evolutionary algorithms [25], by contrast, approached molecular design as an optimization problem over discrete chemical structures. Molecules were treated as evolving populations, with successive generations produced by mutation, crossover, and selection based on heuristic fitness functions. While conceptually appealing, these methods suffered from several key limitations. Their search was often inefficient, as random structural mutations tend to disrupt molecular validity or lead to chemically implausible intermediates. Furthermore, evolutionary algorithms lacked a learned representation of chemical space—each optimization run effectively started from scratch, without leveraging statistical patterns from existing molecular data. As a result, these methods struggled to generalize or to efficiently exploit prior knowledge about molecular structure–property relationships.

Overall, traditional molecular generation can be viewed as an iterative cycle of proposal, evaluation, and refinement. Molecular candidates were proposed through rule-based or stochastic approaches, evaluated using property prediction models, and refined according to expert judgment. While these methods established important paradigms that remain influential today, they were fundamentally limited by their reliance on hand-crafted rules and heuristics rather than learned representations of chemical space. Each approach required extensive domain expertise to define appropriate molecular modifications, reaction templates, or search operators, and these definitions had to be manually adjusted for different design objectives. Furthermore, prior knowledge about molecular structure and properties could not be systematically leveraged across different tasks or domains. These limitations, motivated the development of data-driven generative modeling frameworks, which aim to learn molecular distributions directly from large datasets and provide more flexible, generalizable approaches to molecular design.

Chapter 7

Deep generative modeling

Generative models are probabilistic models of data distributions. They learn a parameterized distribution $p_\theta(\mathbf{x})$ that approximates the underlying data distribution $p_{\text{data}}(\mathbf{x})$ from observed samples. Once trained, they enable *inference*—that is, the generation of new samples or the estimation of probabilities under the learned distribution.

Formally, generative models define $p_\theta(\mathbf{x})$ via a transformation of latent variables $\mathbf{z} \sim p(\mathbf{z})$ drawn from a simple base distribution (e.g., a Gaussian) through a parameterized mapping. The key design question lies in how this transformation is parameterized. Two major paradigms have emerged in modern deep learning for generative modeling: *auto-regressive models* and *probability flow models*, which are compared in Table 7.1. These approaches differ in how they represent densities and construct sampling procedures, but all share the goal of learning expressive, high-dimensional probability distributions. Such generative modeling frameworks are the underpin some of the most significant recent advances in artificial intelligence, such as large language models [46], [47], image [85], [86], [87] and video [88], [89] generation or protein structure prediction models [90], [91], [92].

7.1 Auto-regressive models

Auto-regressive (AR) models [93], [94], [95] decompose the joint distribution over data into a product of conditional probabilities. For a D-dimensional vector $\mathbf{x} = (x_1, \ldots, x_D)$, the joint density can be written as

$$p_\theta(\mathbf{x}) = \prod_{i=1}^{D} p_\theta(x_i \mid x_1, \ldots, x_{i-1}). \tag{7.1}$$

Each conditional distribution is represented by a neural network that predicts the next component given the previously generated ones. Training is performed by *maximum likelihood estimation* (MLE), minimizing the negative

	Training	**Inference**	**Likelihood**
Auto-regressive models	Maximum likelihood	Sequential sampling	Exact, inexpensive
Diffusion models	Score matching	Integration of learned SDE or associated ODE	Intractable
Continous Normalizing Flows	Maximum likelihood or flow matching	Integration of learned ODE	Tractable, expensive

Table 7.1: Comparison of major deep generative model families. Each model type is characterized by its training objective, the nature of its sampling or inference procedure, and whether the model admits a tractable expression for the data likelihood.

log-likelihood of observed data:

$$\mathcal{L}_{\mathrm{AR}}(\theta) = -\mathbb{E}_{\mathbf{x}\sim p_{\mathrm{data}}}\left[\sum_{i=1}^{D} \log p_\theta(x_i \mid x_{<i})\right]. \tag{7.2}$$

Because this factorization yields an exact likelihood, optimization is tractable.

Inference proceeds sequentially: given $x_{<i}$, the model samples $x_i \sim p_\theta(x_i \mid x_{<i})$ until a complete vector $\mathbf{x}$ is produced. While this sequential nature allows fine-grained control over dependencies, it also imposes a computational bottleneck for high-dimensional systems.

In molecular applications, auto-regressive models are used for example in sequential graph generation [14], [15], where atoms and bonds are sampled step-by-step. Common implementations employ masked transformers [5] or graph neural networks [15].

7.2 Probability flow models

Probability flow models [96], [97] define a continuous transformation between a simple, easy-to-sample base distribution p_0 (typically a Gaussian) and the data distribution $p_1 = p_{\mathrm{data}}$. This transformation is governed by a differential equation whose dynamics evolve samples from noise to data.

Depending on whether the dynamics are deterministic or stochastic, two main classes of models emerge: *continuous normalizing flows* [98], [99], using an ordinary differential equation (ODE), and *diffusion models* [100], [101], [102], implementing a stochastic differential equation (SDE):

$$\mathrm{d}\mathbf{x}_t = \mathbf{u}_t^\theta(\mathbf{x}_t)\,\mathrm{d}t \qquad \text{(Continuous normalizing flow)}, \tag{7.3}$$

$$\mathrm{d}\mathbf{x}_t = \mathbf{u}_t^\theta(\mathbf{x}_t)\,\mathrm{d}t + \sigma_t\,\mathrm{d}\mathbf{W}_t \qquad \text{(Diffusion model)}. \tag{7.4}$$

Here, $\mathbf{u}_t^\theta(\mathbf{x}_t)$ denotes a neural network parameterizing the time-dependent vector field, σ_t is a time-dependent diffusion coefficient controlling the noise

magnitude, $\mathbf{W}_t$ represents standard Wiener process, and $t \in [0, 1]$ indexes the transformation from noise to data. Note that the SDE underlying a diffusion model corresponds to a time-dependent (inhomogeneous) overdamped Langevin SDE.

Training a probability flow model involves learning the vector field $\mathbf{u}_t^\theta(\mathbf{x})$. Once trained, inference proceeds by drawing initial samples $\mathbf{x}_0 \sim p_0$ and numerically integrating the corresponding differential equation.

In the following sections, we introduce continuous normalizing flows and diffusion models, and describe how the vector field $\mathbf{u}_t^\theta(\mathbf{x})$ can be trained using samples from p_{data}.

7.2.1 Continuous normalizing flows

Continuous normalizing flows (CNFs) represent the deterministic subclass of probability flow models. Originally, they were originally trained by maximizing the data likelihood,

$$\mathcal{L}_{\text{CNF}}(\theta) = -\,\mathbb{E}_{\mathbf{x}\sim p_{\text{data}}}[\log p_\theta(\mathbf{x}_1)]. \tag{7.5}$$

The model likelihood can be computed via the change-of-variables formula, which describes which describes the evolution of the density $p_t^\theta(\mathbf{x})$ evolves along the flow trajectory:

$$\frac{\mathrm{d}}{\mathrm{d}t}\,\log p_t^\theta(\mathbf{x}_t) = -\,\nabla\cdot\,\mathbf{u}_t^\theta(\mathbf{x}_t). \tag{7.6}$$

This expression corresponds to the continuity equation (Eq. 4.4), evaluated along the flow trajectory rather than at a fixed spatial location. By integrating the divergence term along each trajectory, CNFs enable exact likelihood computation and, in principle, optimization of Eq. 7.5. In practice, however, this approach demands explicit numerical integration of the flow dynamics, which is computationally expensive and often unstable, making training difficult or even impractical.

A more practical alternative is the flow matching framework [103], which avoids explicit likelihood computation. Flow matching defines a conditional interpolation path transporting samples from p_0 to p_{data}. A simple and common choice is the *linear interpolant*,

$$\mathbf{x}_t = (1-t)\mathbf{x}_0 + t\mathbf{x}_1, \quad t \in [0, 1], \tag{7.7}$$

which defines a straight-line trajectory. The corresponding conditional target velocity field is

$$\mathbf{u}_t(\mathbf{x}_t \mid \mathbf{x}_1) = \frac{\mathrm{d}\mathbf{x}_t}{\mathrm{d}t} = \mathbf{x}_1 - \mathbf{x}_0. \tag{7.8}$$

The model vector field $\mathbf{u}_t^\theta$ is trained to approximate this target conditional field under the *conditional flow matching loss*,

$$\mathcal{L}_{\text{FM}}(\theta) = \mathbb{E}_{t,\,\mathbf{x}_t\sim q_t}\left[\|\,\mathbf{u}_t^\theta(\mathbf{x}_t) - \mathbf{u}_t(\mathbf{x}_t \mid \mathbf{x}_1)\,\|^2\right], \tag{7.9}$$

where q_t denotes the reference interpolation between p_0 and p_{data}. Crucially, this objective only requires samples from the base and data distributions—no likelihood computation or ODE solving is needed during training.

7.2.2 Diffusion models

Diffusion models are the stochastic variant of probability flow models. Like CNFs, they define a continuous interpolation between samples from the base and target distributions. A common instance is the Denoising Diffusion Probabilistic Model (DDPM) framework [100], which employs a normal interpolant,

$$p_t(\mathbf{x}_t \mid \mathbf{x}_1) = \mathcal{N}(\mathbf{x}_t \mid \sqrt{\alpha_t}\mathbf{x}_1, (1-\alpha_t)\mathbb{I}), \tag{7.10}$$

where the noise scheduler α_t is a continuously differentiable and monotonic function of t satisfying $\alpha_0 = 0$ and $\alpha_1 = 1$.

An associated SDE bridges the base and data distributions [101], [104],

$$d\mathbf{x}_t = -\frac{1}{2}\beta_t \left(\mathbf{x} + 2\nabla \log p_t(\mathbf{x})\right) dt + \sqrt{\beta}_t d\mathbf{W}_t, \tag{7.11}$$

where $\beta_t = -\frac{d \log \alpha_t}{dt}$, and $\nabla \log p_t(\mathbf{x})$ is the marginal score. Alternatively, the corresponding ODE [101], [104] may be used for inference,

$$d\mathbf{x}_t = -\frac{1}{2}\beta_t \left(\mathbf{x} + \nabla \log p_t(\mathbf{x})\right) dt. \tag{7.12}$$

In both cases, learning the marginal score also yields an expression for $\mathbf{u}_t^\theta$.

Because the true score is intractable, it is approximated via *score-matching loss* [101] by regressing a model $\boldsymbol{\epsilon}_\theta$ against the conditional score,

$$\mathcal{L}_{\text{score}}(\theta) = \mathbb{E}_{t,\mathbf{x}_1,\mathbf{x}_t}\left[\|\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t) - \nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t \mid \mathbf{x}_1)\|^2\right]. \tag{7.13}$$

Moreover, by writing $\mathbf{x}_t$ as $\sqrt{\alpha_t}\mathbf{x}_1 + \sqrt{1-\alpha_t}\boldsymbol{\epsilon}$, with $\epsilon \sim \mathcal{N}(0, \mathbb{I})$ (Eq. 7.10), this loss simplifies to a denoising objective,

$$\mathcal{L}_{\text{score}}(\theta) = \mathbb{E}_{t,\mathbf{x}_1,\mathbf{x}_t}\left[\left\|\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t) - \frac{\mathbf{x}_t - \sqrt{\alpha_t}\mathbf{x}_1}{\sqrt{1-\alpha_t}}\right\|^2\right] = \mathbb{E}_{t,\mathbf{x}_1,\boldsymbol{\epsilon}}\left[\|\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t) - \boldsymbol{\epsilon}\|^2\right], \tag{7.14}$$

revealing that training a diffusion model corresponds to learning to denoise progressively corrupted data. This loss can be viewed as a simplified version of the variational lower bound optimization [100], [101].

Standard diffusion models operate in Euclidean space, but molecular systems often include *periodic* variables, such as dihedral angles, which lie on the n-dimensional torus $\mathbb{T}^n$. In such cases, Euclidean diffusion produces discontinuities at angular boundaries. *Riemannian diffusion models* [105], [106] generalize diffusion to manifolds such as the torus, sphere with minor modifications. On the torus, the forward diffusion process is defined by a rescaled Brownian motion on the manifold, with the key difference that the base distribution $p_0(\boldsymbol{\theta})$ is

uniform over $\mathbb{T}^m$ rather than Gaussian [107]. The corresponding interpolant for this process is a *wrapped normal distribution*, which accounts for the periodic topology by summing Gaussian contributions over all equivalent points under the identification $\boldsymbol{\theta} \sim \boldsymbol{\theta} + 2\pi\mathbf{d}$ for $\mathbf{d} \in \mathbb{Z}^m$,

$$p(\boldsymbol{\theta}_t \mid \boldsymbol{\theta}_1) \propto \sum_{\mathbf{d}\in\mathbb{Z}^m} \exp\left(-\frac{\|\boldsymbol{\theta}_1 - \boldsymbol{\theta}_t + 2\pi\mathbf{d}\|^2}{2\gamma_t^2}\right), \tag{7.15}$$

where γ contains noise levels.

The reverse-time sampling then proceeds as a geodesic random walk on the torus, ensuring that generated conformations respect the angular boundary conditions.

Chapter 8

Generative molecular modeling

As discussed in the previous chapters, many of the key challenges in molecular design can be reformulated as problems of sampling from complex probability distributions. Deep generative models (DGMs) offer a powerful and flexible framework for learning and approximating such distributions. In this work, we investigate how these models can be leveraged to accelerate molecular property prediction and molecular generation by serving as efficient surrogates of hard-to-sample molecular distributions.

8.1 Generative molecular dynamics

As discussed in Chapter 5.1, estimating molecular properties fundamentally relies on sampling from the equilibrium and transition probability distributions that govern molecular dynamics. Traditionally, these samples are obtained through MD simulations. However, MD is computationally expensive due to its need for very small integration time steps to maintain numerical stability, leading to slow mixing and poor sampling efficiency.

A promising alternative is to replace explicit simulation with a DGM that approximates the relevant probability distributions directly. This family of methods, often referred to as *generative molecular dynamics* (GMD), aims to learn surrogate models for either the equilibrium distribution or the transition probability distribution. Once trained, such models can generate statistically valid molecular configurations or trajectories at a fraction of the cost of conventional simulation.

8.1.1 Surrogate models of the equilibrium distribution

Surrogate models that approximate the equilibrium (Boltzmann) distribution are known as *Boltzmann Generators* (BGs) [6]. Boltzmann Generators use a deep generative model—typically a normalizing flow—to learn a mapping

between a simple latent distribution (e.g., a multivariate Gaussian) and molecular configurations in equilibrium. Samples from the generative model are then reweighted to recover unbiased estimates with respect to the target Boltzmann distribution,

$$w_i = \frac{\mu(\mathbf{x}_i)}{p_\theta(\mathbf{x}_i)} \propto \frac{\exp(-\beta U(\mathbf{x}_i))}{p_\theta(\mathbf{x}_i)}, \tag{8.1}$$

where $p_\theta(\mathbf{x})$ is the DGM density.

To be effective, Boltzmann Generators must (i) approximate closely the Boltzmann distribution, (ii) allow efficient, i.i.d. sampling from p_θ, thereby avoiding iterative simulation, and (iii) permit exact likelihood evaluation to enable unbiased importance reweighting. This combination enables direct computation of thermodynamic observables and free energy differences without requiring long MD trajectories. Early Boltzmann Generators [6], [108], [109], [110] demonstrated the feasibility of learning complex equilibrium distributions but suffered from limited transferability across molecular systems. Recent work [111], [112], [113] has improved generalization, though applicability remains largely restricted to small molecules and peptides.

Another related approach is Boltzmann emulators, which do not aim to capture full atomistic detail but instead focus on key degrees of freedom, such as dihedral angles of rotatable bonds [26], [107] in small molecules, or Cartesian coordinates of alpha carbons in peptides and proteins [114]. In particular, BioEMU [114] demonstrated remarkable transferability across medium-sized soluble proteins, highlighting the potential of generative equilibrium models in biomolecular systems. While Boltzmann emulators do not permit exact reweighting with respect to the full Boltzmann distribution, they offer dimensionality reduction benefits, including lighter inference and likelihood evaluation, and approximate reweighting can still, in principle, be performed [26].

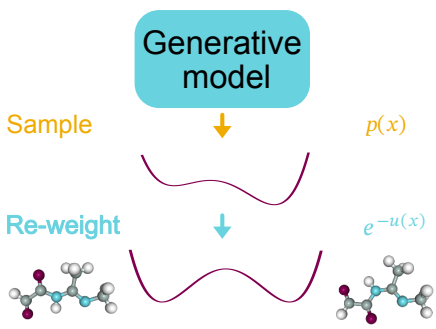


Figure 8.1: Boltzmann Generators: a generative model samples a surrogate equilibrium distribution and associated sample probabilities. The generated ensemble is subsequently reweighted with respect to the Boltzmann distribution to recover unbiased estimates of thermodynamic observables.

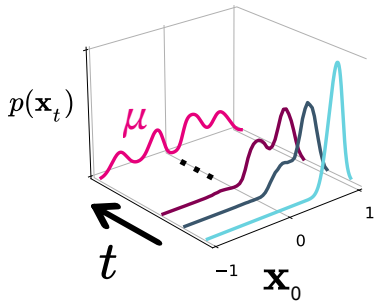


Figure 8.2: Implicit transfer operators learn surrogate models of molecular dynamics at arbitrary time lags.

8.1.2 Surrogate models of the transition probability

While Boltzmann Generators target equilibrium statistics, they neglect temporal correlations present in molecular trajectories and therefore cannot be used to estimate dynamical observables, such as relaxation times or transition rates. To address this limitation, an alternative line of work focuses on learning surrogate models of the transition probability distribution [113], [115], [116], [117]. By learning this distribution directly from MD trajectories, these models can generate dynamically consistent samples and enable estimation of both thermodynamic and kinetic properties without requiring explicit numerical integration of the equations of motion.

Most existing surrogates of the transition probability distribution use a fixed lag time Δt. However, molecular systems exhibit rich multi-scale behavior and an ideal surrogate for the transition density should accommodate arbitrary lag times. The *Implicit Transfer Operator* (ITO) framework [118] addresses this challenge by training a conditional generative model that explicitly depends on the lag time Δt as an input variable. During training, the model is exposed to trajectory segments with varying time intervals, learning to generate configurations $\mathbf{x}_{t+\Delta t}$ conditioned on both the initial state $\mathbf{x}_t$ and the desired lag time Δt. This yields a continuous family of transfer operators $\mathcal{T}_{\Delta t}$ parameterized by a neural network that side-step numerical integration of MD trajectories.

8.2 Generative chemistry

As presented in Chapter 6, classical approaches to molecular structure generation maintained a separation between the generation and evaluation stages, limiting their ability to efficiently explore chemical space and capture the complex distributions governing molecular structures and properties. Evolutionary algorithms enabled automation, but treated molecular design as an optimization over discrete structures, evolving populations through mutation, crossover, and selection, often producing invalid molecules. Moreover, the absence of learned chemical representations made their search inefficient and poorly generaliz-

able, hindering the effective use of prior knowledge about structure–property relationships.

In contrast, modern deep generative and reinforcement learning frameworks address many of these shortcomings by learning differentiable, data-driven representations of molecular distributions and optimizing objectives directly in continuous latent spaces. These models can adaptively explore chemical space, balance exploitation and exploration, and incorporate gradients from predictive models or experimental feedback to guide the generation process more effectively.

DGMs are increasingly being adopted for *de novo* design, leading to the emerging field of generative chemistry. Most approaches start from training a prior generative model on large chemical datasets, which can be seen as 'foundation models' for chemistry. However, these models may not generate molecules that satisfy the design requirements. Therefore, transfer learning [119], [120] is used to bias the prior model towards regions of chemical space that are more desirable. There are two main approaches. The first one consists of fine-tuning the prior model on a smaller dataset of molecules satisfying the design constraints [121]. The second one relies on the availability of a scoring model, assessing the desirability of molecules, and consists of biasing the prior model to promote the generation of highly scored compounds [14]. If target data are scarce, usually the first approach is preferred, but if a reliable model is available, it may be exploited by following the second approach.

Several types of generative models have been proposed to model molecular structure distributions [122]. Previous work based on SMILES strings has used Recurrent Neural Networks (RNNs) [14], [121], [123], Variational Autoencoders (VAEs) [124], [125], Transformers, or Generative Adversarial Networks (GANs) [126], [127]. Methods using a graph representation have used different types of GNNs [128] such as Gated Graph Neural Networks (GGNN) [15] or Graph Convolutional Neural Networks (GCNNs) [129] to iteratively sample actions that build up a chemical graph. More recently, 3D-generative models [130], [131] have explored molecular generation directly in the binding pocket, potentially accounting explicitly for physical interactions. Prevalent methods are diffusion models [101], [132], [133] and normalizing flows [99] powered by equivariant models [52], [53], [134], [135].

The convergence of deep generative modeling with molecular simulation and design represents a fundamental shift in computational chemistry. Rather than relying solely on physics-based simulation or heuristic search, we can now learn probabilistic surrogates that capture the complex distributions governing molecular structure, dynamics, and properties. This thesis explores how such models can be leveraged to address key bottlenecks in molecular science: from accelerating equilibrium and dynamical sampling in molecular dynamics, to enabling efficient exploration of chemical space for molecular design. The papers presented in the following chapters investigate specific instantiations of this paradigm, demonstrating how generative models can serve as flexible, data-driven alternatives to traditional computational approaches.

Chapter 9

Summary of included papers

9.1 Generation of conformational ensembles of small molecules via Surrogate Model-Assisted Molecular Dynamics

The first paper addresses the challenge of developing transferable surrogate models for molecular equilibrium distributions. We introduce Surrogate Model-Assisted Molecular Dynamics (SMA-MD), a novel method for generating equilibrium conformational ensembles of small molecules. SMA-MD integrates a deep generative model (DGM) that samples slow molecular degrees of freedom with a subsequent reweighting and short simulation step that equilibrates the fast degrees of freedom. This approach produces conformational ensembles that are both more diverse and more physically realistic than those obtained through conventional MD simulations alone.

We use a two-step procedure to generate molecular conformations. First, we use a deterministic algorithm to generate the local structure of each atom, and then we use a diffusion model to sample the torsion angles of rotatable bonds [107]. The diffusion model is trained on MD simulations of small non-cyclic molecules. Second, we reweight the generated conformations against the Boltzmann distribution and run short parallel MD simulations to thermalize and mix the fast degrees of freedom.

We evaluate our method by comparing it with MD and Replica Exchange (RE) simulations on various metrics, such as conformer generation, potential energy, free energy of solvation, and slow transitions. We show that SMA-MD outperforms MD in generating more diverse and energetically favorable ensembles, and matches RE in capturing the relevant states and properties of molecules.

We conclude that SMA-MD is an efficient and robust method for sampling from the Boltzmann distribution of molecules. We highlight the advantages of

SMA-MD over MD, such as data aggregation, parallelization, and independence of initial conditions. We also discuss the limitations and future directions of SMA-MD, such as extending it to cyclic molecules, improving the computational cost of sampling, and training Boltzmann surrogates with large-scale data.

SMA-MD shows promising results toward accelerating the generation of representative conformational ensembles of molecules with DGMs. As such, SMA-MD is a step toward faster methods for predicting molecular properties, which is fundamental in drug design.

9.2 Transferable Generative Models Bridge Femtosecond to Nanosecond Time-Step Molecular Dynamics

The second paper introduces Transferable Implicit Transfer Operators (TITO), a transferable surrogate model of molecular dynamics that learns the transition probability distribution directly from simulation data. Unlike conventional MD simulations constrained by femtosecond time steps, TITO models the effective rules of molecular motion across arbitrary lag times, accelerating sampling by up to four orders of magnitude while preserving physical accuracy.

As a surrogate model, TITO replaces explicit numerical integration with learned transition probabilities. The framework employs continuous normalizing flows with equivariant flow matching to parametrize these distributions, capturing how atomic configurations evolve over time without step-by-step propagation. Trained on MD trajectories of small organic molecules and tetrapeptides, TITO learns transferable transition statistics that generalize across both chemical composition and temporal scale. This surrogate approach maintains consistency with the underlying stochastic process, preserving critical statistical properties including Boltzmann equilibrium, Markovianity, and relaxation dynamics.

We benchmark TITO's performance as a dynamics surrogate across thermodynamic and kinetic observables. The model quantitatively reproduces Boltzmann distributions and relaxation timescales for unseen molecules and peptides, matching or exceeding the conformational coverage achieved by regular MD. Remarkably, although trained only on nanosecond-scale data, this transferable surrogate correctly identifies metastable states and predicts exchange timescales on the microsecond scale—states that remain inaccessible to conventional MD within practical computational limits. The model also demonstrates qualitative transferability to peptides twice the size of training systems through simple physical scaling corrections.

TITO's transferable surrogate framework offers distinct advantages: explicit control over the accuracy-efficiency trade-off, generalization across molecular systems and sizes, and the ability to generate milliseconds of simulation time per day on a single GPU. However, current limitations include restriction to implicit solvent, system sizes below a few hundred atoms, and degraded performance for very large extrapolations. Nevertheless, TITO establishes a new paradigm by learning transferable transition operators directly, providing a plausible pathway to bridge the persistent gap between atomistic resolution and experimentally relevant timescales.

9.3 Boltzmann priors for Implicit Transfer Operators

The third paper introduces Boltzmann Priors for Implicit Transfer Operator (BoPITO) learning, combining Boltzmann Generators with Implicit Transfer Operators. BoPITO presents a framework that enhances long-term dynamics data efficiency of deep generative surrogate models for molecular dynamics. While ITO shows promise in accelerating MD simulations by learning transition probability densities at multiple time resolutions, it requires extensive unbiased simulation data, which is scarce for long-term dynamics. BoPITO addresses this limitation by leveraging pre-trained Boltzmann Generators as priors, achieving an order of magnitude reduction in required training data while guaranteeing asymptotically unbiased equilibrium statistics.

BoPITO enhances ITO learning through four key mechanisms. First, it uses Boltzmann Generators to efficiently initialize MD trajectories, ensuring broad sampling across configuration space proportional to the equilibrium distribution. Second, it separates the equilibrium contribution from time-dependent components in the learned transition density, fixing the stationary part. Third, it embeds an inductive bias that asymptotically samples from the equilibrium model for long time horizons, ensuring convergence to the correct Boltzmann distribution. Fourth, it enables tunable sampling protocols that interpolate between models trained on off-equilibrium data and unbiased equilibrium distributions.

The framework is implemented using score-based diffusion models, where the score function separates, equilibrium and dynamical contributions,

$$s_\theta(\mathbf{x}^{t_{\text{diff}}}_{t+N\tau}, \mathbf{x}_t, N, t_{\text{diff}}) = s_{\text{eq}}(\mathbf{x}^{t_{\text{diff}}}, t_{\text{diff}}) + \hat{\lambda}^N s_{\text{dyn}}(\mathbf{x}^{t_{\text{diff}}}_{t+N\tau}, \mathbf{x}_t, N, t_{\text{diff}}),$$

where $s_{\text{eq}}(\mathbf{x}^t_{\text{diff}}, t_{\text{diff}})$ is the score of a pre-trained surrogate of the equilibrium distribution model, $0 < \hat{\lambda} < 1$ is a hyper-parameter and $s_{\text{dyn}}(\mathbf{x}^{t_{\text{diff}}}_{t+N\tau}, \mathbf{x}_t, N, t_{\text{diff}}, \boldsymbol{\theta})$ accounts for the time-dependent components. This principled factorization corresponds to the spectral decomposition of the transfer operator, with the first eigenfunction (equilibrium density) treated as known and the remaining dynamical components learned from data. The hyperparameter $\hat{\lambda}$ controls the timescale at which the model transitions to equilibrium sampling.

We benchmark BoPITO across systems of increasing complexity: the Prinz potential, alanine dipeptide, and chignolin protein. Results demonstrate that BoPITO models achieve higher accuracy for long-term dynamics compared to standard ITO when training data is scarce. Furthermore, we introduce BoPITO interpolators that can recover approximate dynamics from off-equilibrium simulation data by framing interpolation parameter selection as an inverse problem constrained by unbiased experimental observables.

BoPITO represents the first method enabling integration of multiple information sources—off-equilibrium simulations, biased enhanced sampling data, and experimental measurements—into deep generative surrogates of molecular dynamics. Current limitations include the need for careful hyperparameter tuning and lack of Chapman-Kolmogorov guarantees for biased training data.

Moreover, the current formulation does not address chemically transferable models. Nevertheless, BoPITO establishes a principled approach to embedding prior knowledge of stationary distributions as inductive biases for learning long-term dynamical behavior in molecular systems.

9.4 *De Novo* Drug Design Using Reinforcement Learning with Graph-Based Deep Generative Models

In the fourth article, we propose a new training scheme to fine-tune graph-based DGMs for *de novo* molecular design tasks. We show how our computational framework can successfully guide a pre-trained generative model toward the generation of molecules with a specific property profile, even when such molecules are not present in the training set and unlikely to be generated by the pre-trained model. We explored the following tasks: generating molecules of decreasing/increasing size, increasing drug-likeness, and increasing bioactivity.

We use GraphINVENT [15] as a graph-based molecular DGM. GraphINVENT is based on a Gated Graph Neural Network (GGNN) that generates molecules by iteratively sampling actions that build upon an input graph. The action space is divided into three possible actions: add atom, add bond, and terminate graph. The model is trained by minimizing the Kullback-Leibler divergence between target and predicted action probability distributions (APDs).

Our Reinforcement Learning framework uses a memory-aware loss that keeps track of the best agent so far and is updated every few learning steps. By doing so, we remind the current agent of sets of actions that can lead to high-scoring compounds, in turn accelerating and improving agent learning. The scoring model is designed for each specific optimization task and can be based on simple rules or more complex models such as QSAR models.

We tested our framework by fine-tuning a pre-trained graph-based DGM to favor property profiles relevant to drug design, including increasing pharmacological activity. We model bioactivity using a QSAR model for dopamine receptor type D2 (DRD2) activity. Optimization for DRD2 activity is a widely used *de novo* design bioactivity benchmark and allows us to easily compare to previous work. We achieve models that generate diverse compounds with predicted DRD2 activity for 97 % of sampled molecules, outperforming previously reported graph-based methods on this metric.

Our contribution is an important stepping stone toward the design of more advanced molecular DGMs which will allow scientists to efficiently traverse the chemical space in search of promising molecules. We believe the use of DGMs in fields such as drug design has the potential to help chemists come up with new ideas and accelerate the complex process of molecular discovery.

Chapter 10

Discussion and Future Work

This thesis investigates how deep generative models can address fundamental challenges in drug discovery by learning to sample from complex molecular probability distributions. It comprises four papers exploring complementary facets of this problem: three focusing on molecular dynamics and one on molecular design.

The first paper introduces Surrogate Model-Assisted Molecular Dynamics (SMA-MD), which combines a DGM with statistical reweighting and short MD simulations to efficiently sample Boltzmann ensembles of small molecules, producing more diverse and lower-energy configurations than conventional simulations. The second paper presents Transferable Implicit Transfer Operators (TITO), a transferable generative surrogate that learns time-integrated molecular dynamics directly from data, enabling propagation at arbitrarily large time steps with up to four orders of magnitude acceleration while maintaining thermodynamic and kinetic fidelity. The third paper introduces Boltzmann Priors for Implicit Transfer Operator learning (BoPITO), which integrates equilibrium knowledge into deep generative surrogates of molecular dynamics, improving data efficiency by an order of magnitude and enabling interpolation between models trained on biased off-equilibrium data and the equilibrium distribution. Finally, the fourth paper develops a reinforcement learning scheme to fine-tune graph-based DGMs for *de novo* molecular design, guiding models toward molecules with desired properties even when such examples are rare or absent in the training data.

These contributions constitute important stepping stones toward the automation of the drug discovery process. They demonstrate the potential of DGMs to sample molecular distributions with different applications: accelerating molecular dynamics, the prediction of molecular properties, and optimizing molecules for multiple design criteria. At the same time, they highlight key challenges and open directions for future research.

First, Boltzmann emulators—such as SMA-MD—illustrate a growing trend in generative molecular dynamics to avoid fully atomistic representations of

molecular systems. Reduced representations accelerate both sampling and likelihood evaluation. However, these approximations preclude exact reweighting with respect to the Boltzmann distribution and neglect the entropic contributions of the omitted degrees of freedom. Ideally, generative molecular dynamics methods would generate fully atomistic conformations. Nevertheless, in some contexts, sacrificing exact reweighting in exchange for improved sampling efficiency can be justified. Striking a balance between representational fidelity and computational tractability remains a central design consideration.

Second, while TITO demonstrates qualitative transferability to peptides twice the size of training systems, and SMA-MD generalizes across small organic molecules, systematic transferability across diverse chemical compositions and system sizes remains elusive. Extending these frameworks to larger biomolecular systems—such as proteins, nucleic acids, or explicitly solvated drug-like molecules—will require careful treatment of periodic boundary conditions [136], [137], innovations in neural architectures [116], [117], [138], hierarchical modeling strategies [139], [140], and potentially coarse-graining [141], [142], [143], [144] approaches that preserve thermodynamic and kinetic consistency.

Equally important is the careful curation of training datasets to ensure adequate coverage of relevant regions of chemical space. Active learning techniques [145], [146] are likely to play a central role in guiding targeted data acquisition. Furthermore, achieving thermodynamic transferability across temperatures, pressures, and solvation conditions—aking to the reweighting strategies employed in transition-based reweighting analysis (TRAM) methods [147], [148], [149] within the Markov state models framework—would enable the study of perturbations and phase transitions, unlocking new opportunities in materials science and biophysics. Ultimately, the widespread adoption of these machine-learning-based molecular simulation frameworks will depend on their generalization capability, the diversity of available training data, and the accessibility of efficient, open-source implementations—much as GROMACS [150], [151] and OpenMM [152], [153] have become foundational platforms for traditional Molecular Dynamics.

Third, BoPITO demonstrates that incorporating equilibrium priors into dynamical models improves both data efficiency and long-term accuracy. The framework provides a natural mechanism for combining enhanced sampling techniques—potentially augmented with machine-learned collective variables [154], [155], [156]—with short unbiased simulations. Furthermore, by integrating experimental observables, BoPITO enables limited temporal extrapolation beyond transition statistics in the training data. These findings suggest that multiple information sources—unbiased and biased simulations, experimental data, and physics-based priors—can be systematically combined within generative frameworks. Future work should explore principled strategies for such integration while ensuring key properties such as detailed balance and Chapman–Kolmogorov consistency. Establishing these guarantees would strengthen the theoretical foundations of learned dynamical models and improve their reliability for quantitative prediction. Currently, BoPITO is not transferable across molecular systems, which limits its applicability and developing transferable models equipped with equilibrium priors remains an open problem.

Fourth, the reinforcement learning framework introduced in the fourth paper effectively steers graph-based DGMs toward molecules with desired properties, achieving a predicted activity of 97% on the dopamine receptor D2 benchmark. Nonetheless, the QSAR model used for both fine-tuning and evaluation was based exclusively on graph-level representations, disregarding three-dimensional structure and binding geometry. Since *de novo* molecular design is fundamentally a three-dimensional problem, approaches that generate molecules conditioned on binding pockets [138], [157], [158], [159] while explicitly optimizing target interactions constitute a promising avenue for future research. In addition, synthesizability was addressed here as an optimization objective, whereas recent work advocates a more integrated paradigm—designing molecules that are inherently synthesizable by construction [160], [161].

Finally, a natural extension of this work is using MD acceleration methods developed in this thesis to predict molecular properties in a close loop to molecular design framework. Most current design methods either optimize molecules based on single conformations or simple descriptors, ignoring conformational flexibility and the ensemble nature of molecular behavior, or rely on slow optimization where molecular scoring is the main bottleneck. Using MD surrogates within the molecular design loop could allow for fast molecular optimization under a high-accuracy property estimation signal. Furthermore, one could integrate conformational sampling with molecular design. Developing frameworks that jointly optimize molecular graphs and their conformational ensembles—for instance, by combining the reinforcement learning scheme from the fourth paper with MD surrogates from the first three—could enable more reliable property optimization.

In conclusion, this thesis demonstrates that deep generative models offer a viable path toward accelerating molecular property prediction and design by learning flexible, data-driven surrogates of molecular distributions. While significant challenges remain, the foundations established here provide a roadmap for future developments. By continuing to bridge machine learning, statistical mechanics, and computational chemistry, we can work toward a future where the discovery of new therapeutics is both faster and more systematic, ultimately benefiting human health and society.

Bibliography

[1] J. V. Diez, *Sampling From Molecular Unnormalized Distributions With Deep Generative Models Toward the Acceleration of Molecular Design and Conformational Sampling With Deep Learning*. Chalmers Tekniska Hogskola (Sweden), 2024 (cit. on p. ii).

[2] J. A. DiMasi, H. G. Grabowski and R. W. Hansen, "Innovation in the pharmaceutical industry: New estimates of R&D costs," *Journal of Health Economics*, vol. 47, pp. 20–33, 2016, ISSN: 0167-6296. DOI: https://doi.org/10.1016/j.jhealeco.2016.01.012. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167629616000291 (cit. on p. 3).

[3] F. W. Pun, I. V. Ozerov and A. Zhavoronkov, "AI-powered therapeutic target discovery," *Trends in Pharmacological Sciences*, vol. 44, no. 9, pp. 561–572, 2023, ISSN: 0165-6147. DOI: https://doi.org/10.1016/j.tips.2023.06.010. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0165614723001372 (cit. on p. 3).

[4] Y. You et al., "Artificial intelligence in cancer target identification and drug discovery," *Signal Transduction and Targeted Therapy*, vol. 7, no. 1, May 2022, ISSN: 2059-3635. DOI: 10.1038/s41392-022-00994-0. [Online]. Available: http://dx.doi.org/10.1038/s41392-022-00994-0 (cit. on p. 3).

[5] H. H. Loeffler et al., "Reinvent 4: Modern AI–driven generative molecule design," *Journal of Cheminformatics*, vol. 16, no. 1, Feb. 2024, ISSN: 1758-2946. DOI: 10.1186/s13321-024-00812-5. [Online]. Available: http://dx.doi.org/10.1186/s13321-024-00812-5 (cit. on pp. 3, 26).

[6] F. Noé, S. Olsson, J. Köhler and H. Wu, *Boltzmann Generators – Sampling Equilibrium States of Many-Body Systems with Deep Learning*, 2019. arXiv: 1812.01729 [stat.ML] (cit. on pp. 3, 31, 32).

[7] S. Chmiela et al., "Accurate global machine learning force fields for molecules with hundreds of atoms," *Science Advances*, vol. 9, no. 2, Jan. 2023, ISSN: 2375-2548. DOI: 10.1126/sciadv.adf0873. [Online]. Available: http://dx.doi.org/10.1126/sciadv.adf0873 (cit. on p. 3).

[8] P.-C. Kotsias, J. Arús-Pous, H. Chen, O. Engkvist, C. Tyrchan and E. J. Bjerrum, "Direct Steering of de novo Molecular Generation using Descriptor Conditional Recurrent Neural Networks (cRNNs)," Nov. 2019. DOI: 10.26434/chemrxiv.9860906.v2. [Online]. Available: http://dx.doi.org/10.26434/chemrxiv.9860906.v2 (cit. on pp. 3, 19, 23).

[9] Keyvanpour, Mohammad Reza and Shirzad, Mehrnoush Barani, "An Analysis of QSAR Research Based on Machine Learning Concepts," en, *Curr Drug Discov Technol*, vol. 18, no. 1, 17–30, 2021 (cit. on pp. 3, 19, 23).

[10] S. Genheden, A. Thakkar, V. Chadimová, J.-L. Reymond, O. Engkvist and E. Bjerrum, "AiZynthFinder: a fast, robust and flexible open-source software for retrosynthetic planning," *Journal of Cheminformatics*, vol. 12, no. 1, Nov. 2020, ISSN: 1758-2946. DOI: 10.1186/s13321-020-00472-1. [Online]. Available: http://dx.doi.org/10.1186/s13321-020-00472-1 (cit. on p. 3).

[11] R. Qureshi et al., "AI in drug discovery and its clinical relevance," *Heliyon*, vol. 9, no. 7, e17575, Jul. 2023, ISSN: 2405-8440. DOI: 10.1016/j.heliyon.2023.e17575. [Online]. Available: http://dx.doi.org/10.1016/j.heliyon.2023.e17575 (cit. on p. 3).

[12] M. Lu, E. Weinberger, C. Kim and S.-I. Lee, *CellCLIP – Learning Perturbation Effects in Cell Painting via Text-Guided Contrastive Learning*, 2025. arXiv: 2506.06290 [cs.LG]. [Online]. Available: https://arxiv.org/abs/2506.06290 (cit. on p. 3).

[13] Z. Navidi et al., "MorphoDiff: Cellular Morphology Painting with Diffusion Models," Dec. 2024. DOI: 10.1101/2024.12.19.629451. [Online]. Available: http://dx.doi.org/10.1101/2024.12.19.629451 (cit. on p. 3).

[14] M. Olivecrona, T. Blaschke, O. Engkvist and H. Chen, "Molecular de-novo design through deep reinforcement learning," *Journal of Cheminformatics*, vol. 9, no. 1, Sep. 2017, ISSN: 1758-2946. DOI: 10.1186/s13321-017-0235-x. [Online]. Available: http://dx.doi.org/10.1186/s13321-017-0235-x (cit. on pp. 3, 26, 34).

[15] R. Mercado et al., "Graph Networks for Molecular Design," Aug. 2020. DOI: 10.26434/chemrxiv.12843137.v1. [Online]. Available: http://dx.doi.org/10.26434/chemrxiv.12843137.v1 (cit. on pp. 3, 26, 34, 40).

[16] C. Zeni et al., *MatterGen: a generative model for inorganic materials design*, 2024. arXiv: 2312.03687 [cond-mat.mtrl-sci]. [Online]. Available: https://arxiv.org/abs/2312.03687 (cit. on p. 3).

[17] T. Xie, X. Fu, O.-E. Ganea, R. Barzilay and T. Jaakkola, *Crystal Diffusion Variational Autoencoder for Periodic Material Generation*, 2022. arXiv: 2110.06197 [cs.LG]. [Online]. Available: https://arxiv.org/abs/2110.06197 (cit. on p. 3).

[18] P. Dhariwal and A. Nichol, *Diffusion Models Beat GANs on Image Synthesis*, 2021. arXiv: 2105.05233 [cs.LG]. [Online]. Available: https://arxiv.org/abs/2105.05233 (cit. on p. 3).

[19] J. Ho and T. Salimans, *Classifier-Free Diffusion Guidance*, 2022. arXiv: 2207.12598 [cs.LG]. [Online]. Available: https://arxiv.org/abs/2207.12598 (cit. on p. 3).

[20] S. A. Hollingsworth and R. O. Dror, "Molecular Dynamics Simulation for All," *Neuron*, vol. 99, no. 6, 1129–1143, Sep. 2018, ISSN: 0896-6273. DOI: 10.1016/j.neuron.2018.08.011. [Online]. Available: http://dx.doi.org/10.1016/j.neuron.2018.08.011 (cit. on p. 3).

[21] J. D. Durrant and J. A. McCammon, "Molecular dynamics simulations and drug discovery," *BMC Biology*, vol. 9, no. 1, Oct. 2011, ISSN: 1741-7007. DOI: 10.1186/1741-7007-9-71. [Online]. Available: http://dx.doi.org/10.1186/1741-7007-9-71 (cit. on p. 3).

[22] D. van Ravenzwaaij, P. Cassey and S. D. Brown, "A simple introduction to Markov Chain Monte–Carlo sampling," *Psychonomic Bulletin & Review*, vol. 25, no. 1, 143–154, Mar. 2016, ISSN: 1531-5320. DOI: 10.3758/s13423-016-1015-8. [Online]. Available: http://dx.doi.org/10.3758/s13423-016-1015-8 (cit. on pp. 3, 21).

[23] G. Casella, C. P. Robert and M. T. Wells, "Generalized Accept-Reject sampling schemes," in *A Festschrift for Herman Rubin*. Institute of Mathematical Statistics, 2004, 342–347. DOI: 10.1214/lnms/1196285403. [Online]. Available: http://dx.doi.org/10.1214/lnms/1196285403 (cit. on pp. 3, 21).

[24] W. K. Hastings, "Monte Carlo sampling methods using Markov chains and their applications," *Biometrika*, vol. 57, no. 1, 97–109, Apr. 1970, ISSN: 0006-3444. DOI: 10.1093/biomet/57.1.97. [Online]. Available: http://dx.doi.org/10.1093/biomet/57.1.97 (cit. on pp. 3, 21).

[25] J. H. Holland, *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*. The MIT Press, Apr. 1992, ISBN: 9780262275552. DOI: 10.7551/mitpress/1090.001.0001. [Online]. Available: http://dx.doi.org/10.7551/mitpress/1090.001.0001 (cit. on pp. 3, 24).

[26] J. Viguera Diez, S. Romeo Atance, O. Engkvist and S. Olsson, "Generation of conformational ensembles of small molecules via Surrogate Model-Assisted Molecular Dynamics," Nov. 2023. DOI: 10.26434/chemrxiv-2023-sx61w. [Online]. Available: http://dx.doi.org/10.26434/chemrxiv-2023-sx61w (cit. on pp. 4, 32).

[27] J. V. Diez, M. Schreiner and S. Olsson, *Transferable Generative Models Bridge Femtosecond to Nanosecond Time-Step Molecular Dynamics*, 2025. arXiv: 2510.07589 [physics.chem-ph]. [Online]. Available: https://arxiv.org/abs/2510.07589 (cit. on p. 4).

[28] J. V. Diez, M. Schreiner, O. Engkvist and S. Olsson, *Boltzmann priors for Implicit Transfer Operators*, 2025. arXiv: 2410.10605 [physics.chem-ph]. [Online]. Available: https://arxiv.org/abs/2410.10605 (cit. on p. 4).

[29] S. R. Atance, J. V. Diez, O. Engkvist, S. Olsson and R. Mercado, "De Novo Drug Design Using Reinforcement Learning with Graph-Based Deep Generative Models," *Journal of Chemical Information and Modeling*, vol. 62, no. 20, 4863–4872, Oct. 2022, ISSN: 1549-960X. DOI: 10.1021/acs.jcim.2c00838. [Online]. Available: http://dx.doi.org/10.1021/acs.jcim.2c00838 (cit. on p. 4).

[30] J. Hughes, S Rees, S. Kalindjian and K. Philpott, "Principles of early drug discovery," *British Journal of Pharmacology*, vol. 162, no. 6, 1239–1249, Feb. 2011, ISSN: 1476-5381. DOI: 10.1111/j.1476-5381.2010.01127.x. [Online]. Available: http://dx.doi.org/10.1111/j.1476-5381.2010.01127.x (cit. on p. 5).

[31] Y. Tabana, D. Babu, R. Fahlman, A. G. Siraki and K. Barakat, "Target identification of small molecules: an overview of the current applications in drug discovery," *BMC Biotechnology*, vol. 23, no. 1, Oct. 2023, ISSN: 1472-6750. DOI: 10.1186/s12896-023-00815-4. [Online]. Available: http://dx.doi.org/10.1186/s12896-023-00815-4 (cit. on p. 5).

[32] V. Pattan, R. Kashyap, V. Bansal, N. Candula, T. Koritala and S. Surani, "Genomics in medicine: A new era in medicine," *World Journal of Methodology*, vol. 11, no. 5, 231–242, Sep. 2021, ISSN: 2222-0682. DOI: 10.5662/wjm.v11.i5.231. [Online]. Available: http://dx.doi.org/10.5662/wjm.v11.i5.231 (cit. on p. 5).

[33] S. Al-Amrani, Z. Al-Jabri, A. Al-Zaabi, J. Alshekaili and M. Al-Khabori, "Proteomics: Concepts and applications in human medicine," *World Journal of Biological Chemistry*, vol. 12, no. 5, 57–69, Sep. 2021, ISSN: 1949-8454. DOI: 10.4331/wjbc.v12.i5.57. [Online]. Available: http://dx.doi.org/10.4331/wjbc.v12.i5.57 (cit. on p. 5).

[34] P. Wah Tang et al., "A Review of Gene Knockout Strategies for Microbial Cells," *Recent Patents on Biotechnology*, vol. 9, no. 3, 176–197, Jun. 2016, ISSN: 1872-2083. DOI: 10.2174/1872208310666160517115047. [Online]. Available: http://dx.doi.org/10.2174/1872208310666160517115047 (cit. on p. 5).

[35] A. Lavecchia and C. Giovanni, "Virtual Screening Strategies in Drug Discovery: A Critical Review," *Current Medicinal Chemistry*, vol. 20, no. 23, 2839–2860, Jun. 2013, ISSN: 0929-8673. DOI: 10.2174/09298673113209990001. [Online]. Available: http://dx.doi.org/10.2174/092986731132099900001 (cit. on p. 5).

[36] V. D. Mouchlis et al., "Advances in De Novo Drug Design: From Conventional to Machine Learning Methods," *International Journal of Molecular Sciences*, vol. 22, no. 4, p. 1676, Feb. 2021, ISSN: 1422-0067. DOI: 10.3390/ijms22041676. [Online]. Available: http://dx.doi.org/10.3390/ijms22041676 (cit. on p. 5).

[37] U. Muhammad, A. Uzairu and D. Ebuka Arthur, "Review on: quantitative structure activity relationship (QSAR) modeling," *Journal of Analytical & Pharmaceutical Research*, vol. 7, no. 2, Apr. 2018, ISSN: 2473-0831. DOI: 10.15406/japlr.2018.07.00232. [Online]. Available: http://dx.doi.org/10.15406/japlr.2018.07.00232 (cit. on p. 6).

[38] P. Gramatica, "Principles of QSAR models validation: internal and external," *QSAR & Combinatorial Science*, vol. 26, no. 5, 694–701, May 2007, ISSN: 1611-0218. DOI: 10.1002/qsar.200610151. [Online]. Available: http://dx.doi.org/10.1002/qsar.200610151 (cit. on p. 6).

[39] A. T. Plowright, C. Johnstone, J. Kihlberg, J. Pettersson, G. Robb and R. A. Thompson, "Hypothesis driven drug design: improving quality and effectiveness of the design-make-test-analyse cycle," *Drug Discovery Today*, vol. 17, no. 1–2, 56–62, Jan. 2012, ISSN: 1359-6446. DOI: 10.1016/j.drudis.2011.09.012. [Online]. Available: http://dx.doi.org/10.1016/j.drudis.2011.09.012 (cit. on p. 6).

[40] S. S. Wesolowski and D. G. Brown, *The Strategies and Politics of Successful Design, Make, Test, and Analyze (DMTA) Cycles in Lead Generation*, Mar. 2016. DOI: 10.1002/9783527677047.ch17. [Online]. Available: http://dx.doi.org/10.1002/9783527677047.ch17 (cit. on p. 6).

[41] P. G. Polishchuk, T. I. Madzhidov and A. Varnek, "Estimation of the size of drug-like chemical space based on GDB-17 data," *Journal of Computer-Aided Molecular Design*, vol. 27, no. 8, 675–679, Aug. 2013, ISSN: 1573-4951. DOI: 10.1007/s10822-013-9672-4. [Online]. Available: http://dx.doi.org/10.1007/s10822-013-9672-4 (cit. on p. 8).

[42] L. Seep, A. Bonin, K. Meier, H. Diedam and A. H. Göller, "Ensemble completeness in conformer sampling: the case of small macrocycles," *Journal of Cheminformatics*, vol. 13, no. 1, Jul. 2021, ISSN: 1758-2946. DOI: 10.1186/s13321-021-00524-0. [Online]. Available: http://dx.doi.org/10.1186/s13321-021-00524-0 (cit. on p. 11).

[43] D. Weininger, "SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules," *Journal of Chemical Information and Computer Sciences*, vol. 28, no. 1, 31–36, Feb. 1988, ISSN: 1520-5142. DOI: 10.1021/ci00057a005. [Online]. Available: http://dx.doi.org/10.1021/ci00057a005 (cit. on p. 12).

[44] D. E. Rumelhart, G. E. Hinton and R. J. Williams, "Learning internal representations by error propagation," in *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Vol. 1: Foundations*, D. E. Rumelhart and J. L. McClelland, Eds., Cambridge, MA: MIT Press, 1986, pp. 318–362 (cit. on p. 12).

[45] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, 1735–1780, Nov. 1997, ISSN: 1530-888X. DOI: 10.1162/neco.1997.9.8.1735. [Online]. Available: http://dx.doi.org/10.1162/neco.1997.9.8.1735 (cit. on p. 12).

[46] A. Vaswani et al., *Attention Is All You Need*, 2023. arXiv: 1706.03762 [cs.CL]. [Online]. Available: https://arxiv.org/abs/1706.03762 (cit. on pp. 12, 25).

[47] A. Radford, K. Narasimhan, T. Salimans and I. Sutskever, "Improving language understanding by generative pre-training," 2018. [Online]. Available: https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf (cit. on pp. 12, 25).

[48] X.-C. Zhang et al., "Pushing the Boundaries of Molecular Property Prediction for Drug Discovery with Multitask Learning BERT Enhanced by SMILES Enumeration," *Research*, vol. 2022, Jan. 2022, ISSN: 2639-5274. DOI: 10.34133/research.0004. [Online]. Available: http://dx.doi.org/10.34133/research.0004 (cit. on p. 13).

[49] T. Blaschke et al., "REINVENT 2.0: An AI Tool for De Novo Drug Design," *Journal of Chemical Information and Modeling*, vol. 60, no. 12, 5918–5922, Oct. 2020, ISSN: 1549-960X. DOI: 10.1021/acs.jcim.0c00915. [Online]. Available: http://dx.doi.org/10.1021/acs.jcim.0c00915 (cit. on p. 13).

[50] J. Zhou et al., *Graph Neural Networks: A Review of Methods and Applications*, 2021. arXiv: 1812.08434 [cs.LG] (cit. on p. 13).

[51] B. Khemani, S. Patil, K. Kotecha and S. Tanwar, "A review of graph neural networks: concepts, architectures, techniques, challenges, datasets, applications, and future directions," *Journal of Big Data*, vol. 11, no. 1, Jan. 2024, ISSN: 2196-1115. DOI: 10.1186/s40537-023-00876-4. [Online]. Available: http://dx.doi.org/10.1186/s40537-023-00876-4 (cit. on p. 13).

[52] V. G. Satorras, E. Hoogeboom, F. B. Fuchs, I. Posner and M. Welling, *E(n) Equivariant Normalizing Flows*, 2022. arXiv: 2105.09016 [cs.LG] (cit. on pp. 13, 34).

[53] M. Geiger and T. Smidt, *e3nn: Euclidean Neural Networks*, 2022. arXiv: 2207.09453 [cs.LG] (cit. on pp. 13, 14, 34).

[54] K. T. Schütt, P.-J. Kindermans, H. E. Sauceda, S. Chmiela, A. Tkatchenko and K.-R. Müller, *SchNet: A continuous-filter convolutional neural network for modeling quantum interactions*, 2017. arXiv: 1706.08566 [stat.ML] (cit. on p. 13).

[55] T. S. Cohen and M. Welling, *Group Equivariant Convolutional Networks*, 2016. arXiv: 1602.07576 [cs.LG] (cit. on p. 13).

[56] S. L. Batzner et al., "E(3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials," *Nature Communications*, vol. 13, 2021. [Online]. Available: https://api.semanticscholar.org/CorpusID:231418897 (cit. on p. 14).

[57] P. Langevin, "Sur la théorie du mouvement brownien," *C. R. Acad. Sci. (Paris)*, vol. 146, 530—533, 1908 (cit. on p. 15).

[58] N. Wiener, "Differential-Space," *Journal of Mathematics and Physics*, vol. 2, no. 1–4, 131–174, Oct. 1923, ISSN: 0097-1421. DOI: 10.1002/sapm192321131. [Online]. Available: http://dx.doi.org/10.1002/sapm192321131 (cit. on p. 15).

[59] D. T. Limmer, *Statistical Mechanics and Stochastic Thermodynamics: A Textbook on Modern Approaches in and out of Equilibrium*. Oxford University Press, 2024, ch. 7.8, Oxford Graduate Texts, ISBN: 9780198919858. [Online]. Available: https://global.oup.com/academic/product/statistical-mechanics-and-stochastic-thermodynamics-9780198919858 (cit. on p. 15).

[60] A. D. Fokker, "Die mittlere Energie rotierender elektrischer Dipole im Strahlungsfeld," *Annalen der Physik*, vol. 348, no. 5, 810–820, Jan. 1914, ISSN: 1521-3889. DOI: 10.1002/andp.19143480507. [Online]. Available: http://dx.doi.org/10.1002/andp.19143480507 (cit. on p. 16).

[61] M. Planck, "Über einen satz der statistischen dynamik und seine erweiterung in der quantentheorie," German, *Sitzungsberichte der Preussischen Akademie der Wissenschaften zu Berlin*, vol. 24, pp. 324–341, 1917. [Online]. Available: https://www.biodiversitylibrary.org/page/29213319#page/364/mode/1up (cit. on p. 16).

[62] H. Kramers, "Brownian motion in a field of force and the diffusion model of chemical reactions," *Physica*, vol. 7, no. 4, 284–304, Apr. 1940, ISSN: 0031-8914. DOI: 10.1016/s0031-8914(40)90098-2. [Online]. Available: http://dx.doi.org/10.1016/S0031-8914(40)90098-2 (cit. on p. 16).

[63] M. von Smoluchowski, "Zur kinetischen Theorie der Brownschen Molekularbewegung und der Suspensionen," *Annalen der Physik*, vol. 326, no. 14, 756–780, Jan. 1906, ISSN: 1521-3889. DOI: 10.1002/andp.19063261405. [Online]. Available: http://dx.doi.org/10.1002/andp.19063261405 (cit. on p. 16).

[64] K. Sharp and F. Matschinsky, "Translation of Ludwig Boltzmann's Paper "On the Relationship between the Second Fundamental Theorem of the Mechanical Theory of Heat and Probability Calculations Regarding the Conditions for Thermal Equilibrium" Sitzungberichte der Kaiserlichen Akademie der Wissenschaften. Mathematisch-Naturwissen Classe. Abt. II, LXXVI 1877, pp 373-435 (Wien. Ber. 1877, 76:373-435). Reprinted in Wiss. Abhandlungen, Vol. II, reprint 42, p. 164-223, Barth, Leipzig, 1909," *Entropy*, vol. 17, no. 4, 1971–2009, Apr. 2015, ISSN: 1099-4300. DOI: 10.3390/e17041971. [Online]. Available: http://dx.doi.org/10.3390/e17041971 (cit. on p. 17).

[65] J. C. Maxwell, "II. Illustrations of the dynamical theory of gases," *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, vol. 20, no. 130, 21–37, Jul. 1860, ISSN: 1941-5990. DOI: 10.1080/14786446008642902. [Online]. Available: http://dx.doi.org/10.1080/14786446008642902 (cit. on p. 17).

[66] O. Perron, "Zur Theorie der Matrices," *Mathematische Annalen*, vol. 64, no. 2, 248–263, Jun. 1907, ISSN: 1432-1807. DOI: 10.1007/bf01449896. [Online]. Available: http://dx.doi.org/10.1007/BF01449896 (cit. on p. 17).

[67] G. Frobenius, "Über matrizen aus nicht negativen elementen," in *Sitzungsberichte der Königlich Preußischen Akademie der Wissenschaften : Jahrgang 1912 ; Erster Halbband Januar bis Juni*, 2025 (cit. on p. 17).

[68] J.-H. Prinz et al., "Markov models of molecular kinetics: Generation and validation," *The Journal of Chemical Physics*, vol. 134, no. 17, May 2011, ISSN: 1089-7690. DOI: 10.1063/1.3565032. [Online]. Available: http://dx.doi.org/10.1063/1.3565032 (cit. on p. 18).

[69] Z. Cournia, C. Chipot, B. Roux, D. M. York and W. Sherman, "Free Energy Methods in Drug Discovery—Introduction," in *ACS Symposium Series*. American Chemical Society, Nov. 2021, 1–38, ISBN: 9780841298057. DOI: 10.1021/bk-2021-1397.ch001. [Online]. Available: http://dx.doi.org/10.1021/bk-2021-1397.ch001 (cit. on p. 19).

[70] N. Schapin, M. Majewski, A. Varela-Rial, C. Arroniz and G. D. Fabritiis, "Machine learning small molecule properties in drug discovery," *Artificial Intelligence Chemistry*, vol. 1, no. 2, p. 100 020, Dec. 2023, ISSN: 2949-7477. DOI: 10.1016/j.aichem.2023.100020. [Online]. Available: http://dx.doi.org/10.1016/j.aichem.2023.100020 (cit. on p. 19).

[71] Z. Wu et al., "MoleculeNet: a benchmark for molecular machine learning," *Chemical Science*, vol. 9, no. 2, 513–530, 2018, ISSN: 2041-6539. DOI: 10.1039/c7sc02664a. [Online]. Available: http://dx.doi.org/10.1039/C7SC02664A (cit. on p. 19).

[72] S. Olsson, *Markov State Models of Protein–Protein Encounters*, Nov. 2022. DOI: 10.1002/9783527830503.ch9. [Online]. Available: http://dx.doi.org/10.1002/9783527830503.ch9 (cit. on p. 19).

[73] Y. I. Yang, Q. Shao, J. Zhang, L. Yang and Y. Q. Gao, "Enhanced sampling in molecular dynamics," *The Journal of Chemical Physics*, vol. 151, no. 7, Aug. 2019, ISSN: 1089-7690. DOI: 10.1063/1.5109531. [Online]. Available: http://dx.doi.org/10.1063/1.5109531 (cit. on p. 21).

[74] Earl, David J. and Deem, Michael W., "Parallel tempering: Theory, applications, and new perspectives," *Phys. Chem. Chem. Phys.*, vol. 7, pp. 3910–3916, 23 2005. DOI: 10.1039/B509983H. [Online]. Available: http://dx.doi.org/10.1039/B509983H (cit. on p. 21).

[75] D. Sidler, A. Schwaninger and S. Riniker, "Replica exchange enveloping distribution sampling (RE-EDS): A robust method to estimate multiple free-energy differences from a single simulation," *The Journal of Chemical Physics*, vol. 145, no. 15, Oct. 2016, ISSN: 1089-7690. DOI: 10.1063/1.4964781. [Online]. Available: http://dx.doi.org/10.1063/1.4964781 (cit. on p. 21).

[76] A. P. Pasarkar, G. M. Bencomo, S. Olsson and A. B. Dieng, "Vendi sampling for molecular simulations: Diversity as a force for faster convergence and better exploration.," eng, *The Journal of chemical physics*, vol. 159, no. 14, Oct. 2023, ISSN: 1089-7690 (Electronic). DOI: 10.1063/5.0166172 (cit. on p. 21).

[77] H. Grubmüller, "Predicting slow structural transitions in macromolecular systems: Conformational flooding," *Phys. Rev. E*, vol. 52, pp. 2893–2906, 3 Sep. 1995. DOI: 10.1103/PhysRevE.52.2893. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevE.52.2893 (cit. on p. 21).

[78] A. Laio and M. Parrinello, "Escaping free-energy minima," *Proceedings of the National Academy of Sciences*, vol. 99, no. 20, pp. 12 562–12 566, 2002 (cit. on p. 21).

[79] G. Torrie and J. Valleau, "Nonphysical sampling distributions in Monte Carlo free-energy estimation: Umbrella sampling," *Journal of Computational Physics*, vol. 23, no. 2, pp. 187–199, Feb. 1977. DOI: 10.1016/0021-9991(77)90121-8. [Online]. Available: https://doi.org/10.1016/0021-9991(77)90121-8 (cit. on p. 21).

[80] R. Liu, X. Li and K. S. Lam, "Combinatorial chemistry in drug discovery," *Current Opinion in Chemical Biology*, vol. 38, 117–126, Jun. 2017, ISSN: 1367-5931. DOI: 10.1016/j.cbpa.2017.03.017. [Online]. Available: http://dx.doi.org/10.1016/j.cbpa.2017.03.017 (cit. on p. 23).

[81] V. Blay, B. Tolani, S. P. Ho and M. R. Arkin, "High-Throughput Screening: today's biochemical and cell-based approaches," *Drug Discovery Today*, vol. 25, no. 10, 1807–1821, Oct. 2020, ISSN: 1359-6446. DOI: 10.1016/j.drudis.2020.07.024. [Online]. Available: http://dx.doi.org/10.1016/j.drudis.2020.07.024 (cit. on p. 23).

[82] W. Tong, W. J. Welsh, L. Shi, H. Fang and R. Perkins, "Structure-activity relationship approaches and applications," *Environmental Toxicology and Chemistry*, vol. 22, no. 8, 1680–1695, Aug. 2003, ISSN: 1552-8618. DOI: 10.1897/01-198. [Online]. Available: http://dx.doi.org/10.1897/01-198 (cit. on p. 23).

[83] Y. Momin and V. Beloshe, "Pharmacophore modeling in drug design," in *Revolutionizing Drug Discovery:Cutting-Edge Computational Techniques*. Elsevier, 2025, 313–324, ISBN: 9780443346491. DOI: 10.1016/bs.apha.2025.01.010. [Online]. Available: http://dx.doi.org/10.1016/bs.apha.2025.01.010 (cit. on p. 24).

[84] A. Lavecchia and C. Giovanni, "Virtual Screening Strategies in Drug Discovery: A Critical Review," *Current Medicinal Chemistry*, vol. 20, no. 23, 2839–2860, Jun. 2013, ISSN: 0929-8673. DOI: 10.2174/09298673113209990001. [Online]. Available: http://dx.doi.org/10.2174/092986731132099900001 (cit. on p. 24).

[85] R. Rombach, A. Blattmann, D. Lorenz, P. Esser and B. Ommer, *High-Resolution Image Synthesis with Latent Diffusion Models*, 2022. arXiv: 2112.10752 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2112.10752 (cit. on p. 25).

[86] P. Esser et al., *Scaling Rectified Flow Transformers for High-Resolution Image Synthesis*, 2024. arXiv: 2403.03206 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2403.03206 (cit. on p. 25).

[87] A. Ramesh et al., *Zero-Shot Text-to-Image Generation*, 2021. arXiv: 2102.12092 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2102.12092 (cit. on p. 25).

[88] J. Ho, T. Salimans, A. Gritsenko, W. Chan, M. Norouzi and D. J. Fleet, *Video Diffusion Models*, 2022. arXiv: 2204.03458 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2204.03458 (cit. on p. 25).

[89] A. Blattmann et al., *Align your Latents: High-Resolution Video Synthesis with Latent Diffusion Models*, 2023. arXiv: 2304.08818 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2304.08818 (cit. on p. 25).

[90] J. Abramson et al., "Accurate structure prediction of biomolecular interactions with AlphaFold 3," *Nature*, vol. 630, no. 8016, 493–500, May 2024, ISSN: 1476-4687. DOI: 10.1038/s41586-024-07487-w. [Online]. Available: http://dx.doi.org/10.1038/s41586-024-07487-w (cit. on p. 25).

[91] J. Wohlwend et al., "Boltz-1 Democratizing Biomolecular Interaction Modeling," Nov. 2024. DOI: 10.1101/2024.11.19.624167. [Online]. Available: http://dx.doi.org/10.1101/2024.11.19.624167 (cit. on p. 25).

[92] G. Ahdritz et al., "OpenFold: Retraining AlphaFold2 yields new insights into its learning mechanisms and capacity for generalization," *bioRxiv*, 2022. DOI: 10.1101/2022.11.20.517210. eprint: https://www.biorxiv.org/content/early/2022/11/22/2022.11.20.517210.full.pdf. [Online]. Available: https://www.biorxiv.org/content/10.1101/2022.11.20.517210 (cit. on p. 25).

[93] Y. Bengio, R. Ducharme and P. Vincent, "A Neural Probabilistic Language Model," in *Advances in Neural Information Processing Systems*, T. Leen, T. Dietterich and V. Tresp, Eds., vol. 13, MIT Press, 2000. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2000/file/728f206c2a01bf572b5940d7d9a8fa4c-Paper.pdf (cit. on p. 25).

[94] A. Graves, *Generating Sequences With Recurrent Neural Networks*, 2014. arXiv: 1308.0850 [cs.NE]. [Online]. Available: https://arxiv.org/abs/1308.0850 (cit. on p. 25).

[95] I. Sutskever, O. Vinyals and Q. V. Le, *Sequence to Sequence Learning with Neural Networks*, 2014. arXiv: 1409.3215 [cs.CL]. [Online]. Available: https://arxiv.org/abs/1409.3215 (cit. on p. 25).

[96] P. Holderrieth and E. Erives, *An Introduction to Flow Matching and Diffusion Models*, Lecture notes for MIT Class 6.S184: Generative AI With Stochastic Differential Equations, Accessed: 2025-11-03, 2025. [Online]. Available: https://diffusion.csail.mit.edu/docs/lecture-notes.pdf (cit. on p. 26).

[97] M. S. Albergo, N. M. Boffi and E. Vanden-Eijnden, *Stochastic Interpolants: A Unifying Framework for Flows and Diffusions*, 2025. arXiv: 2303.08797 [cs.LG]. [Online]. Available: https://arxiv.org/abs/2303.08797 (cit. on p. 26).

[98] W. Grathwohl, R. T. Q. Chen, J. Bettencourt, I. Sutskever and D. Duvenaud, *FFJORD: Free-form Continuous Dynamics for Scalable Reversible Generative Models*, 2018. arXiv: 1810.01367 [cs.LG]. [Online]. Available: https://arxiv.org/abs/1810.01367 (cit. on p. 26).

[99] G. Papamakarios, E. Nalisnick, D. J. Rezende, S. Mohamed and B. Lakshminarayanan, *Normalizing Flows for Probabilistic Modeling and Inference*, 2021. arXiv: 1912.02762 [stat.ML] (cit. on pp. 26, 34).

[100] J. Ho, A. Jain and P. Abbeel, *Denoising Diffusion Probabilistic Models*, 2020. arXiv: 2006.11239 [cs.LG]. [Online]. Available: https://arxiv.org/abs/2006.11239 (cit. on pp. 26, 28).

[101] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon and B. Poole, *Score-Based Generative Modeling through Stochastic Differential Equations*, 2021. arXiv: 2011.13456 [cs.LG] (cit. on pp. 26, 28, 34).

[102] T. Karras, M. Aittala, T. Aila and S. Laine, *Elucidating the Design Space of Diffusion-Based Generative Models*, 2022. arXiv: 2206.00364 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2206.00364 (cit. on p. 26).

[103] Y. Lipman, R. T. Q. Chen, H. Ben-Hamu, M. Nickel and M. Le, *Flow Matching for Generative Modeling*, 2023. arXiv: 2210.02747 [cs.LG]. [Online]. Available: https://arxiv.org/abs/2210.02747 (cit. on p. 27).

[104] B. D. Anderson, "Reverse-time diffusion equation models," *Stochastic Processes and their Applications*, vol. 12, no. 3, 313–326, May 1982, ISSN: 0304-4149. DOI: 10.1016/0304-4149(82)90051-5. [Online]. Available: http://dx.doi.org/10.1016/0304-4149(82)90051-5 (cit. on p. 28).

[105] V. D. Bortoli, E. Mathieu, M. Hutchinson, J. Thornton, Y. W. Teh and A. Doucet, *Riemannian Score-Based Generative Modelling*, 2022. arXiv: 2202.02763 [cs.LG]. [Online]. Available: https://arxiv.org/abs/2202.02763 (cit. on p. 28).

[106] C.-W. Huang, M. Aghajohari, A. J. Bose, P. Panangaden and A. Courville, *Riemannian Diffusion Models*, 2022. arXiv: 2208.07949 [cs.LG]. [Online]. Available: https://arxiv.org/abs/2208.07949 (cit. on p. 28).

[107] B. Jing, G. Corso, J. Chang, R. Barzilay and T. Jaakkola, *Torsional Diffusion for Molecular Conformer Generation*, 2023. arXiv: 2206.01729 [physics.chem-ph] (cit. on pp. 29, 32, 35).

[108] M. Dibak, L. Klein, A. Krämer and F. Noé, *Temperature Steerable Flows and Boltzmann Generators*, 2022. arXiv: 2108.01590 (cit. on p. 32).

[109] H. Wu, J. Köhler and F. Noé, *Stochastic Normalizing Flows*, 2020. arXiv: 2002.06707 [stat.ML]. [Online]. Available: https://arxiv.org/abs/2002.06707 (cit. on p. 32).

[110] J. Köhler, A. Krämer and F. Noé, *Smooth Normalizing Flows*, 2021. arXiv: 2110.00351 [stat.ML]. [Online]. Available: https://arxiv.org/abs/2110.00351 (cit. on p. 32).

[111] C. B. Tan et al., *Amortized Sampling with Transferable Normalizing Flows*, 2025. arXiv: 2508.18175 [cs.LG]. [Online]. Available: https://arxiv.org/abs/2508.18175 (cit. on p. 32).

[112] L. Klein and F. Noé, *Transferable Boltzmann Generators*, 2024. DOI: 10.48550/ARXIV.2406.14426. [Online]. Available: https://arxiv.org/abs/2406.14426 (cit. on p. 32).

[113] L. Klein et al., *Timewarp: Transferable Acceleration of Molecular Dynamics by Learning Time-Coarsened Dynamics*, 2023. arXiv: 2302.01170 [stat.ML]. [Online]. Available: https://arxiv.org/abs/2302.01170 (cit. on pp. 32, 33).

[114] S. Lewis et al., "Scalable emulation of protein equilibrium ensembles with generative deep learning," *Science*, vol. 389, no. 6761, Aug. 2025, ISSN: 1095-9203. DOI: 10.1126/science.adv9817. [Online]. Available: http://dx.doi.org/10.1126/science.adv9817 (cit. on p. 32).

[115] B. Jing, H. Stärk, T. Jaakkola and B. Berger, *Generative Modeling of Molecular Dynamics Trajectories*, 2024. arXiv: 2409.17808 [q-bio.BM]. [Online]. Available: https://arxiv.org/abs/2409.17808 (cit. on p. 33).

[116] A. dos Santos Costa et al., *EquiJump: Protein Dynamics Simulation via SO(3)-Equivariant Stochastic Interpolants*, 2024. arXiv: 2410.09667 [cs.LG]. [Online]. Available: https://arxiv.org/abs/2410.09667 (cit. on pp. 33, 42).

[117] A. dos Santos Costa, M. Ponnapati, D. Rubin, T. Smidt and J. Jacobson, *Accelerating Protein Molecular Dynamics Simulation with DeepJump*, 2025. arXiv: 2509.13294 [q-bio.BM]. [Online]. Available: https://arxiv.org/abs/2509.13294 (cit. on pp. 33, 42).

[118] M. Schreiner, O. Winther and S. Olsson, *Implicit Transfer Operator Learning: Multiple Time-Resolution Surrogates for Molecular Dynamics*, 2023. DOI: 10.48550/ARXIV.2305.18046. [Online]. Available: https://arxiv.org/abs/2305.18046 (cit. on p. 33).

[119] J. Yosinski, J. Clune, Y. Bengio and H. Lipson, *How transferable are features in deep neural networks?* 2014. arXiv: 1411.1792 [cs.LG] (cit. on p. 34).

[120] M. E. Peters, S. Ruder and N. A. Smith, *To Tune or Not to Tune? Adapting Pretrained Representations to Diverse Tasks*, 2019. arXiv: 1903.05987 [cs.CL] (cit. on p. 34).

[121] M. Moret, L. Friedrich, F. Grisoni, D. Merk and G. Schneider, "Generative molecular design in low data regimes," *Nature Machine Intelligence*, vol. 2, no. 3, 171–180, Mar. 2020, ISSN: 2522-5839. DOI: 10.1038/s42256-020-0160-y. [Online]. Available: http://dx.doi.org/10.1038/s42256-020-0160-y (cit. on p. 34).

[122] V. D. Mouchlis et al., "Advances in De Novo Drug Design: From Conventional to Machine Learning Methods," *International Journal of Molecular Sciences*, vol. 22, no. 4, p. 1676, Feb. 2021, ISSN: 1422-0067. DOI: 10.3390/ijms22041676. [Online]. Available: http://dx.doi.org/10.3390/ijms22041676 (cit. on p. 34).

[123] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, 1735–1780, Nov. 1997, ISSN: 1530-888X. DOI: 10.1162/neco.1997.9.8.1735. [Online]. Available: http://dx.doi.org/10.1162/neco.1997.9.8.1735 (cit. on p. 34).

[124] J. Lim, S. Ryu, J. W. Kim and W. Y. Kim, "Molecular generative model based on conditional variational autoencoder for de novo molecular design," *Journal of Cheminformatics*, vol. 10, no. 1, Jul. 2018, ISSN: 1758-2946. DOI: 10.1186/s13321-018-0286-7. [Online]. Available: http://dx.doi.org/10.1186/s13321-018-0286-7 (cit. on p. 34).

[125] D. P. Kingma and M. Welling, *Auto-Encoding Variational Bayes*, 2022. arXiv: 1312.6114 [stat.ML] (cit. on p. 34).

[126] L. Maziarka, A. Pocha, J. Kaczmarczyk, K. Rataj, T. Danel and M. Warchol, "Mol-CycleGAN: a generative model for molecular optimization," *Journal of Cheminformatics*, vol. 12, no. 1, Jan. 2020, ISSN: 1758-2946. DOI: 10.1186/s13321-019-0404-1. [Online]. Available: http://dx.doi.org/10.1186/s13321-019-0404-1 (cit. on p. 34).

[127] I. J. Goodfellow et al., *Generative Adversarial Networks*, 2014. arXiv: 1406.2661 [stat.ML] (cit. on p. 34).

[128] J. Zhou et al., *Graph Neural Networks: A Review of Methods and Applications*, 2021. arXiv: 1812.08434 [cs.LG] (cit. on p. 34).

[129] J. You, B. Liu, R. Ying, V. Pande and J. Leskovec, *Graph Convolutional Policy Network for Goal-Directed Molecular Graph Generation*, 2019. arXiv: 1806.02473 [cs.LG] (cit. on p. 34).

[130] J. Guan, W. W. Qian, X. Peng, Y. Su, J. Peng and J. Ma, *3D Equivariant Diffusion for Target-Aware Molecule Generation and Affinity Prediction*, 2023. arXiv: 2303.03543 [q-bio.BM] (cit. on p. 34).

[131] W. Feng et al., "Generation of 3D molecules in pockets via a language model," *Nature Machine Intelligence*, vol. 6, no. 1, 62–73, Jan. 2024, ISSN: 2522-5839. DOI: 10.1038/s42256-023-00775-6. [Online]. Available: http://dx.doi.org/10.1038/s42256-023-00775-6 (cit. on p. 34).

[132] J. Ho, A. Jain and P. Abbeel, *Denoising Diffusion Probabilistic Models*, 2020. arXiv: 2006.11239 [cs.LG] (cit. on p. 34).

[133] Y. Song, C. Durkan, I. Murray and S. Ermon, *Maximum Likelihood Training of Score-Based Diffusion Models*, 2021. arXiv: 2101.09258 [stat.ML] (cit. on p. 34).

[134] E. Hoogeboom, V. G. Satorras, C. Vignac and M. Welling, *Equivariant Diffusion for Molecule Generation in 3D*, 2022. arXiv: 2203.17003 [cs.LG]. [Online]. Available: https://arxiv.org/abs/2203.17003 (cit. on p. 34).

[135] K. T. Schütt, O. T. Unke and M. Gastegger, *Equivariant message passing for the prediction of tensorial properties and molecular spectra*, 2021. arXiv: 2102.03150 [cs.LG]. [Online]. Available: https://arxiv.org/abs/2102.03150 (cit. on p. 34).

[136] D. Nagel and T. Bereau, "Fokker–Planck Score Learning: Efficient Free-Energy Estimation under Periodic Boundary Conditions," *The Journal of Physical Chemistry B*, Oct. 2025, ISSN: 1520-5207. DOI: 10.1021/acs.jpcb.5c04579. [Online]. Available: http://dx.doi.org/10.1021/acs.jpcb.5c04579 (cit. on p. 42).

[137] A. Havens et al., *Adjoint Sampling: Highly Scalable Diffusion Samplers via Adjoint Matching*, 2025. arXiv: 2504.11713 [cs.LG]. [Online]. Available: https://arxiv.org/abs/2504.11713 (cit. on p. 42).

[138] R. Irwin, A. Tibo, J. P. Janet and S. Olsson, *SemlaFlow – Efficient 3D Molecular Generation with Latent Attention and Equivariant Flow Matching*, 2025. arXiv: 2406.07266 [cs.LG]. [Online]. Available: https://arxiv.org/abs/2406.07266 (cit. on pp. 42, 43).

[139] Y. Zhu et al., *MolHF: A Hierarchical Normalizing Flow for Molecular Graph Generation*, 2023. arXiv: 2305.08457 [cs.LG]. [Online]. Available: https://arxiv.org/abs/2305.08457 (cit. on p. 42).

[140] J. A. Weller and R. Rohs, "Structure-Based Drug Design with a Deep Hierarchical Generative Model," *Journal of Chemical Information and Modeling*, vol. 64, no. 16, 6450–6463, Jul. 2024, ISSN: 1549-960X. DOI: 10.1021/acs.jcim.4c01193. [Online]. Available: http://dx.doi.org/10.1021/acs.jcim.4c01193 (cit. on p. 42).

[141] T. Bereau and M. Deserno, "Generic Coarse-Grained Model for Protein Folding and Aggregation," *Biophysical Journal*, vol. 96, no. 3, 405a, Feb. 2009, ISSN: 0006-3495. DOI: 10.1016/j.bpj.2008.12.2061. [Online]. Available: http://dx.doi.org/10.1016/j.bpj.2008.12.2061 (cit. on p. 42).

[142] M. Arts et al., "Two for One: Diffusion Models and Force Fields for Coarse-Grained Molecular Dynamics," *Journal of Chemical Theory and Computation*, vol. 19, no. 18, 6151–6159, Sep. 2023, ISSN: 1549-9626. DOI: 10.1021/acs.jctc.3c00702. [Online]. Available: http://dx.doi.org/10.1021/acs.jctc.3c00702 (cit. on p. 42).

[143] N. E. Charron et al., "Navigating protein landscapes with a machine-learned transferable coarse-grained model," *Nature Chemistry*, vol. 17, no. 8, 1284–1292, Jul. 2025, ISSN: 1755-4349. DOI: 10.1038/s41557-025-01874-0. [Online]. Available: http://dx.doi.org/10.1038/s41557-025-01874-0 (cit. on p. 42).

[144] B. E. Husic et al., "Coarse graining molecular dynamics with graph neural networks," *The Journal of Chemical Physics*, vol. 153, no. 19, Nov. 2020, ISSN: 1089-7690. DOI: 10.1063/5.0026133. [Online]. Available: http://dx.doi.org/10.1063/5.0026133 (cit. on p. 42).

[145] B. Settles, "Active Learning Literature Survey," University of Wisconsin–Madison, Computer Sciences Department, Tech. Rep. 1648, 2009. [Online]. Available: https://minds.wisconsin.edu/handle/1793/60660 (cit. on p. 42).

[146] D. A. Cohn, L. E. Atlas and R. E. Ladner, "Improving Generalization with Active Learning," *Machine Learning*, vol. 15, pp. 201–221, 1994. [Online]. Available: https://api.semanticscholar.org/CorpusID:8483688 (cit. on p. 42).

[147] H. Wu, F. Paul, C. Wehmeyer and F. Noé, "Multiensemble Markov models of molecular thermodynamics and kinetics," *Proceedings of the National Academy of Sciences*, vol. 113, no. 23, May 2016, ISSN: 1091-6490. DOI: 10.1073/pnas.1525092113. [Online]. Available: http://dx.doi.org/10.1073/pnas.1525092113 (cit. on p. 42).

[148] A. S. Mey, H. Wu and F. Noé, "xTRAM: Estimating Equilibrium Expectations from Time-Correlated Simulation Data at Multiple Thermodynamic States," *Physical Review X*, vol. 4, no. 4, Oct. 2014, ISSN: 2160-3308. DOI: 10.1103/physrevx.4.041018. [Online]. Available: http://dx.doi.org/10.1103/PhysRevX.4.041018 (cit. on p. 42).

[149] H. Wu, A. S. J. S. Mey, E. Rosta and F. Noé, "Statistically optimal analysis of state-discretized trajectory data from multiple thermodynamic states," *The Journal of Chemical Physics*, vol. 141, no. 21, Dec. 2014, ISSN: 1089-7690. DOI: 10.1063/1.4902240. [Online]. Available: http://dx.doi.org/10.1063/1.4902240 (cit. on p. 42).

[150] M. J. Abraham et al., "GROMACS: High performance molecular simulations through multi-level parallelism from laptops to supercomputers," *SoftwareX*, vol. 1–2, 19–25, Sep. 2015, ISSN: 2352-7110. DOI: 10.1016/j.softx.2015.06.001. [Online]. Available: http://dx.doi.org/10.1016/j.softx.2015.06.001 (cit. on p. 42).

[151] D. Van Der Spoel, E. Lindahl, B. Hess, G. Groenhof, A. E. Mark and H. J. C. Berendsen, "GROMACS: Fast, flexible, and free," *Journal of Computational Chemistry*, vol. 26, no. 16, 1701–1718, Oct. 2005, ISSN: 1096-987X. DOI: 10.1002/jcc.20291. [Online]. Available: http://dx.doi.org/10.1002/jcc.20291 (cit. on p. 42).

[152] P. Eastman et al., "OpenMM 8: Molecular Dynamics Simulation with Machine Learning Potentials," *The Journal of Physical Chemistry B*, vol. 128, no. 1, 109–116, Dec. 2023, ISSN: 1520-5207. DOI: 10.1021/acs.jpcb.3c06662. [Online]. Available: http://dx.doi.org/10.1021/acs.jpcb.3c06662 (cit. on p. 42).

[153] P. Eastman et al., "OpenMM 7: Rapid development of high performance algorithms for molecular dynamics," *PLOS Computational Biology*, vol. 13, no. 7, R. Gentleman, Ed., e1005659, Jul. 2017, ISSN: 1553-7358. DOI: 10.1371/journal.pcbi.1005659. [Online]. Available: http://dx.doi.org/10.1371/journal.pcbi.1005659 (cit. on p. 42).

[154] E. Chiavazzo et al., "Intrinsic map dynamics exploration for uncharted effective free-energy landscapes," *Proceedings of the National Academy of Sciences*, vol. 114, no. 28, Jun. 2017, ISSN: 1091-6490. DOI: 10.1073/pnas.1621481114. [Online]. Available: http://dx.doi.org/10.1073/pnas.1621481114 (cit. on p. 42).

[155] H. Jung et al., "Machine-guided path sampling to discover mechanisms of molecular self-organization," *Nature Computational Science*, vol. 3, no. 4, 334–345, Apr. 2023, ISSN: 2662-8457. DOI: 10.1038/s43588-023-00428-z. [Online]. Available: http://dx.doi.org/10.1038/s43588-023-00428-z (cit. on p. 42).

[156] H. Jung, R. Covino and G. Hummer, *Artificial Intelligence Assists Discovery of Reaction Coordinates and Mechanisms from Molecular Dynamics Simulations*, 2019. arXiv: 1901.04595 [physics.chem-ph]. [Online]. Available: https://arxiv.org/abs/1901.04595 (cit. on p. 42).

[157] T. Le, J. Cremer, F. Noé, D.-A. Clevert and K. Schütt, *Navigating the Design Space of Equivariant Diffusion-Based Generative Models for De Novo 3D Molecule Generation*, 2023. arXiv: 2309.17296 [cs.LG]. [Online]. Available: https://arxiv.org/abs/2309.17296 (cit. on p. 43).

[158] I. Dunn and D. R. Koes, *Mixed Continuous and Categorical Flow Matching for 3D De Novo Molecule Generation*, 2024. arXiv: 2404.19739 [q-bio.BM]. [Online]. Available: https://arxiv.org/abs/2404.19739 (cit. on p. 43).

[159] N. T. Runcie and A. S. Mey, "SILVR: Guided Diffusion for Molecule Generation," *Journal of Chemical Information and Modeling*, vol. 63, no. 19, 5996–6005, Sep. 2023, ISSN: 1549-960X. DOI: 10.1021/acs.jcim.3c00667. [Online]. Available: http://dx.doi.org/10.1021/acs.jcim.3c00667 (cit. on p. 43).

[160] W. Gao, R. Mercado and C. W. Coley, *Amortized Tree Generation for Bottom-up Synthesis Planning and Synthesizable Molecular Design*, 2022. arXiv: 2110.06389 [cs.LG]. [Online]. Available: https://arxiv.org/abs/2110.06389 (cit. on p. 43).

[161] T. Shen et al., *Compositional Flows for 3D Molecule and Synthesis Pathway Co-design*, 2025. arXiv: 2504.08051 [cs.LG]. [Online]. Available: https://arxiv.org/abs/2504.08051 (cit. on p. 43).

Part II

Appended Papers

Paper I

Generation of conformational ensembles of small molecules via Surrogate Model-Assisted Molecular Dynamics

J. Viguera Diez, S. Romeo Atance, O. Engkvist, S. Olsson

2024 Mach. Learn.: Sci. Technol. 5 025010

https://doi.org/10.1088/2632-2153/ad3b64

Abstract

The accurate prediction of thermodynamic properties is crucial in various fields such as drug discovery and materials design. This task relies on sampling from the underlying Boltzmann distribution, which is challenging using conventional approaches such as simulations. In this work, we introduce Surrogate Model-Assisted Molecular Dynamics (SMA-MD), a new procedure to sample the equilibrium ensemble of molecules. First, SMA-MD leverages Deep Generative Models to enhance the sampling of slow degrees of freedom. Subsequently, the generated ensemble undergoes statistical reweighting, followed by short simulations. Our empirical results show that SMA-MD generates more diverse and lower energy ensembles than conventional Molecular Dynamics simulations. Furthermore, we showcase the application of SMA-MD for the computation of thermodynamical properties by estimating implicit solvation free energies.

Generation of conformational ensembles of small molecules via Surrogate Model-Assisted Molecular Dynamics

Juan Viguera Diez[1,2], Sara Romeo Atance[1,2], Ola Engkvist[1,2], and Simon Olsson[1*]

[1] Department of Computer Science and Engineering, Chalmers University of Technology, Rännvägen 6 412 58, Göteborg, Sweden.
[2] Molecular AI, Discovery Sciences, R&D, AstraZeneca Gothenburg, Pepparedsleden 1, 431 50 Mölndal, Sweden

E-mail: simonols@chalmers.se

24 November 2023

Abstract.

The accurate prediction of thermodynamic properties is crucial in various fields such as drug discovery and materials design. This task relies on sampling from the underlying Boltzmann distribution, which is challenging using conventional approaches such as simulations. In this work, we introduce Surrogate Model-Assisted Molecular Dynamics (SMA-MD), a new procedure to sample the equilibrium ensemble of molecules. First, SMA-MD leverages Deep Generative Models to enhance the sampling of slow degrees of freedom. Subsequently, the generated ensemble undergoes statistical reweighting, followed by short simulations. Our empirical results show that SMA-MD generates more diverse and lower energy ensembles than conventional Molecular Dynamics simulations. Furthermore, we showcase the application of SMA-MD for the computation of thermodynamical properties by estimating implicit solvation free energies.

1. Introduction

Accurately predicting molecular properties is an important task with applications across the sciences. Some prominent examples are drug discovery and material design. Estimating such properties relies on sampling from the underlying Boltzmann distribution. However, generating unbiased and independent samples from the Boltzmann distribution efficiently remains a challenging open problem.

Currently, Molecular Dynamics (MD) [1] and Markov chain Monte Carlo (MCMC) [2] are the key techniques to draw samples from the Boltzmann distribution. While these techniques asymptotically generate samples from the Boltzmann distribution, many simulation steps are often needed to generate just one independent sample. This problem is particularly prescient for high-dimensional and meta-stable molecular

systems. Despite their limitations, these techniques are widely used, especially in combination with enhanced sampling methods [3], which offer different strategies to speed up the generation of independent samples. Important enhanced sampling methods include replica-based approaches [4, 5], flooding [6], meta-dynamics [7], and umbrella sampling [8].

With the advent of Deep Generative Models (DGMs) [9, 10, 11, 12, 13], a family of new methods to generate unbiased one-shot equilibrium samples of the Boltzmann distribution were proposed under the name of *Boltzmann Generators* (BG) [14, 15, 16, 17]. These methods approximate the Boltzmann distribution of a molecular system with a DGM which allows efficient sampling and exact likelihood evaluation, commonly Normalizing Flows [11]. Efficient sampling allows to side-step iterative simulation methods, and exact likelihood evaluation allows to recover unbiased samples through importance sampling or importance weighing [14]. Previous work has successfully used BGs to sample from large molecules such as proteins [14] or solids [18]. However, these approaches currently do not generalize to different molecular systems, and designing models that are transferable across different molecules remains a challenging task. Another family of recent related methods, such as Implicit Transfer Operator Learning (ITO) [19] or Timewarp [20], tackles the sampling problem by modeling the generative process in Molecular Dynamics simulations, yet on much longer time-scales. Nevertheless, currently, these approaches similarly suffer from limited transferability.

Other approaches focus on enumerating the local minima of a potential energy function, so-called conformers. Different architectures have been proposed, such as CGVAE [21], GeoMol [22], or GeoDiff [23]. These methods are transferable across different molecular systems, however, as the generated states represent local minima of potential energy, they are unable to capture entropic effects due to thermal fluctuations, which makes them unsuitable for computing many molecular properties.

Large-scale conformational rearrangements in molecules can be represented by changes in torsion angles (dihedrals), which correspond to rotations occurring around flexible bonds. For example, the structures of biomolecules such as proteins or RNA are compared and analyzed in terms of such angles, using the Ramachandran plot and probabilistic models of local structure [24, 25] as prominent examples. Transitions between different conformations usually account for the slowest processes in simulations. Therefore, generating representative ensembles of torsions is time-consuming and challenging. For this reason, models focusing on torsion angles are useful means to conformational sampling. Recent work encoding conformations in small molecules using torsions include GeoMol [22], Torsional Diffusion [26], Tora3D [27], and VonMisesNet [28]. Apart from capturing major conformational changes, torsion angles are attractive as they reduce the dimensionality of conformational space and are intrinsically invariant to rigid body symmetries. However, even if some of these methods have been adapted to generate equilibrium samples, they still cannot model stochastic fluctuations in the local structure, and their evaluation as surrogates of the Boltzmann distribution is limited.

In this work, we present Surrogate Model-Assisted Molecular Dynamics (SMA-MD), a method for generating equilibrium ensembles of molecules. In SMA-MD, generative models are used to sample a diverse ensemble of initial conditions for short molecular simulations. SMA-MD follows a two-step procedure: First, a generative model mixes efficiently across degrees of freedom which exchange slowly in molecular simulations. Second, we reweight samples against the Boltzmann distribution and run short Molecular Dynamics simulations to equilibrate the local structure and ensure sampling statistics are unbiased with respect to the target Boltzmann distribution. In this manner, SMA-MD is able to capture entropic effects occurring in all degrees of freedom in a molecule, which are critical for the computation of thermodynamic quantities such as free energy differences. We implement SMA-MD using torsional surrogate models and restrict ourselves to working with small non-cyclic molecules. We probe our method by measuring geometric and thermodynamical (potential and implicit solvation free energies) properties and comparing them to the baseline of classic MD simulations, used as the source of training data. We empirically show that our method can generate diverse and physically realistic ensembles. Equilibrium ensembles generated with SMA-MD present higher conformational coverage and lower average energy than those obtained with conventional MD simulations of similar runtime, closely matching long Replica Exchange (RE) simulations.

Our main contributions are:

- Introducing SMA-MD: A new approach that combines generative models for slow degrees of freedom with statistical reweighting and short simulations to produce equilibrium ensembles for molecules.
- Evaluation of our method by comparison with ensembles generated by Molecular Dynamics and Replica Exchange simulations.
- Generating a new dataset: *MDQM9-nc*, consisting of MD simulations of $\sim$ 12.5 k non-cyclic small molecules from the QM9 dataset that we use as training data, along with data-splits for benchmarking.
- Showcasing a downstream application: estimating relevant observables such as geometrical quantities and free energies of solvation.

2. Methods

2.1. Sampling and molecular properties

Experimental observables (molecular properties) often correspond to averages over the ensemble of 3D arrangements (conformations, $\mathbf{x}$) molecules can adopt, which follow the Boltzmann distribution,

$$\mu(\mathbf{x}) = \mathcal{Z}^{-1} \exp(-\beta U(\mathbf{x})), \text{ with } \mathcal{Z} = \int \mathrm{d}\mathbf{x} \exp(-\beta U(\mathbf{x})), \tag{1}$$

where $U(\mathbf{x})$ is the potential energy and β is the inverse temperature. We conveniently define the reduced potential $u(x) = \beta U(\mathbf{x})$. Given independent and

identically distributed (i.i.d.) conformations sampled from the Boltzmann distribution, thermodynamical quantities can be computed with the *Monte Carlo estimator*:

$$O = \mathbb{E}_{\mathbf{x}\sim\mu(\mathbf{x})}[o(\mathbf{x})] \approx \frac{1}{N}\sum_{i=1}^{N} o(\mathbf{x}_i),\ \ \mathbf{x}_i \overset{\text{i.i.d.}}{\sim} \mu(\mathbf{x}), \tag{2}$$

where $o(\mathbf{x})$ computes the microscopic contribution of a property for a conformation. Generating i.i.d. samples from the Boltzmann distribution is challenging. However, if a surrogate model $p(\mathbf{x})$ overlaps with $\mu(\mathbf{x})$, unbiased Boltzmann-distributed samples can be generated by reweighting the surrogate distribution [14]. This reweighing or resampling, can be achieved with various algorithms, but the simplest one is known as importance sampling, in which samples from $p(\mathbf{x})$ are assigned statistical weight $w = e^{-u(\mathbf{x})}/p(\mathbf{x})$.

2.2. Boltzmann surrogate model

A Boltzmann surrogate is a generative model trained to generate samples from the Boltzmann distribution. In this work, we consider models that generalize across different molecular systems. We use a torsional generative model and therefore, we structure the generation process in the following two sub-steps.

2.2.1. Local structure generation We define the local structure of a non-terminal atom as the relative geometry of the atoms connected to it. The local structure of a given atom can be specified as a set of internal coordinates (distances, angles, and dihedral angles). In this work, we restrict ourselves to molecules that don't have rings as cyclic molecules would require special considerations. Local structures around non-terminal atoms are very constrained and highly dependent on the hybridization of the central atom. For this reason, we use a simple method for generating the local structure of molecules:

- Distances and angles are set to the equilibrium force field parameters.
- Dihedral angles are chosen based on the hybridization of the central atom. For example, if an atom is sp^2 hybridized, its local structure will be planar but if it is sp^3 hybridized, it will follow a tetrahedron shape. We provide further details in the Appendix 9.1.

2.2.2. Rotatable bond generation: Torsional diffusion In this work, following conventions from previous contributions in the context of small molecules [22, 26], we consider a bond to be rotatable if it connects two non-terminal atoms in the chemical graph. In this step, we generate the remaining degrees of freedom, the torsion angles of rotatable bonds, using a DGM to model potentially complex multi-modal distributions. We choose this DGM to be a diffusion model [12, 13, 29]. Torsion angles lie on the circle and therefore the set of torsions of rotatable bonds within a molecule lies on a

hyper-torus. Previous work adapted the formalism of diffusion models to operate on this Riemannian manifold [30]. Based on this work, Torsional Diffusion [26], a diffusion model tailored for modeling torsions, was proposed. One of the main innovations behind Torsional Diffusion is the use of 3D-aware torsional updates which are invariant to the choice of reference atoms. Moreover, a new model architecture exploiting symmetries around rotatable bonds, the Pseudo-Torque layer, was introduced. In this work, we use Torsional Diffusion as a model for torsion angles around rotatable bonds.

2.2.3. Training Since the local structure generation module has no learnable parameters, in this section we solely elaborate on how we train the rotatable bonds model. We train Torsional Diffusion models against a set of target conformations with torsions $\tau_0 \sim p_0$. During training, noise with intensity t is added, resulting in noisy torsions τ_t. Model optimization is performed by minimizing the denoising score matching loss [13, 31]

$$J_{DSM}(\theta) = \mathbb{E}_t \left[\lambda(t) \mathbb{E}_{\boldsymbol{\tau}_0 \sim p_0, \boldsymbol{\tau}_t \sim p_{t|0}(\cdot|\ \boldsymbol{\tau}_0)} \left[||s(\boldsymbol{\tau}_t, t) - \nabla_{\boldsymbol{\tau}_t} \log p_{t|0}(\boldsymbol{\tau}_t|\boldsymbol{\tau}_0)||^2 \right] \right], \qquad (3)$$

where noise level t is uniformly sampled, $\lambda(t) = 1/\mathbb{E}_{\boldsymbol{\tau} \sim p_{t|0}(\cdot|0)} \left[||\nabla_{\boldsymbol{\tau}_t} \log p_{t|0}(\boldsymbol{\tau}|\mathbf{0})||^2 \right]$, $s(\boldsymbol{\tau}_t, t)$ is the neural network prediction for the score and $p_{t|0}(\boldsymbol{\tau}_t|\boldsymbol{\tau}_0)$ is the perturbation kernel. More details about training can be found in the Appendix 9.3.

Please note that, if we directly trained a model using molecular conformations obtained from simulations (or other methods), we would create a distribution shift between training and inference. Therefore, to compensate for this, during training, we substitute the local structures from target conformations with the ones generated by our local structure model.

2.3. Surrogate Model-Assisted Molecular Dynamics

In this work, we present Surrogate Model-Assisted Molecular Dynamics (SMA-MD) as an approach to efficiently sample from the Boltzmann distribution. The overall workflow of SMA-MD is summarized in Figure 1 and consists of two main steps. In the first step, we use a generative model trained to emulate the Boltzmann distribution of molecules (the Boltzmann surrogate) to generate conformations. These conformations are constructed in two sub-steps: starting by generating local geometries with a deterministic algorithm and then sampling the torsion angles using a DGM to mix across the molecules' 'slow degrees of freedom'. Next, in the second step, we use short parallel MD simulations to thermalize and mix the fast degrees of freedom. Combining simulations with an exact reweighting scheme enables us to generate unbiased samples from the Boltzmann distribution of the molecule. These ensembles can then be used to compute thermodynamic properties of interest.

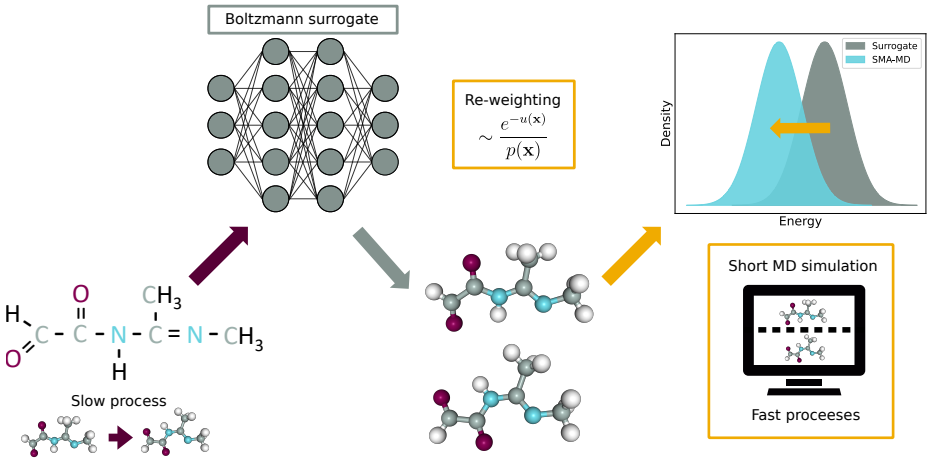


Figure 1: The SMA-MD workflow: Starting from the chemical graph describing a molecule, SMA-MD generates its conformational equilibrium ensemble. First, we generate conformations using a generative model trained to generate samples from the Boltzmann distribution. Then, we reweight against the Boltzmann distribution and run short MD simulations. The generated ensembles can then be used for computing downstream properties of interest.

2.3.1. Sampling from the surrogate model We first generate the degrees of freedom corresponding to the local structure and then we sample the torsions, $\boldsymbol{\tau}$. The torsions are sampled through integration of the probability flow (neural) ODE (ordinary differential equation), corresponding to the torsional diffusion model, which further enables exact reweighting through sample likelihood calculation. We denote $p_0(\boldsymbol{\tau})$ the neural ODE sampler likelihood in torsional space. However, the Boltzmann distribution is generally specified as a function of the 3D coordinates of the atoms in a molecule, $\mathbf{x}$. Therefore, to allow for compatibility with the Boltzmann measure, we need to express $p_0(\boldsymbol{\tau})$ in Euclidean space instead. As all the generated geometric quantities correspond to internal coordinates, and the local structure is generated deterministically, the Euclidean likelihood can be computed as

$$p(\mathbf{x}) = p_0(\boldsymbol{\tau})/|\det(J_{\text{int}\to\text{euc}}(\mathbf{x}))|, \tag{4}$$

where $J_{\text{int}\to\text{euc}}(\mathbf{x})$ is the Jacobian of the transformation from internal to Euclidean coordinates. For details about the sampling procedure, see Appendix 9.4.

2.3.2. Reweighting and MD fine-tuning After generating molecular conformations with the surrogate model, we post-process them by taking two extra steps which we have observed to be critical to generate physically realistic structures.

Reweighting according to Boltzmann weights Given a set of sampled conformers, we compute per-sample weights with

$$w(\mathbf{x}) = \frac{e^{-u(\mathbf{x})}}{p(\mathbf{x})}. \tag{5}$$

We note that reweighting only guarantees improvement in the case of complete domain coverage, which is not fulfilled in general. However, we observe that the surrogate model tends to broadly cover the domain, even generating high-energy metastable conformations. One important role of the reweighting step is ruling out these states as we illustrate in Appendix 9.7.

MD fine-tuning: short parallel simulations Generally, we expect reweighting to align the generated ensemble more closely to the target Boltzmann distribution. However, the ensemble still misses two fundamental ingredients in our implementation: stochastic fluctuations in the local structure and the coupling of these fluctuations with the rotatable bonds. We hypothesize that these two features can be recovered in short simulations. Therefore, we run parallel simulations on the reweighted ensemble using the REFORM library extending OpenMM [32, 33]. For the experiments performed in this work, we run 1 *ns* simulations per sample.

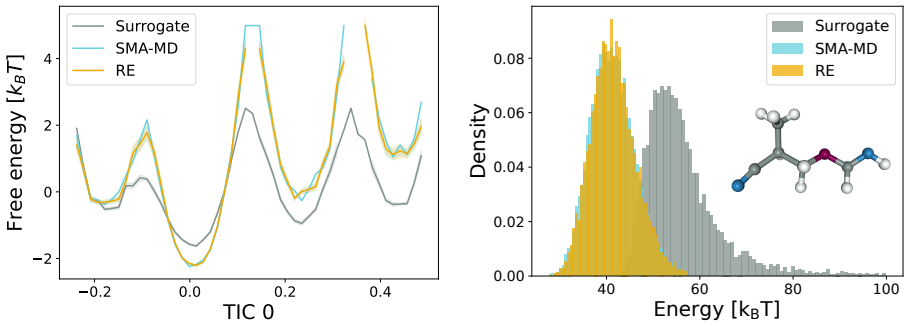
2.4. The MDQM9-nc dataset

We generated approximately Boltzmann-distributed samples using MD by simulating 12,530 non-cyclic molecules from QM9 in vacuum and room temperature using the GAFF force field [34]. We carried out these simulations using the `openmmforcefields` [35] and OpenMM packages [33]. All initial conditions were generated by energy minimizing the QM9 geometry in the corresponding GAFF force field. We sampled different molecules proportionally to their number of heavy atoms with a median sampling time of 36.5 ns. Moreover, for 100 molecules from the test set (10%), we run longer 100 ns Replica Exchange (RE) simulations. These long RE simulations are used as ground truth in our experiments. We provide further details about the dataset generation and training, validation, and test splits in the Appendices 9.8 and 9.9.

3. Results and discussion

3.1. The impact of the different components of SMA-MD for sampling equilibrium conformations of molecular systems

We showcase the contribution of the different components of SMA-MD by analyzing state populations and potential energies for a molecule in the test set at the two different sampling stages (sampling from the surrogate model and applying post-processing). To find the slowest transitions between metastable states, we use time-lagged independent component analysis (TICA) [36], a linear dimensionality reduction



(a) (b)

Figure 2: Free energy of slowest TICA component (a) and density of ensemble energies (b). The shaded areas in (a) correspond to the standard deviation of 10-fold cross-validation. The Boltzmann surrogate captures the relevant states, potentially transitioning to each other in slow processes. Moreover, reweighting and running short simulations adjust the populations and reduce the energy of the ensemble, matching ground-truth long RE simulations

technique that identifies the linear combinations of molecular features that maximizes the autocorrelation. In Figure 2 (a) we observe that the Boltzmann surrogate captures the relevant states, potentially transitioning to each other in slow processes. Moreover, reweighting and running short simulations adjust the populations and reduce the energy of the ensemble, matching ground-truth long RE simulations 2(b).

3.2. SMA-MD generates similar local structures to MD

As the marginal distributions of the degrees of freedom in the local structure are often unimodal, in order to compare local structures generated by different methods, we compute the Mean Absolute Error (MAE) of the estimated mean and standard deviation of these distributions. In Table 1 we observe that local structures generated by SMA-MD and MD exhibit remarkable similarity.

Table 1: Average relative difference of distribution parameters in the local structure between generated and MD for all molecules in the test set. Errors are not shown for being smaller than the last digit.

Distances		Angles		Dihedrals	
Average (%)	Std (%)	Average (%)	Std (%)	Average (%)	Std (%)
0.04	2.6	0.13	1.3	0.14	1.4

3.3. SMA-MD outperforms MD in conformer generation

As previously introduced, a conformer is a local minimum in the molecular energy landscape. Therefore, they are also local maxima of the density landscape. Given an ensemble, we can extract its conformers by finding these local minima. Following this logic, we retrieve conformers from ensembles generated by SMA-MD and MD.

To obtain a set of conformers from an ensemble, we first find modes in the marginal distributions of torsions. Then, we check how many of the possible combinations of modes in the marginal distributions appear in the ensemble. Next, we refine the set of extracted conformers eliminating duplicates corresponding to atom permutations or global inversions of the geometry. Finally, we check that all conformers satisfy a dissimilarity threshold in their relative Root Mean Square Deviation (RMSD). We provide additional details about this procedure in the Appendix 9.10.

In Table 2 we show the Average Minimum RMSD (AMR) and Coverage (COV) of our method in precision and recall modes, see Appendix 9.11 for precise definitions. We report remarkable agreement of SMA-MD with RE, clearly outperforming the training-set-like MD trajectories (MD (15 min)). Because the runtime of SMA-MD (using a single GPU) is longer (20 min) than that of the training set simulations (15 min), see Appendix 9.12 for details, we further challenged SMA-MD by comparing to longer MD simulations of comparable runtime. We observe that SMA-MD still clearly outperforms these longer simulations (MD (20 min)) by recovering 6 % more conformers in the ground-truth ensemble. In Figure 3 we compare the Coverage for different threshold values. Here we observe that SMA-MD achieves the best Coverage among the three methods independently of the choice of threshold and the improvement margin becomes greater as we reduce the threshold.

Table 2: Average Minimum Root Square Deviation (AMR) and Coverage (COV) of the ensembles generated by SMA-MD and MD simulations of different runtimes against Replica Exchange. For COV, the threshold is set to $\delta = 0.75$ Å.

	MD (15 min)		SMA-MD (20 min)		MD (20 min)	
	Precision	Recall	Precision	Recall	Precision	Recall
AMR (Å,↓)	0.14±0.02	0.24±0.03	**0.08±0.01**	**0.10±0.02**	0.13±0.01	0.21±0.02
COV (↑)	0.95±0.02	0.87±0.02	**0.98±0.01**	**0.96±0.01**	0.97±0.01	0.90±0.02

3.4. Covering conformations separated by high free-energy barriers

The conformer generation results above, suggest that our generative model covers the local free energy minima of small molecules well and that we may cover the conformational space of small molecules faster than regular MD simulations. We illustrate that this indeed is the case through the following example. We find SMA-MD samples states present in the long RE simulations separated by barriers that are

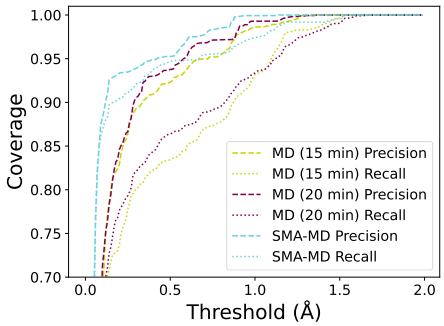


Figure 3: Coverage (precision and recall) as a function of threshold.

never overcome in MD simulations of comparable runtime to SMA-MD. In Figure 4, we show how SMA-MD samples a state (torsion 8) that is not sampled by MD. Overall, the marginal distributions of MD do not match RE, suggesting the simulated ensembles are not fully equilibrated. However, SMA-MD shows remarkable agreement. The difference in mean ensemble energy w.r.t. to RE is -0.37 k_BT for SMA-MD and +3.15 k_BT for MD.

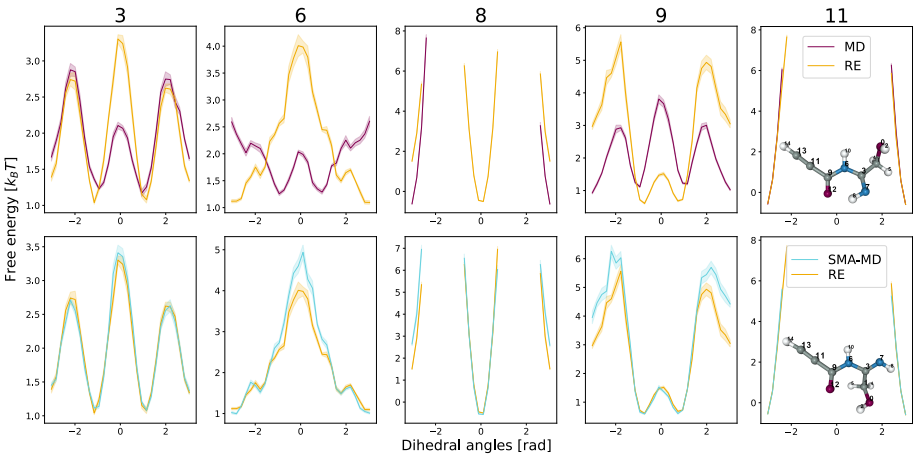
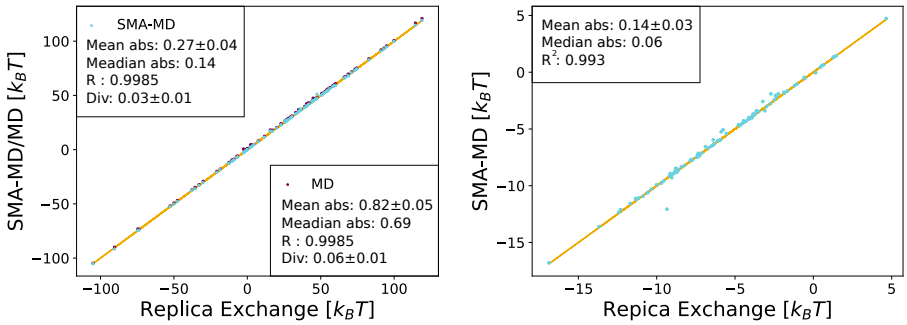


Figure 4: Comparison of dihedral angles of rotatable bonds generated by Molecular Dynamics simulations of similar runtime to SMA-MD (MD, top) and our method (SMA-MD, bottom) vs Replica Exchange (RE). The shaded areas correspond to the standard deviation of 10-fold cross-validation. While MD shows different marginal distributions, our method recovers the distributions generated with RE. Torsions of atoms 13 and 14 are omitted for clarity of presentation, having similar marginal distributions under the three methods.



(a) (b)

Figure 5: Average potential energy (a) and free energy from vacuum to implicit water (b) of ensembles generated using different methods versus Replica Exchange. Error bars are smaller than dots. MD here is an MD simulation of comparable runtime to SMA-MD.

3.5. SMA-MD generates more diverse and energetically favorable ensembles than MD

In section 3.1 we have shown that SMA-MD outperforms conventional MD simulations in conformer generation, presenting better precision and recall w.r.t. ground-truth simulations. Moreover, we have illustrated in section 3.4 how SMA-MD is able to sample across high-energy barriers, where MD falls short.

Furthermore, we show that SMA-MD generates samples in a more statistically efficient manner by comparing the ensemble potential energies. We do this by computing the difference in average ensemble energies and the Jensen-Shanon divergence (div) between energy histograms. We show in Figure 5 (a) that SMA-MD generates more similar energy averages and distributions to RE than MD, and that the agreement with RE is very high.

We attribute a number of factors to the improved performance of SMA-MD compared to MD. First, SMA-MD combines data aggregation and post-processing. On the one hand, even if simulations used as training data are not fully converged, it is possible to learn relevant structures from similar molecules. On the other, reweighting and short simulations help to equilibrate the populations and compensate for small deviations in the geometries generated by the surrogate. Combining these two elements, SMA-MD shows robustness against training on biased data (non-converged simulations). Second, SMA-MD does not need initial conditions. One important limitation of MD simulations is their sensitivity to the initial coordinates, often obtained from an experimental crystal structure. Indeed, a main success of Markov state modeling [37] is its ability to use simulation data from different initial conditions to make quantitative predictions. In contrast, SMA-MD not only does not require an initial condition but

provides a way of initializing simulations with several different representative initial structures to boost convergence. Third, SMA-MD allows for parallelization. MD simulations are intrinsically sequential, however, all the individual components in SMA-MD allow for parallelization, which makes SMA-MD a more suitable method for modern computing hardware.

3.6. Prediction of molecular properties using SMA-MD: solvation free energy

Finally, after finding great agreement between SMA-MD and RE in the previous analyses, we illustrate how the equilibrium ensembles generated by SMA-MD could be used for downstream tasks by estimating solvation free energies. We use the improved Generalized Born model (GB-Neck2) [38] available in OpenMM and experimentally validated with the GAFF force field in previous work [39]. We set an effective number of samples threshold of 100. Details are available in the Appendix 9.13. Results in Figure 5 (right) show remarkable agreement between the two methods.

4. Limitations and future work

A major bottleneck of SMA-MD remains the computational cost per sample in comparison to MD. However, as previously discussed, while MD is intrinsically sequential, the SMA-MD framework is fully parallelizable (except for the short individual simulations) and therefore allows for an efficient way to use modern computing hardware, including GPUs. Indeed, this divide-and-conquer strategy is successfully applied in the Markov state modeling community [40, 41, 42, 43]. Nevertheless, SMA-MD, in the context presented here does not outperform RE in terms of runtime, see Appendix 9.14. This limitation is related to the high cost of sampling from the generative model and the low sample efficiency. Due to continual improvements in the field of DGMs, we believe both of these issues will be resolved in the near future.

Currently, SMA-MD is limited to non-cyclic molecules. This limitation comes from the difficulty of generating realistic ring structures when representing molecules in internal coordinates, e.g. distance, angles, and torsions. This problem is particularly prescient for non-aromatic ring structures, including sugars, which are highly restrained but may undergo concerted slow conformational transitions. While heuristics are available to overcome this problem in e.g. rule-based conformer generations, these methods do not readily allow for the extraction of equilibrium statistics. Therefore, we leave accurate modeling of rings for future work. If modeling non-aromatic rings does not entail significant extra computational cost, we foresee that SMA-MD will be scalable to drug-like molecules. Drug-like molecules in the GEOM-Drugs dataset [44] contain 1.7 aromatic rings and 1.3 non-aromatic rings per molecule. Consequently, the number of rotatable bonds in molecules studied here and those of drug-like molecules, 6.3 and 7.9 respectively, remain comparable.

Finally, we consider training Boltzmann surrogates with large-scale data a

promising direction. With data sharing taking an increasing priority, simulation data using similar protocols are deposited in scientific data repositories. Although these simulations may not be fully converged, they still contain valuable information to build general transferable Boltzmann surrogates. Indeed, the data we used to train our model is clearly not converged, however, we can still learn a useful surrogate. Mining this data to train Boltzmann surrogates could increase the transferability and therefore the usefulness of these systems.

5. Conclusions

In this work, we have introduced Surrogate Model-Assisted Molecular Dynamics, SMA-MD, an efficient method for generating equilibrium conformational ensembles for molecules. SMA-MD combines a transferable surrogate from the Boltzmann distribution with a statistical reweighting and short simulation post-processing step. The goal is to mix between slow degrees of freedom using the surrogate model and refine the ensemble through reweighting and brief simulations. Here, we implement SMA-MD using torsional generative models and show that this method outperforms conventional MD simulations in diversity and ensemble energy. We showcase the applications of this method in downstream tasks by successfully estimating solvation free energies. Even if the work presented here is limited to small non-cyclic molecules, we believe it motivates further research on solving remaining scientific problems and paves the way towards the adoption of similar methods in practical applications, especially drug-discovery pipelines.

6. Data and Software

Data and software will be made available upon publication.

7. Acknowledgments

This work was partially supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation. The computations in this work were enabled by the Berzelius resource provided by the Knut and Alice Wallenberg Foundation at the National Supercomputer Centre. The authors thank Rocío Mercado for discussions and early contributions to this work.

8. Bibliography

[1] Scott A. Hollingsworth and Ron O. Dror. "Molecular Dynamics Simulation for All". In: *Neuron* 99.6 (2018), pp. 1129–1143. ISSN: 0896-6273. DOI: https://doi.org/10.1016/j.neuron.2018.08.011. URL: https://www.sciencedirect.com/science/article/pii/S0896627318306846.

[2] Don van Ravenzwaaij, Pete Cassey, and Scott D. Brown. "A simple introduction to Markov Chain Monte–Carlo sampling". In: *Psychonomic Bulletin & Review* 25.1 (Feb. 2018), pp. 143–154. ISSN: 1531-5320. DOI: 10.3758/s13423-016-1015-8. URL: https://doi.org/10.3758/s13423-016-1015-8.

[3] Jérôme Hénin et al. "Enhanced Sampling Methods for Molecular Dynamics Simulations [Article v1.0]". In: *Living Journal of Computational Molecular Science* 4.1 (2022). DOI: 10.33011/livecoms.4.1.1583. URL: https://doi.org/10.33011/livecoms.4.1.1583.

[4] David J. Earl and Michael W. Deem. "Parallel tempering: Theory, applications, and new perspectives". In: *Phys. Chem. Chem. Phys.* 7 (23 2005), pp. 3910–3916. DOI: 10.1039/B509983H. URL: http://dx.doi.org/10.1039/B509983H.

[5] Amey P Pasarkar et al. "Vendi sampling for molecular simulations: Diversity as a force for faster convergence and better exploration." eng. In: *The Journal of chemical physics* 159.14 (Oct. 2023). ISSN: 1089-7690 (Electronic). DOI: 10.1063/5.0166172.

[6] Helmut Grubmüller. "Predicting slaw structural transitions in macromolecular systems: Conformational flooding Helmut". In: *Physical Review E* 52.3 (1995), pp. 251–275. DOI: 10.1002/9783527611591.ch15.

[7] Alessandro Laio and Michele Parrinello. "Escaping free-energy minima". In: *Proceedings of the National Academy of Sciences* 99.20 (2002), pp. 12562–12566.

[8] G.M. Torrie and J.P. Valleau. "Nonphysical sampling distributions in Monte Carlo free-energy estimation: Umbrella sampling". In: *Journal of Computational Physics* 23.2 (Feb. 1977), pp. 187–199. DOI: 10.1016/0021-9991(77)90121-8. URL: https://doi.org/10.1016/0021-9991(77)90121-8.

[9] Ian J. Goodfellow et al. *Generative Adversarial Networks*. 2014. arXiv: 1406.2661 [stat.ML].

[10] Diederik P Kingma and Max Welling. *Auto-Encoding Variational Bayes*. 2022. arXiv: 1312.6114 [stat.ML].

[11] George Papamakarios et al. *Normalizing Flows for Probabilistic Modeling and Inference*. 2021. arXiv: 1912.02762 [stat.ML].

[12] Jonathan Ho, Ajay Jain, and Pieter Abbeel. *Denoising Diffusion Probabilistic Models*. 2020. arXiv: 2006.11239 [cs.LG].

[13] Yang Song et al. *Score-Based Generative Modeling through Stochastic Differential Equations*. 2021. arXiv: 2011.13456 [cs.LG].

[14] Frank Noé et al. "Boltzmann generators: Sampling equilibrium states of many-body systems with deep learning". In: *Science* 365.6457 (2019), eaaw1147. DOI: 10.1126/science.aaw1147. eprint: https://www.science.org/doi/pdf/10.1126/science.aaw1147. URL: https://www.science.org/doi/abs/10.1126/science.aaw1147.

[15] Jonas Köhler, Andreas Krämer, and Frank Noé. *Smooth Normalizing Flows*. 2021. arXiv: 2110.00351 [stat.ML].

[16] Manuel Dibak et al. *Temperature Steerable Flows and Boltzmann Generators*. 2022. arXiv: 2108.01590 [cond-mat.stat-mech].

[17] Hao Wu, Jonas Köhler, and Frank Noé. *Stochastic Normalizing Flows*. 2020. arXiv: 2002.06707 [stat.ML].

[18] Jonas Köhler et al. *Rigid Body Flows for Sampling Molecular Crystal Structures*. 2023. arXiv: 2301.11355 [cs.LG].

[19] Mathias Schreiner, Ole Winther, and Simon Olsson. *Implicit Transfer Operator Learning: Multiple Time-Resolution Surrogates for Molecular Dynamics*. 2023. arXiv: 2305.18046 [physics.chem-ph].

[20] Leon Klein et al. *Timewarp: Transferable Acceleration of Molecular Dynamics by Learning Time-Coarsened Dynamics*. 2023. arXiv: 2302.01170 [stat.ML].

[21] Elman Mansimov et al. "Molecular Geometry Prediction using a Deep Generative Graph Neural Network". In: *Scientific Reports* 9.1 (Dec. 2019), p. 20381. ISSN: 2045-2322. DOI: 10.1038/s41598-019-56773-5. URL: https://doi.org/10.1038/s41598-019-56773-5.

[22] Octavian-Eugen Ganea et al. *GeoMol: Torsional Geometric Generation of Molecular 3D Conformer Ensembles*. 2021. arXiv: 2106.07802 [physics.chem-ph].

[23] Minkai Xu et al. *GeoDiff: a Geometric Diffusion Model for Molecular Conformation Generation*. 2022. arXiv: 2203.02923 [cs.LG].

[24] Wouter Boomsma et al. "A generative, probabilistic model of local protein structure". en. In: *Proc Natl Acad Sci U S A* 105.26 (June 2008), pp. 8932–8937.

[25] Jes Frellsen et al. "A probabilistic model of RNA conformational space". en. In: *PLoS Comput Biol* 5.6 (June 2009), e1000406.

[26] Bowen Jing et al. *Torsional Diffusion for Molecular Conformer Generation*. 2023. arXiv: 2206.01729 [physics.chem-ph].

[27] Zimei Zhang et al. "Tora3D: an autoregressive torsion angle prediction model for molecular 3D conformation generation". In: *Journal of Cheminformatics* 15.1 (June 2023), p. 57. ISSN: 1758-2946. DOI: 10.1186/s13321-023-00726-8. URL: https://doi.org/10.1186/s13321-023-00726-8.

[28] Kirk Swanson, Jake Williams, and Eric Jonas. *Von Mises Mixture Distributions for Molecular Conformation Generation*. 2023. arXiv: 2306 . 07472 [physics.chem-ph].

[29] Yang Song et al. *Maximum Likelihood Training of Score-Based Diffusion Models*. 2021. arXiv: 2101.09258 [stat.ML].

[30] Valentin De Bortoli et al. *Riemannian Score-Based Generative Modelling*. 2022. arXiv: 2202.02763 [cs.LG].

[31] Aapo Hyvärinen. "Estimation of Non-Normalized Statistical Models by Score Matching". In: *Journal of Machine Learning Research* 6.24 (2005), pp. 695–709. URL: http://jmlr.org/papers/v6/hyvarinen05a.html.

[32] Yaoyi Chen. *Replica Exchange FOR OpenmM: REFORM*. Date accessed 17-08-2023. URL: https://github.com/noegroup/reform.

[33] *OpenMM: High performance, customizable molecular simulation.* Date accessed 17-08-2023. URL: https://openmm.org/.

[34] Junmei Wang et al. "Development and testing of a general amber force field". en. In: *J Comput Chem* 25.9 (July 2004), pp. 1157–1174.

[35] *OpenMM force fields: AMBER and CHARMM force fields for OpenMM*. Date accessed 17-08-2023. URL: https://github.com/openmm/openmmforcefields.

[36] Guillermo Pérez-Hernández et al. "Identification of slow molecular order parameters for Markov model construction". In: *The Journal of Chemical Physics* 139.1 (July 2013). DOI: 10.1063/1.4811489. URL: https://doi.org/10.1063%2F1.4811489.

[37] Jan-Hendrik Prinz et al. "Markov models of molecular kinetics: generation and validation". en. In: *J Chem Phys* 134.17 (May 2011), p. 174105.

[38] Hai Nguyen, Daniel R Roe, and Carlos Simmerling. "Improved Generalized Born Solvent Model Parameters for Protein Simulations". en. In: *J Chem Theory Comput* 9.4 (Apr. 2013), pp. 2020–2034.

[39] Martin Brieg et al. "Generalized Born implicit solvent models for small molecule hydration free energies". In: *Phys. Chem. Chem. Phys.* 19 (2 2017), pp. 1677–1685. DOI: 10.1039/C6CP07347F. URL: http://dx.doi.org/10.1039/C6CP07347F.

[40] Kalyan S. Chakrabarti et al. "A litmus test for classifying recognition mechanisms of transiently binding proteins". In: *Nature Communications* 13.1 (July 2022). DOI: 10.1038/s41467-022-31374-5. URL: https://doi.org/10.1038%2Fs41467-022-31374-5.

[41] Frank Noé et al. "Constructing the equilibrium ensemble of folding pathways from short off-equilibrium simulations". In: *Proceedings of the National Academy of Sciences* 106.45 (Nov. 2009), pp. 19011–19016. DOI: 10.1073/pnas.0905466106. URL: https://doi.org/10.1073%2Fpnas.0905466106.

[42] Gregory R. Bowman et al. "Discovery of multiple hidden allosteric sites by combining Markov state models and experiments". In: *Proceedings of the National Academy of Sciences* 112.9 (Feb. 2015), pp. 2734–2739. DOI: 10.1073/pnas.1417811112. URL: https://doi.org/10.1073%2Fpnas.1417811112.

[43] Simon Olsson and Frank Noé. "Dynamic graphical models of molecular kinetics". In: *Proceedings of the National Academy of Sciences* 116.30 (July 2019), pp. 15001–15006. DOI: 10.1073/pnas.1901692116. URL: https://doi.org/10.1073%2Fpnas.1901692116.

[44] Simon Axelrod and Rafael Gómez-Bombarelli. "GEOM, energy-annotated molecular conformations for property prediction and molecular generation". In: *Scientific Data* 9.1 (Apr. 2022), p. 185. ISSN: 2052-4463. DOI: 10.1038/s41597-022-01288-4. URL: https://doi.org/10.1038/s41597-022-01288-4.

[45] Greg Landrum. *RDKit: Open-source cheminformatics*. Date accessed 17-08-2023. URL: http://www.rdkit.org.

[46] Raghunathan Ramakrishnan et al. "Quantum chemistry structures and properties of 134 kilo molecules". In: *Scientific Data* 1.1 (Aug. 2014), p. 140022. ISSN: 2052-4463. DOI: 10.1038/sdata.2014.22. URL: https://doi.org/10.1038/sdata.2014.22.

9. Appendix

9.1. Local structure model

In this work, we define the local structure of a non-terminal (central) atom as the relative geometry of the atoms connected to it. We use a breadth-fist algorithm to generate atomic labels. Atoms with higher atomic ranking (generated with RDkit [45]) and which are non-terminal have a preference to be selected as the next atom. Particularly, every time the algorithm reaches a non-terminal atom, it generates labels for all its non-labeled neighboring nodes. The non-terminal node and its neighbors form a group. In a group of atoms, the first atom is the central one. The next is the one that had been previously visited by the algorithm. Finally, the remaining atoms are added to the group based on their ranking. This is illustrated in Figure 6.

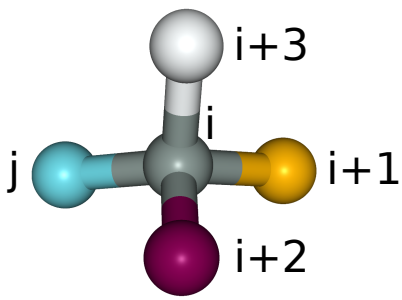


Figure 6: Atomic groups and order.

We can specify the local structure within a group (g_i) as a set of internal coordinates. Distances between pairs of connected atoms ((i,k) with $k \in g_i$) are generated using the force field equilibrium distance parameters. Angles between triplets of atoms are set using the force field equilibrium angle parameter. Triplets of atoms are chosen using the atoms (j,i,k) in the group, where $k \in g_i$ and $k > i$. If g_i has more than 3 atoms, dihedral angles are needed. Dihedral angles are assigned based on the hybridization of the central atom i.

- If i is sp^2 hybridized, the dihedral angle formed by $(j,i,i+1)$ and $(j,i,i+2)$ is set to π.
- If i is sp^3 hybridized, the dihedral angle formed by $(j,i,i+1)$ and $(j,i,i+2)$ is set to $+-2\pi/3$ and, if necessary, the one formed by $(j,i,i+1)$ and $(j,i,i+3)$ is set to $-+2\pi/3$.

Chirality is specified depending on the sign of the previous dihedral angles. In cases

where the central atom is sp^3 hybridized and the groups has 4 atoms, chirality inversion is possible. For simplicity, we set the probability of the dihedral angle being $2\pi/3$ or $-2\pi/3$ to 0.5. In our dataset, *MDQM9-nc*, no other hybridizations occur, but our framework could be easily extended to other types of hybridizations such as sp^3d or sp^3d^2, which are present in some drug-like molecules.

9.2. Diffusion generative models

Diffusion models are a type of generative model and aim to generate samples from the data distribution $p_0(x)$. Starting from samples from p_0, the *forward diffusion process* is defined as an Ito stochastic differential equation (SDE),

$$d\mathbf{x} = \mathbf{f}(\mathbf{x}, t)dt + g(t)\ d\mathbf{w}, \tag{6}$$

where $d\mathbf{w}$ is a Wiener process, $\mathbf{f}(\mathbf{x}, t)$ and $g(t)$ are chosen functions and $t \in (0, T)$ is referred to as the time variable. The distribution at time T, p_T, asymptotically approaches a simple Gaussian distribution. Diffusion models are often trained to approximate the so-called score $\nabla_x \log p_t(\mathbf{x})$ [13]. Once a diffusion model is trained to approximate the score, there are different ways to sample from it. In this work, we use the *probability flow* ordinary differential equation (ODE),

$$d\mathbf{x} = \left[\mathbf{f}(\mathbf{x}, t)dt - \frac{1}{2}g^2(t)\ \nabla_x \log p_t(\mathbf{x})\right] dt, \tag{7}$$

Generating samples from p_T and solving the *probability flow* ODE yields samples from the data distribution, p_0.

9.3. Training details

We train a Torsional Diffusion model by feeding batches of conformations of molecules to the Pseudo-torque network, computing the loss in Equation 3, and optimizing using Adam. The target score of a set of noisy torsions $\boldsymbol{\tau'}$ at time t obtained from a data sample $\boldsymbol{\tau}$ can be computed with the perturbation kernel [30, 26],

$$p_{t|0}(\boldsymbol{\tau'}|\boldsymbol{\tau}) \propto \sum_{\boldsymbol{d}\in\mathbb{Z}^m} \exp -\frac{||\boldsymbol{\tau} - \boldsymbol{\tau'} + 2\pi\boldsymbol{d}||^2}{2\sigma^2(t)}, \tag{8}$$

where $\sigma(t)$ is the noise scale.

We tuned the following hyper-parameters on the validation set: initial leaning rate ($5 \cdot 10^{-3}$, $\mathbf{1 \cdot 10^{-3}}$ and $5 \cdot 10^{-2}$) and number of layers (3, **4** and 5). The rest of the hyper-parameters were set as in [26]. We trained our models for around 750 epochs until the log-log plot of the loss was converged taking about 10 days to train in a single GPU.

9.4. Sampling details

In this work, we use the *probability flow* ODE 7 of the Torsional Diffusion model to sample torsions of rotatable bonds. Crucially, this method allows for likelihood computations of the generated samples. The likelihood in torsional space under this sampling mechanism can be computed as

$$\log p_0(\boldsymbol{\tau}_0) = \log p_T(\boldsymbol{\tau}_T) - \frac{1}{2}\int_0^T g^2(t)\nabla_{\boldsymbol{\tau}} \cdot s(\boldsymbol{\tau}_t, t)dt. \tag{9}$$

We sample 10k conformations for each molecule in our empirical evaluations.

9.5. Reweighting and Boltzmann Distribution

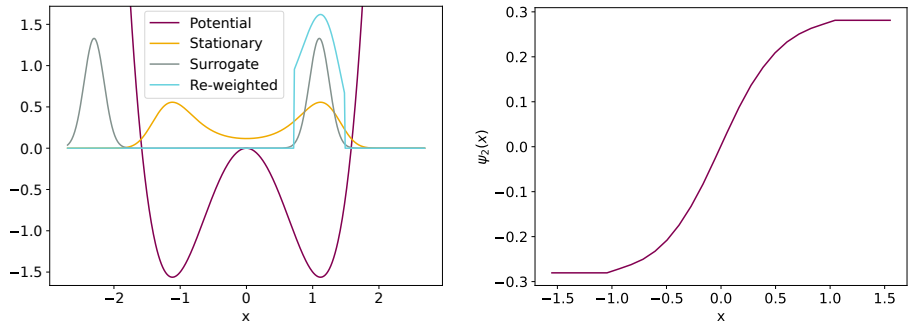
In this section, we first show how under ideal conditions of full coverage and infinite sampling, reweighting returns the target distribution for an arbitrary surrogate. Finally, we discuss the differences between the ideal case and ours and reflect how this can affect the performance of our model.

A probability density is the stationary (Boltzmann) distribution if and only if its corresponding μ-weighted density u fulfills $\langle u|\phi_i\rangle = \delta_{i,1}$. That is, $p(x) = \phi_1(x) = \mu(x)\psi_1(x) = \mu(x)$, where ϕ_i and ψ_i are, respectively, the eigenfunctions of the propagator and transfer operator [37]. Therefore, in the language of these operators, we want to show that the contributions corresponding to non-stationary distributions ($i \neq 1$) get smaller after reweighting. Under the assumptions of infinite sampling and domain coverage (meaning that the surrogate model assigns a non-zero probability to all possible states in state pace), the reweighted distribution (p_r) is exactly the Boltzmann distribution. We confirm this by computing

$$\begin{aligned}\langle u_r|\phi_i\rangle &= \int_\Omega p_0 \frac{\mu(x)}{p_0(x)}\frac{1}{\mu(x)}\phi_i(x)dx = \int_\Omega \phi_i(x)dx \\ &= \int_\Omega \mu(x)\psi_i(x)dx = \int_\Omega \phi_1(x)\psi_i(x)dx = \delta_{i,1},\end{aligned} \tag{10}$$

where we have used the orthonormality of the eigenfunctions.

In practice, the previous assumptions do not necessarily hold. Sampling time is limited by our computing budget and we cannot guarantee domain coverage. However, we do observe empirically that our model does not under-sample, but, if anything, tends to over-sample states not observed in long simulations. After reweighting, the distribution will not be exactly Boltzmann distributed, in contrast to our derivation, but states with lower energy will be favored as more efficient starting points for our simulation while balancing state populations. Furthermore, after the short simulation, we expect that fast-decaying contributions of the distribution vanish pushing our samples further closer to the stationary distribution.



(a) (b)

Figure 7: Scenario in which reweighting can cause longer convergence times. In the left (a), we show a double-well potential with its corresponding stationary distribution and possible surrogate density. In the right (b), we show the second (slowest) eigenfunction of the system. When the surrogate distribution is reweighted, the states on the negative part are assigned very small statistical weight. However, running a short simulation would transport those steps to a high-density area in a few steps, following the steep potential.

9.6. When reweighting can be detrimental

In general, it is not guaranteed that reweighting improves a surrogate distribution. In this section, we provide a counter-example to prove this. Consider the situation depicted in Figure 7 (a).

Here, we show a double-well potential and the corresponding stationary distribution. In this figure, we plot as well the density corresponding to a surrogate distribution. This consists of a clipped mixture of two Gaussian distributions. The positive component of the Gaussian mixture covers values of x with high stationary density, but the negative component covers a high-energy region. The surrogate distribution is clipped in the sense that the density is set to 0 for values of x in which the density of the mixture is below a threshold. This defines a simple surrogate that does not have full support over the domain. The surrogate distribution is depicted in 7 (a), see section 9.6.1 for exact equations.

Intuitively, because negative samples from p_0 have high energy, if we reweight the ensemble, they will be assigned smaller (negligible in this example) statistical weight. Therefore, p_r, the reweighted distribution, contains fewer (almost none) samples in the negative region. However, if a simulation starts from a negative sample from the surrogate, it would approach the left node of the stationary distribution, following the steep

energy gradient, so, intuitively, reweighting should slow down the convergence in this constructed example.

We can prove this formally by computing the relaxation time of the slowest process, t_2, which is proportional to the projection of the surrogate distribution to the corresponding eigenfunction, ψ_2. As ψ_2 resembles the slowest process in the system, it must change sign in 0 as shown in Figure 7 (b). Let's say that we choose ψ_2 in such a way that is negative on the left and positive on the right. Then,

$$\begin{aligned} |\langle p_r|\psi_2\rangle| &= |\int_+ p_r(x)\psi_2(x)dx + \int_- p_r(x)\psi_2(x)dx| = |\int_+ p_r(x)\psi_2(x)dx| - |\int_- p_r(x)\psi_2(x)dx| \\ &> |\int_+ p_0(x)\psi_2(x)dx| - |\int_- p_0(x)\psi_2(x)dx| = |\langle p_0|\psi_2\rangle|, \end{aligned} \tag{11}$$

where $\int_+$ and $\int_+ -$ denote, respectively, the integral over the positive and negative regions. The previous inequality holds because $p_r(x) > p_0(x)$ in the positive region while the opposite occurs in the negative one. Finally,

$$\frac{t_{r,2}}{t_{0,2}} = \frac{|\langle p_r|\psi_2\rangle|}{|\langle p_0|\psi_2\rangle|} > 1, \tag{12}$$

which means that the relaxation time of the slowest process for the reweighted distribution ($t_{r,2}$) is longer than for the non-reweighted one ($t_{0,2}$).

9.6.1. Equations of counter-example Double-well potential with b=1 and c=2.5:

$$V(x) = bx^4 - cx^2. \tag{13}$$

The surrogate distribution is composed of the normal distributions $f_-(x) = \mathcal{N}(\mu = -2.3, \sigma = 0.15)$ and $f_+(x) = \mathcal{N}(\mu = 1.1, \sigma = 0.15)$. The unnormalized density can be computed as a function of $f(x) = 0.5 * f_-(x) + 0.5 * f_+(x)$,

$$\text{unnormalized } p_0(x) = \begin{cases} f(x) \text{ if } f(x) > \delta \\ 0 \text{ else.} \end{cases}$$

δ is chosen to be 0.05.

9.7. Reweighting in practice

In practice, for most molecules, we observe the reweighting steps to be beneficial. We illustrate this with an example molecule in the test set. In Figure 8, we observe:

- Globally, the marginal free energies obtained with the reweighting step fit much better the ones generated by RE.

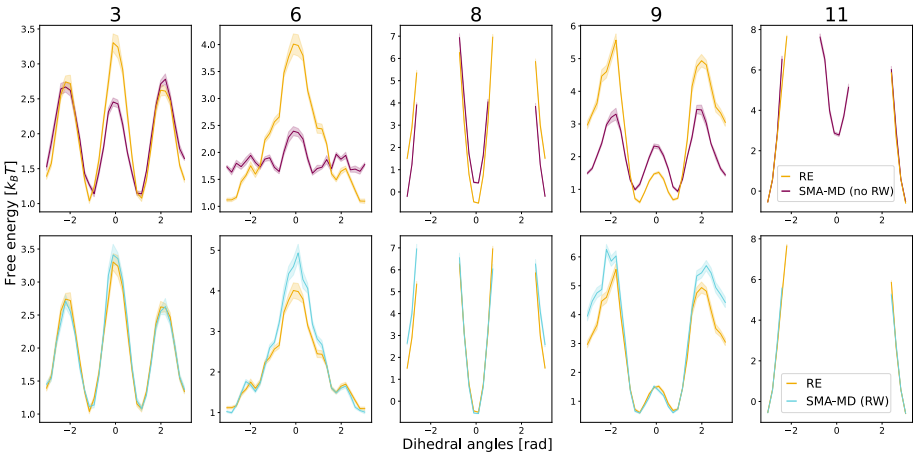


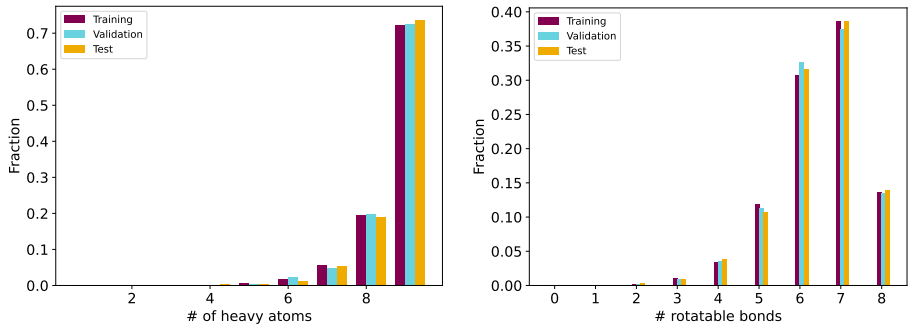
Figure 8: Role of reweighting in practice. Comparison of dihedral angles of rotatable bonds generated by SMA-MD with (SMA-MD (RW), bottom) and without (SMA-MD (no RW), top) the reweighting step vs Replica Exchange (RE). The shaded regions correspond to the standard deviation of 10-fold cross-validation. The reweighting step helps compensate populations and eliminates metastable states not sampled in the RE simulations. Torsions of atoms 13 and 14 are omitted for clarity of presentation, having similar marginal distributions under the three methods.

- Population equilibration: In torsion 8, we observe how SMA-MD without reweighting fails to exactly capture the relative populations of the two minima, while this is mostly corrected when we add the reweighting step.
- High-energy metastable states depletion: In torsion 11, a state around 0 is sampled with low probability when reweighting is not used. This state is not sampled by RE. Using the reweighting step this state is ruled out from the ensemble.

In certain instances, we have observed that performing reweighting prior to executing brief simulations can have an unfavorable impact, resulting in an increase in the mean energy value. In such cases, we omit the reweighting procedure and proceed directly with the brief simulations. Note that these two sampling modalities can be executed in parallel with very limited overhead.

9.8. Dataset generation: MDQM9-nc

Our dataset is generated by running MD simulations of non-cyclic molecules in the QM9 dataset [46]. We were able to simulate 12,530 molecules with the GAFF 2.11 [34] force field in vacuum. We used the `OpenMM` [33] library and the GAFF template generator available at the `openmmforcefields` [35] as recommended in `https://github.com/openmm/openmmforcefields`. Our workflow consists of:



(a) (b)

Figure 9: Distribution of heavy atoms (a) and rotatable bonds (b) for different splits in the dataset.

- Selecting the non-cyclic molecules of QM9.
- Minimizing the structures in QM9 using the GAFF 2.11 force field.
- Running simulations: We choose the simulation time depending on the size of the molecule. The simulation time is heuristically computed as $0.5+4*n_h$ ns, where n_h is the number of heavy atoms. This way we aim to perform longer simulations for molecules that are bigger and potentially more flexible. We generate a total of $16k$ samples per molecule with a time step of $1fs$. The minimum threshold of 0.5 ns is chosen so that the minimum time between frames is 0.3 ps. The simulation time for the median QM9 molecule (9 heavy atoms) is 36.5 ns. Partial charges are taken from the QM9 dataset.

9.9. Dataset statistics and splits

We use random $70-20-10$ train-validation-test splitting in this work. We show in Figure 9 how the number of heavy atoms and rotatable bonds is evenly distributed across the different splits.

9.10. Extracting conformers from a small molecule ensemble

Here we propose a simple method to generate a set of conformers from an equilibrium ensemble. As torsions of rotatable bonds are the main source of flexibility, we look for local maxima in the densities of torsions. Our workflow consists of:

- We first find modes in the marginal distributions of torsions. We do this based on modes' prominence and considering that torsion angles are wrapped around the circle. We heuristically set the density prominence threshold (minimum threshold so that a node is considered to be so) to 0.02.

Table 3: Average runtime of different components of our implementation of SMA-MD for one molecule. Note that components do not sum to total because CPU → GPU requires to be done twice. MDFT (MD fine-tuning) corresponds to short simulations.

	Sampling	CPU → GPU	Energy evaluation	MDFT	Total
Time (min)	11.3	1.5	0.4	5.6	20.3

- As we have analyzed the marginal distributions of torsions (and not the global distribution), we need to check which combinations of local maxima appear in the ensemble. To do that, we associate each sample in the ensemble with a combination of local maxima and only keep the combinations found in the ensemble.
- Finally, we add an extra dissimilarity threshold depending on the relative Root Mean Square Deviation (RMSD) between candidate conformers. We set this threshold to 0.1 Å. This prevents us from selecting multiple times conformers that are too similar, helping us to discard repetitions due to atom labels permutations. Finally, we also apply the same filter to inverted structures to avoid considering global geometric inversions as different.

9.11. Conformer metrics

We use the benchmark metrics used in [22] and followed in posterior works for evaluating the quality of the conformers generated with different methods. The metrics consist of the so-called Average Minimum Root Mean Square Deviation (AMR) and Coverage (COV). These metrics are computed in two modes, Precision (P) and recovery (R). Let $\{C_l^*\}_{l\in[1,L]}$ and $\{C_k^*\}_{k\in[1,K]}$ be respectively the set of ground-truth and generated conformers. Then,

$$\text{AMR-R} := \frac{1}{L}\sum_{l\in[1..L]} \min_{k\in[1.K]} \text{RMSD}\left(C_k, C_l^*\right) \tag{14}$$

$$\text{COV-R} := \frac{1}{L}|\{l \in [1..L] : \exists k \in [1..K], \text{RMSD}\left(C_k, C_l^*\right)\}|, \tag{15}$$

where δ is the coverage threshold, which is usually set to 0.75 Å. Precision metrics are obtained by swapping ground truth and generated conformer sets.

9.12. Runtime

We show the runtime of different components in our implementation of SMA-MD running on a single NVIDIA A100 GPU in Table 3

Table 4: Average Minimum Root minimum square deviation (AMR) and Coverage (COV) of the ensembles generated by SMA-MD and Replica exchange comparable runtime against long Replica Exchange with $\delta = 0.75$ Å.

	SMA-MD		RE	
	Precision	Recall	Precision	Recall
AMR(Å)	0.08±0.01	0.10±0.02	**0.05±0.01**	**0.06±0.01**
COV	0.98±0.01	0.96±0.02	**0.99±0.01**	**0.99±0.01**

9.13. Solvation free energy details

The free energy of solvation is estimated in our experiments using free energy perturbation,

$$\Delta F(v \to s) = -k_B T \log \left\langle \exp\left(-\frac{U_s(x) - U_v(x)}{k_B T}\right)\right\rangle_v, \tag{16}$$

where v denotes vacuum and s denotes solvent. The average over v is evaluated by generating vacuum ensembles and computing the average of the exponential term in Equation 16. The solvent energy U_s is evaluated using the improved Generalized Born model (GB-Neck2). GAFF and (GB-Neck2) have been experimentally validated for predicting free energies of solvation [39]. A prediction of very high solvation energy can be due to both the molecule being not soluble (small overlap between vacuum and solvent ensemble) or that an ensemble that we generated (but not the ground-true vacuum ensemble) has low overlap. To be certain to have some statistical significance in our estimates we set a threshold of 100 effective samples to rely on the predicted free energy. This only discards 3% of the molecules.

9.14. Comparison: SMA-MD vs RE

Even if we show that SMA-MD outperforms conventional MD simulations, SMA-MD does not yet outperform RE simulations of the same runtime. We show this in Table 4 and Figure 10, reporting the benchmark metrics used in the main text.

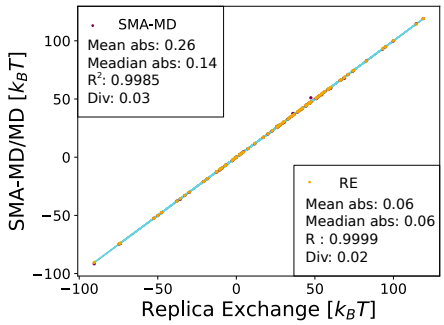


Figure 10: Average potential energy of ensembles generated using RE of comparable runtime to SMA-MD versus long Replica Exchange. Error bars are smaller than dots.

Paper II

Transferable Generative Models Bridge Femtosecond to Nanosecond Time-Step Molecular Dynamics

J. Viguera Diez, M. Schreiner, S. Olsson

Submitted, under revision

https://doi.org/10.48550/arXiv.2510.07589

Abstract

Understanding molecular structure, dynamics, and reactivity requires bridging processes that occur across widely separated time scales. Conventional molecular dynamics simulations provide atomistic resolution, but their femtosecond time steps limit access to the slow conformational changes and relaxation processes that govern chemical function. Here, we introduce a deep generative modeling framework that accelerates sampling of molecular dynamics by four orders of magnitude while retaining physical realism. Applied to small organic molecules and peptides, the approach enables quantitative characterization of equilibrium ensembles and dynamical relaxation processes that were previously only accessible by costly brute-force simulation. Importantly, the method generalizes across chemical composition and system size, extrapolating to peptides larger than those used for training, and captures chemically meaningful transitions on extended time scales. By expanding the accessible range of molecular motions without sacrificing atomistic detail, this approach opens new opportunities for probing conformational landscapes, thermodynamics, and kinetics in systems central to chemistry and biophysics.

Transferable Generative Models Bridge Femtosecond to Nanosecond Time-Step Molecular Dynamics

Juan Viguera Diez,[1,2], Mathias Schreiner,[1] and Simon Olsson[1,*]

[1]Department of Computer Science and Engineering,
Chalmers University of Technology and University of Gothenburg,
SE-41296 Gothenburg, Sweden
[2]Molecular AI, Discovery Sciences, R&D,
AstraZeneca Gothenburg,
Pepparedsleden 1, 431 50 Mölndal, Sweden.

[*]To whom correspondence should be addressed; E-mail: simonols@chalmers.se.

Understanding molecular structure, dynamics, and reactivity requires bridging processes that occur across widely separated time scales. Conventional molecular dynamics simulations provide atomistic resolution, but their femtosecond time steps limit access to the slow conformational changes and relaxation processes that govern chemical function. Here, we introduce a deep generative modeling framework that accelerates sampling of molecular dynamics by four orders of magnitude while retaining physical realism. Applied to small organic molecules and peptides, the approach enables quantitative characterization of equilibrium ensembles and dynamical relaxation processes that were previously only accessible by costly brute-force simulation. Importantly, the method generalizes across chemical composition and system size, extrapolating to peptides larger than those used for training, and captures chemically

meaningful transitions on extended time scales. By expanding the accessible range of molecular motions without sacrificing atomistic detail, this approach opens new opportunities for probing conformational landscapes, thermodynamics, and kinetics in systems central to chemistry and biophysics.

Introduction

Many of the most important observables in statistical mechanics—such as the stability of a folded protein, the conformational transitions underlying allosteric regulation, or the unbinding rate of a drug from its targets—are central to understanding chemical and biological function. These processes span timescales from nanoseconds to seconds, and while they are directly accessible through experiments such as spectroscopy (*1*) and single-molecule techniques (*2*), their atomistic origins are often hidden.

Molecular dynamics (MD) offers a powerful complement to such experiments. By simulating the trajectories of atoms and molecules at atomic resolution, MD connects the fundamental interatomic forces that govern molecular motion to the statistical behavior observed in bulk. In this way, simulations provide a mechanistic bridge between microscopic physics and macroscopic phenomena (*3*).

Yet MD comes with a fundamental limitation. To ensure numerical stability, simulations must take time steps small enough to resolve the fastest motions in the system, such as bond and angle vibrations. This requirement restricts MD to femtosecond update steps, even though many processes of chemical and biological interest — protein folding, conformational transitions, ligand binding — unfold over microseconds to seconds. These processes are typically governed by rare transitions between metastable states (*4*), creating a persistent gap between simulation and experiment that limits our ability to characterize slow molecular processes with statistical confidence (*5*).

This challenge, known as the *'sampling problem'*, continues to inspire a growing array of strategies aiming at accelerating the observation of rare events. Most approaches either bias the underlying dynamics or simulate multiple coupled replicas in parallel, both designed to make infrequent transitions occur more often (*6*). Biasing methods rely on the definition of collective variables (CVs), that capture the progress of a process of interest (*7–9*). However, identifying suitable CVs for complex, high-dimensional systems remains difficult: variables that accelerate one process may obscure others, and their design has become a discipline in its own right with numerous options (*10*) including ones derived using machine learning-based strategies (*11–13*). Further, since these methods bias the dynamic behavior of the system, estimation of kinetic properties is only possible under restrictive conditions (*14*). A complementary line of work seeks to increase the integration time-step directly, reducing the number of integration steps per unit time. Despite decades of intense research in this direction (*15–17*), integration steps remain on the femtosecond scale, leaving even the most efficient simulations orders of magnitude too slow to capture experimentally relevant molecular processes.

A parallel line of progress has focused on harnessing ever-larger computational resources. Specialized compute architectures (*18, 19*) have achieved continuous millisecond-scale trajectories for small proteins, revealing mechanistic detail inaccessible to conventional hardware (*20*). Distributed platforms such as Folding@home leverage millions of short trajectories contributed by volunteers (*21*), while modern GPU-based algorithms have brought comparable acceleration to widely used MD engines (*22–25*). Together with statistical frameworks such as Markov state models (MSMs) (*26, 27*), these efforts have enabled the reconstruction of long-timescale kinetics from massive ensembles of short simulations. Yet all remain bound by the need to generate femtosecond-resolved trajectories, keeping progress tied to extreme computational resources. A conceptually different approach would be to model the effective long-lag dynamics directly, without resorting to brute-force sampling or biasing.

MD simulations function through the numerical integration of the Langevin equation (*28*). As we move along the simulation trajectory, MD generates statistical samples from a transition probability distribution $p(\mathbf{x}_{t+\tau} \mid \mathbf{x}_t)$ where τ is on the order of femtoseconds, and $\mathbf{x}_t$ and $\mathbf{x}_{t+\tau}$ are points in the phase-space. Crucially, this distribution is not *ad hoc*: it approximates the Green's function of the Fokker-Planck equation governing Langevin dynamics, providing the theoretical foundation for viewing MD trajectories as stochastic samples from an underlying probabilistic process (*29*). It follows that analogous transition probability distributions exist for much larger Δt, and that these can, in principle, be learned directly for a given molecular system (*30*). Learning such long-lag transition densities offers a direct route to coarse-grained yet statistically faithful dynamics, sidestepping the need for explicit time integration.

Here, we introduce Transferable Implicit Transfer Operators (TITO), a deep generative framework that learns these transition probability distributions across molecular systems. TITO allows us to choose the simulation step size freely, whether to match the characteristic timescales of experiments or to accelerate sampling of slow conformational transitions. Trained on MD data from small molecules and short peptides, TITO simultaneously learns transitions at multiple step sizes, ensuring consistency with the underlying stochastic process. As a result, it preserves key statistical properties such as Boltzmann equilibrium, Markovianity, and relaxation dynamics, suggesting approximate energy conservation and equipartition.

TITO demonstrates quantitative transferability to molecular systems of similar size as in the training data, and provides qualitative insights for molecules twice as large. Unlike conventional simulation-based sampling, TITO offers explicit control over the trade-off between accuracy and computational cost, enabling speedups of up to 15,000-fold. By learning effective long-lag dynamics directly, TITO takes a step toward bridging the longstanding gap between atomistic resolution and experimentally relevant timescales. More broadly, it establishes a new paradigm for accelerating molecular simulations, with the potential to extend atomistic modeling to processes previously beyond reach.

Transferable Implicit Transfer Operators

At its core, TITO (Transferable Implicit Transfer Operators, Fig. 1) learns the effective rules of molecular motion: predicting how atomic configurations evolve over time without explicit time integration. Rather than advancing dynamics step by step, TITO draws statistical samples directly from the transition distribution $p(\mathbf{x}_{t+\Delta t} \mid \mathbf{x}_t)$, capturing how configurations change over a specified lag time Δt. Trained across diverse molecular systems and lag times, TITO generalizes both across chemistry and temporal scale.

Training proceeds from reference molecular dynamics trajectories simulated with a small integration step τ:

$$\mathbf{X} = \{\mathbf{x}_\tau, \ldots, \mathbf{x}_{N\tau}\}, \quad \mathbf{x}_{n\tau} \sim p(\mathbf{x}_{n\tau} \mid \mathbf{x}_{(n-1)\tau}), \quad n = 1, \ldots, N,$$

collected across a diverse set of molecules. From these data, the model learns to reproduce the time-integrated transition statistics that would arise if the dynamics were propagated at much larger effective steps $\Delta t = m\tau$, where m is an arbitrary large integer.

We parametrize the transition probability distribution, using a continuous normalizing flow (CNF) through the equivariant *flow matching* (*31, 32*) objective. A CNF consists of an ordinary differential equation (ODE) and an easy-to-sample 'base distribution,' p_0, such as a Gaussian (*33*). The velocity field of the ODE is parameterized with a neural network model which is trained to ensure that the resulting flow transports samples from p_0 to a distribution p_1 closely matching the target data distribution, here, the transition probability distribution. The flowis then the set of all integral paths $\mathbf{x}^T_{t+\Delta t}$, where $T \in [0, 1]$ is the ODE integration time. Throughout this work, superscripts denote ODE integration time, while subscripts indicate MD simulation time.

In practice, we learn the weights θ of a neural network, $\mathbf{v}_\theta\left(\mathbf{x}^T_{t+\Delta t};\ \mathbf{x}_t,\ \Delta t,\ T\right)$, to match a conditional flow which approximates the transition probability distribution, by minimizing the

conditional flow matching loss,

$$\mathcal{L}(\theta) = \mathbb{E}_{\mathbf{x}_t, \mathbf{x}_{t+\Delta t} \sim \mathbf{X},\, T \sim \mathcal{U}(0,1)} \left[\left\| \mathbf{v}_\theta \left(\mathbf{x}^T_{t+\Delta t};\ \mathbf{x}_t,\ \Delta t,\ T \right) - \left(\mathbf{x}^1_{t+\Delta t} - \mathbf{x}^0_{t+\Delta t} \right) \right\|^2 \right].$$

During training, we sample molecules and lag times jointly, enabling TITO to generalize across both chemical composition and temporal scale. After training, new trajectories are generated by sampling from p_0 and integrating the learned ODE defined by $\mathbf{v}_\theta$. Full model details are provided in Section Model.

We train TITO models on two datasets. The first, **MDQM9-nc** (*34*), contains MD simulations of small organic molecules, while the second, **Timewarp** (*35*), provides tetra-peptides trajectories. Together, these datasets enable training across a range of molecular sizes and chemistries. We provide details on dataset generation and pre-processing in Section Data.

Results

Integrity of the Boltzmann distribution under TITO dynamics in unseen small molecules and peptides

A defining property of molecular systems undergoing Langevin dynamics is convergence to the Boltzmann distribution. In contrast, when a generative model is trained to approximate time–integrated transition probabilities, this guarantee is no longer automatic. The central question is therefore whether TITO preserves physical realism—whether it samples configurations consistent with the Boltzmann distribution—or instead produces unphysical states, analogous to large language models generating text that is fluent but factually incorrect.

To test this, we examined whether the Boltzmann distribution, $\mu \propto \exp(-\beta U(\mathbf{x}))$, of the potential energy function, U, at inverse temperature β is the invariant measure (i.e., stationary distribution) of the transfer operator implicitly learned by TITO (*36*). Because the learned operator is not directly accessible, we assessed this property numerically. Specifically, we drew

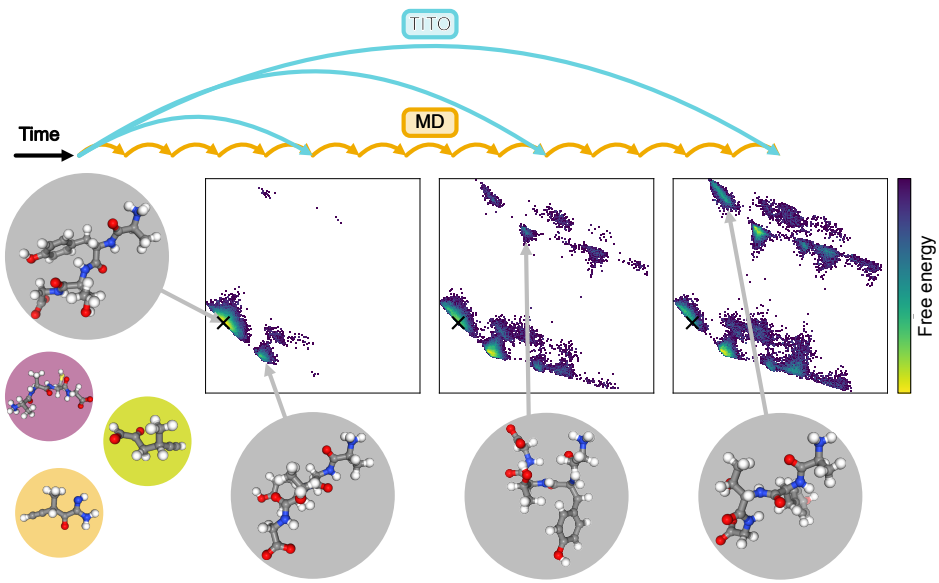


Figure 1: **Transferable Implicit Transfer Operators (TITO)**: A multi–time-scale surrogate model for molecular dynamics that is transferable across systems. Starting from an initial condition (black cross), TITO generates molecular dynamics ensembles for diverse molecules at arbitrary lag times.

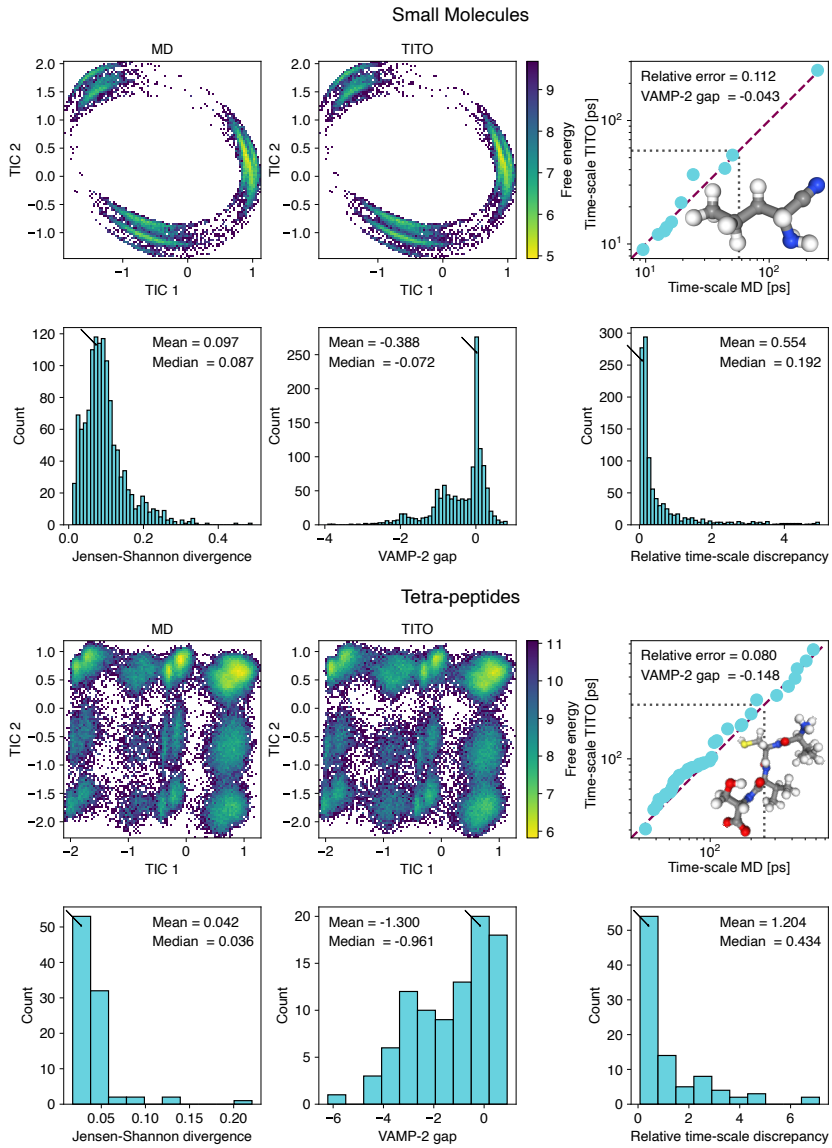


Figure 2: **TITO accurately predicts both thermodynamics and kinetics.** Small molecules (top) and tetra-peptides (bottom). Top row: Projection onto the first two TICA components and comparison of VAMP timescales between MD and TITO-generated samples for a representative molecule. Bottom row: Aggregated evaluation across systems: Jensen–Shannon divergence of TICA projections (left), VAMP-2 gap (center), and top-10 relative error (right). Black arrows denote the position of the example molecule within each histogram.

initial conditions $\mathbf{x}_0 \sim \mu$ from long unbiased MD simulations, generated trajectories using $p(\mathbf{x}_{\Delta t} \mid \mathbf{x}_0)$ with TITO, and compared the empirical distributions of $\mathbf{x}_0$ and $\mathbf{x}_{\Delta t}$ using the Jensen–Shannon divergence (JSD) (*37*). Across a broad set of small molecules and tetrapeptides unseen during training, TITO reproduced the Boltzmann distribution obtained from reference MD simulations (Fig. 2). A small fraction of cases exhibited elevated JSD values, indicating discrepancies that could reflect either spurious (hallucinatory) samples or genuine metastable states not explored by reference simulations (Suppl. Fig. S1).

To probe these outliers, we constructed Koopman operator models from long unbiased MD and from TITO trajectories. These models characterize the system's slowest dynamical modes and associated metastable states. We found that TITO generally reproduced relaxation timescales of MD, suggesting that much of the JSD tail arises from minor numerical mismatches. However, a subset of systems displayed substantially slower relaxation times under TITO, as revealed by significantly larger VAMP (Variational Approach to Markov Processes) scores (*38*). Because theoretical results show that timescales are bounded from above (*39*), this suggests that TITO sampled metastable states not observed in the MD trajectories.

To evaluate whether these new states were physically meaningful, we performed extensive replica exchange (RE) MD simulations (*40*). On average, TITO covered all density regions visited by RE MD, whereas long conventional MD failed to do so in a significant fraction of systems (Fig. 3A). States detected by TITO but absent in long MD were consistently recovered in RE MD (Suppl. Fig. S2), confirming that they correspond to genuine metastable basins. In one representative example, propiolamide, TITO uncovered a metastable basin absent from long MD but corroborated by both ultra-long unbiased MD and RE MD simulations (Fig. 3C and Suppl. Figs. S3 and S4). Remarkably, although trained only on nanosecond MD data, TITO correctly inferred an exchange timescale between basins on the order of microseconds—consistent with estimates from ultra-long trajectories.

Intriguingly, TITO also predicted additional states not observed in either reference method (Fig. 3B). To further investigate these cases, we initialized ensembles of nanosecond-length unbiased MD trajectories from TITO-generated configurations. These simulations exhibited a small but systematic improvement in agreement with RE MD (Suppl. Fig. S5), indicating that TITO samples near-physical configurations capable of relaxing into correct basins under explicit dynamics. We then assessed the stability of these newly identified states in ensemble simulations and found that the configurations remained metastable (Fig. S6), suggesting that they represent physically valid states rather than artifacts of the learned dynamics. Collectively, these findings demonstrate that TITO not only preserves the integrity of the Boltzmann distribution but also uncovers metastable states that would likely remain undetected using conventional methods within practical computational limits.

We further examined whether TITO's generalization correlates with chemical similarity between training and test molecules. Surprisingly, no such correlation was observed in either of the data-sets (Suppl. Figs. S7 and S8 and Suppl. Table S1). This lack of correlation suggests that chemical composition alone provides limited signal for guiding iterative refinement or active learning of generative dynamical models, underscoring the need for alternative strategies to improve generalization.

TITO faithfully reproduces relaxation transients in unseen molecular systems

Next, we investigate whether the dynamics generated by TITO is statistically equivalent to that generated by numerical MD simulations. Since we here target MD in the NVT ensemble, the dynamics are stochastic, and consequently, we use statistical tools to quantitatively compare the two approaches.

A stringent test of dynamical fidelity is whether a model can reproduce relaxation processes

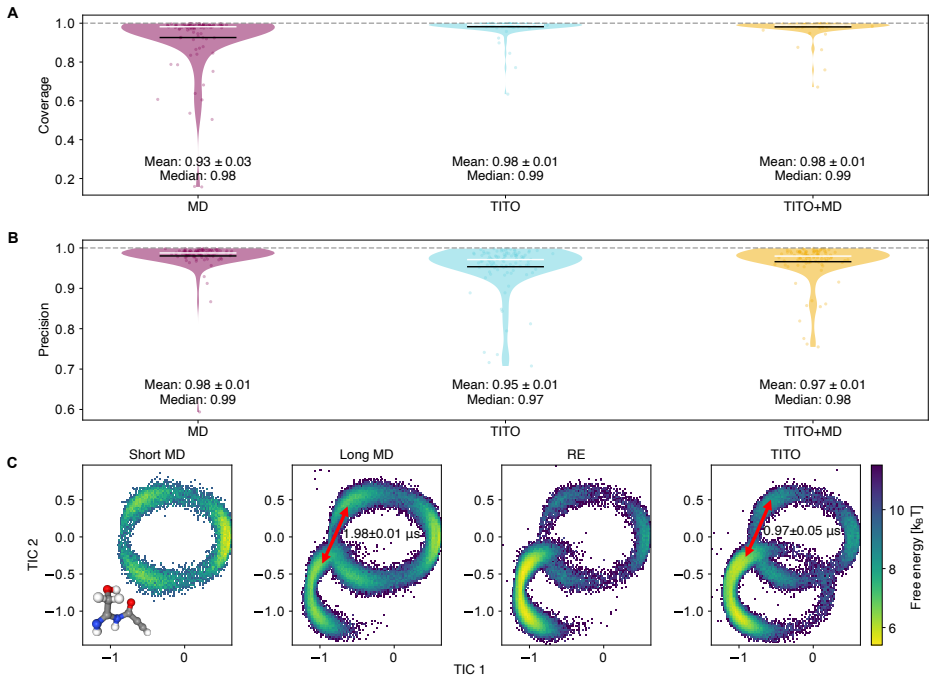


Figure 3: **TITO accurately samples two orders of magnitude slower dynamics than the training data.** (A) Coverage and (B) precision, see Evaluation metrics section, w.r.t to RE of projections onto the first two TICs for training set-like simulations (MD, 36.5 ns), TITO (32 μs) and an ensemble of ultra-short MD initialized with TITO samples (10 ps per sample). TITO achieves better coverage of the conformational space than MD. (C) Example propiolamide test-set molecule: MD fails to sample the most meta-stable basin, while TITO (160 μs, initialized from MD) successfully transitions into the dominant basin, recovers the Boltzmann distribution and estimates the right order of magnitude for the corresponding VAMP implied time-scale, estimated from ultra-long MD (16 ms). Structural insights of this transition and alternative TICA projections are provided in Suppl. Figs. S3 and S4, respectively.

across systems that differ vastly in their intrinsic timescales. As noted TITO accurately recapitulates relaxation dynamics in molecules and peptides outside its training set, and crucially provides quantitative predictions spanning over three orders of magnitude in characteristic times (Suppl. Fig. S9). This level of agreement indicates that TITO has learned an effective and generalizable representation of the underlying stochastic dynamics rather than merely fitting short-time correlations.

Still, matching timescales alone does not guarantee that the associated motions are physically meaningful. To examine this, we performed extensive TITO simulations and compared the full relaxation transients of slow dynamical modes with those obtained from long, unbiased MD trajectories. Across two orders of magnitude in timescale, the agreement was striking, suggesting that TITO reproduces both the rate and the mechanism of the underlying molecular dynamics (Fig. 4A). This demonstrates that the learned transition operator generalizes dynamically, faithfully capturing the hierarchy of molecular motions that govern relaxation kinetics.

A further requirement is internal consistency across time resolutions. Because TITO predicts time-integrated dynamics at multiple timescales, the resulting transients must be consistent regardless of whether they are generated in a single long step or as a sequence of shorter steps (nested sampling). We find that the relaxation transients remain self-consistent under this test (Fig. 4A), suggesting that the learned transition density satisfies the Chapman–Kolmogorov equation and thus encodes genuinely Markovian dynamics.

Finally, we investigated whether the high fidelity of slow dynamics is achieved at the expense of accuracy in fast, rapidly relaxing modes. Remarkably, despite operating at timesteps orders of magnitude larger than those of bond and angle vibrations, TITO accurately reproduced equilibrium properties (Fig. 4B) and generated conformers with potential energies closely matching those from unbiased MD (Fig. 4C). The main deviation we observed was a slight underestimation of the variance of fast modes, which in turn leads to systematically lower potential

energies relative to reference simulations.

Qualitative extrapolation to larger peptide systems

We next examined whether TITO can extrapolate beyond the molecular sizes represented in its training data. This setting provides a stringent test of transferability: a model trained on short peptides must infer effective dynamics at new length scales, where both the number of atoms and the hierarchy of internal motions increase substantially. Specifically, we applied a model trained only on tetrapeptides to generate trajectories for penta-, hexa-, hepta-, and octapeptides.

Extrapolation introduces a scale mismatch in the latent base distribution p_0, whose variance depends on system size. To mitigate this, we rescaled the standard deviation of p_0 according to Flory's scaling law for the radius of gyration of random polymers, $\langle R_g \rangle \propto N^{0.688}$ (*41*), where N denotes the number of residues. Guided by this simple physical prior, TITO produced configurations with realistic local geometry and global compactness across peptide lengths; without this correction, stable extrapolation beyond pentapeptides was not achievable.

With the scaling correction in place, TITO approximately recovered the conformational landscapes of larger peptides and reproduced relaxation times qualitatively consistent with explicit MD, even when the sequence length was doubled relative to the training systems (Fig. 5). For the largest peptides, however, the generated trajectories exhibited mild structural compaction (Suppl. Fig. S10) and a systematic downward drift in potential energy (Suppl. Fig. S11), leading to instability in long nested-sampling runs. These deviations likely arise from cumulative local errors that are amplified at increasing system sizes. Nevertheless, the generated configurations remain physically meaningful and can be readily refined by short low-cost MD equilibrations.

Together, these results demonstrate that TITO captures transferable physical principles sufficient to generalize far beyond its training domain, while also delineating the limits of such

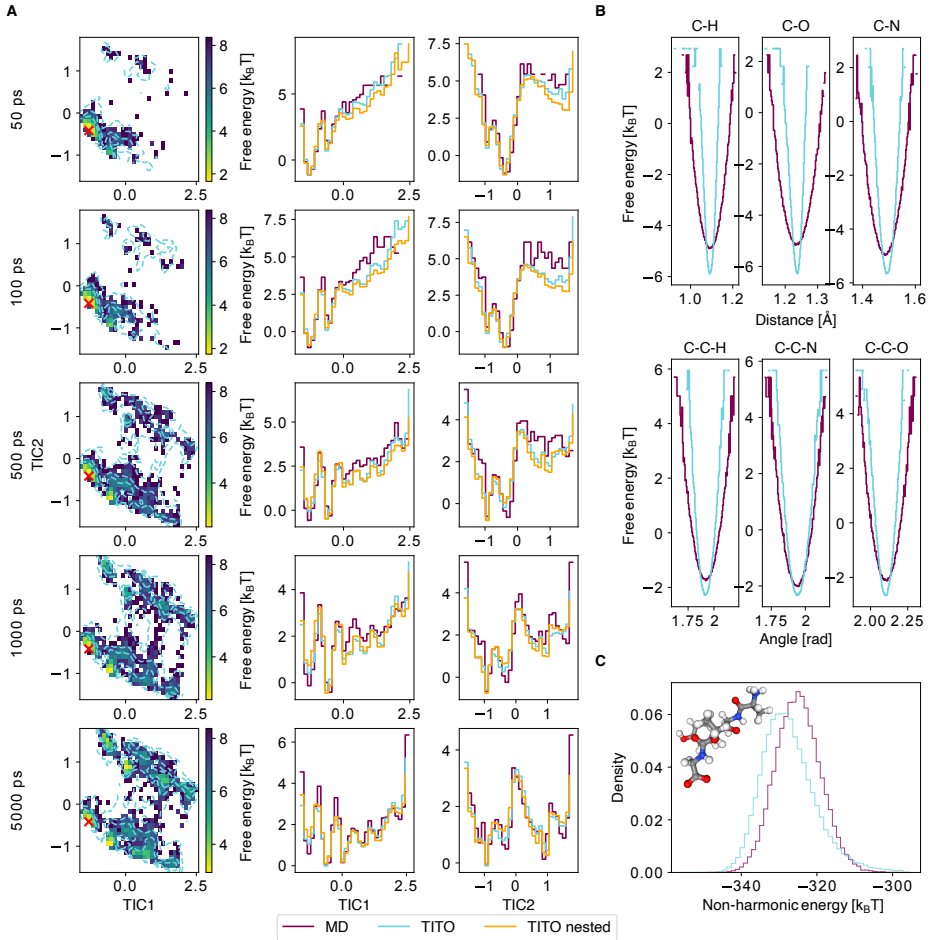


Figure 4: **TITO recapitulates transients, fast vibrations and potential energies.** (**A**) Free energy of the transition probability estimated with TITO for AYTG (test set) at increasing lag times (top to bottom). The first column shows conditional free energies projected onto the first two TICA components (TICs). Contours represent TITO samples, while 2D histograms correspond to Markov state model estimates from MD data. The red cross marks the simulation's initial state. The second and third columns display marginal distributions along each TIC. Nested samples are generated with 5 steps. (**B**) Free energy profiles of bond distances (top) and angles (bottom). (**C**) Probability density of the non-harmonic potential energy.

extrapolation. The observed degradation at large sizes naturally motivates hybrid divide-and-conquer strategies, in which long TITO propagation steps are interleaved with brief MD equilibration phases—akin to hybrid Monte Carlo or multi-resolution simulation schemes (*34,42,43*).

Calibration of simulation accuracy to compute budget

For practical impact, TITO must deliver substantially higher throughput than conventional MD while retaining quantitative accuracy in equilibrium and dynamical properties. Two factors determine the effective simulation throughput: (i) the molecular size, which constrains the number of simulations that can be run in parallel on a GPU with fixed memory, and (ii) the number of ODE solver steps required for each CNF evaluation, which controls the cost per TITO step.

We find that equilibrium properties can be reproduced at comparatively low computational budgets, whereas accurate estimation of relaxation time scales requires additional solver steps and hence higher cost (Suppl. Suppl. Fig. S12). This trade-off implies that the compute budget can be calibrated to match the target application, for example, prioritizing structural ensemble generation versus reproducing kinetic observables measured in experiment.

To quantify achievable throughput, we report the maximum simulation time reached on a single GPU. As shown in Table 1, TITO attains approximately 10 miliseconds of physical simulation time per day of computation, representing a four-order-of-magnitude improvement relative to standard unbiased MD simulations using the same resources. These results suggests that TITO can be tuned flexibly: users may trade simulation fidelity against throughput depending on the level of accuracy required. Further, we emphasize that these gains can potentially be even larger if the model is trained to predict larger time-steps, paving the way to study ultra-slow processes in biology and and material science.

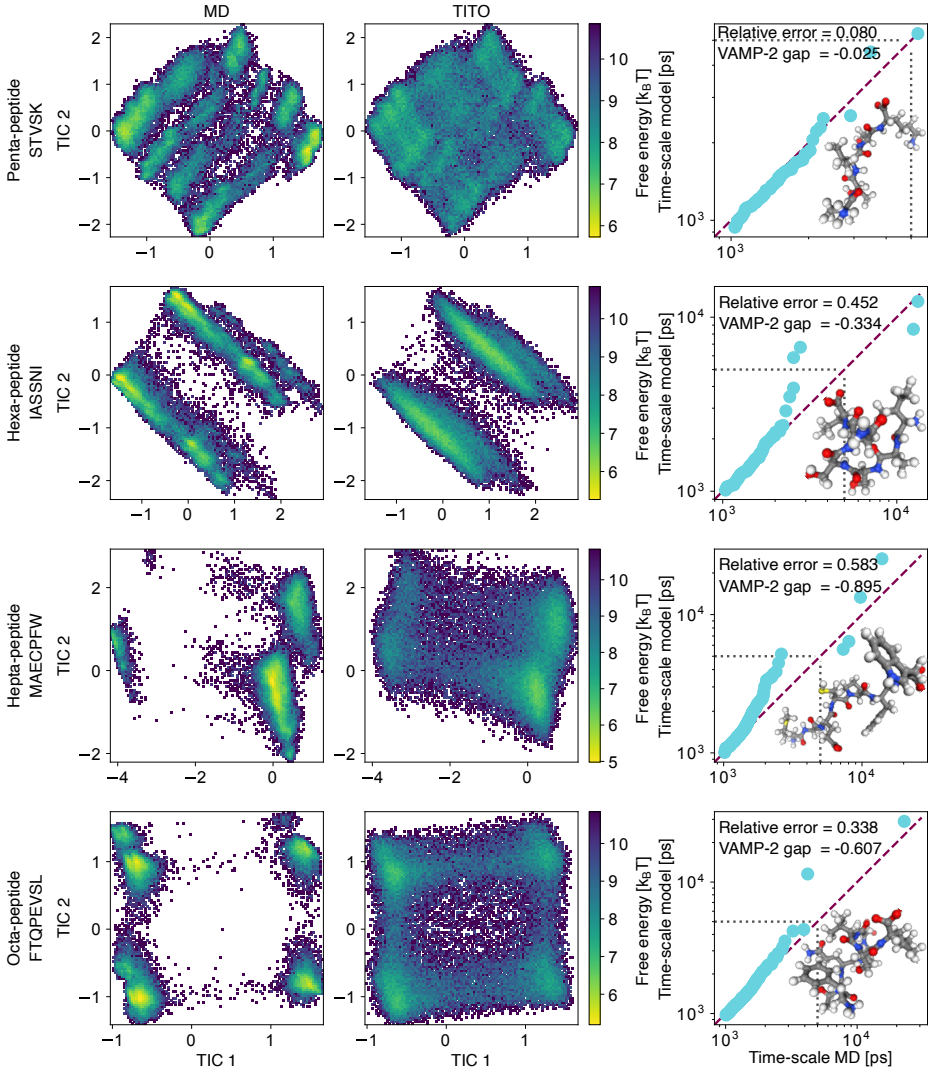


Figure 5: **Extrapolation to larger systems.** Free energy landscape and VAMP time scales of a TITO model trained on tetra-peptides and performing 5 ns single-step sampling for penta, hexa and hepta and octa- peptides.

Systems	Method	Simulation throughput/day
Small molecules	MD	3.5 μs
	TITO	11.3 ms
Tetra-peptides	MD	0.67 s
	TITO	10.3 ms

Table 1: **Maximum simulation time throughput.** Estimates using one NVIDIA A100 80 GB GPU for one day.

Discussion and Conclusion

We introduce Transferable Implicit Transfer Operators (TITO), a chemically generalizable generative model that reproduces molecular dynamics at a fraction of the computational cost of traditional simulations. TITO quantitatively recovers both the equilibrium probabilities of molecular configurations and the rates and mechanisms of conformational exchange across diverse chemistries, from small molecules to peptides. In essence, it delivers the fidelity of molecular dynamics at the cost of sampling a deep generative model.

TITO achieves this by learning the statistics of time-integrated dynamics directly from simulation data, allowing propagation over arbitrarily long lag times without explicit numerical integration at femtosecond resolution. This formulation yields an acceleration of up to four orders of magnitude in the quantitative characterization of equilibrium states and relaxation kinetics at compute cost parity. Furthermore, TITO retains predictive power beyond its training regime, qualitatively reproducing thermodynamic and kinetic behavior in peptides up to twice the size of those used for training highlighting its ability to extrapolate across molecular size.

TITO differs fundamentally from dominant paradigms in generative models of molecular dynamics, Boltzmann Generators which aim to quantitatively sample the independent equilibrium samples from the Boltzmann distribution (*44*) and Boltzmann Emulators which sacrifice quantitative alignment with MD to boost efficiency and scaling (*34, 45, 46*). These methods, and in particular their transferable variants (*34, 46, 47*), are rapidly becoming a viable complement

to MD simulations when equilibrium properties are the target of investigation. However, these approaches cannot capture dynamic properties such as rates or mechanisms.

Other methods that aim to predict the dynamics provide mainly qualitative insights and at fixed time scales (*35, 48*). To our knowledge, TITO is the first framework to achieve physically realistic, multi–timescale sampling with demonstrated transferability across both chemical composition and molecular size. Other complementary strategies, include machine-learning infused path-sampling (*49*) strategies or latent space simulators (*50–52*) show promise in scaling to larger systems, but generalization remains an open challenge requiring careful modeling and calibration for every specific process of interest.

Machine learned interatomic potentials (*53,54*) and coarse-grained force-fields (*55,56*) have similarly shown impressive strides towards general purpose transferability. These models can guarantee realistic physical dynamics, depending on the integration strategy chosen. So while they might boost the accuracy over current force-fields and coarse-grained models, they still rely on iterative numerical integration with tiny time-steps making their computational footprint significant. TITO instead offers a paradigm shift: bypassing iterative integration altogether.

Despite these advances, important limitations remain. At present, TITO is restricted to implicit solvent representations and system sizes of at most a few hundred atoms. Extending the method to explicitly solvated biomolecules with tens to hundreds of thousands of degrees of freedom will require innovations in neural architectures (*57–59*) and/or hierarchical strategies such as coarse-graining (*60*). In addition, periodic boundary conditions—essential for realistic modeling of solvated systems—are not yet supported. While our experiments show promising extrapolation to larger and chemically distinct molecules, generalization performance still depends on the chemical similarity between target and training systems. Achieving broad chemical coverage will necessitate larger and more diverse training datasets. Finally, TITO is presently limited to a single thermodynamic state (NVT ensemble at room temperature). Ex-

tending thermodynamic transferability, including across temperatures or pressures, would enable the study of the influence of thermodynamic perturbations on molecular systems stationary and dynamic properties (*61, 62*).

Intriguingly, we find that TITO's generalization performance shows no clear relationship to the chemical similarity between training and test systems. This observation challenges the prevailing assumption that broader chemical coverage alone ensures generalization. Instead, it suggests that the structure and diversity of training data—how well they represent relevant dynamical motifs and energy landscapes—may be more critical than sheer data volume. In this view, progress may hinge less on scaling to ever-larger simulation datasets and more on carefully curated, mechanistically diverse benchmarks that capture the essential physics of molecular dynamics.

In summary, TITO establishes a new paradigm for transferable generative modeling of molecular dynamics, unifying thermodynamic sampling and dynamical prediction in a singular generative surrogate. By enabling accelerated and chemically transferable estimation of stationary and dynamic properties—such as free energies and rates—TITO paves the way toward practical deep-learning–based acceleration of molecular simulations.

Acknowledgements

This work was partially supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation. Model training and inference was made possible by an allocation on the Berzelius resource provided by the Knut and Alice Wallenberg Foundation at the National Supercomputer Centre hosted by the National Academic Infrastructure for Supercomputing in Sweden (NAISS) (project: Berzelius-2025-189), partially funded by the Swedish Research Council through grant agreement no. 2022-06725.

Materials and methods

Data

We use three datasets covering different regions of chemical space:

1. **Small molecules**: The MDQM9-nc dataset (*34*) contains MD simulations for 12,530 small non-cyclic molecules from the QM9 dataset (*63*). Simulations are performed in a vacuum, at room temperature using the GAFF force field (*64*). Simulation time is dependent on the molecule size with a median sampling time of 36.5 ns. We perform extra 1 $\mu\mathrm{s}$ RE simulations across 8 temperatures (300, 400, 500, 600, 700, 800, 900 and 1000 K). The average exchange rate 58 $\%$.

2. **Tetra-peptides**: The Timewarp dataset (*35*) contains MD simulations for tetra-peptides. It contains two tetra-peptides sub-datasets, large, which contains train 1457 molecules and huge, with 92 larger molecules . We use large as training data and huge as test set. The simulations are performed in implicit water and at room temperature. Simulation time is 50 mns for training set molecules and 1 $\mu\mathrm{s}$ for test set molecules.

3. **Larger systems**: We performed 1 $\mu\mathrm{s}$ simulation of penta-, hexa-, hepta- and octa-peptides with the same simulation parameters as in the Timewarp dataset. For each peptide length, six sequences were randomly sampled based on vertebrate amino acid frequencies.

Model

TITO uses equivariant optimal transport flow matching to parameterize the transition probability. Flow matching (*31*) provides an efficient framework for training continuous normalizing flows by aligning a learnable velocity field with the optimal transport velocity field between an easy to sample base distribution p_0 and a target distribution p_1. Rather than directly minimizing a divergence between the generated distribution and p_1, flow matching constructs in-

termediate states along an interpolation between p_0 and p_1. The neural ODE vector field is then trained to predict the conditional displacement between these paired states. We use linear interpolants (*65*),

$$\mathbf{x}^T = (1 - T)\,\mathbf{x}^0 + T\mathbf{x}^1, \quad T \in [0, 1],$$

with target velocity,

$$\mathbf{v}^T = \frac{d\mathbf{x}^T}{dT} = \mathbf{x}^1 - \mathbf{x}^0.$$

Molecular distributions live in a space with inherent symmetries, such as rotational and permutational invariances of atomic coordinates. Equivariant optimal transport (*32*) incorporates these symmetries to construct shorter paths by aligning sample pairs, $\mathbf{x}^0$ and x^1, along their symmetry orbits. In practice, this minimization is approximated sequentially: the optimal permutation by solving a linear sum assignment problem (*66*), followed by an optimal superpositioning through solving a Procrustes problem (*67*).

For the velocity field model, $\mathbf{v}_\theta$, we use a modified SE3-ITO architecture (*30*) enriched with edge features encoding interaction types between atoms. Specifically, we distinguish single, double, triple, and through-space bonds or interactions. As in SE3-ITO, we assume a complete interaction graph where every atom interacts with all others, with bonded interactions prioritized according to the order listed above.

Model training and inference parameters for different experiments are included in Suppl. Tables S2 and S3, respectively.

Evaluation metrics

Jensen-Shannon Divergence (JSD) The Jensen-Shannon Divergence provides a symmetric measure of similarity between two probability distributions. Given two distributions p and q, the JSD is defined as

$$\mathrm{JSD}(p \parallel q) = \frac{1}{2} D_{\mathrm{KL}}(p \parallel m) + \frac{1}{2} D_{\mathrm{KL}}(q \parallel m),$$

where $m = \frac{1}{2}(p + q)$ and D_{KL} denotes the Kullback–Leibler divergence. Unlike the KL divergence, the JSD is bounded between 0 and 1.

Free energy The free energy (negative log-likelihood) of a sub-space of the conformational space $\Omega_i \in \Omega$ is

$$F(\Omega_i) = -k_B T \ \log(p(\Omega_i)) \text{ with } p(\Omega_i) = \int_{\Omega_i} p(\mathbf{x})\, \mathrm{d}\mathbf{x}.$$

Coverage and precision. Coverage and precision quantify the degree of overlap between the probability distributions generated by two sampling methods. Given two methods, m_1 and m_2, the coverage of m_1 with respect to m_2 is defined as

$$\mathrm{COV}_{m_1,m_2} = \int_{\Omega_{p_{m_1} \cap p_{m_2}}} p_{m_2}(\mathbf{x})\, \mathrm{d}\mathbf{x},$$

where p_{m_i} denotes the probability mass sampled from method m_i, and $\Omega_{p_{m_1} \cap p_{m_2}} = \{\mathbf{x} : p_{m_1}(\mathbf{x}) > \delta \text{ and } p_{m_2}(\mathbf{x}) > \delta\}$. Precision is defined analogously as

$$\mathrm{PRE}_{m_1,m_2} = \int_{\Omega_{p_{m_1} \cap p_{m_2}}} p_{m_1}(\mathbf{x})\, \mathrm{d}\mathbf{x}.$$

Intuitively, coverage measures how much of the probability mass of m_2 is captured by m_1, while precision measures the fraction of m_1's probability mass supported by m_2. In practice, we estimate these quantities from empirical histograms, defining $\Omega_{p_{m_1} \cap p_{m_2}}$ as the set of discrete states where both methods have nonzero counts.

Variational Approach for Markov Processes (VAMP) The VAMP framework provides a principled method for evaluating the quality of dynamical models based on the variational principle of conformation dynamics. We employed the following VAMP-based metrics:

Implied timescales The eigenvalues of the estimated transfer operator were used to compute implied timescales, defined as

$$t_i = -\frac{\tau}{\ln \sigma_i},$$

where σ_i is the i-th singular value of the Koopman operator approximation and τ the lag time.

Relative time-scale discrepancy We define the relative time-scale discrepancy as

$$\epsilon_{rel} = \frac{1}{N} \sum_{i}^{N} \frac{|t_i^{MD} - t_i^{TITO}|}{t_i^{MD}},$$

where t_i^{MD} and t_i^{TITO} are implied time scales predicted with MD and TITO respectively and are sorted in decreasing order. We use $N = 10$ implied-time scales throughout this work.

VAMP-2 score The VAMP-2 score is the squared Frobenius norm of the singular value spectrum,

$$\text{VAMP-2}^{(k)} = \sum_{i=1}^{k} \sigma_i^2.$$

Higher scores indicate that the model captures slow dynamical modes.

VAMP-gap We define the VAMP-gap as the difference in VAMP2-scores between TITO and the MD,

$$\text{VAMP-gap} = \text{VAMP2-score}_{\text{MD}} - \text{VAMP2-score}_{\text{TITO}}.$$

Negative VAMP-gaps indicate TITO predicts slower dynamics and vice versa.

Together, these evaluation metrics provide complementary insights: the Jensen-Shannon Divergence and free energy excess measures how well the model reproduces equilibrium distributions, while VAMP metrics assess the model's fidelity in capturing slow dynamical processes and metastability.

References

1. A. G. Palmer III, *Annual Review of Biophysics and Biomolecular Structure* **30**, 129–155 (2001).

2. B. Schuler, *Journal of Nanobiotechnology* **11**, S2 (2013).

3. B. Leimkuhler, C. Matthews, *Molecular Dynamics: With Deterministic and Stochastic Numerical Methods* (Springer International Publishing, 2015).

4. M. Karplus, J. A. McCammon, *Nature Structural Biology* **9**, 646–652 (2002).

5. C. Schütte, F. Noe, E. Meerbach, P. Metzner, C. Hartmann, *Proc. Int. Congr. ICIAM* pp. 297–336 (2009).

6. J. Hénin, T. Lelièvre, M. R. Shirts, O. Valsson, L. Delemotte, *Living J. Comput. Mol. Sci* **4** (2022).

7. A. Laio, M. Parrinello, *Proceedings of the National Academy of Sciences* **99**, 12562 (2002).

8. G. Torrie, J. Valleau, *Journal of Computational Physics* **23**, 187 (1977).

9. H. Grubmüller, *Phys. Rev. E* **52**, 2893 (1995).

10. D. Branduardi, F. L. Gervasio, M. Parrinello, *The Journal of Chemical Physics* **126** (2007).

11. G. Pérez-Hernández, F. Paul, T. Giorgino, G. De Fabritiis, F. Noé, *The Journal of Chemical Physics* **139** (2013).

12. P. Tiwary, B. J. Berne, *Proceedings of the National Academy of Sciences* **113**, 2839–2844 (2016).

13. W. Chen, A. L. Ferguson, *Journal of Computational Chemistry* **39**, 2079–2102 (2018).

14. A. F. Voter, *Physical Review Letters* **78**, 3908–3911 (1997).

15. M. Tuckerman, B. J. Berne, G. J. Martyna, *The Journal of Chemical Physics* **97**, 1990–2001 (1992).

16. K. A. Feenstra, B. Hess, H. J. C. Berendsen, *Journal of Computational Chemistry* **20**, 786–798 (1999).

17. B. Leimkuhler, C. Matthews, *The Journal of Chemical Physics* **138** (2013).

18. D. E. Shaw, *et al.*, *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, SC '21 (ACM, 2021), p. 1–11.

19. I. Ohmura, G. Morimoto, Y. Ohno, A. Hasegawa, M. Taiji, *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* **372**, 20130387 (2014).

20. D. E. Shaw, *et al.*, *Science* **330**, 341–346 (2010).

21. M. Shirts, V. S. Pande, *Science* **290**, 1903–1904 (2000).

22. P. Eastman, *et al.*, *The Journal of Physical Chemistry B* **128**, 109–116 (2023).

23. M. J. Abraham, *et al.*, *SoftwareX* **1–2**, 19–25 (2015).

24. M. J. Harvey, G. Giupponi, G. D. Fabritiis, *Journal of Chemical Theory and Computation* **5**, 1632–1639 (2009).

25. D. A. Case, *et al.*, *Journal of Chemical Information and Modeling* **63**, 6183–6191 (2023).

26. J.-H. Prinz, *et al.*, *The Journal of Chemical Physics* **134**, 174105 (2011).

27. G. R. Bowman, V. S. Pande, F. Noé, *An Introduction to Markov State Models and Their Application to Long Timescale Molecular Simulation* (Springer, Dordrecht, Netherlands, 2014).

28. P. Langevin, *C. R. Acad. Sci. (Paris)* **146**, 530– (1908).

29. H. Risken, *The Fokker-Planck Equation: Methods of Solution and Applications* (Springer Berlin Heidelberg, 1996).

30. M. Schreiner, O. Winther, S. Olsson, *Advances in Neural Information Processing Systems*, A. Oh, *et al.*, eds. (Curran Associates, Inc., 2023), vol. 36, pp. 36449–36462.

31. Y. Lipman, R. T. Q. Chen, H. Ben-Hamu, M. Nickel, M. Le, *The Eleventh International Conference on Learning Representations* (2023).

32. L. Klein, A. Krämer, F. Noe, *Advances in Neural Information Processing Systems*, A. Oh, *et al.*, eds. (Curran Associates, Inc., 2023), vol. 36, pp. 59886–59910.

33. R. T. Q. Chen, Y. Rubanova, J. Bettencourt, D. K. Duvenaud, *Advances in Neural Information Processing Systems*, S. Bengio, *et al.*, eds. (Curran Associates, Inc., 2018), vol. 31.

34. J. Viguera Diez, S. Romeo Atance, O. Engkvist, S. Olsson, *Machine Learning: Science and Technology* **5**, 025010 (2024).

35. L. Klein, *et al.*, *Advances in Neural Information Processing Systems*, A. Oh, *et al.*, eds. (Curran Associates, Inc., 2023), vol. 36, pp. 52863–52883.

36. J. V. Diez, M. J. Schreiner, O. Engkvist, S. Olsson, *The Thirteenth International Conference on Learning Representations* (2025).

37. J. Lin, *IEEE Transactions on Information Theory* **37**, 145–151 (1991).

38. H. Wu, F. Noé, *Journal of Nonlinear Science* **30**, 23–66 (2019).

39. F. Nüske, B. G. Keller, G. Pérez-Hernández, A. S. J. S. Mey, F. Noé, *Journal of Chemical Theory and Computation* **10**, 1739–1752 (2014).

40. D. J. Earl, M. W. Deem, *Phys. Chem. Chem. Phys.* **7**, 3910 (2005).

41. P. Flory, *Principles of Polymer Chemistry*, Baker lectures 1948 (Cornell University Press, 1953).

42. J. P. Nilmeier, G. E. Crooks, D. D. L. Minh, J. D. Chodera, *Proceedings of the National Academy of Sciences* **108** (2011).

43. G. A. Ross, *et al.*, *Journal of Chemical Theory and Computation* **16**, 6061–6076 (2020).

44. F. Noé, S. Olsson, J. Köhler, H. Wu, *Science* **365** (2019).

45. B. Jing, G. Corso, J. Chang, R. Barzilay, T. Jaakkola, *Advances in Neural Information Processing Systems*, S. Koyejo, *et al.*, eds. (Curran Associates, Inc., 2022), vol. 35, pp. 24240–24253.

46. S. Lewis, *et al.*, *Science* **389** (2025).

47. L. Klein, F. Noé, *Advances in Neural Information Processing Systems*, A. Globerson, *et al.*, eds. (Curran Associates, Inc., 2024), vol. 37, pp. 45281–45314.

48. B. Jing, H. Stärk, T. Jaakkola, B. Berger, Generative modeling of molecular dynamics trajectories (2024).

49. H. Jung, *et al.*, *Nature Computational Science* **3**, 334–345 (2023).

50. H. Sidky, W. Chen, A. L. Ferguson, *Chemical Science* **11**, 9459–9467 (2020).

51. P. R. Vlachas, J. Zavadlav, M. Praprotnik, P. Koumoutsakos, *Journal of Chemical Theory and Computation* **18**, 538–549 (2021).

52. D. Wang, Y. Wang, L. Evans, P. Tiwary, *Journal of Chemical Theory and Computation* **20**, 3503–3513 (2024).

53. A. Kabylda, *et al.*, *Journal of the American Chemical Society* (2025).

54. I. Batatia, *et al.*, A foundation model for atomistic materials chemistry (2023).

55. N. E. Charron, *et al.*, *Nature Chemistry* **17**, 1284–1292 (2025).

56. M. Majewski, *et al.*, *Nature Communications* **14** (2023).

57. J. T. Frank, S. Chmiela, K.-R. Müller, O. T. Unke, Euclidean fast attention: Machine learning global atomic representations at linear cost (2024).

58. R. Irwin, A. Tibo, J. P. Janet, S. Olsson, Semlaflow – efficient 3d molecular generation with latent attention and equivariant flow matching (2025).

59. J. Cremer, *et al.*, Flowr: Flow matching for structure-aware de novo, interaction- and fragment-based ligand generation (2025).

60. J. Wang, *et al.*, *ACS Central Science* **5**, 755–767 (2019).

61. S. Moqvist, W. Chen, M. Schreiner, F. Nüske, S. Olsson, *Journal of Chemical Theory and Computation* **21**, 2535–2545 (2025).

62. L. Herron, K. Mondal, J. S. Schneekloth, P. Tiwary, *Proceedings of the National Academy of Sciences* **121** (2024).

63. M. Rupp, A. Tkatchenko, K.-R. Müller, O. A. von Lilienfeld, *Physical Review Letters* **108**, 058301 (2012).

64. J. Wang, R. M. Wolf, J. W. Caldwell, P. A. Kollman, D. A. Case, *Journal of Computational Chemistry* **25**, 1157–1174 (2004).

65. M. S. Albergo, N. M. Boffi, E. Vanden-Eijnden, Stochastic interpolants: A unifying framework for flows and diffusions (2023).

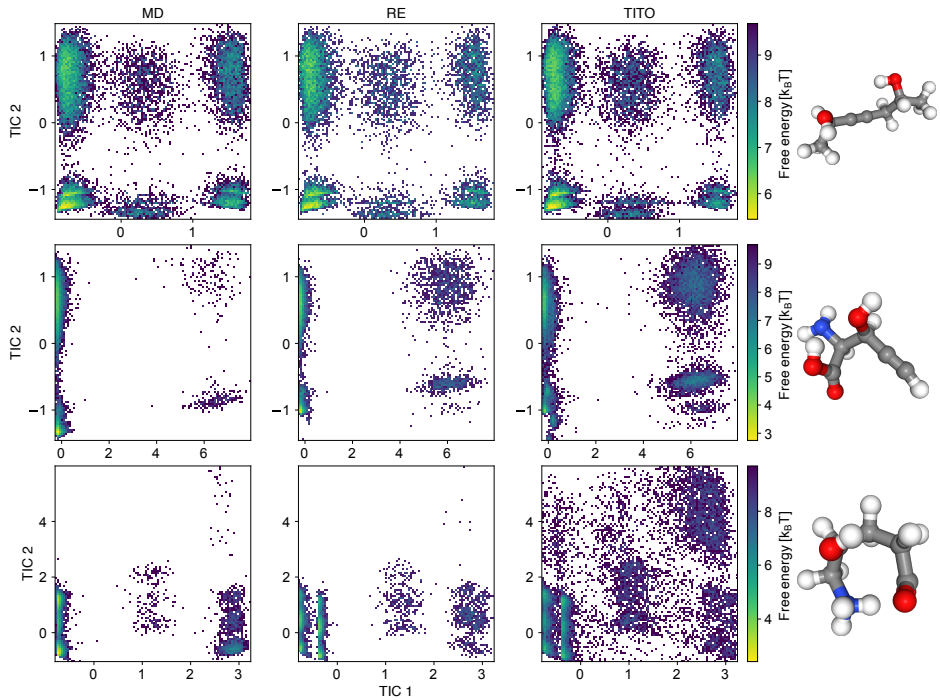
66. D. F. Crouse, *IEEE Transactions on Aerospace and Electronic Systems* **52**, 1679–1696 (2016).

67. W. Kabsch, *Acta Crystallographica Section A* **32**, 922–923 (1976).

Supplementary material

Examples of molecules with different Jensen-Shannon divergences

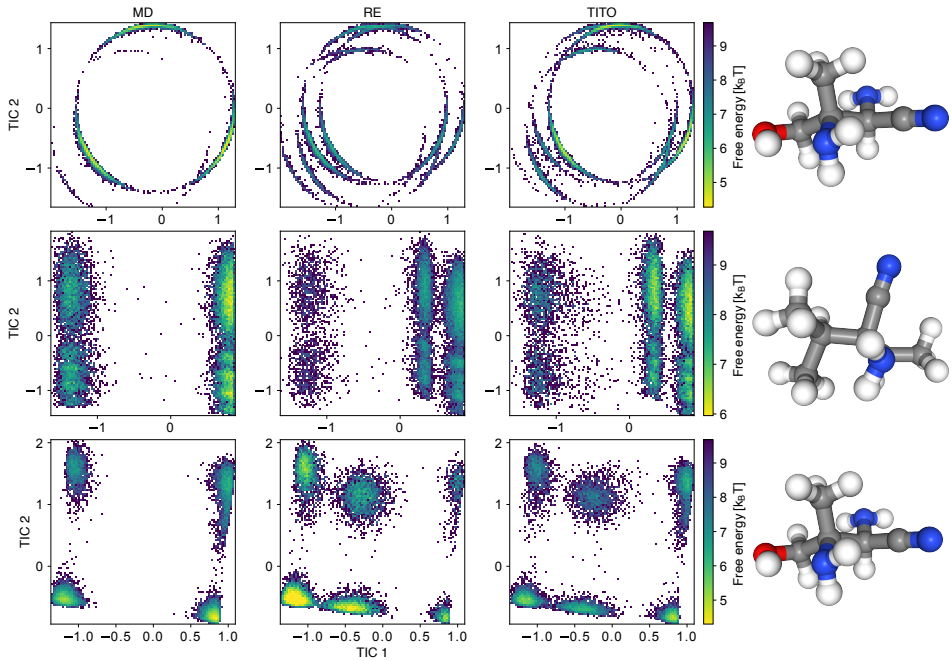
In Suppl. Fig. S1 we compare free energy landscape of MD, RE and TITO for different regions of JSD.



Supplementary Figure S1: Free energy landscape comparison between MD, RE and TITO for different regions of JSD. From top to bottom, the JSD are 0.09, 0.21 and 0.35.

TITO recovers states not accessible by training set-like simulations

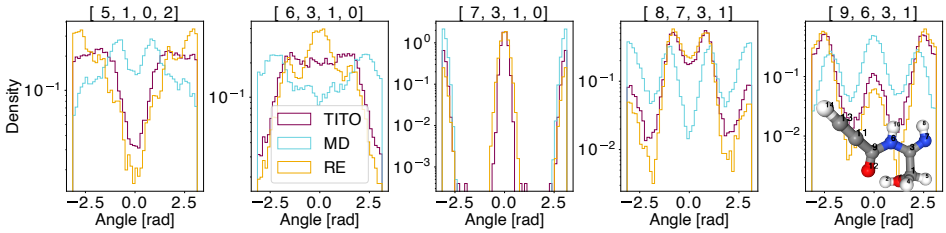
In Suppl. Fig. S2 we provide several examples of test molecules for which TITO is able to recover states sampled by RE, which are not accessible by training set-like simulations.



Supplementary Figure S2: Free energy landscape comparison between MD, RE and TITO of test set molecules for which TITO recovers states sampled by RE, but not accessible by training set-like MD simulations .

Overcoming MD time scales with TITO: Structural insights for propiolamide

In Suppl. Fig. S3 we show that the slowest process of propiolamide (Fig. 3C) involves a dihedral angle sign inversion over the bond 3-1 and a global re-arrangement of other dihedral angles in the molecule. Short MD fails to sample the transition, but TITO samples it and recovers an equilibrium distribution in high agreement with RE.



Supplementary Figure S3: Dihedral angles involved in slowest process of example molecule in Fig. 3 C. MD fails to sample the transition. TITO recovers the equilibrium distribution predicted by RE.

Alternative TICA projections

In Suppl. Fig. S4 we show alternative TICA projections using ultra long MD simulations to estimate TICA models of propiolamide.

MD fine-tuning time vs JS divergence

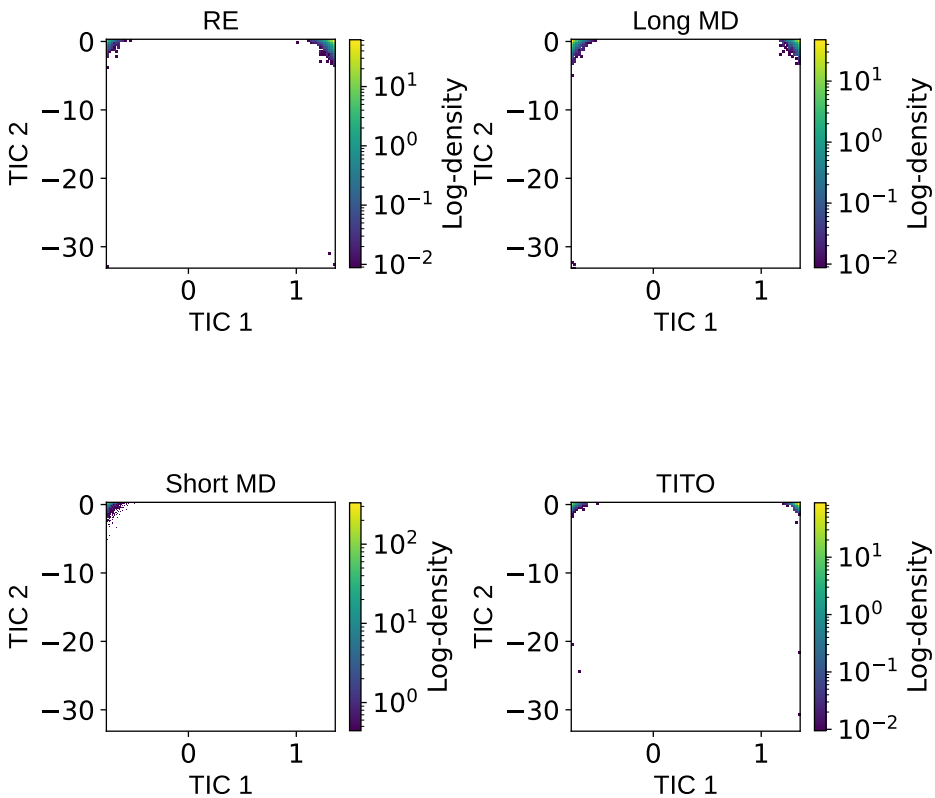
In Suppl. Fig. S5 we show the evolution of average Jensen-Shannon divergence w.r.t. the time that TITO samples are simulated with standard MD. Most of the reduction is achieved during the first 10 ps.

Examples of nanosecond meta-stable state predictions of TITO missing in RE

In Suppl. Fig. S6 we collect 4 example test set molecules for which TITO samples states are not present in RE or MD, but are stable after 1 ns (per sample) ensemble simulation.

Performance vs chemical similarity

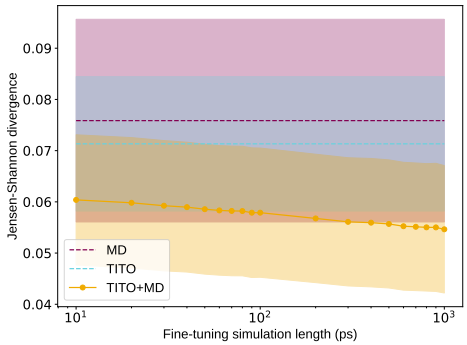
In Suppl. Figs. S7 and S8, and Table S1 we show that equilibrium distribution errors do not correlate with chemical dissimilarity.



Supplementary Figure S4: Alternative TICA projections using long MD simulations to estimate TICA models in Fig. 3C. MD fails to sample the transition. TITO recovers the equilibrium distribution predicted by RE and long MD.

Accurately sampling different fast and slow molecules

In Suppl. Fig. S9 we show that TITO accurately samples thermodynamic and kinetic properties of molecules whose slowest process ranges from ps to ns.



Supplementary Figure S5: Mean JS divergence and 95 % confindence interval for TITO, TITO + short MD and MD versus simulation time applied to sampled from TITO.

Hamming distance	0.25	0.75
Mean JS divergence	0.043	0.040
% of test peptides	64	26

Supplementary Table S1: Minimum Hamming dissimilarity of test set molecules w.r.t. training set vs Jensen-Shannon divergence. No correlation is observed

Peptide size extrapolation results

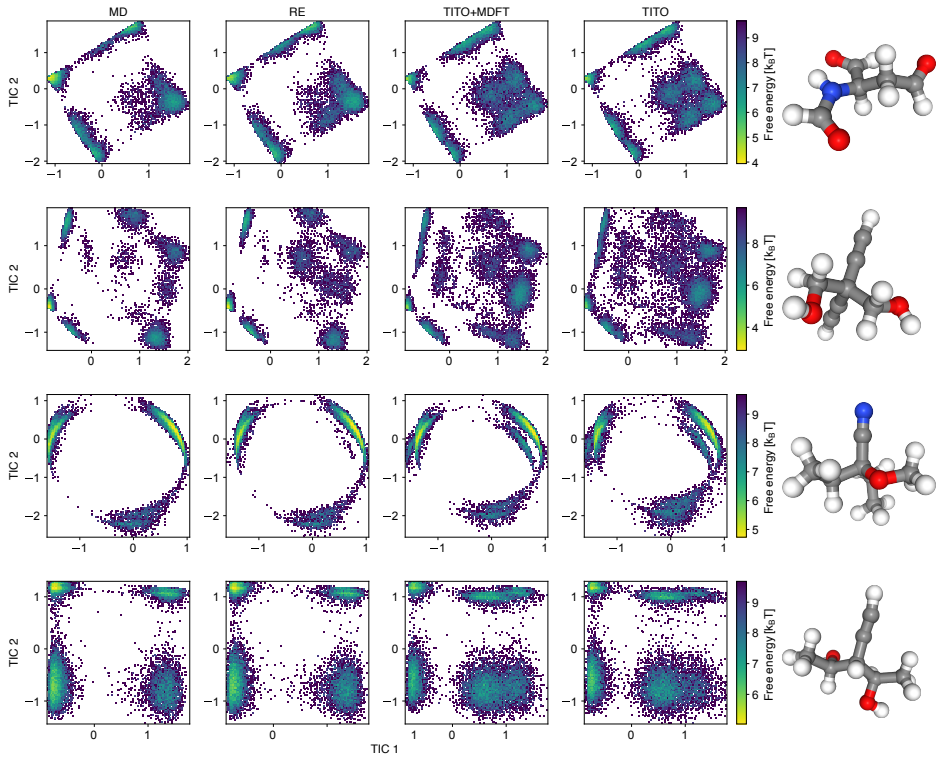
Scaling of radius of gyration with number of heavy atoms (S10) and potential energy distributions comparisons (S11) for MD and TITO models trained on tetra-peptides in the pentapeptides, hexapeptides, heptapeptides and octapeptides.

Compute calibration example

In Suppl. Fig. S12 we show how VAMP implied time scales agreement improves when increasing the number of ODE steps.

Experimental parameters

We show training and sampling parameters in Suppl. Tables S2) and S3, respectively. When two numbers are shown separated by a forward slash (/), the first number refers to the small



Supplementary Figure S6: Free energy landscape comparison between MD, RE and TITO+MDFT and TITO of test set molecules for which TITO predicts nanosecond meta-stable states missing or poorly sampled in RE.

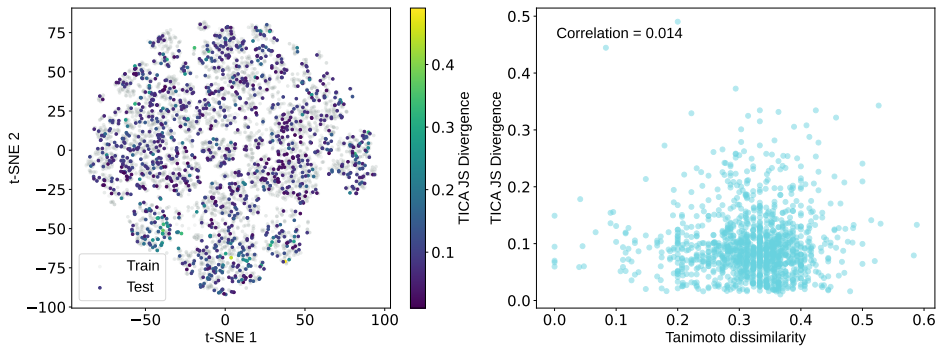
molecules dataset and the second to the tetra-peptides.

Layers	5
Feature vector size	64
Embedding layers	2
Learning rate	0.01
Batch size	750
Max lag	1 ns/ 5 ns
Lag distribution	Uniform

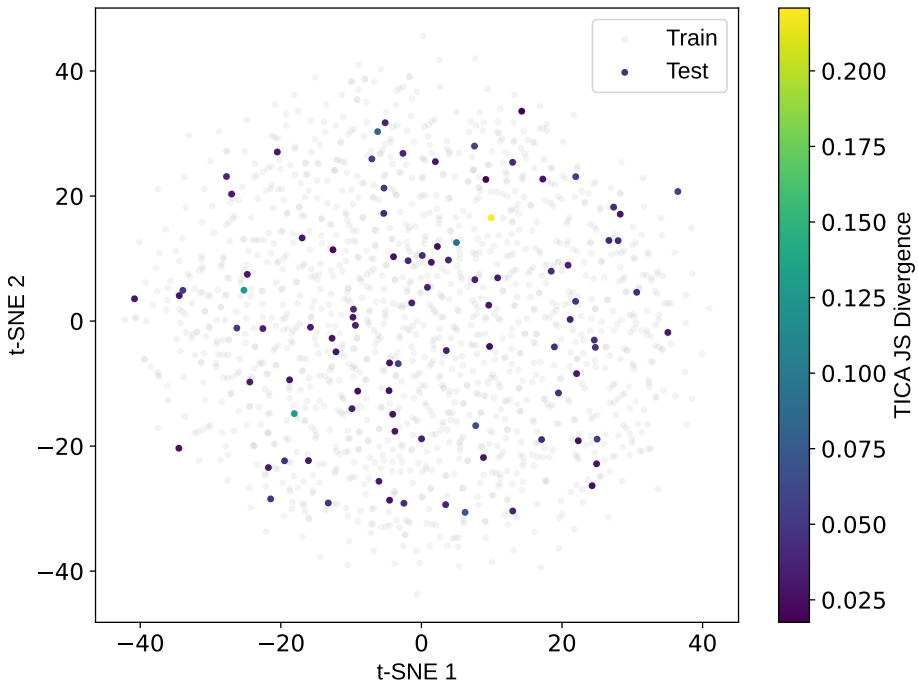
Supplementary Table S2: Training hyper-parameters of TITO models for small molecules/tetra-peptides.

	Figure 2	Figure 3 A and B	Figure 3 C	Figure 4	Figure 5
Lag	57 ps/250 ps	1 ns	1 ns	See figure	5 ns
Nested samples	640/500	1000	50,000	1 or 5 (nested)	1
ODE steps	20/40	20	20	40	100
Integrator	Euler	Euler	Euler	Euler	Euler
Batch size	32	128	32	50 000	51200
MD fine-tuning simulation time	-	-	10 ps	-	-
MD fine-tuning replicates	-	-	32,000	-	-
Ultra long MD simulation time	-	-	500 ns	-	-
Ultra long MD replicates	-	-	32,000	-	-

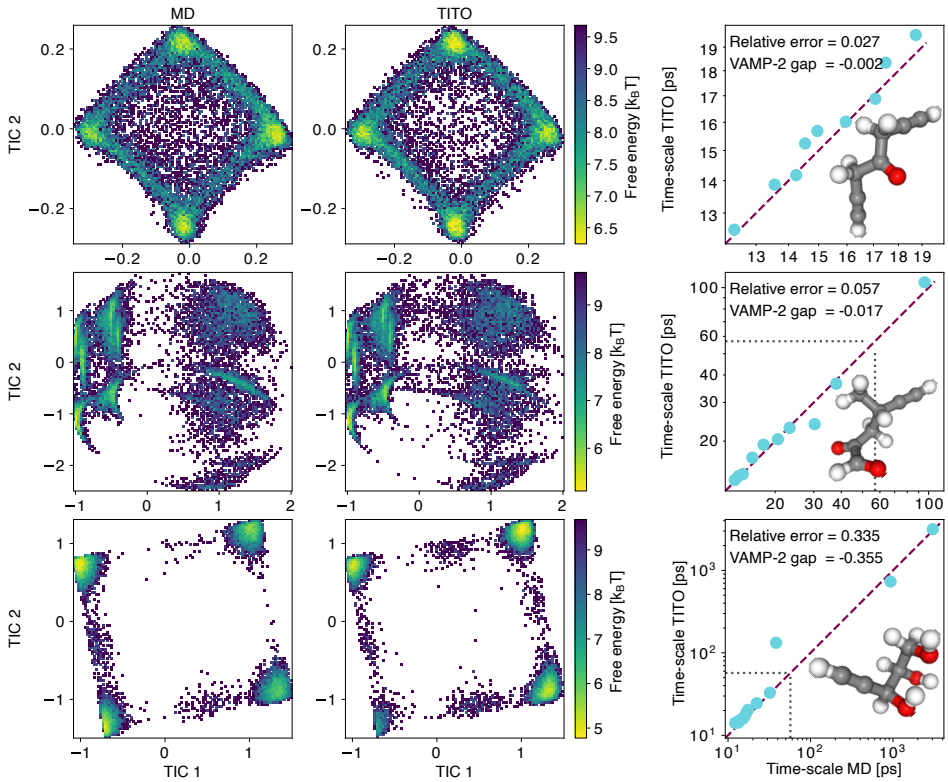
Supplementary Table S3: TITO sampling parameters for results in different figures in the main text for small molecules/tetra-peptides.



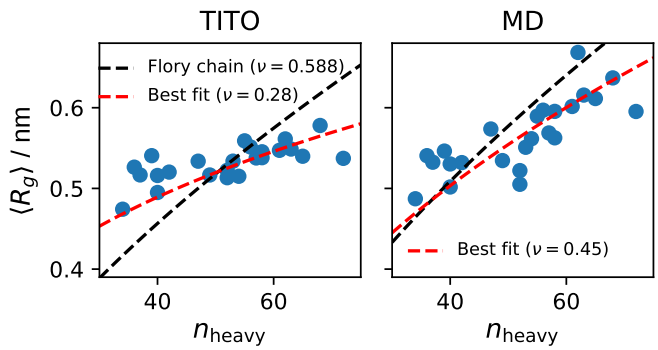
Supplementary Figure S7: TICA Jensen-Shannon divergence vs chemical similarity for small molecules dataset. (Left) First two t-SNE projections for molecules in training and test set using Tanimoto dissimilarity as distance. Test set molecules are colored with Jensen-Shannon divergence of samples generated with TITO vs reference MD simulations. Test set is well covered by training set. (Right) Minimum Tanimoto dissimilarity of test set molecules w.r.t. training set vs Jensen-Shannon divergence. No correlation is observed and Tanimoto dissimilarities are low.



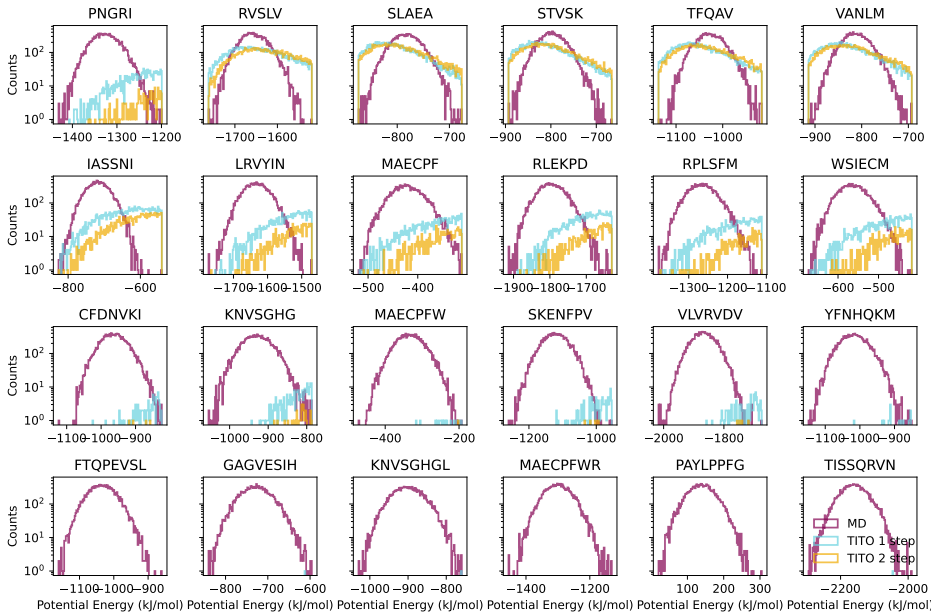
Supplementary Figure S8: TICA Jensen-Shannon divergence vs chemical similarity for tetrapeptides dataset. First two t-SNE projections for molecules in training and test set using Hamming distance. Test set molecules are colored with Jensen-Shannon divergence of samples generated with TITO vs reference MD simulations. Test set is well covered by training set.



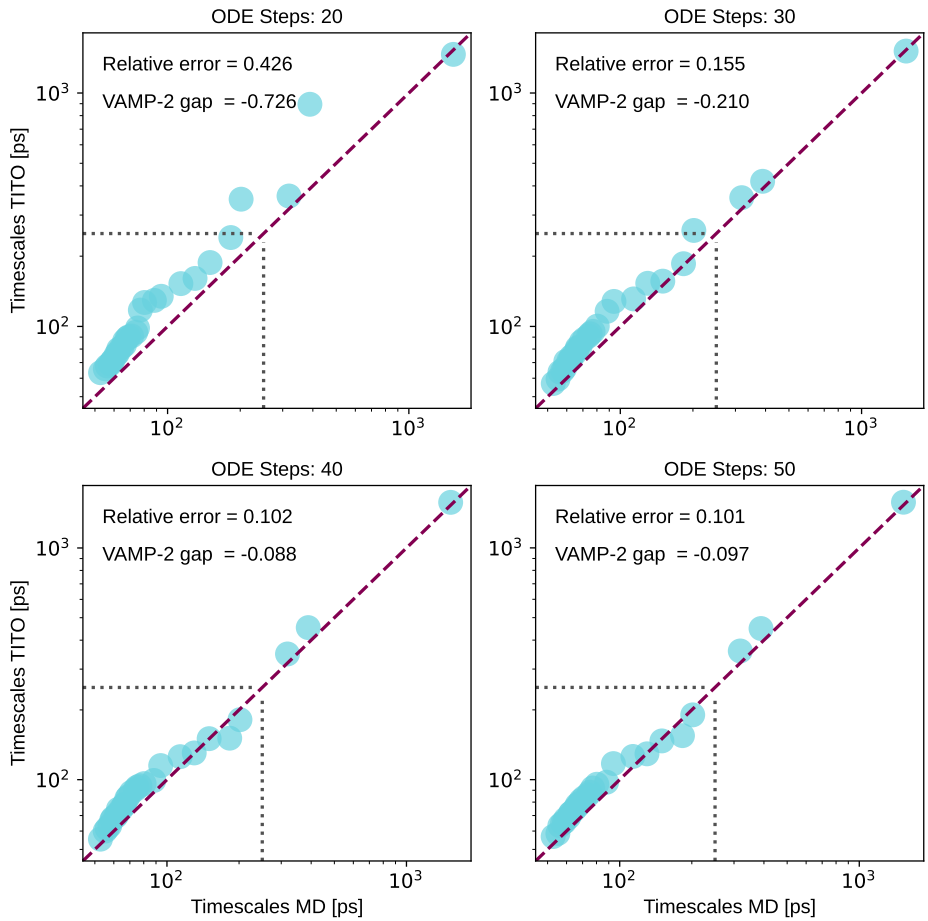
Supplementary Figure S9: TITO accurately samples thermodynamic and kinetic properties of molecules whose slowest process ranges from ps to ns.



Supplementary Figure S10: Scaling of radius of gyration $\langle R_g \rangle$ with number of heavy atoms in the pentapeptides, hexapeptides, heptapeptides and octapeptides. TITO results (left) are size extrapolations. MD values are computed on $100\,\mathrm{ns}$ simulation trajectories for each of the extrapolation test systems. Dashed lines show ideal Flory chain (black) and best fit (red) scaling exponent ν.



Supplementary Figure S11: Potential energy distributions comparisons between MD (purple), single (blue) and two step (gold) sampling with TITO ($\Delta t = 0.5\,\mathrm{ns}$) for pentapeptides, hexapeptides, heptapeptides and octapeptides. Only samples in the range of sampled potential energies in the MD are included in the plot, if no samples are visible it means no samples generated fall within this potential energy range.



Supplementary Figure S12: Correlation plot of MD and TITO time scales for different number of ODE steps. Error decrases with number of ODE steps.

Paper III

Boltzmann priors for Implicit Transfer Operators

J. Viguera Diez, M. Schreiner, O. Engkvist, S. Olsson

International Conference of Learning Representations 2025

https://doi.org/10.48550/arXiv.2410.10605

Abstract

Accurate prediction of thermodynamic properties is essential in drug discovery and materials science. Molecular dynamics (MD) simulations provide a principled approach to this task, yet they typically rely on prohibitively long sequential simulations. Implicit Transfer Operator (ITO) Learning offers a promising approach to address this limitation by enabling stable simulation with time steps orders of magnitude larger than MD. However, to train ITOs, we need extensive, unbiased MD data, limiting the scope of this framework. Here, we introduce Boltzmann Priors for ITO (BoPITO) to enhance ITO learning in two ways. First, BoPITO enables more efficient data generation, and second, it embeds inductive biases for long-term dynamical behavior, simultaneously improving sample efficiency by one order of magnitude and guaranteeing asymptotically unbiased equilibrium statistics. Furthermore, we showcase the use of BoPITO in a new tunable sampling protocol interpolating between ITOs trained on off-equilibrium simulations and an equilibrium model by incorporating unbiased correlation functions.

BOLTZMANN PRIORS FOR IMPLICIT TRANSFER OPERATORS

Juan Viguera Diez[1,2] **Mathias Schreiner**[1] **Ola Engkvist**[1,2] **Simon Olsson**[1] *

[1] Department of Computer Science and Engineering Chalmers University of Technology and University of Gothenburg SE-41296 Gothenburg, Sweden.
[2] Molecular AI, Discovery Sciences, R&D, AstraZeneca Gothenburg, Pepparedsleden 1, 431 50 Mölndal, Sweden.

ABSTRACT

Accurate prediction of thermodynamic properties is essential in drug discovery and materials science. Molecular dynamics (MD) simulations provide a principled approach to this task, yet they typically rely on prohibitively long sequential simulations. Implicit Transfer Operator (ITO) Learning offers a promising approach to address this limitation by enabling stable simulation with time steps orders of magnitude larger than MD. However, to train ITOs, we need extensive, unbiased MD data, limiting the scope of this framework. Here, we introduce Boltzmann Priors for ITO (BoPITO) to enhance ITO learning in two ways. First, BoPITO enables more efficient data generation, and second, it embeds inductive biases for long-term dynamical behavior, simultaneously improving sample efficiency by one order of magnitude and guaranteeing asymptotically unbiased equilibrium statistics. Furthermore, we showcase the use of BoPITO in a new tunable sampling protocol interpolating between ITOs trained on off-equilibrium simulations and an equilibrium model by incorporating unbiased correlation functions. Code is available at `https://github.com/olsson-group/bopito`.

1 INTRODUCTION

Efficient molecular dynamics (MD) simulation on long time-scales is critical to a large number of scientific and engineering applications. Stable simulations rely on solvers taking tiny integration time-steps, making the simulation of most phenomena impractical with current methods. Since these simulations are stochastic, a simulation step corresponds to drawing a sample from a transition probability density $p(\mathbf{x}_{t+\tau} \mid \mathbf{x}_t)$, where τ is a tiny time-step. Recently, deep generative models have emerged as a promising strategy to potentially speed up these simulations by learning transition probability densities where τ is much larger (Schreiner et al., 2023; Klein et al., 2023; Hsu et al., 2024; Fu et al., 2023) and thereby allow efficient sampling at long time-scales. Implicit Transfer Operator (ITO) learning (Schreiner et al., 2023) learns such surrogate models at multiple time-resolutions. While ITO has shown promise in accelerating simulations, it relies on extensive unbiased simulation data which may not always be available. In Markov state models, this limitation can be mitigated by integrating off-equilibrium simulations and enhanced sampling simulations (Trendelkamp-Schroer et al., 2015; Rosta & Hummer, 2014; Wu et al., 2014; 2016). However, such estimators are so far unavailable for deep generative surrogates of the transition density.

Here, we introduce Boltzmann Priors for Implicit Transfer Operator (BoPITO) learning (Figure 1). BoPITO leverages pre-trained Boltzmann Generators (BG) (Noé et al., 2019; Viguera Diez et al., 2024; Klein & Noé, 2024; Köhler et al., 2023; Köhler et al., 2020; Midgley et al., 2024; 2023) as priors to enable data-efficient training of ITO models, leading to one order of magnitude reduction in the simulation data needed for training. As the BG encodes the *invariant measure* or Boltzmann distribution of the dynamics encoded by the transition density, BoPITO, by construction, guarantees asymptotically unbiased equilibrium statistics. In this way, BoPITO can combine off-equilibrium data and biased data encoded into a BG prior to train ITO models that predict MD across multiple time-scales. Using BoPITO we introduce a new sampling strategy to recover approximate dynamics from biased off-equilibrium data, a BG prior, and unbiased time-correlation data, providing a new method for inverse problems for molecular systems.

*email: simonols@chalmers.se

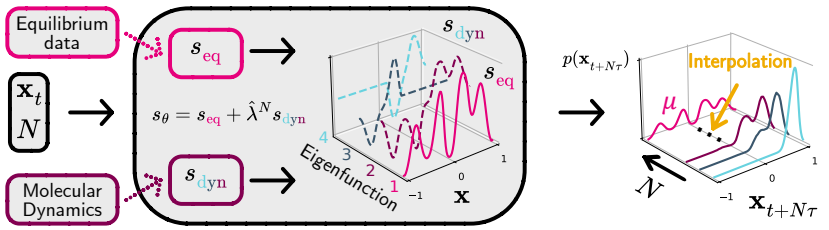


Figure 1: **Boltzmann Priors for Implicit Transfer Operators (BoPITO)** leverage pre-trained Boltzmann Generators to enable data-efficient training of surrogate models of the transition density after N simulation steps. BoPITO is implemented using Score-Based Diffusion Models. The score of the model, s_θ, separates the contributions from the first eigenfunction of the transfer operator (equilibrium density), s_{eq}, from the rest, s_{dyn}. BoPITO embeds inductive biases for long-term dynamical behavior and enables interpolation between off-equilibrium and equilibrium models.

Our main contributions are

1. **Boltzmann Priors for Implicit Transfer Operator Learning (BoPITO)**: We provide a principled way to leverage Boltzmann Generators as priors for Implicit Transfer Operator Learning, boosting sample efficiency, ensuring the generation of uncorrelated data and allowing asymptotically unbiased equilibrium statistics for $\tau \to \infty$.
2. **BoPITO interpolators**: A tunable sampling protocol facilitating interpolation of BoPITO models trained on off-equilibrium simulation data to an unbiased equilibrium distribution.
3. We show that BoPITO interpolators can **recover approximate dynamics from models trained on biased simulations**. We frame the optimization of interpolation parameters as an inverse problem and **select the interpolated ensemble that is most consistent with unbiased observables**.

2 Background and Preliminaries

2.1 Molecular Dynamics and Observables

Molecular dynamics (MD) is a widely used simulation method in chemistry, physics, and biology. It combines a mathematical model of the dynamics — e.g. Langevin dynamics (Langevin, 1908) — with a potential energy model, $U(\mathbf{x}) : \Omega \to \mathbb{R}$, of a system of interest, in turn providing detailed mechanistic insights of molecular systems, through the time evolution of particles in configuration space, $\mathbf{x} \in \Omega$. Practically, MD is solved by numerical integration, and time is discretized, with step τ. Consequently, we can understand MD as a Markov process with a Normal transition density $p(\mathbf{x}_{t+\tau} \mid \mathbf{x}_t)$, whose associated Markov operator has the *Boltzmann distribution*,

$$\mu(\mathbf{x}) = \mathcal{Z}^{-1} \exp(-\beta U(\mathbf{x})), \text{ with } \mathcal{Z} = \int \mathrm{d}\mathbf{x} \, \exp(-\beta U(\mathbf{x})), \tag{1}$$

as its *invariant measure* or stationary distribution, where β is the inverse temperature.

One important application of MD is to compute expectations or *observables* (Olsson, 2022):

1. **Stationary observables**:
$$O_a = \mathbb{E}_\mu \left[a(\mathbf{x}) \right]. \tag{2}$$
2. **Dynamic observables / Time correlation functions**:
$$O_{a(t),b(t+N\tau)} = \mathbb{E}_{\mathbf{x}_t \sim \mu} \left[\mathbb{E}_{\mathbf{x}_{t+N\tau} \sim p_\tau(\mathbf{x}_{t+N\tau}|\mathbf{x}_t)} \left[a(\mathbf{x}_t) \cdot b(\mathbf{x}_{t+N\tau}) \right] \right], \tag{3}$$

where $p_\tau(\mathbf{x}_{t+N\tau}|\mathbf{x}_t)$ is the conditional probability density after N simulation steps with time-step τ. The maps a and $b : \Omega \to \mathbb{R}^L$ serve as observable functions or *'forward models'* characterizing microscopic observation processes, e.g. indicating whether a drug is bound or not, or an interatomic distance, leading to observables such binding affinities and off-rates of a drug to a target protein.

Unfortunately, stable MD simulations rely on using time discretizations, τ, on the order of 10^{-15} s, whereas properties such as protein folding or ligand unbinding occur on much longer time scales, $\sim 10^{-3} - 10^{-1}$ s. Due to the sequential nature of these simulations, an impractical number of simulation steps are necessary to predict these properties in an unbiased fashion. As a result, most MD data will be *'off-equilibrium'*, e.g. trajectories exploring one or a few of the modes of the Boltzmann distribution $\mu(\mathbf{x})$. See Appendix A.1 for more precise definitions of configuration space, off-equilibrium, unbiased and biased MD.

2.2 Boltzmann Generators

Boltzmann Generators (BG) (Noé et al., 2019) are a generative machine learning approach to draw samples i.i.d. from the Boltzmann distribution, $\mu(\mathbf{x})$, of physical many-body systems (eq. 1). BGs learn a diffeomorphic map, $\mathcal{F}_{\boldsymbol{\theta}} : \mathbb{R}^N \to \mathbb{R}^N$, from a latent space, equipped with a simple base density $p(\mathbf{z})$, to configurational space of a physical system, such that the push-forward density $\hat{p}(\mathbf{x}) = \mathcal{F}_{\boldsymbol{\theta}} \# p(\mathbf{z})$ closely approximates the Boltzmann distribution (eq. 1). In practice, $\mathcal{F}_{\boldsymbol{\theta}}$ is implemented using an invertible neural network architecture with tractable Jacobian determinants to enable efficient sampling and exact sample likelihood computation (Chen et al., 2018; Papamakarios et al., 2021). BGs are trained either with approximately equilibrated simulation data or with biased simulations, from i.e. enhanced sampling, by employing appropriate reweighting (Ferrenberg & Swendsen, 1989; Shirts & Chodera, 2008). Unbiased samples can then be generated by importance re-sampling (Nicoli et al., 2020). Similarly, unbiased expectations can be computed using the importance weights $w(\mathbf{x}) = e^{-\beta U(\mathbf{x})}/\hat{p}(\mathbf{x})$. As such, BGs are *surrogates* of the target Boltzmann distribution, but do not model time-correlation statistics.

2.3 Transfer operators

Transfer operators (Ruelle, 1978; Schütte et al., 2009) provide a framework to describe the evolution of probability densities over time. Let p denote an initial probability density function on Ω, and ρ its μ-weighted version, $p = \mu\rho$. The Markov operator, T_Ω, is defined using a transition density $p(\mathbf{x}_{t+\tau}|\mathbf{x}_t)$,

$$[T_\Omega(\tau) \circ \rho](\mathbf{x}_{t+\tau}) \equiv \frac{1}{\mu(\mathbf{x}_{t+\tau})} \int_\Omega \mu(\mathbf{x}_t)\rho(\mathbf{x}_t)p(\mathbf{x}_{t+\tau}|\mathbf{x}_t)\,\mathrm{d}\mathbf{x}_t. \tag{4}$$

This operator describes the μ-weighted evolution of absolutely convergent probability density functions on Ω by a discrete-time increment τ given by the dynamics encoded in a transition density $p(\mathbf{x}_{t+\tau}|\mathbf{x}_t)$. In the context of molecular dynamics, T_Ω is a μ-weighted equivalent to the Markov operator discussed above (sec. 2.1). The spectral form of the transfer operator is expressed as

$$[T_\Omega(\tau) \circ \rho](\mathbf{x}_{t+\tau}) = \sum_{i=1}^{\infty} \lambda_i(\tau)\langle\rho|\phi_i\rangle\ \psi_i(\mathbf{x}_{t+\tau}), \tag{5}$$

where $\lambda_i(\tau)$ are the eigenvalues, ψ_i and ϕ_i are the corresponding right and left eigenfunctions, respectively, and $\langle f|g\rangle = \int_\Omega f(\mathbf{x})g(\mathbf{x})\,\mathrm{d}\mathbf{x}$. The eigenvalues $\lambda_i(\tau)$ depend on the parameter τ and are related to the characteristic relaxation rates κ_i by $|\lambda_i(\tau)| = \exp(-\tau\kappa_i)$. The right and left eigenfunctions are related by the stationary density, such that $\phi_i(\mathbf{x}) = \mu(\mathbf{x})\psi_i(\mathbf{x})$. In reversible dynamics, all eigenvalues λ_i are real and lie within the interval $-1 < \lambda_i \leq 1$. Notably, there is one eigenvalue $\lambda_1 = 1$, with corresponding eigenfunctions $\psi_1(\mathbf{x}) = \mathbf{1}$ and $\phi_1(\mathbf{x}) = \mu(\mathbf{x})$.

2.4 Diffusion models

Diffusion models (DMs) (Ho et al., 2020; Song et al., 2020) are a popular generative modeling framework that approximates data densities $p(\mathbf{x}^0)$ by learning to invert a noising process (*'forward diffusion'*). The forward diffusion process is pre-specified and incrementally transforms the data distribution into a simple prior distribution, $p(\mathbf{x}^T)$, through simulation of a time-inhomogenous Markov process represented by the SDE,

$$\mathrm{d}\mathbf{x}^{t_{\mathrm{diff}}} = f(\mathbf{x}^{t_{\mathrm{diff}}}, t_{\mathrm{diff}})\,\mathrm{d}t_{\mathrm{diff}} + g(t_{\mathrm{diff}})\,\mathrm{d}W, \tag{6}$$

where ($0 < t_{\mathrm{diff}} < T$) is the diffusion time, f and g are chosen functions, and $\mathrm{d}W$ is a Wiener process. To generate samples from the data distribution, $p(x^0)$, we can sample from $p(\mathbf{x}^T)$ and

solve the backward diffusion process (*'denoising process'*) (Anderson, 1982),

$$\mathrm{d}\mathbf{x}^{t_{\mathrm{diff}}} = \left[f(\mathbf{x}^{t_{\mathrm{diff}}}, t_{\mathrm{diff}}) - g^2(t_{\mathrm{diff}})\nabla_{\mathbf{x}^{t_{\mathrm{diff}}}} \log p(\mathbf{x}^{t_{\mathrm{diff}}}|t_{\mathrm{diff}})\right] \mathrm{d}t_{\mathrm{diff}} + g(t_{\mathrm{diff}})\,\mathrm{d}W. \tag{7}$$

This backward process is approximated using a deep neural network $\nabla_{\mathbf{x}^{t_{\mathrm{diff}}}} \log p_{\boldsymbol{\theta}}(\mathbf{x}^{t_{\mathrm{diff}}}|t_{\mathrm{diff}}) = s_{\boldsymbol{\theta}}(\mathbf{x}^{t_{\mathrm{diff}}}, t_{\mathrm{diff}})$. The learned backward SDE can be recast into a probability flow ODE (Maoutsa et al., 2020; Song et al., 2021), which in turn can be interpreted as a continuous normalizing flow (Chen et al., 2018) which facilitates efficient sampling and enables sample likelihood evaluation.

2.5 Implicit Transfer Operator Learning

Implicit Transfer Operator (ITO) Learning (Schreiner et al., 2023) is a framework for learning surrogate models of the transition density, $p(\mathbf{x}_{N\tau}|\mathbf{x}_0)$, from MD data, where N is an arbitrarily large integer. ITO leverages that the transfer operator framework allows us to express the transition density as

$$p(\mathbf{x}_{t+N\tau}|\mathbf{x}_t) = \sum_{i=1}^{\infty} \lambda_i^N(\tau)\psi_i(\mathbf{x}_t)\phi_i(\mathbf{x}_{t+N\tau}), \tag{8}$$

where ψ_i and ϕ_i are independent of τ and N. See Appendix A.2 for a complete derivation.

This decomposition inspired a strategy to learn a conditional generative model $\mathbf{x}_{t+N\tau} \sim p_{\boldsymbol{\theta}}(\mathbf{x}_{t+N\tau}|\mathbf{x}_t, N)$ by sampling tuples $(\mathbf{x}_{t_i}, \mathbf{x}_{t_i+N_i\tau}, N_i)$ and training a generative model to mimic the empirical transition density at multiple time-horizons, $N\tau$. Here, we approximate ITO models with a conditional denoising diffusion probabilistic model (cDDPM) of the form

$$p(\mathbf{x}^0_{t+N\tau}|\mathbf{x}_t, N) \equiv \int p(\mathbf{x}^{0:T}_{t+N\tau}|\mathbf{x}_t, N)\,\mathrm{d}\mathbf{x}^{1:T}, \tag{9}$$

where $\mathbf{x}_{1:T}$ are latent variables of the same dimension as our output, and follow a joint density describing the backward diffusion process, eq. 7, and $\mathbf{x}^T \sim \mathcal{N}(0, \mathbb{I})$. We denote diffusion time in Diffusion Models using superscripts, while physical time is represented using subscripts. The conditional sample likelihood is given by

$$\ell(\mathbf{I}; \boldsymbol{\theta}) \equiv \prod_{i\in\mathbf{I}} p_{\boldsymbol{\theta}}(\mathbf{x}^0_{t_i+N_i\tau} \mid \mathbf{x}_{t_i}, N_i) \tag{10}$$

where $\mathbf{I}$ is a list of generated indices i specifying a time t_i and a time-lag (τ) integer multiple N_i, associating two time-points in a MD trajectory of length $M\tau$, $\mathbf{x} = \{\mathbf{x}_0, \mathbf{x}_\tau, \ldots, \mathbf{x}_{(M-1)\tau}\}$.

Training is performed by optimizing an approximation of the variational bound of the log-likelihood (Ho et al., 2020),

$$\mathcal{L}(\boldsymbol{\theta}) = \mathbb{E}_{i\sim\mathbf{I},\epsilon\sim\mathcal{N}(0,\mathbb{I}),t_{\mathrm{diff}}\sim\mathcal{U}(0,T)}\left[\|\epsilon - \hat{\epsilon}_{\boldsymbol{\theta}}(\widetilde{\mathbf{x}}^{t_{\mathrm{diff}}}_{t_i+N_i\tau}, \mathbf{x}_{t_i}, N_i, t_{\mathrm{diff}})\|_2\right], \tag{11}$$

where $\widetilde{\mathbf{x}}^{t_{\mathrm{diff}}}_t = \sqrt{\bar{\alpha}^{t_{\mathrm{diff}}}}\mathbf{x}_t + \sqrt{1-\bar{\alpha}^{t_{\mathrm{diff}}}}\epsilon$, with $\bar{\alpha}^{t_{\mathrm{diff}}} = \prod_j^{t_{\mathrm{diff}}}(1-\beta_j)$ and β_j is the variance of the forward diffusion process at diffusion time, j. $\hat{\epsilon}_{\boldsymbol{\theta}}(\cdot)$ is one of the two architectures presented by Schreiner et al. (2023). Following Arts et al. (2023) we express the score as

$$s_{\boldsymbol{\theta}}(\mathbf{x}^{t_{\mathrm{diff}}}_{t+N\tau}, \mathbf{x}_t, N_i, t_{\mathrm{diff}}) = -\frac{\hat{\epsilon}_{\boldsymbol{\theta}}(\widetilde{\mathbf{x}}^{t_{\mathrm{diff}}}_{t_i+N_i\tau}, \mathbf{x}_{t_i}, N_i, t_{\mathrm{diff}})}{\sqrt{1-\bar{\alpha}^{t_{\mathrm{diff}}}}}. \tag{12}$$

See Appendix A.3 for more details and pseudo-code on training and sampling algorithms.

3 Related work

Sampling the Boltzmann distribution Apart from Boltzmann Generator-based approaches, there are a number of traditional ways to draw statistical samples from the Boltzmann distribution of molecular systems. Prominent examples include, molecular dynamics or Markov Chain Monte Carlo simulations coupled with enhanced sampling strategies (Hénin et al., 2022; Kamenik et al., 2022), including replica-based approaches (Earl & Deem, 2005; Sidler et al., 2016; Pasarkar et al., 2023), conformational flooding (Grubmüller, 1995), meta-dynamics (Laio & Parrinello, 2002), and

umbrella sampling (Torrie & Valleau, 1977), in particular when paired up with machine-learned collective variables (Chen & Ferguson, 2018; Wang et al., 2019b; Herringer et al., 2023; M. Sultan & Pande, 2017). The success of these approaches relies on choosing the right mechanism to enhance sampling across high free energy barriers, or low probability regions, for any given case (Carter et al., 1989). Finding such mechanisms typically involves substantial manual engineering of collective variables and other hyper-parameters. Finally, there are transition path sampling approaches (Dellago et al., 1998; Bolhuis et al., 2002), which are particularly powerful when combined with reinforcement learning (Jung et al., 2023) or deep generative priors (Plainer et al., 2023).

Latent space simulators and Coarse graining Classical (fine-grained) molecular dynamics simulation can be coarse-grained beyond the Born-Oppenheimer approximation, such that atomic nuclei are merged together into 'beads.'(Noid, 2023) This strategy, in principle enables faster simulations due to the smaller number of particles and the acceleration of kinetics caused by the coarse-graining (CG) operation (Nüske et al., 2019). CG models can be estimated to closely approximate the thermodynamics of the corresponding fine-grained system through the optimization of two equivalent variational bounds (Noid et al., 2008; Lyubartsev & Laaksonen, 1995; Ercolessi & Adams, 1994). A relaxation of this bound was recently proposed to also allow for probabilistic reconstruction of fine-grained configurations (Chennakesavalu et al., 2023). These bounds have been used extensively to build CG force-field models (Husic et al., 2020; Wang et al., 2019a; Majewski et al., 2023; Charron et al., 2023) including implicit solvation models (Chen et al., 2021; Katzberger & Riniker, 2024), using deep neural networks due to their ability to capture multibody terms (Wang et al., 2021) to ultimately accelerate the prediction of equilibrium properties of molecular systems. Similarly, the development of *'latent space simulators'* where a learned, typically low-dimensional, latent space equipped either with a propagator (Sidky et al., 2020; Chennakesavalu et al., 2023; Wang et al., 2024), or not (Wang & Gómez-Bombarelli, 2019), is learned to enable efficient simulation. These approaches, in general aim to accelerate molecular simulations akin to BoPITO, yet due to the CG operation, the molecular dynamics (kinetics) will be accelerated, and detailed knowledge of the unbiased dynamics are needed to correct this (Nüske et al., 2019; Crommelin & Vanden-Eijnden, 2011). Concurrent work, presents MDGen where conformational states are tokenized and in turn used to generate multiple frames of a MD trajectory jointly (Jing et al., 2024).

Transfer Operator surrogates Analysis of MD data often involves building transfer operator surrogates such as Markov state models (MSM) (Schütte et al., 1998; Prinz et al., 2011; Swope et al., 2004; Husic & Pande, 2018), time-lagged independent components analysis (Molgedey & Schuster, 1994; Ziehe & Müller, 1998; Pérez-Hernández et al., 2013), Markov field models or dynamic graphical models, (Olsson & Noé, 2019; Mardt et al., 2022; Hempel et al., 2022), VAMPnets (Mardt et al., 2018; Wu & Noé, 2019), or observable operator models (Wu et al., 2015). Markov state models are time-space discrete approximations of the transfer operator and Deep Generative MSM (Wu et al., 2019) and VAMPnets (Mardt et al., 2018) learn the space discretization through deep neural networks. Dynamic graphical models or Markov field models (Hempel et al., 2022) represent a time-space discrete approximation of the transfer operator injecting a (conditional) independence assumption of molecular subsystems, when modeling the transition probability, leading to better scaling for systems with poor time-scale separation (Olsson & Noé, 2019). Apart from ITO (Schreiner et al., 2023), several other deep generative approaches for modeling the transition density of molecular dynamics have recently proposed. *Timewarp* where a normalizing flow is used to encode the transition density (Klein et al., 2023) with limited transferability to enable metropolized sampling of unbiased equilibrium distributions (Hastings, 1970). *Score dynamics* use a DDPM to model the displacements of an initial configuration towards a time-lagged one, achieving picosecond time-steps simulation and limited transferability (Hsu et al., 2024). However, unlike in the context of MSMs (Trendelkamp-Schroer et al., 2015; Rosta & Hummer, 2014; Wu et al., 2014; 2016), there are no deep generative transition density surrogates available leveraging available information about the equilibrium distribution — BoPITO is one such method.

4 BOLTZMANN PRIORS FOR IMPLICIT TRANSFER OPERATOR LEARNING

ITO training requires extensive, unbiased MD simulations to capture the statistical distribution of rare events. This data-intensive requirement can hinder ITO's practical implementation. Furthermore, models trained on off-equilibrium simulations, which may exhibit non-representative statistics, can lead to inaccurate predictions, compromising their utility in downstream applications. In practice, however, both unbiased off-equilibrium simulations and information about the equilibrium distribution from potentially biased simulations (Hénin et al., 2022) can be cheaply generated, and both encode information about the dynamical behavior of molecular systems. We introduce Boltzmann Priors for Implicit Transfer Operator (BoPITO), as a learning paradigm for ITO models that leverage available information about the equilibrium distribution.

BoPITO uses pre-trained models of the equilibrium distribution to improve ITO learning in four ways. First, it helps ensure broad sampling of Ω proportional to the equilibrium distribution for subsequent MD simulations yielding information about the transition density $p(\mathbf{x}_\tau \mid \mathbf{x}_0)$ from across Ω. Second, we use it to fix the stationary part of the learned transition density, boosting the sample efficiency when learning models of molecular dynamics. Third, it imposes an inductive bias of long-time dynamical behavior allowing for recovery of unbiased Boltzmann distribution for long-time horizons. Fourth, using BoPITO, we introduce a novel tunable sampling protocol interpolating ITO models trained on off-equilibrium simulation data and an unbiased equilibrium distribution.

4.1 EFFICIENT DATA GENERATION

Training data for ITO models consists of several independent unbiased MD simulations. Following the adaptive sampling strategy (Bowman et al., 2010; Doerr & De Fabritiis, 2014; Viguera Diez et al., 2024; Betz & Dror, 2019), used extensively in the molecular dynamics simulation community, we use a pre-trained BG to generate initial conditions to simulations ensuring broad sampling across Ω proportional to the Boltzmann distribution. As long as these trajectories reach a 'local equilibrium' we can in principle recover an unbiased model of the molecular dynamics (Nüske et al., 2017), albeit without relying on running one or a few very long simulations to reach the global equilibrium.

4.2 LONG-TERM DYNAMICS INDUCTIVE BIAS FOR ITO

One important application of ITO is to allow for one-step sampling of long-time-scale dynamics. However, real datasets often contain a limited number of effective samples for long time-scales, leading to potential biases in models. To mitigate this issue, we propose separating the equilibrium contribution from the time-dependent components:

$$p(\mathbf{x}_{t+N\tau}|\mathbf{x}_t) = \mu(\mathbf{x}_{t+N\tau}) + \sum_{i=2}^{\infty} \lambda_i^N(\tau)\phi_i(\mathbf{x}_{t+N\tau})\psi_i(\mathbf{x}_t), \tag{13}$$

where we have used that $\lambda_1 = 1$ and its corresponding eigenfunctions are $\phi_1 = \mu$ and $\psi_1 = \mathbf{1}$. Additionally, we introduce a decay in the time-dependent component, creating an inductive bias that asymptotically samples from an available equilibrium model for long-term dynamics. Using the spectral decomposition of the transition density (eq. 13), we choose the score model as

$$s(\mathbf{x}_{t+N\tau}^{t_{\text{diff}}}, \mathbf{x}_t, N, t_{\text{diff}}, \boldsymbol{\theta}) = s_{\text{eq}}(\mathbf{x}_{t+N\tau}^{t_{\text{diff}}}, t_{\text{diff}}) + \hat{\lambda}^N s_{\text{dyn}}(\mathbf{x}_{t+N\tau}^{t_{\text{diff}}}, \mathbf{x}_t, N, t_{\text{diff}}, \boldsymbol{\theta}), \tag{14}$$

where $s_{\text{eq}}(\mathbf{x}_{\text{diff}}^t, t_{\text{diff}})$ is the score of a pre-trained surrogate of the equilibrium distribution model, $0 < \hat{\lambda} < 1$ is a hyper-parameter and $s_{\text{dyn}}(\mathbf{x}_{t+N\tau}^{t_{\text{diff}}}, \mathbf{x}_t, N, t_{\text{diff}}, \boldsymbol{\theta})$ accounts for the time-dependent components. As $N \to \infty$, s_{eq} dominates and the model samples from the equilibrium model, see Appendix A.4 for an example. By interpreting the score field as a velocity field, we can reformulate the DM as a continuous normalizing flow which we can use for Metropolized sampling of the unbiased equilibrium distribution (Klein et al., 2023). The proposed factorization is principled, corresponding to a separation of the score of the transition density into a stationary and dynamic part, where the first part is considered known, see Appendix A.5 for details. In practice, we first train s_{eq} (if not provided) using equilibrium data. Then we train s_{dyn} with unbiased, possibly off-equilibrium, MD data while keeping s_{eq} fixed and we choose $\hat{\lambda}$ performing hyper-parameter optimization, see Appendix A.6 for details. We discuss alternative formulations of the score in Appendix A.7.

4.3 BoPITO INTERPOLATORS

A significant issue in MD simulations is the unbiased sampling of transitions over high free-energy barriers, e.g. channels in $\mu(\mathbf{x})$ with very little probability mass, typically coinciding with mixing on Ω. This difficulty leads to an inability to predict time-correlation statistics for large $N\tau$. With BoPITO we can define an interpolation between models trained on off-equilibrium simulations and the equilibrium distribution. In this manner, we can approximate dynamics not seen explicitly in the data. For a time-dependent model, s_{dyn}, trained on off-equilibrium simulations with maximum training lag, N_{max}, we define a BoPITO interpolator as a model with the following score function:

$$s(\mathbf{x}^{t_{\mathrm{diff}}}, \mathbf{x}_t, N_{\mathrm{int}}, t_{\mathrm{diff}}) = s_{\mathrm{eq}}(\mathbf{x}^{t_{\mathrm{diff}}}, t_{\mathrm{diff}}) + \hat{\lambda}^{N_{\mathrm{int}}} s_{\mathrm{dyn}}(\mathbf{x}^{t_{\mathrm{diff}}}, \mathbf{x}_t, N_{\mathrm{max}}, t_{\mathrm{diff}}), \tag{15}$$

with $N_{\mathrm{int}} \geq N_{\mathrm{max}}$. This defines an interpolator because $N_{\mathrm{int}} = N_{\mathrm{max}}$ generates samples from the model distribution with maximum lag and $N_{\mathrm{int}} \to \infty$ generates samples from the equilibrium model.

Inspired by methods in molecular biophysics (Kolloff & Olsson, 2023; Olsson et al., 2017; Bottaro & Lindorff-Larsen, 2018; Salvi et al., 2016), we propose to choose the interpolation parameter, N_{int}, as the most consistent with an unbiased dynamical observable, such as experimental data. That is, for a lag $N > N_{\mathrm{max}}$ and an unbiased dynamic observable O_N^*, we choose

$$N_{\mathrm{int},N} = \arg\min_{N_{\mathrm{int}}} |O_N^* - O_{N_{\mathrm{int}}}|, \tag{16}$$

where $O_{N_{\mathrm{int}}}$ is the dynamic observable estimated with samples from the BoPITO interpolator. This way BoPITO can integrate off-equilibrium and equilibrium MD with experimental data. In practice, we find the interpolator to generate some high-energy states. However, we can alleviate the high-energy structures by alternating long-lag interpolation steps with short-lag non-interpolation steps for local relaxation. We discuss how BoPITO interpolators alleviate out-of-distribution issues in Appendix A.8.

5 RESULTS

For detailed parameters of the experiments below, we refer to Appendix A.9.

5.1 SYSTEMS

Prinz potential is a 1D potential commonly used for benchmarking MD sampling methods (Prinz et al., 2011). We set the observable functions, a and b in eq. 3, to be the identity function for computing dynamic observables. For details, see Appendix A.10.

Alanine Dipeptide is a small peptide with 22 atoms. We use publicly available data from Dibak et al. (2022), containing 1 μs simulation time split in 20 trajectories. Simulation is performed in an implicit solvent with 2 fs integration time-step, and data is saved every 1 ps. We choose,

$$a(\mathbf{x}) = b(\mathbf{x}) = \begin{bmatrix} \sin\phi(\mathbf{x}) \\ \cos\phi(\mathbf{x}) \end{bmatrix}, \tag{17}$$

where $\phi(\mathbf{x})$ is a torsion angle involved in the slowest transition in the system. For details, see Appendix A.11.

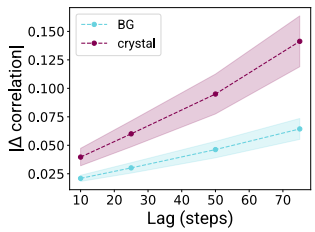
Chignolin (cln025) is a fast folding protein with 10 residues, 166 atoms and 93 heavy atoms. We use molecular dynamics data previously reported by Lindorff-Larsen et al. (2011). The data is proprietary but available upon request for research purposes. The simulations were performed in explicit solvent with a 2.5 fs time-step and the positions was saved at 200 ps intervals. We extract all heavy atoms positions from the simulations and train models on this data. We use the fraction of native contacts Lindorff-Larsen et al. (2011) to define a dynamic observable with,

$$a(\mathbf{x}) = b(\mathbf{x}) = \frac{\sum_{i=1}^{N_{\mathrm{res}}} \sum_{j>i}^{N_i} \frac{1}{1+e^{10(d_{ij}(\mathbf{x}) - d_{ij}^* - 1)}}}{\sum_{i=1}^{N_{\mathrm{res}}} N_i}, \tag{18}$$

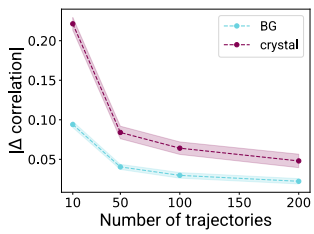
where the first sum in the numerator iterates over all N_{res} residues in the protein, while the second sum considers the N_i native contacts of residue i separated by at least seven residues in the primary sequence. Here, $d_{ij}(\mathbf{x})$ and d_{ij}^* represent the $C_\alpha - C_\alpha$ distances of residues i and j in the structure $\mathbf{x}$ and the native structure, respectively. This observable quantifies the protein's foldedness, with values ranging from 0 (unfolded) to 1 (folded). For a detailed explanation, refer to Appendix A.12.

5.2 BOLTZMANN PRIORS FOR TRAINING DATA GENERATION

Modeling the transition density over Ω requires observing transitions across the state space. In the usual setting, only one or a few initial conditions are available when data collection starts. Here, we explore the case where a BG is available before data collection and compare it against the baseline, where only one initial condition is known. We use a BG trained on equilibrium data of the Prinz potential to sample the initial conditions of our training trajectories. We also generate trajectories using a single starting point (crystal, $\mathbf{x} = 0.75$) and compare their performance with long simulations (Figure 2). We compare the two different data generation strategies using the metric, $|\Delta\text{correlation}|$, which measures the absolute difference of the time-correlation function (dynamic observable) compared to the MD ground truth (Appendix A.13). We find that the performance of models using Boltzmann priors for data generation is superior for different lag times (Figure 2a) and number of trajectories (Figure 2b). The gap between BG and the crystal baseline increases with lag-time, and as expected decreases with the number of generated trajectories.



(a)



(b)

Figure 2: Absolute difference in correlation with respect to long unbiased MD simulations (lower is better) of models trained on trajectories initialized on samples from a Boltzmann Generator (BG) and a single structure (crystal, $\mathbf{x} = 0.75$) for the Prinz Potential under direct sampling. The former presents superior performance for different lag times (number of trajectories = 50) (a) and number of trajectories (b). The shaded areas correspond to 95 % confidence interval.

5.3 BOPITO EFFICIENTLY SAMPLES LONG-TERM DYNAMICS IN A LOW-DATA CONTEXT

Fixing the equilibrium contribution to the score field effectively reduces the number of parameters that need to be estimated. To test whether this prior information manifests as an improved sample efficiency we compared ITO and BoPITO models against each other with varying sizes of training data.

We find that the BoPITO models achieve a higher accuracy for long-term dynamics compared to ITO models when data is scarce, as the equilibrium distribution is known *a priori* and does not need to be learned from simulation data (Figure 3). The inductive bias in BoPITO models enables them to learn long-term dynamics, even in scenarios where ITO models fail. For the Prinz Potential, we find that while ITO suffers from poor performance modeling long-term dynamics when data is scarce, BoPITO models accurately capture long-term dynamics without worsening the performance on short and medium time-scales. The results for Alanine Dipeptide and Chignolin show favorable scaling with increasing system size: we find that an order of magnitude more data is needed to train an ITO to the same accuracy as a BoPITO model trained on the same data. Moreover, we show in Appendix A.14 that the energies of configurations sampled from our prior Boltzmann Generator used in our experiments match closely the energies of MD samples.

5.4 INTERPOLATING BETWEEN MODELS TRAINED ON OFF-EQUILIBRIUM DATA AND THE BOLTZMANN DISTRIBUTION WITH EXPERIMENTAL DATA

In practical settings, our MD data will be off-equilibrium, e.g. having sampled only one or a few of the relevant modes in the Boltzmann distribution $\mu(\mathbf{x})$ in a given trajectory. Consequently, unless extensive data across the domain Ω can be collected, models based on such data will be biased. An alternative to collecting more simulation data is to use a multi-modal strategy where experimental data is used to fill the gaps left by simulation data and bridge to long time-scale dynamics (Salvi et al., 2016; Kolloff & Olsson, 2023). In this section we explore the potential of BoPITO interpolators to integrate dynamic observables with off-equilibrium simulation data to recover unbiased long-term dynamics.

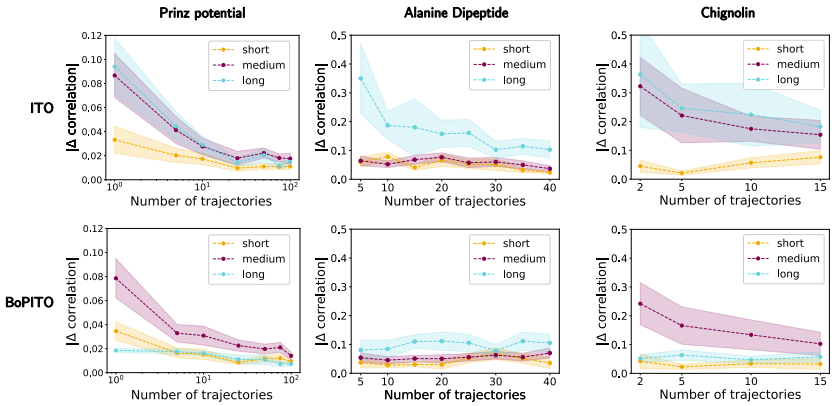


Figure 3: Absolute difference in correlation with respect to long unbiased MD simulations (lower is better, one-step sampling) of ITO (left) and BoPITO (right) split into short, medium, and long timescales against the number of training trajectories for the Prinz potential (top) and Alanine Dipeptide (middle) and Chignolin (bottom). The shaded areas correspond to a 95 % confidence interval.

We showcase BoPITO interpolators on Alanine Dipeptide. By removing the transitions between the modes of the Boltzmann distribution corresponding to the slowest process in the system, e.g. ϕ crossing 0 or 2, we generate a biased simulation data, resembling a realistic scenario. We then train ITO and BoPITO models using these biased trajectories. For lags > 100, we sample the BoPITO interpolator estimated by matching to an unbiased dynamic observable defined by eq. 26, allowing us to overcome a systematic error in the correlation function observed in the biased MD data and for an ITO model trained on these data (Figure 4).

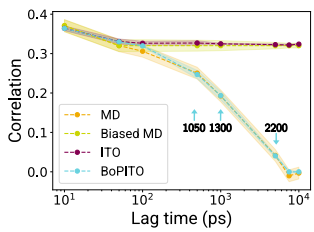


Figure 4: Fitting an interpolator of a biased BoPITO model for Alanine Dipeptide. Both the ITO and BoPITO models are trained on biased MD simulations of Alanine Dipeptide. We fit a BoPITO interpolator selecting the most consistent ensemble with the unbiased correlation function specified in eq. 16. Numbers with blue arrows specify the interpolation parameter, N_{int}.

Beyond reproducing the provided dynamic observable, the interpolator also demonstrates a remarkable ability to capture the underlying microscopic dynamics (Figure 5). Even with excellent agreement, the generated ensembles only slightly overestimates density in the transition state region. However, this effect could easily be alleviated by annealing a short MD simulation to the interpolation as was recently shown (Viguera Diez et al., 2024).

6 Limitations and Future Work

Choice of hyper-parameter $\hat{\lambda}$ The hyper-parameter $\hat{\lambda}$, defines a global relaxation or mixing timescale of the dynamics after which the model is guaranteed to sample equilibrium. In Appendix A.6 we describe a protocol to determine a bound for this parameter. However, developing a similar protocol for a transferable BoPITO model would likely require modifications to accommodate the diversity of global relaxation time-scales across different systems.

No Chapman-Kolmogorov Guarantee When training on biased or off-equilibrium data where we rely on establishing ergodicity through interpolation we cannot guarantee self-consistency of the dynamics in the Chapman-Kolmogorov sense.

Surrogate model BoPITO inherits the current limitations of ITO, such as generalization over chemical space and thermodynamic variables, and scaling. Furthermore, current models cannot guarantee unbiased sampling dynamics for non-equilibrium ensembles, which would require closed-form expressions for the target path probabilities.

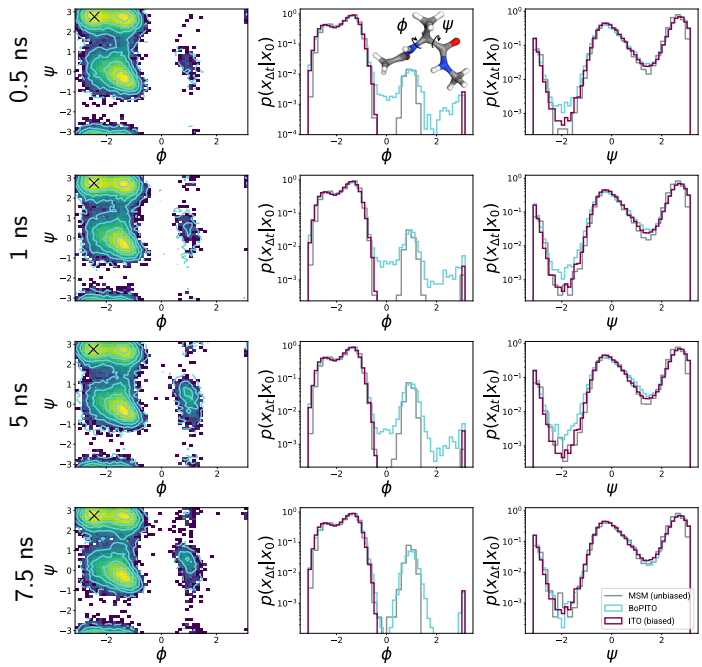


Figure 5: BoPITO can incorporate unbiased dynamic observables to correct a model trained on biased data. Rows of increasing time-lag (from top to bottom). Contour plots correspond to a BoPITO interpolator. The first column shows conditional transition densities projected onto the torsion angles ϕ and ψ (inset). The black cross indicates the initial condition. The second and third columns show marginal distributions of ϕ and ψ, respectively. MSM stands for a Markov State Model of the unbiased MD data.

7 Conclusion

We introduce Boltzmann Priors for Implicit transfer Operator Learning (BoPITO), a framework to enhance ITO learning in three ways. First, a broad sampling of configuration space is used to initialize short off-equilibrium MD simulations. Second, we parameterize the transition density as an interpolation towards a pre-trained Boltzmann Generator, improving sample efficiency by an order of magnitude. BoPITO is a principled approach to embedding prior knowledge of the stationary distribution of Markovian dynamics as an inductive bias for long-term dynamical behavior. Third, our approach enables interpolation between models trained on off-equilibrium data and the equilibrium distribution, and we can recover accurate models of unseen dynamics when informed by unbiased observables. Consequently, BoPITO is the first method to allow for the integration of multiple sources of information into the generation of deep generative surrogates of molecular dynamics.

ACKNOWLEDGMENTS

The authors thank Prof. Rocío Mercado and Prof. Fredrik Johansson for their useful feedback on the manuscript. This work was partially supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation. The computations in this work were enabled by the Berzelius resource provided by the Knut and Alice Wallenberg Foundation at the National Supercomputer Centre.

REFERENCES

Brian D.O. Anderson. Reverse-time diffusion equation models. *Stochastic Processes and their Applications*, 12(3):313–326, May 1982. ISSN 0304-4149. doi: 10.1016/0304-4149(82)90051-5. URL http://dx.doi.org/10.1016/0304-4149(82)90051-5.

Marloes Arts, Victor Garcia Satorras, Chin-Wei Huang, Daniel Zügner, Marco Federici, Cecilia Clementi, Frank Noé, Robert Pinsler, and Rianne van den Berg. Two for one: Diffusion models and force fields for coarse-grained molecular dynamics. *Journal of Chemical Theory and Computation*, 19(18):6151–6159, September 2023. ISSN 1549-9626. doi: 10.1021/acs.jctc.3c00702. URL http://dx.doi.org/10.1021/acs.jctc.3c00702.

Robin M. Betz and Ron O. Dror. How effectively can adaptive sampling methods capture spontaneous ligand binding? *Journal of Chemical Theory and Computation*, 15(3):2053–2063, January 2019. ISSN 1549-9626. doi: 10.1021/acs.jctc.8b00913. URL http://dx.doi.org/10.1021/acs.jctc.8b00913.

Peter G. Bolhuis, David Chandler, Christoph Dellago, and Phillip L. Geissler. Transition path sampling: Throwing ropes over rough mountain passes, in the dark. *Annual Review of Physical Chemistry*, 53(1):291–318, October 2002. ISSN 1545-1593. doi: 10.1146/annurev.physchem.53.082301.113146. URL http://dx.doi.org/10.1146/annurev.physchem.53.082301.113146.

Sandro Bottaro and Kresten Lindorff-Larsen. Biophysical experiments and biomolecular simulations: A perfect match? *Science*, 361(6400):355–360, July 2018. ISSN 1095-9203. doi: 10.1126/science.aat4010. URL http://dx.doi.org/10.1126/science.aat4010.

Gregory R. Bowman, Daniel L. Ensign, and Vijay S. Pande. Enhanced modeling via network theory: Adaptive sampling of markov state models. *Journal of Chemical Theory and Computation*, 6(3):787–794, February 2010. ISSN 1549-9626. doi: 10.1021/ct900620b. URL http://dx.doi.org/10.1021/ct900620b.

E.A. Carter, Giovanni Ciccotti, James T. Hynes, and Raymond Kapral. Constrained reaction coordinate dynamics for the simulation of rare events. *Chemical Physics Letters*, 156(5):472–477, April 1989. ISSN 0009-2614. doi: 10.1016/s0009-2614(89)87314-2. URL http://dx.doi.org/10.1016/S0009-2614(89)87314-2.

Nicholas E. Charron, Felix Musil, Andrea Guljas, Yaoyi Chen, Klara Bonneau, Aldo S. Pasos-Trejo, Jacopo Venturin, Daria Gusew, Iryna Zaporozhets, Andreas Krämer, Clark Templeton, Atharva Kelkar, Aleksander E. P. Durumeric, Simon Olsson, Adrià Pérez, Maciej Majewski, Brooke E. Husic, Ankit Patel, Gianni De Fabritiis, Frank Noé, and Cecilia Clementi. Navigating protein landscapes with a machine-learned transferable coarse-grained model, 2023.

Ricky T. Q. Chen, Yulia Rubanova, Jesse Bettencourt, and David Duvenaud. Neural ordinary differential equations, 2018.

Wei Chen and Andrew L. Ferguson. Molecular enhanced sampling with autoencoders: On-the-fly collective variable discovery and accelerated free energy landscape exploration. *Journal of Computational Chemistry*, 39(25):2079–2102, September 2018. ISSN 1096-987X. doi: 10.1002/jcc.25520. URL http://dx.doi.org/10.1002/jcc.25520.

Yaoyi Chen, Andreas Krämer, Nicholas E. Charron, Brooke E. Husic, Cecilia Clementi, and Frank Noé. Machine learning implicit solvation for molecular dynamics. *The Journal of Chemical Physics*, 155(8):084101, August 2021. ISSN 1089-7690. doi: 10.1063/5.0059915. URL http://dx.doi.org/10.1063/5.0059915.

Shriram Chennakesavalu, David J. Toomer, and Grant M. Rotskoff. Ensuring thermodynamic consistency with invertible coarse-graining. *The Journal of Chemical Physics*, 158(12), March 2023. ISSN 1089-7690. doi: 10.1063/5.0141888. URL http://dx.doi.org/10.1063/5.0141888.

Daan Crommelin and Eric Vanden-Eijnden. Diffusion estimation from multiscale data by operator eigenpairs. *Multiscale Modeling & Simulation*, 9(4):1588–1623, October 2011. ISSN 1540-3467. doi: 10.1137/100795917. URL http://dx.doi.org/10.1137/100795917.

Christoph Dellago, Peter G. Bolhuis, and David Chandler. Efficient transition path sampling: Application to lennard-jones cluster rearrangements. *The Journal of Chemical Physics*, 108(22): 9236–9245, June 1998. ISSN 1089-7690. doi: 10.1063/1.476378. URL http://dx.doi.org/10.1063/1.476378.

Manuel Dibak, Leon Klein, Andreas Krämer, and Frank Noé. Temperature steerable flows and boltzmann generators, 2022. URL https://arxiv.org/abs/2108.01590.

S. Doerr and G. De Fabritiis. On-the-fly learning and sampling of ligand binding by high-throughput molecular simulations. *Journal of Chemical Theory and Computation*, 10(5):2064–2069, April 2014. ISSN 1549-9626. doi: 10.1021/ct400919u. URL http://dx.doi.org/10.1021/ct400919u.

David J. Earl and Michael W. Deem. Parallel tempering: Theory, applications, and new perspectives. *Phys. Chem. Chem. Phys.*, 7:3910–3916, 2005. doi: 10.1039/B509983H. URL http://dx.doi.org/10.1039/B509983H.

F Ercolessi and J. B Adams. Interatomic potentials from first-principles calculations: The force-matching method. *Europhysics Letters (EPL)*, 26(8):583–588, June 1994. ISSN 1286-4854. doi: 10.1209/0295-5075/26/8/005. URL http://dx.doi.org/10.1209/0295-5075/26/8/005.

Alan M. Ferrenberg and Robert H. Swendsen. Optimized monte carlo data analysis. *Physical Review Letters*, 63(12):1195–1198, September 1989.

Xiang Fu, Tian Xie, Nathan J. Rebello, Bradley D. Olsen, and Tommi Jaakkola. Simulate time-integrated coarse-grained molecular dynamics with multi-scale graph networks, 2023. URL https://arxiv.org/abs/2204.10348.

Helmut Grubmüller. Predicting slow structural transitions in macromolecular systems: Conformational flooding. *Phys. Rev. E*, 52:2893–2906, 9 1995. doi: 10.1103/PhysRevE.52.2893. URL https://link.aps.org/doi/10.1103/PhysRevE.52.2893.

W. K. Hastings. Monte carlo sampling methods using markov chains and their applications. *Biometrika*, 57(1):97–109, April 1970. doi: 10.1093/biomet/57.1.97. URL https://doi.org/10.1093/biomet/57.1.97.

Tim Hempel, Simon Olsson, and Frank Noé. Markov field models: Scaling molecular kinetics approaches to large molecular machines. *Current Opinion in Structural Biology*, 77:102458, December 2022. ISSN 0959-440X. doi: 10.1016/j.sbi.2022.102458. URL http://dx.doi.org/10.1016/j.sbi.2022.102458.

Jérôme Hénin, Tony Lelièvre, Michael R Shirts, Omar Valsson, and Lucie Delemotte. Enhanced sampling methods for molecular dynamics simulations [article v1.0]. *Living Journal of Computational Molecular Science*, 4(1), 2022. doi: 10.33011/livecoms.4.1.1583. URL https://doi.org/10.33011/livecoms.4.1.1583.

Nicholas S. M. Herringer, Siva Dasetty, Diya Gandhi, Junhee Lee, and Andrew L. Ferguson. Permutationally invariant networks for enhanced sampling (pines): Discovery of multimolecular and solvent-inclusive collective variables. *Journal of Chemical Theory and Computation*, 20(1):178–198, December 2023. ISSN 1549-9626. doi: 10.1021/acs.jctc.3c00923. URL http://dx.doi.org/10.1021/acs.jctc.3c00923.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models, 2020. URL https://arxiv.org/abs/2006.11239.

Moritz Hoffmann, Martin Scherer, Tim Hempel, Andreas Mardt, Brian de Silva, Brooke E Husic, Stefan Klus, Hao Wu, Nathan Kutz, Steven L Brunton, and Frank Noé. Deeptime: a python library for machine learning dynamical models from time series data. *Machine Learning: Science and Technology*, 3(1):015009, December 2021. ISSN 2632-2153. doi: 10.1088/2632-2153/ac3de0. URL http://dx.doi.org/10.1088/2632-2153/ac3de0.

Tim Hsu, Babak Sadigh, Vasily Bulatov, and Fei Zhou. Score dynamics: Scaling molecular dynamics with picoseconds time steps via conditional diffusion model. *Journal of Chemical Theory and Computation*, 20(6):2335–2348, March 2024. ISSN 1549-9626. doi: 10.1021/acs.jctc.3c01361. URL http://dx.doi.org/10.1021/acs.jctc.3c01361.

Brooke E. Husic and Vijay S. Pande. Markov state models: From an art to a science. *Journal of the American Chemical Society*, 140(7):2386–2396, February 2018. doi: 10.1021/jacs.7b12191. URL https://doi.org/10.1021/jacs.7b12191.

Brooke E. Husic, Nicholas E. Charron, Dominik Lemm, Jiang Wang, Adrià Pérez, Maciej Majewski, Andreas Krämer, Yaoyi Chen, Simon Olsson, Gianni de Fabritiis, Frank Noé, and Cecilia Clementi. Coarse graining molecular dynamics with graph neural networks. *The Journal of Chemical Physics*, 153(19), November 2020. ISSN 1089-7690. doi: 10.1063/5.0026133. URL http://dx.doi.org/10.1063/5.0026133.

Bowen Jing, Hannes Stärk, Tommi Jaakkola, and Bonnie Berger. Generative modeling of molecular dynamics trajectories, 2024.

Hendrik Jung, Roberto Covino, A. Arjun, Christian Leitold, Christoph Dellago, Peter G. Bolhuis, and Gerhard Hummer. Machine-guided path sampling to discover mechanisms of molecular self-organization. *Nature Computational Science*, 3(4):334–345, April 2023. ISSN 2662-8457. doi: 10.1038/s43588-023-00428-z. URL http://dx.doi.org/10.1038/s43588-023-00428-z.

Anna S. Kamenik, Stephanie M. Linker, and Sereina Riniker. Enhanced sampling without borders: on global biasing functions and how to reweight them. *Physical Chemistry Chemical Physics*, 24 (3):1225–1236, 2022. ISSN 1463-9084. doi: 10.1039/d1cp04809k. URL http://dx.doi.org/10.1039/D1CP04809K.

Paul Katzberger and Sereina Riniker. A general graph neural network based implicit solvation model for organic molecules in water. *Chemical Science*, 15(28):10794–10802, 2024. ISSN 2041-6539. doi: 10.1039/d4sc02432j. URL http://dx.doi.org/10.1039/D4SC02432J.

Leon Klein and Frank Noé. Transferable boltzmann generators, 2024. URL https://arxiv.org/abs/2406.14426.

Leon Klein, Andrew Y. K. Foong, Tor Erlend Fjelde, Bruno Mlodozeniec, Marc Brockschmidt, Sebastian Nowozin, Frank Noé, and Ryota Tomioka. Timewarp: Transferable acceleration of molecular dynamics by learning time-coarsened dynamics, 2023. URL https://arxiv.org/abs/2302.01170.

Jonas Köhler, Leon Klein, and Frank Noe. Equivariant flows: Exact likelihood generative learning for symmetric densities. In Hal Daumé III and Aarti Singh (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 5361–5370. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/kohler20a.html.

Jonas Köhler, Michele Invernizzi, Pim De Haan, and Frank Noe. Rigid body flows for sampling molecular crystal structures. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 17301–17326. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/kohler23a.html.

Christopher Kolloff and Simon Olsson. Rescuing off-equilibrium simulation data through dynamic experimental data with dynammo. *Machine Learning: Science and Technology*, 4(4):045050, December 2023. ISSN 2632-2153. doi: 10.1088/2632-2153/ad10ce. URL http://dx.doi.org/10.1088/2632-2153/ad10ce.

Alessandro Laio and Michele Parrinello. Escaping free-energy minima. *Proceedings of the National Academy of Sciences*, 99(20):12562–12566, 2002.

Paul Langevin. Sur la théorie du mouvement brownien. *C. R. Acad. Sci. (Paris)*, 146:530—-533, 1908.

Kresten Lindorff-Larsen, Stefano Piana, Ron O. Dror, and David E. Shaw. How fast-folding proteins fold. *Science*, 334(6055):517–520, October 2011. ISSN 1095-9203. doi: 10.1126/science.1208351. URL http://dx.doi.org/10.1126/science.1208351.

Alexander P. Lyubartsev and Aatto Laaksonen. Calculation of effective interaction potentials from radial distribution functions: A reverse monte carlo approach. *Physical Review E*, 52(4):3730–3737, October 1995. ISSN 1095-3787. doi: 10.1103/physreve.52.3730. URL http://dx.doi.org/10.1103/PhysRevE.52.3730.

Mohammad M. Sultan and Vijay S. Pande. tica-metadynamics: Accelerating metadynamics by using kinetically selected collective variables. *Journal of Chemical Theory and Computation*, 13(6):2440–2447, May 2017. ISSN 1549-9626. doi: 10.1021/acs.jctc.7b00182. URL http://dx.doi.org/10.1021/acs.jctc.7b00182.

Maciej Majewski, Adrià Pérez, Philipp Thölke, Stefan Doerr, Nicholas E. Charron, Toni Giorgino, Brooke E. Husic, Cecilia Clementi, Frank Noé, and Gianni De Fabritiis. Machine learning coarse-grained potentials of protein thermodynamics. *Nature Communications*, 14(1), September 2023. ISSN 2041-1723. doi: 10.1038/s41467-023-41343-1. URL http://dx.doi.org/10.1038/s41467-023-41343-1.

Dimitra Maoutsa, Sebastian Reich, and Manfred Opper. Interacting particle solutions of fokker–planck equations through gradient–log–density estimation. *Entropy*, 22(8):802, July 2020. ISSN 1099-4300. doi: 10.3390/e22080802. URL http://dx.doi.org/10.3390/e22080802.

Andreas Mardt, Luca Pasquali, Hao Wu, and Frank Noé. VAMPnets for deep learning of molecular kinetics. *Nature Communications*, 9(1), jan 2018. doi: 10.1038/s41467-017-02388-1. URL https://doi.org/10.1038%2Fs41467-017-02388-1.

Andreas Mardt, Tim Hempel, Cecilia Clementi, and Frank Noé. Deep learning to decompose macromolecules into independent markovian domains. *Nature Communications*, 13(1), November 2022. ISSN 2041-1723. doi: 10.1038/s41467-022-34603-z. URL http://dx.doi.org/10.1038/s41467-022-34603-z.

Laurence I. Midgley, Vincent Stimper, Javier Antorán, Emile Mathieu, Bernhard Schölkopf, and José Miguel Hernández-Lobato. Se(3) equivariant augmented coupling flows, 2024. URL https://arxiv.org/abs/2308.10364.

Laurence Illing Midgley, Vincent Stimper, Gregor N. C. Simm, Bernhard Schölkopf, and José Miguel Hernández-Lobato. Flow annealed importance sampling bootstrap, 2023. URL https://arxiv.org/abs/2208.01893.

L. Molgedey and H. G. Schuster. Separation of a mixture of independent signals using time delayed correlations. *Phys. Rev. Lett.*, 72:3634–3637, Jun 1994. doi: 10.1103/PhysRevLett.72.3634. URL https://link.aps.org/doi/10.1103/PhysRevLett.72.3634.

Kim A. Nicoli, Shinichi Nakajima, Nils Strodthoff, Wojciech Samek, Klaus-Robert Müller, and Pan Kessel. Asymptotically unbiased estimation of physical observables with neural samplers. *Physical Review E*, 101(2), February 2020. ISSN 2470-0053. doi: 10.1103/physreve.101.023304. URL http://dx.doi.org/10.1103/PhysRevE.101.023304.

W. G. Noid. Perspective: Advances, challenges, and insight for predictive coarse-grained models. *The Journal of Physical Chemistry B*, 127(19):4174–4207, May 2023. ISSN 1520-5207. doi: 10.1021/acs.jpcb.2c08731. URL http://dx.doi.org/10.1021/acs.jpcb.2c08731.

W. G. Noid, Jhih-Wei Chu, Gary S. Ayton, Vinod Krishna, Sergei Izvekov, Gregory A. Voth, Avisek Das, and Hans C. Andersen. The multiscale coarse-graining method. i. a rigorous bridge between atomistic and coarse-grained models. *The Journal of Chemical Physics*, 128(24), June 2008. ISSN 1089-7690. doi: 10.1063/1.2938860. URL http://dx.doi.org/10.1063/1.2938860.

Frank Noé, Simon Olsson, Jonas Köhler, and Hao Wu. Boltzmann generators: Sampling equilibrium states of many-body systems with deep learning. *Science*, 365(6457), September 2019. ISSN 1095-9203. doi: 10.1126/science.aaw1147. URL http://dx.doi.org/10.1126/science.aaw1147.

Feliks Nüske, Hao Wu, Jan-Hendrik Prinz, Christoph Wehmeyer, Cecilia Clementi, and Frank Noé. Markov state models from short non-equilibrium simulations—analysis and correction of estimation bias. *The Journal of Chemical Physics*, 146(9), March 2017. ISSN 1089-7690. doi: 10.1063/1.4976518. URL http://dx.doi.org/10.1063/1.4976518.

Feliks Nüske, Lorenzo Boninsegna, and Cecilia Clementi. Coarse-graining molecular systems by spectral matching. *The Journal of Chemical Physics*, 151(4), July 2019. ISSN 1089-7690. doi: 10.1063/1.5100131. URL http://dx.doi.org/10.1063/1.5100131.

Simon Olsson. Markov state models of protein–protein encounters, November 2022. URL http://dx.doi.org/10.1002/9783527830503.ch9.

Simon Olsson and Frank Noé. Dynamic graphical models of molecular kinetics. *Proceedings of the National Academy of Sciences*, 116(30):15001–15006, July 2019. doi: 10.1073/pnas.1901692116. URL https://doi.org/10.1073/pnas.1901692116.

Simon Olsson, Hao Wu, Fabian Paul, Cecilia Clementi, and Frank Noé. Combining experimental and simulation data of molecular processes via augmented markov models. *Proceedings of the National Academy of Sciences*, 114(31):8265–8270, July 2017. ISSN 1091-6490. doi: 10.1073/pnas.1704803114. URL http://dx.doi.org/10.1073/pnas.1704803114.

George Papamakarios, Eric Nalisnick, Danilo Jimenez Rezende, Shakir Mohamed, and Balaji Lakshminarayanan. Normalizing flows for probabilistic modeling and inference, 2021. URL https://arxiv.org/abs/1912.02762.

Amey P Pasarkar, Gianluca M Bencomo, Simon Olsson, and Adji Bousso Dieng. Vendi sampling for molecular simulations: Diversity as a force for faster convergence and better exploration. *The Journal of chemical physics*, 159(14), 10 2023. ISSN 1089-7690 (Electronic). doi: 10.1063/5.0166172.

Michael Plainer, Hannes Stärk, Charlotte Bunne, and Stephan Günnemann. Transition path sampling with boltzmann generator-based mcmc moves, 2023.

Jan-Hendrik Prinz, Hao Wu, Marco Sarich, Bettina Keller, Martin Senne, Martin Held, John D. Chodera, Christof Schütte, and Frank Noé. Markov models of molecular kinetics: Generation and validation. *The Journal of Chemical Physics*, 134(17):174105, May 2011. doi: 10.1063/1.3565032. URL https://doi.org/10.1063/1.3565032.

Guillermo Pérez-Hernández, Fabian Paul, Toni Giorgino, Gianni De Fabritiis, and Frank Noé. Identification of slow molecular order parameters for markov model construction. *The Journal of Chemical Physics*, 139(1), July 2013. ISSN 1089-7690. doi: 10.1063/1.4811489. URL http://dx.doi.org/10.1063/1.4811489.

Edina Rosta and Gerhard Hummer. Free energies from dynamic weighted histogram analysis using unbiased markov state model. *Journal of Chemical Theory and Computation*, 11(1):276–285, December 2014. ISSN 1549-9626. doi: 10.1021/ct500719p. URL http://dx.doi.org/10.1021/ct500719p.

David Ruelle. *Thermodynamic Formalism*. Encyclopedia of mathematics and its applications. Longman Higher Education, Harlow, England, November 1978.

Nicola Salvi, Anton Abyzov, and Martin Blackledge. Multi-timescale dynamics in intrinsically disordered proteins from nmr relaxation and molecular simulation. *The Journal of Physical Chemistry Letters*, 7(13):2483–2489, June 2016. ISSN 1948-7185. doi: 10.1021/acs.jpclett.6b00885. URL http://dx.doi.org/10.1021/acs.jpclett.6b00885.

Mathias Schreiner, Ole Winther, and Simon Olsson. Implicit transfer operator learning: Multiple time-resolution surrogates for molecular dynamics, 2023. URL https://arxiv.org/abs/2305.18046.

Christof Schütte, Alexander Fischer, Wilhelm Huisinga, and Peter Deuflhard. A hybrid monte carlo method for essential molecular dynamics. Technical Report SC-98-04, ZIB, Takustr. 7, 14195 Berlin, 1998.

Christof Schütte, Frank Noe, Eike Meerbach, Philipp Metzner, and Carsten Hartmann. Conformation dynamics. *Proc. Int. Congr. ICIAM*, pp. 297–336, 2009.

Michael R. Shirts and John D. Chodera. Statistically optimal analysis of samples from multiple equilibrium states. *The Journal of Chemical Physics*, 129(12), September 2008. ISSN 1089-7690. doi: 10.1063/1.2978177. URL http://dx.doi.org/10.1063/1.2978177.

Hythem Sidky, Wei Chen, and Andrew L. Ferguson. Molecular latent space simulators. *Chemical Science*, 11(35):9459–9467, 2020. ISSN 2041-6539. doi: 10.1039/d0sc03635h. URL http://dx.doi.org/10.1039/D0SC03635H.

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. *The Journal of Chemical Physics*, 145(15), October 2016. ISSN 1089-7690. doi: 10.1063/1.4964781. URL http://dx.doi.org/10.1063/1.4964781.

Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations, 2020. URL https://arxiv.org/abs/2011.13456.

Yang Song, Conor Durkan, Iain Murray, and Stefano Ermon. Maximum likelihood training of score-based diffusion models, 2021.

William C. Swope, Jed W. Pitera, and Frank Suits. Describing protein folding kinetics by molecular dynamics simulations. 1. theory. *The Journal of Physical Chemistry B*, 108(21):6571–6581, April 2004. doi: 10.1021/jp037421y. URL https://doi.org/10.1021/jp037421y.

G.M. Torrie and J.P. Valleau. Nonphysical sampling distributions in monte carlo free-energy estimation: Umbrella sampling. *Journal of Computational Physics*, 23(2):187–199, February 1977. doi: 10.1016/0021-9991(77)90121-8. URL https://doi.org/10.1016/0021-9991(77)90121-8.

Benjamin Trendelkamp-Schroer, Hao Wu, Fabian Paul, and Frank Noé. Estimation and uncertainty of reversible markov models. *The Journal of Chemical Physics*, 143(17), November 2015. ISSN 1089-7690. doi: 10.1063/1.4934536. URL http://dx.doi.org/10.1063/1.4934536.

Juan Viguera Diez, Sara Romeo Atance, Ola Engkvist, and Simon Olsson. Generation of conformational ensembles of small molecules via surrogate model-assisted molecular dynamics. *Machine Learning: Science and Technology*, 5(2):025010, April 2024. ISSN 2632-2153. doi: 10.1088/2632-2153/ad3b64. URL http://dx.doi.org/10.1088/2632-2153/ad3b64.

Dedi Wang, Yihang Wang, Luke Evans, and Pratyush Tiwary. From latent dynamics to meaningful representations. *Journal of Chemical Theory and Computation*, 20(9):3503–3513, April 2024. ISSN 1549-9626. doi: 10.1021/acs.jctc.4c00249. URL http://dx.doi.org/10.1021/acs.jctc.4c00249.

Jiang Wang, Simon Olsson, Christoph Wehmeyer, Adrià Pérez, Nicholas E. Charron, Gianni de Fabritiis, Frank Noé, and Cecilia Clementi. Machine learning of coarse-grained molecular dynamics force fields. *ACS Central Science*, 5(5):755–767, April 2019a. ISSN 2374-7951. doi: 10.1021/acscentsci.8b00913. URL http://dx.doi.org/10.1021/acscentsci.8b00913.

Jiang Wang, Nicholas Charron, Brooke Husic, Simon Olsson, Frank Noé, and Cecilia Clementi. Multi-body effects in a coarse-grained protein force field. *The Journal of Chemical Physics*, 154 (16), April 2021. ISSN 1089-7690. doi: 10.1063/5.0041022. URL http://dx.doi.org/10.1063/5.0041022.

Wujie Wang and Rafael Gómez-Bombarelli. Coarse-graining auto-encoders for molecular dynamics. *npj Computational Materials*, 5(1), December 2019. ISSN 2057-3960. doi: 10.1038/s41524-019-0261-5. URL http://dx.doi.org/10.1038/s41524-019-0261-5.

Yihang Wang, João Marcelo Lamim Ribeiro, and Pratyush Tiwary. Past–future information bottleneck for sampling molecular reaction coordinate simultaneously with thermodynamics and kinetics. *Nature Communications*, 10(1), August 2019b. ISSN 2041-1723. doi: 10.1038/s41467-019-11405-4. URL http://dx.doi.org/10.1038/s41467-019-11405-4.

Hao Wu and Frank Noé. Variational approach for learning markov processes from time series data. *Journal of Nonlinear Science*, 30(1):23–66, aug 2019. doi: 10.1007/s00332-019-09567-y. URL https://doi.org/10.1007%2Fs00332-019-09567-y.

Hao Wu, Antonia S. J. S. Mey, Edina Rosta, and Frank Noé. Statistically optimal analysis of state-discretized trajectory data from multiple thermodynamic states. *The Journal of Chemical Physics*, 141(21), December 2014. ISSN 1089-7690. doi: 10.1063/1.4902240. URL http://dx.doi.org/10.1063/1.4902240.

Hao Wu, Jan-Hendrik Prinz, and Frank Noé. Projected metastable markov processes and their estimation with observable operator models. *The Journal of Chemical Physics*, 143(14):144101, October 2015. doi: 10.1063/1.4932406. URL https://doi.org/10.1063/1.4932406.

Hao Wu, Fabian Paul, Christoph Wehmeyer, and Frank Noé. Multiensemble markov models of molecular thermodynamics and kinetics. *Proceedings of the National Academy of Sciences*, 113 (23), May 2016. ISSN 1091-6490. doi: 10.1073/pnas.1525092113. URL http://dx.doi.org/10.1073/pnas.1525092113.

Hao Wu, Andreas Mardt, Luca Pasquali, and Frank Noe. Deep generative markov state models, 2019. URL https://arxiv.org/abs/1805.07601.

Andreas Ziehe and Klaus-Robert Müller. *TDSEP — an efficient algorithm for blind separation using time structure*, pp. 675–680. Springer London, 1998. ISBN 9781447115991. doi: 10.1007/978-1-4471-1599-1_103. URL http://dx.doi.org/10.1007/978-1-4471-1599-1_103.

A APPENDIX

A.1 DEFINITIONS

- **Configuration space**: Mathematical space in which all possible states or positions of a physical system are represented. For example, in a classical MD simulation, the positions of all the atoms in the simulation.

- **Unbiased simulations**: Simulations performed with standard MD, i.e. Langevin Dynamics. They are unbiased because they generate samples from the underlying transition density. Particularly, for asymptotically large simulation times, unbiased simulations sample the Boltzmann distribution $\mu(\mathbf{x})$. However, because of energy barriers in the potential energy landscape, unbiased simulations may not explore some modes of the Boltzmann distribution if the simulation time is not long enough. Therefore, generating samples with unbiased MD often leads to off-equilibrium data.

- **Off-equilibrium data**: Simulation trajectories whose underlying statistics do not represent well those of the equilibrium distribution. One example is lack of ergodicity: when the simulation fails to explore some modes in the Boltzmann distribution. Even if having explored the full state space, it is possible to have off-equilibrium trajectories because of rare events, which require to be sampled "many" times to accurately represent equilibrium statistics.

- **Biased simulation**: Simulations performed with modified versions of naive MD that speed up the exploration of state space. These methods are a subset of enhanced sampling and include meta-dynamics (Laio & Parrinello, 2002), replica exchange (Earl & Deem, 2005) and others (Sidler et al., 2016; Pasarkar et al., 2023). Data generated with enhanced sampling do not resemble the underlying transition density, but, often, can be re-weighted to represent the equilibrium distribution.

A.2 EIGEN DECOMPOSITION OF THE TRANSFER OPERATOR AND TRANSITION DENSITY

The two-fold composition of the transfer operator, eq. 5, acting on an initial μ-weighted density, ρ, is

$$
\begin{aligned}
[T_\Omega^2(\tau) \circ \rho](\mathbf{x}_{t+2\tau}) &= \sum_{i=1}^{\infty} \lambda_i(\tau) \left\langle \sum_{j=1}^{\infty} \lambda_j(\tau) \langle \rho | \phi_j \rangle \ \psi_j | \phi_i \right\rangle \ \psi_i(\mathbf{x}_{t+2\tau}) \\
&= \sum_{i=1}^{\infty} \lambda_i(\tau) \left(\sum_{j=1}^{\infty} \lambda_j(\tau) \langle \rho | \phi_j \rangle \ \langle \psi_j | \phi_i \rangle \right) \ \psi_i(\mathbf{x}_{t+2\tau}) \\
&= \sum_{i=1}^{\infty} \lambda_i^2(\tau) \langle \rho | \phi_i \rangle \ \psi_i(\mathbf{x}_{t+2\tau}),
\end{aligned}
$$

where we have used the orthonormality of the eigenfunctions, that is $\langle \psi_j | \phi_i \rangle = \delta_{ij}$. Similarly,

$$
[T_\Omega^N(\tau) \circ \rho](\mathbf{x}_{t+N\tau}) = \sum_{i=1}^{\infty} \lambda_i^N(\tau) \langle \rho | \phi_i \rangle \ \psi_i(\mathbf{x}_{t+N\tau}). \tag{19}
$$

The transition density can be re-written in terms of the spectral decomposition of the transfer operator by choosing the initial density, p, as a Dirac delta function $\delta_{\mathbf{x}_t}(\mathbf{x})$, that is $\rho(\mathbf{x}) = \frac{\delta_{\mathbf{x}_t}(\mathbf{x})}{\mu(\mathbf{x})}$.

Then,

$$
\begin{aligned}
p(\mathbf{x}_{t+N\tau}|\mathbf{x}_t) &= \mu(\mathbf{x}_{t+N\tau}) \left[T_\Omega^N(\tau) \circ \frac{\delta_{\mathbf{x}_t}}{\mu} \right] (\mathbf{x}_{t+N\tau}) \\
&= \mu(\mathbf{x}_{t+N\tau}) \sum_{i=1}^{\infty} \lambda_i^N(\tau) \left\langle \frac{\delta_{\mathbf{x}_t}}{\mu} \middle| \phi_i \right\rangle \psi_i(\mathbf{x}_{t+N\tau}) \\
&= \sum_{i=1}^{\infty} \lambda_i^N(\tau) \langle \delta_{\mathbf{x}_t}(\mathbf{x}) | \psi_i \rangle \ \psi_i(\mathbf{x}_{t+N\tau}) \ \mu(\mathbf{x}_{t+N\tau}) \\
&= \sum_{i=1}^{\infty} \lambda_i^N(\tau) \psi_i(\mathbf{x}_t) \phi_i(\mathbf{x}_{t+N\tau}).
\end{aligned}
$$

A.3 Implicit Transfer Operator details

Implicit Transfer Operator (ITO) Learning (Schreiner et al., 2023) is a framework for learning surrogate models of the transition density, $p(\mathbf{x}_{N\tau}|\mathbf{x}_0)$. ITO models are trained sampling tuples $(\mathbf{x}_{t_i}, \mathbf{x}_{t_i+N_i\tau}, N_i)$ and training a generative model to mimic the empirical transition density at multiple time-horizons, $N\tau$. Training is done following Algorithm 1.

Algorithm 1 Training. DisExp is defined in Algorithm 4

Input: n MD-trajectories; $\mathcal{X} = \{\mathbf{x}_0^j, \ldots, \mathbf{x}_{t_j}^j\}_{j=0}^n$, ITO score-model; $\hat{\epsilon}_\theta$, max lag; N_{max}
$\mathcal{X}' = \text{Concatenate}(\{\mathbf{x}_0^j, \ldots, \mathbf{x}_{t_j - N_{\text{max}}}^j\}_{j=0}^n)$
while not converged **do**
  $\mathbf{x}_t \sim \text{Choice}(\mathcal{X}')$
  $N \sim \text{DisExp}(N_{\text{max}})$
  $t_{\text{diff}} \sim \text{Uniform}(0, T)$
  Take gradient step on:
  $\nabla_{\boldsymbol{\theta}} \left[\| \epsilon - \hat{\epsilon}_{\boldsymbol{\theta}}(\widetilde{\mathbf{x}}_{t+N\tau}^{t_{\text{diff}}}, \mathbf{x}_t, N, t_{\text{diff}}) \|_2 \right]$
end while
return $\hat{\epsilon}_{\boldsymbol{\theta}}$

Once a model is trained, it can be sampled by following Algorithm 2.

Algorithm 2 Sampling from $\hat{p}_{\boldsymbol{\theta}}(\mathbf{x}_0, N)$

Input: initial condition $\mathbf{x}_0$, lag; N, diffusion steps; T_{diff}, ITO score-model; $\hat{\epsilon}_\theta$
$\mathbf{x}_N^{T_{\text{diff}}} \sim \mathcal{N}(\mathbf{0}, \mathbf{1})$
for $t_{\text{diff}} = T_{\text{diff}} \ldots 1$ **do**
  $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{1})$
  $\mathbf{x}_N^{t_{\text{diff}}-1} = \frac{1}{\sqrt{\alpha^{t_{\text{diff}}}}} \left(\mathbf{x}_N^{t_{\text{diff}}} - \frac{1-\alpha^{t_{\text{diff}}}}{\sqrt{1-\bar{\alpha}^{t_{\text{diff}}}}} \hat{\epsilon}_\theta(\mathbf{x}_N^{t_{\text{diff}}}, \mathbf{x}_0, N, t_{\text{diff}}) \right) + \sqrt{\beta_t}\epsilon$
end for
return $\mathbf{x}_N^0$

Several sampling steps can be annealed to sample longer lag-times as depicted in Algorithm 3.

Algorithm 3 Ancestral sampling. Sampling from $p_{\boldsymbol{\theta}}$ is defined in Algorithm 2

Input: initial condition $\mathbf{x}_0$, lag N, ancestral steps n.
Allocate $\mathcal{T} \in \mathbb{R}^{(n+1)\times \dim(\mathbf{x}_0)}$
$\mathcal{T}[0] = \mathbf{x}_0$
for $i = 1 \dots n$ **do**
  $\mathbf{x}_i \sim \hat{p}_{\boldsymbol{\theta}}(\mathcal{T}[i-1], N)$
  $\mathcal{T}[i] = \mathbf{x}_i$
end for
return $\mathcal{T}$

Algorithm 4 Sampling from DisExp

$N_{\log} \sim \text{Uniform}(0, \log(N_{\max}))$
Return: $\text{floor}(\exp(N_{\log}))$

BoPITO uses Algorithm 1 for training and Algorithm 2 for sampling as well, but uses the score mode in eq. 14.

A.4 Decay of time-dependent score term

In Figure 6, we show the average time-dependent component of the score,

$$s_N = \mathbb{E}_{i\sim\mathbf{I},t_{\text{diff}}\sim\mathcal{U}(0,T)}\left[|\hat{\lambda}^N s_{\text{dyn}}(\mathbf{x}_{t_i+N\tau}^{t_{\text{diff}}}, \mathbf{x}_{t_i}, N, t_{\text{diff}}, \boldsymbol{\theta})|\right], \tag{20}$$

of a trained BoPITO model of the Prinz Potential with $\hat{\lambda} = 0.994$. The expectation for $\mathbf{I}$ corresponding to the sampling of tuples as during training (Alg. 1). The model remains flexible for small lags but eventually decreases to 0, sampling from the equilibrium model.

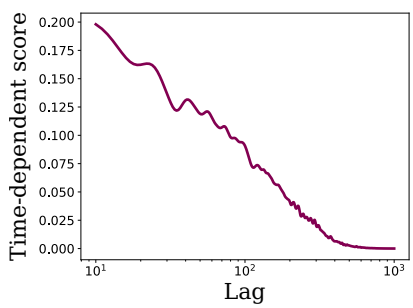


Figure 6: Average time-dependent component of the score, eq. 20, of a trained model BoPITO model of the Prinz Potential.

A.5 THE SCORE OF THE TRANSITION PROBABILITY AND BOPITO

The score of the transition probability can be written as

$$\begin{aligned}\nabla_{\mathbf{x}_{N\tau}} \log p(\mathbf{x}_{N\tau}|\mathbf{x}_0, N) &= \sum_i^\infty \lambda_i(\tau)^N \frac{\psi_i(\mathbf{x}_0)}{p(\mathbf{x}_{N\tau}|\mathbf{x}_0, N)} \nabla_{\mathbf{x}_{N\tau}} \phi_i(\mathbf{x}_{N\tau}) \\ &= \frac{\nabla_{\mathbf{x}_{N\tau}} \mu(\mathbf{x}_{N\tau})}{p(\mathbf{x}_{N\tau}|\mathbf{x}_0, N)} + \sum_{i=2}^\infty \lambda_i(\tau)^N \frac{\psi_i(\mathbf{x}_0)}{p(\mathbf{x}_{N\tau}|\mathbf{x}_0, N)} \nabla_{\mathbf{x}_{N\tau}} \phi_i(\mathbf{x}_{N\tau}) \\ &= \frac{\nabla_{\mathbf{x}_{N\tau}} \mu(\mathbf{x}_{N\tau})}{\mu(\mathbf{x}_{N\tau})} \frac{\mu(\mathbf{x}_{N\tau})}{p(\mathbf{x}_{N\tau}|\mathbf{x}_0, N)} \\ &\quad + \hat{\lambda}^N \sum_{i=2}^\infty \left(\frac{\lambda_i(\tau)}{\hat{\lambda}}\right)^N \frac{\psi_i(\mathbf{x}_0)}{p(\mathbf{x}_{N\tau}|\mathbf{x}_0, N)} \nabla_{\mathbf{x}_{N\tau}} \phi_i(\mathbf{x}_{N\tau}).\end{aligned}$$

$\frac{\nabla_{\mathbf{x}_{N\tau}} \mu(\mathbf{x}_{N\tau})}{\mu(\mathbf{x}_{N\tau})} = \nabla_{\mathbf{x}_{N\tau}} \log \mu(\mathbf{x}_{N\tau})$ is the score of the Boltzmann distribution and can be modeled with the score of a Boltzmann Generator, $s_{\mathrm{eq}}(\mathbf{x}^{t_{\mathrm{diff}}}, t_{\mathrm{diff}})$. $\frac{\mu(\mathbf{x}_{N\tau})}{p(\mathbf{x}_{N\tau}|\mathbf{x}_0, N)} \to 1$ as $N \to \infty$ and can be modeled with a scalar neural network $1 + \hat{\lambda}^N f_{\boldsymbol{\theta}}(\mathbf{x}_{N\tau}^{t_{\mathrm{diff}}}, \mathbf{x}_0, N, t_{\mathrm{diff}})$. The last term can be modeled with $g_{\boldsymbol{\theta}}(\mathbf{x}_{N\tau}^{t_{\mathrm{diff}}}, \mathbf{x}_0, N, t_{\mathrm{diff}})$. The corresponding score model is

$$\begin{aligned}s_{\boldsymbol{\theta}}(\mathbf{x}_{N\tau}^{t_{\mathrm{diff}}}, \mathbf{x}_0, N, t_{\mathrm{diff}}) &= s_{\mathrm{eq}}(\mathbf{x}^{t_{\mathrm{diff}}}, t_{\mathrm{diff}})(1 + \hat{\lambda}^N f_{\boldsymbol{\theta}}(\mathbf{x}_{N\tau}^{t_{\mathrm{diff}}}, \mathbf{x}_0, N, t_{\mathrm{diff}}) + \hat{\lambda}^N g_{\boldsymbol{\theta}}(\mathbf{x}_{N\tau}^{t_{\mathrm{diff}}}, \mathbf{x}_0, N, t_{\mathrm{diff}}) \\ &= s_{\mathrm{eq}}(\mathbf{x}^{t_{\mathrm{diff}}}, t_{\mathrm{diff}}) + \hat{\lambda}^N \underbrace{\left(s_{\mathrm{eq}}(\mathbf{x}^{t_{\mathrm{diff}}}, t_{\mathrm{diff}}) f_{\boldsymbol{\theta}}(\mathbf{x}_{N\tau}^{t_{\mathrm{diff}}}, \mathbf{x}_0, N, t_{\mathrm{diff}}) + g_{\boldsymbol{\theta}}(\mathbf{x}_{N\tau}^{t_{\mathrm{diff}}}, \mathbf{x}_0, N, t_{\mathrm{diff}})\right)}_{s_{\mathrm{dyn}}(\mathbf{x}_{N\tau}^{t_{\mathrm{diff}}}, \mathbf{x}_0, N, t_{\mathrm{diff}})}.\end{aligned}$$

We can aggregate f and g to a single neural network component, $s_{\mathrm{dyn}}(\mathbf{x}_{N\tau}^{t_{\mathrm{diff}}}, \mathbf{x}_0, N, t_{\mathrm{diff}})$, to get

$$s_{\boldsymbol{\theta}}(\mathbf{x}_{N\tau}^{t_{\mathrm{diff}}}, \mathbf{x}_0, N, t_{\mathrm{diff}}) = s_{\mathrm{eq}}(\mathbf{x}^{t_{\mathrm{diff}}}, t_{\mathrm{diff}}) + \hat{\lambda}^N s_{\mathrm{dyn}}(\mathbf{x}_{N\tau}^{t_{\mathrm{diff}}}, \mathbf{x}_0, N, t_{\mathrm{diff}}). \tag{21}$$

For simplicity, we incorporate the structure of the transition density into the diffusion model, not only for $t_{\mathrm{diff}} = 0$ (the learned data distribution), but for all $t_{\mathrm{diff}} > 0$. We do not observe this choice to limit model expressivity in our experiments.

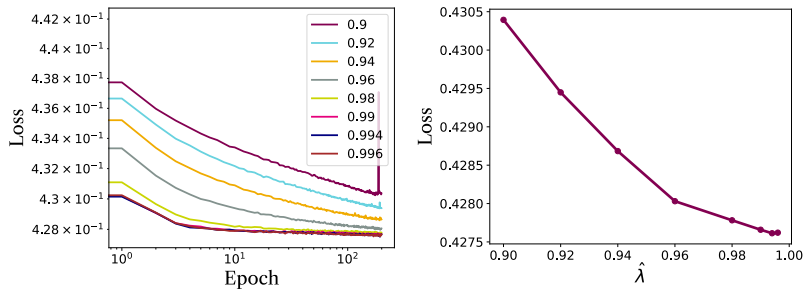
A.6 FITTING $\hat{\lambda}$

The hyperparameter $\hat{\lambda}$ controls the time-scale at which the BoPITO model transitions to sampling the equilibrium model. Ideally, $\hat{\lambda}$ should be similar to the eigenvalue corresponding with the slowest process in the system, λ_2. However, its value is not generally available and requires extensive unbiased simulation data to be accurately estimated. If $\hat{\lambda}$ is too small, the model may prematurely relax to equilibrium, limiting its ability to capture non-equilibrium dynamics. Conversely, if $\hat{\lambda}$ is too large, the benefits of the BoPITO framework could diminish. Therefore, careful tuning of $\hat{\lambda}$ can be crucial for optimal performance.

As illustrated in Figure 7, grid-search hyper-parameter tuning can select an appropriate $\hat{\lambda}$ for a single-system model. Too small values of $\hat{\lambda}$ can hinder the model's ability to capture long-time-scale dynamics, leading to increased loss. We recommend choosing the smallest $\hat{\lambda}$ that yields a plateau in the $\hat{\lambda}$-loss curve (elbow rule), as demonstrated in Figure 7 (b) for the Prinz potential. However, practitioners should be cautious about sampling longer lags than the implied time-scale defined by $\hat{\lambda}$ if they cannot guarantee the system relaxes to equilibrium for those time-scales. Fitting curves for Alanine Dipeptide and Chignolin are shown in Figure 8.

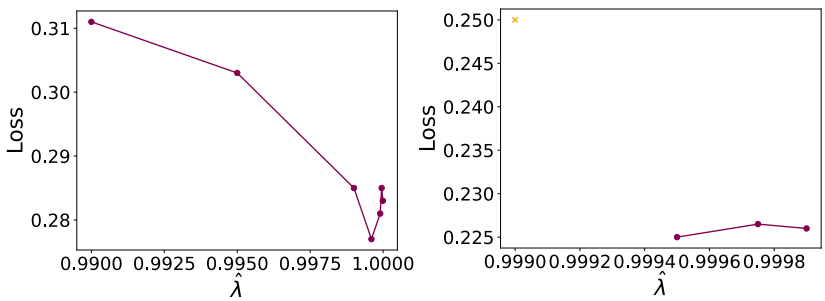
A.7 ALTERNATIVE SCORE FORMULATIONS

The score in eq. 14 is principled and resembles the score of the transition density. However, in practice, we found that this score can lead to suboptimal performance in learning short time-scales if the dynamic component fails to dominate the equilibrium component for small N. We only observed this for Chignolin. We relate this issue to numerical limitations due to very different scale



(a) (b)

Figure 7: Average loss against training epochs for the Prinz potential for different values of $\hat{\lambda}$ (a) and average final loss versus $\hat{\lambda}$ (b). Averages are taken w.r.t. 10 runs. Final losses are computed as the average between epochs 190 and 200. The greatest eigenvalue of the system under our simulation parameters is 0.994.



(a) (b)

Figure 8: Average final loss against different values of $\hat{\lambda}$ for Alanine Dipeptide (a) and Chignolin (b). The smallest value of $\hat{\lambda}$ that allowed numerically stable training for Chignolin was 0.9995. Orange cross symbolizes that the trainings crashed because of numerical instability reasons.

in neural network weights for different values of N. We remark that short time-scales are the least relevant since they can be easily sampled with MD. Still, to mitigate this effect we propose the following alternative factorization,

$$s_{\boldsymbol{\theta}}(\mathbf{x}_{N\tau}^{t_{\text{diff}}}, \mathbf{x}_0, N, t_{\text{diff}}) = f(N) s_{\text{eq}}(\mathbf{x}^{t_{\text{diff}}}, t_{\text{diff}}) + \hat{\lambda}^N s_{\text{dyn}}(\mathbf{x}_{N\tau}^{t_{\text{diff}}}, \mathbf{x}_0, N, t_{\text{diff}}), \tag{22}$$

where $f(N)$ is an increasing function and tends to 1 for large N. One potential option that does not introduce extra hyper-parameters is

$$s_{\boldsymbol{\theta}}(\mathbf{x}_{N\tau}^{t_{\text{diff}}}, \mathbf{x}_0, N, t_{\text{diff}}) = \frac{\hat{\lambda}^{-2N} - \hat{\lambda}^{2N}}{\hat{\lambda}^{-2N} + \hat{\lambda}^{2N}} s_{\text{eq}}(\mathbf{x}^{t_{\text{diff}}}, t_{\text{diff}}) + \hat{\lambda}^N s_{\text{dyn}}(\mathbf{x}_{N\tau}^{t_{\text{diff}}}, \mathbf{x}_0, N, t_{\text{diff}}). \tag{23}$$

Moreover, to ensure adequate convergence to the equilibrium distribution and avoid over-fitting to unbiased MD for long time-scales, we use the following decay score for $N > N_{\text{decay}}$,

$$s_{\boldsymbol{\theta}}(\mathbf{x}_{N\tau}^{t_{\text{diff}}}, \mathbf{x}_0, N, t_{\text{diff}}) = s_{\text{eq}}(\mathbf{x}^{t_{\text{diff}}}, t_{\text{diff}}) + \hat{\lambda}^N s_{\text{dyn}}(\mathbf{x}_{N\tau}^{t_{\text{diff}}}, \mathbf{x}_0, N_{\text{decay}}, t_{\text{diff}}). \tag{24}$$

We use this version for the experiments conducted on Chignolin and set $N_{\text{decay}} = 3t_2$, where t_2 is the unitless characteristic time of the slowest process in the system, $t_2 = \frac{-1}{\log \lambda_2}$.

A.8 How BoPITO interpolators mitigate out-of-distribution effects

BoPITO interpolators mitigate out-of-distribution effects by, first, fixing $N = N_{max}$ as the argument of s_{dyn}. This choice ensures that none of our neural networks are evaluated outside of the training domain. Second, we alter long-lag interpolation steps with short-lag non-interpolation steps for local relaxation. Since we in practice see that the long-lag steps occasionally generates 'off data manifold' states, e.g. structures with potential high energies, the local 'relaxation' steps projects us back onto the data manifold. In practice, just one short-lag non-interpolation step is sufficient to overcome this. Third, we use dynamic observables to fit the interpolation parameter. Without any additional source of information we don't know how to choose N_{int}. However, when we use dynamic observables, we can select the interpolated ensemble which is the most consistent with experimental observables. So this helps calibrate the time-scales to stay "in distribution".

A.9 Experimental parameters

A.9.1 Boltzmann Priors for dataset generation

For different numbers of trajectories, n, we train $5000/n$ ITO models on n Prinz potential trajectories of length 150. Trajectories do not overlap among different trainings. The maximum model lag is 100 steps.

A.9.2 BoPITO samples efficiently long-term dynamics in data-sparse scenarios

Prinz potential 10 models are trained on non-overlapping trajectory sets for different numbers of trajectories. The trajectories length is $10,000$ and the maximum model lag is $1,000$. We consider lags of 10, 25, and 50 as short, 100, 200, and 300 as medium, and 500, 750, and $1,000$ as long time-scales in our experiments, see Figure 10 (b) for a visual reference. $\hat{\lambda}$ is set to 0.994 for all BoPITO experiments.

Alanine Dipeptide 10 models are trained on potentially overlapping random trajectory sets for different numbers of trajectories. The trajectories length is $12,500$ and the maximum model lag is $10,000$. We consider lags of 5, 10, and 50 as short, 100, 500, and 1000, as medium, and 2500, 7500, and 10000 as long time-scales in our experiments, see Figure 11 (b). $\hat{\lambda}$ is set 0.9996 for all BoPITO models.

Chignolin 4 models are trained on potentially overlapping random trajectory sets for different numbers of trajectories. The trajectories length is $35,000$ and the maximum model lag is $30,000$. We consider lags of 10, 50, and 100 as short, 500, 1000, and 5000, as medium, and 10000, 20000, and 30000 as long time-scales in our experiments, see Figure 13 (b). $\hat{\lambda}$ is set 0.9995 for all BoPITO models. BoPITO models are trained using the score in eq. 23 and the decay function in eq. 24.

Diffusion steps	500
Noise schedule	Sigmoidal
Batch size	$2,097,152$
Learning rate	0.001
Layers	3
Embedding dimension	256
Net dimension	256
Optimizer	Adam
Inference ODE steps	50

Diffusion steps	$1,000$
Noise schedule	Polynomial
Batch size	$1,024/32$
Learning rate	0.001
Score layers	5
Embedding layers	2
n_features	64
Optimizer	Adam
Inference ODE steps	100 /50

Table 1: Architectural and training parameters of MB-ITO models (a) and SE3-ITO (b). Batch sizes and inference ODE steps refer to Alanine Dipeptide and Chignolin experiments respectively.

A.9.3 Interpolating between models trained on off equilibrium data and the Boltzmann distribution

We remove the transitions resembling the slowest process in the system, ϕ crossing 0 or 2, to generate biased simulation data. When a transition occurs, we remove one frame before and after the transition and split the trajectory. We train both biased ITO and BoPITO models with a maximum model lag of 100 on the resulting biased dataset. For lags > 100, we sample the BoPITO interpolator fitted on the unbiased dynamic observable defined by eq. 26, see Figure 4. We perform interpolation by sampling with the score model in eq. 15 followed by one round of non-interpolation sampling with lag 100 steps for local relaxation.

A.9.4 Architectural, training and inference details

We use MB-ITO as the architecture of models for the Prinz potential, and SE3-ITO, an $SE(3)-$equivariant neural network, for Alanine Dipeptide, both introduced in Schreiner et al. (2023). We report architectural and training hyper-parameters in Table 1. Models are trained until convergence in the log-log loss plot.

A.10 Prinz potential

The Prinz potential is a 1D potential commonly used for benchmarking MD sampling methods. The potential is defined as

$$U(x) = 4\left(x^8 + 0.8e^{-80x^2} + 0.2e^{-80(x-0.5)^2} + 0.5e^{-40(x+0.5)^2}\right). \tag{25}$$

We generate trajectories using an Euler-Maruyama integrator using the library Deeptime (Hoffmann et al., 2021). We set the integrator time-step to $1 \cdot 10^{-5}$, and the temperature, mass, and damping factor to 1. In Figure 9 we show histograms of the position of a particle after N steps, starting from 0.75, and in Figure 10 (a) we report an implied time-scales plot. We use the identity function to define a dynamic observable/correlation function of this system and show it in Figure 10 (b).

A.11 Alanine Dipeptide

Alanine Dipeptide is a small peptide with 22 atoms. We use publicly available data from Dibak et al. (2022), containing a total of 1 μs simulation time split in 20 trajectories. Simulation is performed in an implicit solvent with 2 fs integration time-step and data is saved every 1 ps. We choose,

$$a(x) = b(x) = \begin{bmatrix} \sin\phi \\ \cos\phi \end{bmatrix}, \tag{26}$$

to define a dynamic observable of this system. The torsion angle ϕ is involved in the slowest transition observed in the simulation, see Figure 12. We combine these vectors taking the inner product for computing dynamic observables. We show this correlation function in Figure 11 (a), and implied time-scales plot in Figure 11 (b) and a histogram of the torsions ϕ and ψ aggregating all simulation data in Figure 12.

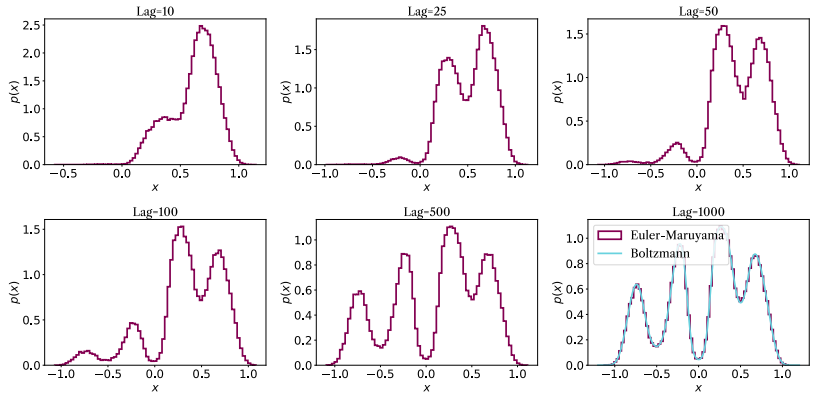
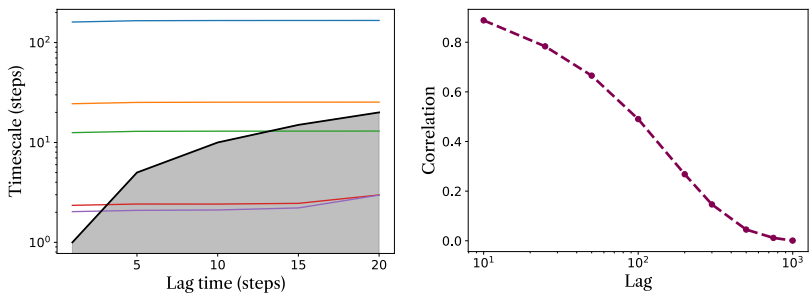
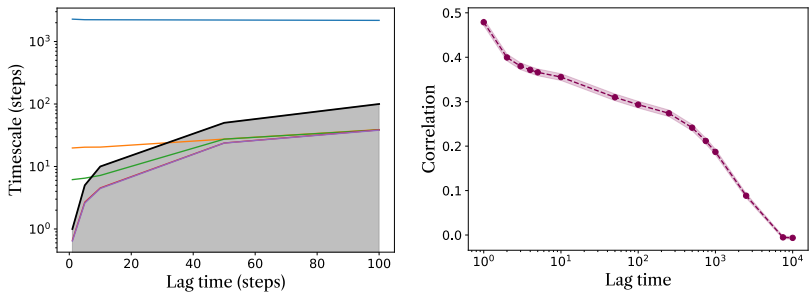


Figure 9: Conditional density of a simulation of the Prinz potential starting at $x = 0.75$ for different lags (steps). Long-term dynamics approaches the Boltzmann distribution.



(a) (b)

Figure 10: Implied time-scales (a) and dynamic observable under the identity function (b) for the Prinz Potential. Errors in (b) are smaller than the dots. In (a), the implied time-scales of the 5 slowest processes in the system are computed for different lags. The color order, from longest to shortest implied time-scale, is blue, orange, green, red, and purple.



(a) (b)

Figure 11: Implied time-scales for Alanine Dipeptide computed with a Markov State Model on torsional angles ϕ and ψ (a) and dynamic observable in eq. 26 (b). In (a), the implied time-scales of the 5 slowest processes in the system are computed for different lags. The color order, from longest to shortest implied time-scale, is blue, orange, green, red, and purple.

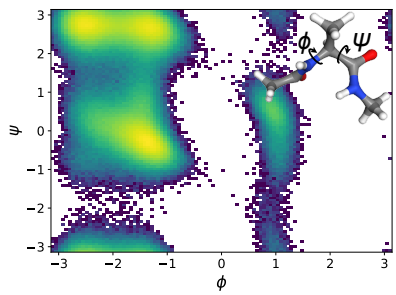
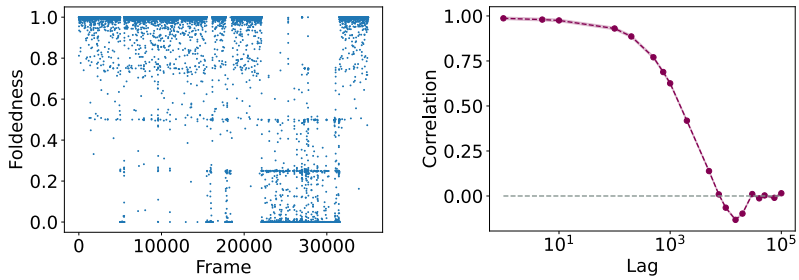


Figure 12: Histogram of the torsional angles ϕ and ψ of Alanine Dipeptide (insert). Data is aggregated among all trajectories in the dataset introduced in Dibak et al. (2022).



(a) (b)

Figure 13: Time evoluation of 'foldedness' in a subset of the reference MD simulation (a) and dynamic observable in eq. 27 (b).

A.12 CHIGNOLIN

Chignolin (cln025) is a fast folding protein with 10 residues, 166 atoms and 93 heavy atoms. We use molecular dynamics data previously reported by Lindorff-Larsen et al. (2011). The data is proprietary but available upon request for research purposes. The simulations were performed in explicit solvent with a 2.5 fs time-step and the positions was saved at 200 ps intervals. We extract all heavy atoms positions from the simulations and train models on this data. We use the reaction coordinate proposed in Lindorff-Larsen et al. (2011) as to define a dynamic observable,

$$a(\mathbf{x}) = b(\mathbf{x}) = \frac{\sum_{i=1}^{N_{\text{res}}} \sum_{j>i}^{N_i} \frac{1}{1+e^{10(d_{ij}(\mathbf{x})-d^*_{ij}-1)}}}{\sum_{i=1}^{N_{\text{res}}} N_i}, \tag{27}$$

where the first sum in the numerator iterates over all N_{res} residues in the protein, while the second sum considers the N_i native contacts of residue i separated by at least seven residues in the primary sequence. We define native contacts as residue pairs separated by at least seven residues in the primary sequence and with C_α atoms closer than 10 Å in the native structure. Moreover, $d_{ij}(\mathbf{x})$ and d^*_{ij} represent the distances between the C_α atoms of residues i and j in the structure $\mathbf{x}$ and the native structure, respectively. This observable quantifies the protein's foldedness, with values ranging from 0 (unfolded) to 1 (folded). In Figure 13 (a) we show the evolution of eq. 27 on a subset of the reference simulation data and we observe how the protein undergoes transformations between the folded and unfolded states. In Figure 13 (b) we visualize the corresponding dynamic observable.

A.13 METRICS

We evaluate models computing differences in the dynamic observables introduced in section 5.1 w.r.t. long MD simulations,

$$|\Delta\text{correlation}|_N = |O^*_N - O^{\text{model}}_N|, \tag{28}$$

where O^*_N is the normalized observable predicted by MD for lag N and O^{model}_N is the model's prediction. See Appendix A.15 for details on normalization. We report the average difference over different training runs.

A.14 ENERGIES FOR ALANINE DIPEPTIDE

In Figure 14 we observe a remarkable agreement in energy densities of samples generated with our Boltzmann Generator and MD. GBSAOBForce is the implicit solvent model energy.

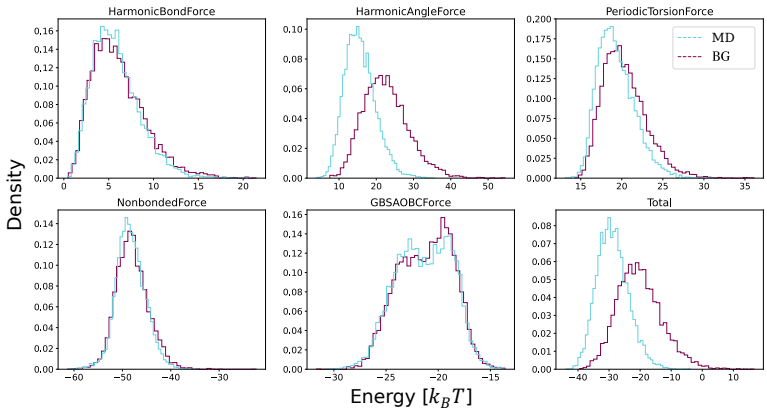


Figure 14: Energy and energy components of samples generated with a Boltzmann Generator (BG) and MD.

A.15 Correlation function normalization

We subtract the mean and divide by correlation at lag 0 to normalize our dynamic observables. Subtracting the mean guarantees that the correlation function asymptotically decays to 0, and we divide by the correlation at lag 0 so that correlation can be read as a fraction of correlation at time 0. The resulting normalized correlation function is

$$\frac{E[(f(\mathbf{x}_t) - E[f(\mathbf{x}_t)])\,(g(\mathbf{x}_{t+\Delta t}) - E[g(\mathbf{x}_{t+\Delta t})])]}{E[(f(\mathbf{x}_t) - E[f(\mathbf{x}_t)])\,(g(\mathbf{x}_t) - E[g(\mathbf{x}_t)])]}. \tag{29}$$

When comparing different methods, we compute $E[f(\mathbf{x}_t)]$, $E[g(\mathbf{x}_{t+\Delta t})]$ and $E[(f(\mathbf{x}_t) - E[f(\mathbf{x}_t)])\,(g(\mathbf{x}_t) - E[g(\mathbf{x}_t)])]$ using long MD simulations and use these same normalizing factors for all methods.

Paper IV

De Novo Drug Design Using Reinforcement Learning with Graph-Based Deep Generative Models

S. Romeo Atance, **J. Viguera Diez**, O. Engkvist, S. Olsson, R. Mercado

J. Chem. Inf. Model. 2022, 62, 20, 4863–4872

https://doi.org/10.1021/acs.jcim.2c00838

Abstract

Machine learning (ML) provides effective computational tools for exploring the chemical space via deep generative models. Here, we propose a new reinforcement learning scheme to fine-tune graph-based deep generative models for *de novo* molecular design tasks. We show how our computational framework can successfully guide a pre-trained generative model towards the generation of molecules with a specific property profile, even when such molecules are not present in the training set and unlikely to be generated by the pre-trained model. We explored the following tasks: generating molecules of decreasing/increasing size, increasing drug-likeness, and increasing bioactivity. Using the proposed approach, we achieve a model which generates diverse compounds with predicted DRD2 activity for 95% of sampled molecules, outperforming previously-reported methods on this metric.

De Novo Drug Design Using Reinforcement Learning with Graph-Based Deep Generative Models

Sara Romeo Atance,[*,†,‡,¶] Juan Viguera Diez,[†,‡,¶] Ola Engkvist,[†,‡] Simon Olsson,[‡] and Rocío Mercado[*,†]

†*Molecular AI, Discovery Sciences, R&D, AstraZeneca Gothenburg*

Pepparedsleden 1, 431 50 Mölndal, Sweden

‡*Department of Computer Science and Engineering, Chalmers University of Technology*

Rännvägen 6 412 58, Göteborg, Sweden

¶*Contributed equally to this work*

E-mail: sararromeo@gmail.com; rociomer@mit.edu

Abstract

Machine learning (ML) provides effective computational tools for exploring the chemical space via deep generative models. Here, we propose a new reinforcement learning scheme to fine-tune graph-based deep generative models for *de novo* molecular design tasks. We show how our computational framework can successfully guide a pre-trained generative model towards the generation of molecules with a specific property profile, even when such molecules are not present in the training set and unlikely to be generated by the pre-trained model. We explored the following tasks: generating molecules of decreasing/increasing size, increasing drug-likeness, and increasing bioactivity. Using the proposed approach, we achieve a model which generates diverse compounds with predicted DRD2 activity for 95% of sampled molecules, outperforming previously-reported methods on this metric.

Introduction

Deep generative models (DGM) are a class of machine learning models which seek to approximate the underlying distribution of data. DGMs importantly allow for sampling — or *generation* — of new, synthetic data from a learned distribution. Consequently, DGMs can aid in creative scientific processes, such as the design of new medicines, by helping researchers design molecules which optimise over a desired property profile, as has been experimentally demonstrated in several studies.[1–4]

DGMs have already seen success across several domains and support diverse tasks such as text,[5] music[6] and image[7] synthesis. Chemists are also increasingly adopting DGMs for applications to complex design tasks such as drug discovery[3] and materials design,[8] leading to the emergence of the sub-field of *generative chemistry*.[9] Previous work has already explored a broad array of neural network 'architectures,' including recurrent neural networks (RNN),[10–13] variational autoencoders (VAE),[14–18] and generative adversarial networks (GAN),[19–21] which have all successfully been used in DGMs for *de novo* molecular design.

Reinforcement learning (RL) is a machine learning technique where a computational agent interacts with an environment and obtains a reward by measuring the quality of its actions. Broadly, RL explores algorithms which train agents to encourage some desired behaviour, e.g., playing a game or designing molecules with specific properties. Indeed, RL has been used to fine-tune DGMs towards generation of compounds with targeted properties using a variety of molecular representations. Previous RL efforts have focused on fine-tuning models that represent molecules as strings.[22–27] However, RL has also been applied to fingerprint-[28] and graph-based[29] models. The latter, compared to string-based methods, allow direct learning from the graph structure, better handling of complex molecular structures (e.g., aromatic heterocycles), and more direct integration of 3D information.[30] Previous work applying RL to molecular DGMs which explicitly treat molecules as graphs is limited, and consists of (1) a graph convolutional network (GCN)-based model for targeted molecular graph generation using policy gradient methods,[29] and (2) a Deep-Q Network which

generates molecular graphs from scratch and without model pre-training.[28]

Two closely-related molecular DGMs have inspired this work:

- REINVENT,[27] a string-based DGM, which uses RNNs to generate targeted molecular strings via policy gradient RL, and
- GraphINVENT,[31] which uses graph neural networks (GNNs) to generate molecular graphs.

Here, we combine the graph-based DGMs from GraphINVENT with a policy gradient RL framework as in REINVENT. We describe a RL strategy for fine-tuning graph-based DGMs for drug discovery applications. We test the proposed RL framework by fine-tuning a pre-trained GraphINVENT model to favour property profiles relevant in drug design, including increasing pharmacological activity. We model bioactivity using a quantitative structure-activity relationship (QSAR) ML model for dopamine receptor type D2 (DRD2) activity. Optimisation for DRD2 activity is a widely used *de novo* design bioactivity benchmark and allows us to easily compare to previous work.[22,27,32] While RL has been applied to many string-based methods for *de novo* molecular design,[22–26] our results encourage the possibility of future work in RL for graph-based molecular design targeting more complex design objectives, as well as prospective experimental validation studies.

Methods

Our graph-based *de novo* design model consists of three main components:

1. a graph-based molecular DGM,
2. a RL framework with a memory-aware loss,
3. and a scoring model.

Graph-based molecular DGM

Following the GraphINVENT[33] approach we use a model based on a Gated Graph Neural Network (GGNN) to generate molecules by iteratively sampling actions which build upon an input graph. The action space is divided into three possible actions: 'add atom', 'add bond', and 'terminate graph'. Using such a set of actions, $\mathcal{A}$, we can formulate the task of generating a molecular graph, $\mathcal{G}$, as a *Markov decision process*, where an 'agent' generates molecules by sampling a sequence of actions from statistical distributions over these actions, leading to $\mathcal{G}$. These statistical distributions are called the *action probability distributions* (APD) (see Supporting Information). Here we model the APD at any given step in the generation sequence using a DGM. That is, given the current graph state, which includes what atoms and bonds have been placed in previous steps, our DGM returns the probabilities of the next available actions.

More formally, we build molecules using a sequence of n actions $A = \{a_0, a_1, \ldots, a_{n-1}\}$, where $a_i \sim \text{APD}_i$ and $f : \mathcal{G}_i \mapsto \text{APD}_i$. Here, f represents our GGNN-based model, which maps a sub-graph built with i actions to APD_i, shorthand for $\text{APD}(\mathcal{A}|\mathcal{G}_i)$. Starting from an empty graph $\mathcal{G}_0$, and ending with the final graph $\mathcal{G}_n$, the graph generation process proceeds as follows: $\mathcal{G}_0 \rightarrow a_0 \sim \text{APD}_0 \rightarrow \mathcal{G}_1 \rightarrow \cdots \rightarrow a_{n-1} \sim \text{APD}_{n-1} \rightarrow \mathcal{G}_n$.

Molecules are broken into a sequence of actions using a breadth-first algorithm to generate the target APD. The model is trained by imitating the inverse process. Particularly, the average Kullback-Leibler divergence (D_{KL}) between target (APD_t) and predicted (APD_p) APDs (Eq. 1) is minimised during training.

$$\text{D}_{\text{KL}}(\text{APD}_t||\text{APD}_p) = \sum_{a \in \mathcal{A}} \text{APD}_t(a) \log \frac{\text{APD}_t(a)}{\text{APD}_p(a)} \tag{1}$$

The set of chosen hyperparameters is the result of an exhaustive search and is detailed in the Supporting Information. The best model was selected at the epoch which minimised the validation loss and used as the 'prior' in the RL framework.

Memory-aware RL framework

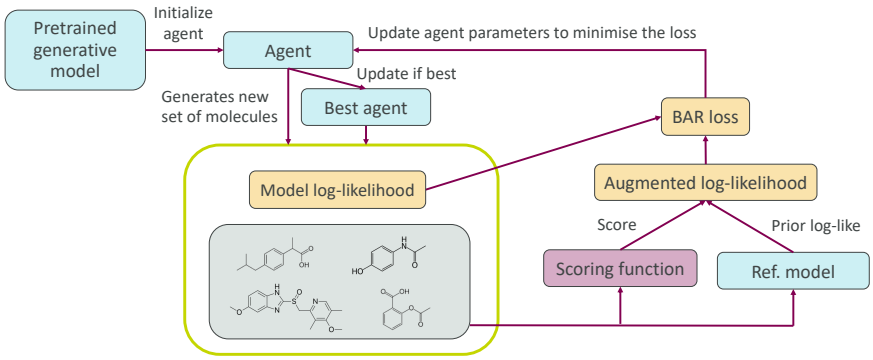


Figure 1: RL loop. The BAR loss is defined in Eq. 2. *Augmented log-likelihood* refers to the second term in Eq. 3 and consists of the reference likelihood and the weighted score.

We build on the previously reported REINVENT algorithm for fine-tuning.[22] The goal is to update the agent policy π from the prior policy π_{Prior} to increase the expected score for the action sequences used to build more desirable graphs. Here, the policy is parameterised using our graph-based model that predicts an APD given an input graph.

The loss we propose here uses a reward shaping mechanism.[34] Compared to REINVENT, we introduce a loss term which keeps track of the best agent so far and is updated every few learning steps. By doing so, we remind the current agent of sets of actions that can lead to high-scoring compounds, in turn accelerating agent learning. The best agent reminder (BAR) loss takes the form

$$J(\boldsymbol{\theta}) = (1-\alpha)J_{\mathbb{A}}(\mathbb{A},\mathbb{P};\boldsymbol{\theta}) + \alpha J_{\tilde{\mathbb{A}}}(\mathbb{A},\tilde{\mathbb{A}};\boldsymbol{\theta}). \tag{2}$$

Above, $\alpha \in [0,1]$ is a scaling factor that we treat as a hyperparameter. $\mathbb{P}$ is the prior model. The current agent and best agent are respectively $\mathbb{A}$ and $\tilde{\mathbb{A}}$. Then,

$$J_{\mathbb{S}}(\mathbb{B},\mathbb{R};\boldsymbol{\theta}) = \mathbb{E}_{p_{\mathbb{S}}}\left[\left[\log P(A)_{\mathbb{B}} - (\log P(A)_{\mathbb{R}} + \sigma S(A))\right]^2\right]. \tag{3}$$

Above, $\mathbb{S}$ represents the agent used for sampling and $\mathbb{R}$ represents the reference model. σ is a scaling factor for the score that we treat as a hyperparameter. $\mathbb{E}_{p_{\mathbb{S}}}$ denotes the average with respect to molecules generated by agent $\mathbb{S}$. A refers to the sequence of actions taken to build a molecule by agent $\mathbb{S}$. $\mathrm{APD}_{\mathbb{B}}(a_i|\mathcal{G}_i)$ is the probability of sampling action a_i given the input graph $\mathcal{G}_i$ and model $\mathbb{B}$, $P(A)_{\mathbb{B}} = \prod_{i=0}^{n-1} \mathrm{APD}_{\mathbb{B}}(a_i|\mathcal{G}_i)$ is the probability of taking the sequence of actions A computed with model $\mathbb{B}$, and $P(A)_{\mathbb{R}}$ is the analogous probability given by the reference model $\mathbb{R}$ for the same sequence of actions. $S(A)$ is the score for the molecule generated following actions A. Eq. 3 corresponds to a mean square error loss of the log-likelihood of model $\mathbb{B}$ and the augmented log-likelihood, $\log P(A)_{\mathbb{R}} + \sigma S(A)$. Intuitively, the score modulates the log-probabilities given by the reference model and ensures that those of poorly scoring molecules are lowered relative to those of highly scoring molecules.

The learning process (Fig. 1) consists of the following steps:

1. Initialise the current and best agents to the prior model. For the prior, we use the pre-trained DGM.
2. Generate a batch of molecules with the current and the best agents, saving the actions.
3. Score all generated molecules.
4. Compute the following:
    i. the probabilities which the prior model $\mathbb{P}$ and current agent $\mathbb{A}$ assign to A, the set of actions taken by the current agent,
    ii. and the probabilities which the current agent $\mathbb{A}$ and best agent $\tilde{\mathbb{A}}$ assign to $\tilde{A}$, the set of actions taken by the best agent.
5. Compute the BAR loss (Eq. 2).
6. Update the current agent parameters so as to minimise the loss.

7. Repeat from step 2, potentially updating the best agent every 5 learning steps.[a]

Scoring model

The scoring model should be designed for each specific optimisation task. Here, we implemented four different scoring functions. The goals of the scoring functions were to:

1. change the average size of molecules by either increasing or decreasing the number of heavy atoms,
2. promote 'drug-likeness' in molecules,
3. and promote DRD2 activity.

The first two scoring functions were used to test the operation of the RL framework. The final scoring function was designed to be representative of a drug discovery optimisation task.

During scoring, molecules which are invalid, improperly terminated and/or duplicates are assigned a score of 0 so as to discourage their sampling in future iterations.

Reducing and increasing the average size of the molecules

On average, molecules sampled from the prior contain 26 heavy atoms. As such, we began exploring the RL framework with the simple task of shifting the distribution of the number of nodes in the sampled molecules towards smaller and larger molecules.

We accomplished these two tasks by defining a scoring function that creates a maximum reward for molecules with 10 and 40 heavy atoms, respectively. More specifically:

$$S_{\text{size}}(A) = \begin{cases} 0 & \text{if not } \{\text{PT, valid } \textit{and} \text{ unique}\}, \\ 1 - \frac{\left|n_{nodes} - n^{\star}_{nodes}\right|}{\max_{nodes} - n^{\star}_{nodes}} & \text{otherwise,} \end{cases} \tag{4}$$

[a]The best agent is updated if the average score of 1K generated molecules is the largest observed, with 1K chosen as a trade-off between speed and sufficient sampling.

where $n^{\star}_{nodes}$ is the target number of heavy atoms in sampled molecules and was set to 10 or 40 for the tasks of reducing and increasing molecular size, respectively. Here, A is the set of actions taken to build the molecule, PT stands for *properly terminated*, n_{nodes} is the number of heavy atoms in the molecule, and $\max_{nodes}$ is the maximum number of nodes allowed in the model (72 here).

Promoting drug-like molecules

The next scoring function is based on the quantitative estimate of drug-likeness (QED)[35] implementation from RDKit:[36]

$$S_{\mathrm{QED}}(A) = \begin{cases} 0 & \text{if not } \{\text{PT, valid } \textit{and} \text{ unique}\}, \\ \mathrm{QED}(\mathrm{Mol}(A)) & \text{otherwise.} \end{cases} \tag{5}$$

Here, $\mathrm{Mol}(A)$ refers to the molecule generated via actions A. QED values can range between 0 and 1, with higher values indicating a molecule is more drug-like and thus more desirable. It should be noted that QED does not necessarily correlate with pharmacological activity.

Promoting DRD2 active molecules

Finally, we investigated a scoring model to fine-tune our DGM towards the generation of drug-like, DRD2-active molecules. Here, we used a QSAR model[37] to predict DRD2 activity in sampled compounds. It consists of a probabilistic Support Vector Machine Classification model trained to discriminate active compounds from inactive ones based on their 2048-bit radius 2 Morgan fingerprint representations and the ExCAPE-DB dataset.[38] This dataset has been proven to share similar physico-chemical property distributions to ChEMBL, and this QSAR model has been previously used to predict the DRD2 activity of compounds in this dataset.[39]

Our scoring model for this task takes the form:

$$S_{\text{activity}}(A) = \begin{cases} 1 \text{ if PT, valid, unique, QED} > 0.5 \textit{ and } \text{activity} > 0.5, \\ 0 \text{ otherwise.} \end{cases} \tag{6}$$

Like QED, predicted activity ranges from 0 to 1, with 1 indicating that a molecule is likely active. However, as the QED and QSAR models are not perfect indicators of pharmacological activity,[35] we used a threshold of 0.5 to classify molecules as either 'active' (QED *and* activity > 0.5) or 'inactive' (QED *or* activity < 0.5). We found the chosen threshold values to work well for ensuring stable training and the subsequent generation of high-scoring molecules.

We compare the molecules generated after RL using this scoring function with a dataset of predicted DRD2 active molecules, which consists of 3627 predicted active molecules according to Eq. 6. Comparison to this set allows us to evaluate if the model can learn to generate true, known DRD2 actives despite that known actives were removed from the original training set.

Dataset details

The dataset used to train the prior was downloaded from[40] and is a subset of ChEMBL[41] with known DRD2 active molecules removed. Molecules in the remaining set are made up of {H, C, N, O, F, S, Cl, Br} and < 50 heavy atoms.[39] $5 \cdot 10^5$ molecules were randomly selected from ChEMBL to create the training set, with $5 \cdot 10^4$ for validation and $5 \cdot 10^4$ for testing. The DRD2 'predicted actives' dataset was downloaded from `github.com/pcko1/Deep-Drug-Coder/`.[40]

Results

Using the BAR loss function

When analysing the behaviour of the reinforcement learning framework using different values of α in the loss function (Eq. 2) with the activity scoring function (Eq. 6), we observe that a value of $\alpha = 0.5$ helps to significantly improve learning (Fig. 2). As the score is discrete, the model learns only when molecules satisfying all the desired criteria are sampled, and the model does not generate many active molecules initially (see $\alpha = 0.0$ in Fig. 2). Therefore, it is especially helpful in this setting to have introduced a memory mechanism to the loss via the α-modulated term which depends on the best recent agent. Without this term, the agent is more likely to forget combinations of actions which result in high activities/scores. We found that using $\alpha = 0.5$ not only accelerated and stabilised learning, but also led to a greater fraction of predicted actives sampled.

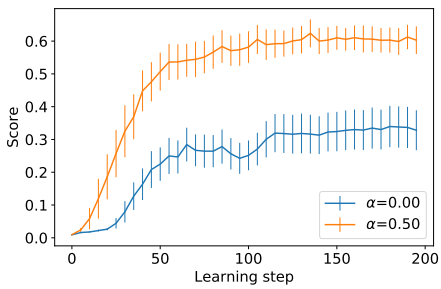


Figure 2: Comparison of the average score of the generated molecules as a function of learning step during bioactivity optimisation. The results in blue are analogous to using the loss proposed in REINVENT,[27] which is recovered when $\alpha = 0.0$. The results in orange correspond to keeping contributions from the best recent agent in the loss with $\alpha = 0.5$ (Eq. 2).

Tuning desired properties via the scoring function

We show here some results for the scoring functions defined above. To prove the ability of the RL framework to fine-tune the DGM towards the generation of molecules with de-

sired properties, we used the scoring functions previously defined in Eqs. 4, 5, and 6. For hyperparameters, see the Supporting Information.

Figure 3: Learning curves for the four different scoring functions investigated. Left: Evolution of the average score of the generated molecules during learning. Centre: Evolution of the fraction of generated molecules which are both valid *and* properly terminated (PT) during learning. Right: Evolution of the fraction of unique molecules generated during learning. The values are computed in all cases for 1000 molecules, taking averages over 10 runs. The error bars correspond to the standard deviation. The hyperparameter values used are $\alpha = 0.5$ for all four scoring functions, $\sigma = 10$ for {Reduce, Increase, and QED}, and $\sigma = 20$ for Activity.

In Fig. 3 we show the evolution of the average score, the fraction of valid and properly terminated molecules (those which do not violate any chemical rules and for which the last sampled action was 'terminate'), and the fraction of unique molecules sampled during learning. Several observations can be made:

- Our model improves the average score of sampled molecules using all four scoring functions. We highlight that the model was able to learn how to generate well-scoring molecules even when we searched for active DRD2 molecules, of which no known true positive examples were given during training.

- The percentage of valid and properly terminated molecules improves during learning as we penalise invalid and improperly terminated molecules.

- The fraction of unique molecules decreases during learning when reducing molecular size or promoting drug-like and active compounds. This behaviour is undesirable but unsurprising, as we are updating towards a smaller chemical space.

- The results are robust as most metrics exhibit very little noise.

We illustrate examples of molecules from the training set, samples from the pre-trained model, and samples from the fine-tuned models in Fig. 4. We find that most generated molecules look reasonable although some of them may be less stable due to the large macrocycles present in them. Less stable molecules are sampled more often from the models which aim to either increase the size or drug-likeness of molecules.

Our approach is successful because it can generate molecules which improve on all four scores tested here, compared to the pre-trained GraphINVENT model. One example is the substantial change in the average size of the molecules sampled when reducing and increasing the number of atoms in the molecules (Eq. 4). Similarly, the approach can also yield models that generate molecules with higher QED scores (Eq. 5) and improved activity scores (Eq. 6). The latter is particular remarkable, as 95% of sampled molecules are predicted to be active by the QSAR model. Nonetheless, some of these predicted actives are not predicted to be synthesizable, as highlighted in Fig. 4.

We break down the performance of the DGM optimising for DRD2 activity in Table 1. We compared the fine-tuned models to the prior as follows:

1. First, we sampled 10K molecules from the prior model.
2. Then, we sampled 10K molecules from a single fine-tuned model.
3. Finally, we sampled and collected 1K molecules from 10 different fine-tuned models (same set of hyperparameters, but different training/sampling runs).

For each set of sampled molecules, we computed their average QED, average DRD2 activity, and how many are predicted actives. We also computed the number of known true actives generated by each model. We observe that both sets of fine-tuned molecules show similar values for the first metrics, and are substantially improved compared to the pre-trained model. Most importantly, while the prior model is not able to generate any

known true DRD2 actives, both fine-tuned models are indeed able to sample known actives. Notably, when the 10K molecules come from 10 different fine-tuned models, the number of known actives sampled is 10-fold higher than when 10K molecules are sampled from a single model. This follows from the previous reasoning about the RL-trained models being heavily-dependent on the initial learning steps; as such, there is little overlap in sets of molecules generated during different RL runs.

Table 1: Comparison of various evaluation metrics for three sets of 10K generated molecules: one in which all are sampled from the pre-trained DGM without fine-tuning (*Prior*), another in which all are sampled from a single fine-tuned model (*Single*), and another in which 1K molecules are sampled from 10 separate fine-tuned models and combined (*Comb.*). *Active* refers to the percentage of molecules which have predicted QED and activity scores > 0.5. *Known true actives* refers to the small percentage of molecules from the DRD2 dataset which have been re-generated by each model.

Evaluation metric	**Prior**	**Single**	**Comb.**
Average QED	0.59	0.72	0.76
Average DRD2 activity	0.03	0.92	0.94
Active (%)	1	94	97
Unique (%)	100	48	60
Active and unique (%)	1	45	58
Known true actives (%)	0	0.08	0.83

Synthesizability of generated molecules

To investigate the synthesizability of molecules generated by our model, we used AiZynthFinder,[42] an open-software platform for retrosynthetic planning based on a Monte Carlo tree search guided by a policy neural network. The tree search recursively breaks down the molecule to purchasable precursors in the ZINC dataset[43] using reactions from the USPTO dataset.[44] The result of the search can be either positive, if a synthetic route is found, or negative otherwise. Our analysis was performed using defaults settings. However, small variations in our results could be observed depending on the available building blocks, as has been previously reported.[42]

In Table 2 we compare the fraction of synthesizable molecules in the training set with those generated by the pre-trained model and the four fine-tuned models. Notably, the synthesizability of molecules sampled from the pre-trained model is lower than measured for the reference data (ChEMBL). This might be an indication that the graph-based models implemented in GraphINVENT don't optimally learn this property (synthesizability). However, we observe how the synthesizability of the generated molecules improves for all the fine-tuned models, except in the scenario where the model aims to increase molecular size. There are a few possible explanations for this. On the one hand, decreasing the size can improve synthesizablity since fewer purchasable precursors are needed to build the molecule. On the other hand, both drug-likeness and DRD2 activity are connected to greater synthetic accessibility since molecules with these attributes tend to share common chemical moieties. As such, increasing molecular size can lead to a decrease in the synthesis score due to both the presence of more unique moieties in sampled molecules, and the fact that there were few large molecules in the training set. This can be confirmed by visualising the sampled molecules (Figure 4).

We compare the effect of using different surrogate models for synthesizability in the Supporting Information.

Table 2: Synthesizability of molecules in the training set and in the various learning scenarios. The pre-trained model is a GraphINVENT model trained on the ChEMBL dataset. For the different RL tasks, 1K molecules are sampled from 10 separate fine-tuned models and combined.

Reference dataset/model	**Synthesizable molecules**
Training set (ChEMBL)	51.4%
Pre-trained model	35.8%
Reduce size scenario (RL)	69.5%
Augment size scenario (RL)	33.1%
QED optimisation scenario (RL)	71.3%
Bioactivity optimisation scenario (RL)	81.0%

Diversity of generated molecules

To analyse the diversity of the generated molecules, we first compute the average Tanimoto similarity of the generated molecules for the four different scenarios, as well as for three reference sets, in Table 3. As expected, ChEMBL and molecules generated by the pre-trained model show similar values. In a realistic task, one could expect the similarity to increase after fine-tuning the model to optimise a certain property, as we observe for the activity optimisation and size augmentation scenarios. However, in artificial tasks such as reducing the size of molecules or promoting only QED, is it possible to observe a decline. Particularly, the decrease of the similarity in the size reduction scenario might be due to the relatively big structural change caused by changing a small number of atoms in a small molecule.

Table 3: Average Tanimoto similarity of molecules in different reference sets and learning scenarios. The pre-trained model is a GraphINVENT model trained on the ChEMBL dataset. 'True actives' refers to the set of known true actives from ExCAPE-DB with QED > 0.5. For the different RL tasks, 1K molecules are sampled from 10 separate fine-tuned models and combined. Error is the standard deviation.

Scenario	**Similarity**
Training set (ChEMBL)	0.293 ± 0.06
Pre-trained model	0.278 ± 0.07
Reduce size scenario (RL)	0.095 ± 0.03
Augment size scenario (RL)	0.325 ± 0.06
QED optimisation scenario (RL)	0.208 ± 0.03
Bioactivity optimisation scenario (RL)	0.366 ± 0.09
True actives	0.301 ± 0.04

In evaluating the molecules generated in the bioactivity optimisation scenario, we observe lower, yet comparable, diversity in the set of generated molecules with respect to the set of true actives. Computing the Tanimoto similarity between the generated actives and the true actives, we obtain a value of 0.278 ± 0.04, suggesting that the new predicted actives are on

average quite different from known actives.

We also computed the number of unique Murcko scaffolds[45] per molecule in the two sets, obtaining 0.45 in the set of true actives and 0.39 for the generated set, suggesting the generated set is less structurally diverse than the set of known actives. Moreover, 4.0% of the scaffolds in the true actives were recovered from the generated set, such that 1.7% of scaffolds in the generated set appear in that of true actives. These results suggest low overlap between the set of generated actives and the set of true actives, in agreement with the conclusions from Table 1.

As a case study, in Table 4 we report the similarity of sampled molecules from the bioactivity optimisation task to 1-(o-anisyl)piperazine, which is a substituted arylpiperazine and a known pharmacophore, by computing the number of molecules containing this substructure in the generated and true actives sets. We observe that our model predicts the different ether substitution patterns in the pharmacophore in the right ranking, with 55% of the generated molecules containing the pharmacophore with the ether in the ortho substitution pattern.

Table 4: Percentage of molecules presenting the 1-(*-anisyl)piperazine pharmacophore generated by our model in the activity task and in the set of true actives with QED > 0.5. The * indicates the different possible ether substitution patterns.

Ether substitution pattern	**Generated**	**True actives**
ortho	55.08 %	9.8 %
meta	3.17 %	0.88 %
para	0.19 %	0.55 %

Discussion

Improvements from previous work

In contrast to the graph convolutional policy network (GCPN),[29] the action space used by our underlying model, GraphINVENT, is split into three possible action types, while the

GCPN uses four possible action types, which in both cases are concatenated to make the 'overall' action space. This difference is more of a design choice, and ultimately both models encode the action space similarly. While GCPN uses the GCN implementation, our models use the GGNN,[46] which was recently reported to outperform other GNN implementations in graph-based molecular generation applications.[31] With the exception of QED optimisation, the other design tasks explored in this work are distinct from those explored previously; namely, the generation of potential bioactive molecules was not explored with the GCPN.

Of all the tasks explored, our model shows particular promise for the task of generating DRD2 actives, a popular benchmark for molecular DGMs as it simulates a 'real' drug discovery task. Compared to previous work,[39] our model is able to generate a much greater fraction of predicted active molecules after removal of duplicates: 95% predicted active compounds, compared to only 54% in the best model from the aforementioned work. Although the percentage of known true actives that our models are able to recover is very small ($< 1\%$), we highlight that the DGM has not seen any examples of known true active molecules at any point during training and that there were no predicted actives generated before fine-tuning. This finding suggests that the model could be used to generate actives in a challenging but realistic drug discovery setting where little to no actives are known.

Memory mechanism

The goal of our approach is to enable the exploration of chemical space in a prioritised manner, using it to search for promising new molecules that demonstrate pharmacological activity. Use of the memory mechanism allows our model to train more smoothly, at the cost of introducing some bias to it. However, by keeping track of the best agent rather than the best molecules generated so far (another popular memory mechanism in DGMs[23,26,27]), we believe that the model is less biased, and thus better able to balance exploration and the generation of novel structures without forgetting actions that led to good molecules.

We believe the good performance of our model is due to the memory augmented loss

function. By keeping track of the best agent seen so far during training, we were able to stabilise learning and achieve better performance for all models. We speculate the origin of the improved performance of the BAR loss is similar to that seen in momentum-based optimisers in stochastic gradient descent. We leave a rigorous theoretical analysis of the loss for future work. The trained models are robust, and show little variation between runs in terms of the metrics of interest (Fig. 3), and only the fraction of unique samples varies notably between runs when aiming to generate DRD2 actives. This task is extremely difficult, as it depends strongly on the first active molecules generated by the model, which means the sets of actives generated by a model during different runs generally have negligible overlap.

We can compare our model to previous work, the GCPN,[29] for the task of QED optimisation. Here, the authors report the top 3 QED values obtained from molecules generated by their fine-tuned model: 0.948, 0.947, and 0.946. Similarly, we find the top 3 QED values out of 1000 molecules sampled by our model after QED fine-tuning to be 0.948, 0.947, and 0.947. Furthermore, for 10 different runs of 1000 samples each, all top 3 QEDs are in the range of 0.940-0.948. The models thus show similar performance for this task, where 0.948 is observed to be an upper limit for the task of QED optimisation.

Limitations

On par with that needed for other state-of-the-art molecular DGMs,[47] the main drawback of the proposed model is the amount of time and computational power needed to pre-train the underlying GraphINVENT model (a few days on an NVIDIA Tesla K80); however, this only has to be done once per dataset. After pre-training, fine-tuning the model with RL is comparatively quick and requires only between $10 - 40$ minutes, where scoring the model is the main bottleneck. Furthermore, the same pre-trained model can be fine-tuned for multiple tasks, making our model competitive with other tools.

Some molecules generated by the models when increasing molecular size and QED appear to have a larger fraction of (undesirable) macrocycles and unstable moieties. Additionally,

the percent valid and properly terminated does not increase as much when fine-tuning the model towards larger molecules as for the other scores (Fig. 3). We believe in these cases, the model has not seen many examples on how to predict reasonable APDs, making it difficult for it to learn actions that lead to large, *stable* molecules. We do not observe this trend when reducing the size of the molecules, and we believe it is because the model sees significantly more small sub-graphs during pre-training. QED is an equally challenging property to optimise as it is highly non-linear. These challenges motivated the use of the 0.5 threshold in Eq. 6, which proved to work well. However, exploring better estimates of molecular stability, drug-likeness, and synthetic accessibility in the scoring function are possible ways to minimise the sampling of undesirable molecules, and is a topic of future work.

Furthermore, approaches such as beam search[48] and temperature sampling[49] could be used to boost sampling from a fine-tuned GraphINVENT model and provide automatic prioritisation of the designs. While these techniques have been demonstrated to enhance sampling in chemical language models, they could also be adapted to graph generation under an autoregressive framework like GraphINVENT.

When optimising for bioactivity, we cannot guarantee all explored regions of the chemical space will lie under the applicability domain of the QSAR model. However, we don't consider this to be especially problematic in this setting, given that the model was deliberately trained using an imbalanced dataset in which only 6.7% of the compounds are active. There are two advantages to having a small number of actives in the training set; first, it is closer to a practical situation, and second, a molecule far from the applicability domain will have a small probability of being predicted as active. Finally, in future studies we would like to experimentally validate the properties of the generated molecules.

Conclusions

Here, we have used policy-gradient RL to extend a graph-based *de novo* molecular design tool for the generation of drug-like molecules with desired properties. As part of the model, we proposed the best agent reminder (BAR) loss and show that it significantly improves model training. Using this loss, the proposed RL framework shows a remarkable ability for fine-tuning the pre-trained DGM towards production of molecules with desired sets of properties, even in challenging situations where only a few or no examples of compounds with the desired properties were seen by the initial model. We have shown how our model is able to perform well in several tasks, most notably promoting the generation of DRD2 active molecules. While favouring certain properties, our RL framework also improves other performance metrics, including increasing the percentage of valid and properly terminated molecules, reaching validity rates comparable to that of state-of-the-art models.[47,50,51]

Many properties a molecule exhibits directly depend on its molecular graph. As such, we believe the development of graph-based methods is key for the next generation of *de novo* design tools, as graphs can naturally encode information about connectivity. Our tool is thus an important stepping stone towards the design of more advanced molecular DGMs and tools which will allow scientists to efficiently traverse the chemical space in search of promising molecules. We believe the use of DGMs in fields like drug design has the potential to help chemists come up with new ideas, and to accelerate the complex process of molecular discovery.

Data and Software Availability

Code for this work is available at `https://github.com/olsson-group/RL-GraphINVENT`.

Acknowledgement

We thank Dr. Atanas Patronov and Vendy Fialkova for useful discussions on previous work on REINVENT. We also thank Prof. Morteza Haghir Chehreghani for his useful feedback on the manuscript. This work was partially supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation (to S.O.). R.M. was supported by the AstraZeneca Postdoc Program.

Supporting Information Available

Supporting information is available online and includes a detailed description of abbreviations, hyperparameter settings, and generative model details.

References

(1) Merk, D.; Friedrich, L.; Grisoni, F.; Schneider, G. De Novo Design of Bioactive Small Molecules by Artificial Intelligence. *Mol. Inf.* **2018**, *37*, 1700153.

(2) Grisoni, F.; Neuhaus, C. S.; Hishinuma, M.; Gabernet, G.; Hiss, J. A.; Kotera, M.; Schneider, G. De novo design of anticancer peptides by ensemble artificial neural networks. *J. Mol. Model.* **2019**, *25*, 112.

(3) Stokes, J. M.; Yang, K.; Swanson, K.; Jin, W.; Cubillos-Ruiz, A.; Donghia, N. M.; MacNair, C. R.; French, S.; Carfrae, L. A.; Bloom-Ackermann, Z., et al. A deep learning approach to antibiotic discovery. *Cell* **2020**, *180*, 688–702.

(4) Zhavoronkov, A.; Ivanenkov, Y. A.; Aliper, A.; Veselov, M. S.; Aladinskiy, V. A.; Aladinskaya, A. V.; Terentiev, V. A.; Polykovskiy, D. A.; Kuznetsov, M. D.; Asadulaev, A., et al. Deep learning enables rapid identification of potent DDR1 kinase inhibitors. *Nat. Biotechnol.* **2019**, *37*, 1038–1040.

(5) McKeown, K. *Text generation*; Cambridge University Press, 1992.

(6) Briot, J.-P.; Hadjeres, G.; Pachet, F.-D. Deep Learning Techniques for Music Generation – A Survey. 2017; `https://arxiv.org/abs/1709.01620`.

(7) Gregor, K.; Danihelka, I.; Graves, A.; Rezende, D. J.; Wierstra, D. DRAW: A Recurrent Neural Network For Image Generation. 2015; `https://arxiv.org/abs/1502.04623`.

(8) Yao, Z.; Sánchez-Lengeling, B.; Bobbitt, N. S.; Bucior, B. J.; Kumar, S. G. H.; Collins, S. P.; Burns, T.; Woo, T. K.; Farha, O. K.; Snurr, R. Q., et al. Inverse design of nanoporous crystalline reticular materials with deep generative models. *Nat. Mach. Intell.* **2021**, *3*, 76–86.

(9) Chen, H.; Engkvist, O.; Wang, Y.; Olivecrona, M.; Blaschke, T. The rise of deep learning in drug discovery. *Drug Discovery Today* **2018**, *23*, 1241–1250.

(10) Segler, M. H.; Kogej, T.; Tyrchan, C.; Waller, M. P. Generating focused molecule libraries for drug discovery with recurrent neural networks. *ACS Cent. Sci.* **2018**, *4*, 120–131.

(11) Li, Y.; Zhang, L.; Liu, Z. Multi-objective de novo drug design with conditional graph generative model. *J. Cheminf.* **2018**, *10*, 33.

(12) Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780.

(13) Cho, K.; van Merrienboer, B.; Bahdanau, D.; Bengio, Y. On the Properties of Neural Machine Translation: Encoder-Decoder Approaches. 2014; `https://arxiv.org/abs/1409.1259`.

(14) Gómez-Bombarelli, R.; Wei, J. N.; Duvenaud, D.; Hernández-Lobato, J. M.; Sánchez-Lengeling, B.; Sheberla, D.; Aguilera-Iparraguirre, J.; Hirzel, T. D.; Adams, R. P.;

Aspuru-Guzik, A. Automatic Chemical Design Using a Data-Driven Continuous Representation of Molecules. *ACS Cent. Sci.* **2018**, *4*, 268–276, PMID: 29532027.

(15) Ma, T.; Chen, J.; Xiao, C. Constrained Generation of Semantically Valid Graphs via Regularizing Variational Autoencoders. 2018; https://arxiv.org/abs/1809.02630.

(16) Jin, W.; Barzilay, R.; Jaakkola, T. Hierarchical Generation of Molecular Graphs using Structural Motifs. 2020; https://arxiv.org/abs/2002.03230.

(17) Kingma, D. P.; Welling, M. Auto-Encoding Variational Bayes. 2013; https://arxiv.org/abs/1312.6114.

(18) Rezende, D. J.; Mohamed, S.; Wierstra, D. Stochastic Backpropagation and Approximate Inference in Deep Generative Models. 2014; https://arxiv.org/abs/1401.4082.

(19) Sanchez-Lengeling, B.; Outeiral, C.; Guimaraes, G. L.; Aspuru-Guzik, A. Optimizing distributions over molecular space. An Objective-Reinforced Generative Adversarial Network for Inverse-design Chemistry (ORGANIC). 2017; https://chemrxiv.org/engage/chemrxiv/article-details/60c73d91702a9beea7189bc2.

(20) De Cao, N.; Kipf, T. MolGAN: An implicit generative model for small molecular graphs. 2018; https://arxiv.org/abs/1805.11973.

(21) Goodfellow, I. J.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative Adversarial Networks. 2014; https://arxiv.org/abs/1406.2661.

(22) Olivecrona, M.; Blaschke, T.; Engkvist, O.; Chen, H. Molecular de-novo design through deep reinforcement learning. *J. Cheminf.* **2017**, *9*, 1–14.

(23) Popova, M.; Isayev, O.; Tropsha, A. Deep reinforcement learning for de novo drug design. *Sci. Adv.* **2018**, *4*, eaap7885.

(24) Guimaraes, G. L.; Sanchez-Lengeling, B.; Outeiral, C.; Farias, P. L. C.; Aspuru-Guzik, A. Objective-Reinforced Generative Adversarial Networks (ORGAN) for Sequence Generation Models. 2017; https://arxiv.org/abs/1705.10843.

(25) Neil, D.; Segler, M.; Guasch, L.; Ahmed, M.; Plumbley, D.; Sellwood, M.; Brown, N. Exploring deep recurrent models with reinforcement learning for molecule design. *OpenReview.net* **2018**,

(26) Putin, E.; Asadulaev, A.; Ivanenkov, Y.; Aladinskiy, V.; Sanchez-Lengeling, B.; Aspuru-Guzik, A.; Zhavoronkov, A. Reinforced adversarial neural computer for de novo molecular design. *J. Chem. Inf. Model.* **2018**, *58*, 1194–1204.

(27) Blaschke, T.; Arús-Pous, J.; Chen, H.; Margreitter, C.; Tyrchan, C.; Engkvist, O.; Papadopoulos, K.; Patronov, A. REINVENT 2.0: An AI Tool for De Novo Drug Design. *J. Chem. Inf. Model.* **2020**,

(28) Zhou, Z.; Kearnes, S.; Li, L.; Zare, R. N.; Riley, P. Optimization of molecules via deep reinforcement learning. *Sci. Rep.* **2019**, *9*, 1–10.

(29) You, J.; Liu, B.; Ying, R.; Pande, V.; Leskovec, J. Graph Convolutional Policy Network for Goal-Directed Molecular Graph Generation. 2018; https://arxiv.org/abs/1806.02473.

(30) Jiménez-Luna, J.; Grisoni, F.; Weskamp, N.; Schneider, G. Artificial intelligence in drug discovery: Recent advances and future perspectives. *Expert Opin. Drug Discovery* **2021**, 1–11.

(31) Mercado, R.; Rastemo, T.; Lindelöf, E.; Klambauer, G.; Engkvist, O.; Chen, H.; Bjerrum, E. J. Graph networks for molecular design. *Mach. Learn.: Sci. Technol.* **2021**, *2*, 025023.

(32) Arús-Pous, J.; Patronov, A.; Bjerrum, E. J.; Tyrchan, C.; Reymond, J.-L.; Chen, H.; Engkvist, O. SMILES-based deep generative scaffold decorator for de-novo drug design. *J. Cheminf.* **2020**, *12*, 1–18.

(33) Mercado, R.; Rastemo, T.; Lindelöf, E.; Klambauer, G.; Engkvist, O.; Bjerrum, E. J. GraphINVENT. 2021; `https://github.com/MolecularAI/GraphINVENT/releases/tag/v1.0`, Date accessed 01-02-2021.

(34) Buhmann, M. D. et al. *Encyclopedia of Machine Learning*; Springer US, 2011; pp 863–865.

(35) Bickerton, G. R.; Paolini, G. V.; Besnard, J.; Muresan, S.; Hopkins, A. L. Quantifying the chemical beauty of drugs. *Nat. Chem.* **2012**, *4*, 90–98.

(36) Landrum, G. RDKit: Open-source cheminformatics. `http://www.rdkit.org`, Date accessed 01-02-2021.

(37) Kotsias, P.-C.; Arús-Pous, J.; Chen, H.; Engkvist, O.; Tyrchan, C.; Bjerrum, E. J. DeepDrugCoder (DDC): Heteroencoder for molecular encoding and de novo generation. 2020; `https://github.com/pcko1/Deep-Drug-Coder/blob/master/models/qsar_model.pickle`, Date accessed 01-02-2021.

(38) Sun, J.; Jeliazkova, N.; Chupakhin, V.; Golib-Dzib, J.-F.; Engkvist, O.; Carlsson, L.; Wegner, J.; Ceulemans, H.; Georgiev, I.; Jeliazkov, V.; Kochev, N.; Ashby, T. J.; Chen, H. ExCAPE-DB: an integrated large scale dataset facilitating Big Data analysis in chemogenomics. *J. Cheminf.* **2017**, *9*, 17.

(39) Kotsias, P.-C.; Arús-Pous, J.; Chen, H.; Engkvist, O.; Tyrchan, C.; Bjerrum, E. J. Direct steering of de novo molecular generation with descriptor conditional recurrent neural networks. *Nat. Mach. Intell.* **2020**, *2*, 254–265.

(40) Kotsias, P.-C.; Arús-Pous, J.; Chen, H.; Engkvist, O.; Tyrchan, C.; Bjerrum, E. J. DeepDrugCoder (DDC): Heteroencoder for molecular encoding and de novo generation. 2020; `https://github.com/pcko1/Deep-Drug-Coder/tree/master/datasets`, Date accessed 01-02-2021.

(41) Mendez, D. et al. ChEMBL: Towards direct deposition of bioassay data. *Nucleic Acids Res.* **2018**, *47*, D930–D940.

(42) Genheden, S.; Thakkar, A.; Chadimová, V.; Reymond, J.-L.; Engkvist, O.; Bjerrum, E. AiZynthFinder: a fast, robust and flexible open-source software for retrosynthetic planning. *J. Cheminf.* **2020**, *12*.

(43) Irwin, J. J.; Shoichet, B. K. ZINC–a free database of commercially available compounds for virtual screening. *J. Chem. Inf. Model.* **2005**, *45*, 177–182.

(44) Lowe, D. Chemical reactions from US patents (1976-Sep2016). **2017**,

(45) Bemis, G. W.; Murcko, M. A. The Properties of Known Drugs. 1. Molecular Frameworks. *J. Med. Chem.* **1996**, *39*, 2887–2893, PMID: 8709122.

(46) Li, Y.; Tarlow, D.; Brockschmidt, M.; Zemel, R. Gated Graph Sequence Neural Networks. 2015; `https://arxiv.org/abs/1511.05493`.

(47) Zhang, J.; Mercado, R.; Engkvist, O.; Chen, H. Comparative study of deep generative models on chemical space coverage. *J. Chem. Inf. Model.* **2021**,

(48) Moret, M.; Helmstädter, M.; Grisoni, F.; Schneider, G.; Merk, D. Beam Search Sampling for Molecular Design and Intrinsic Prioritization with Machine Intelligence. 2021; `https://chemrxiv.org/engage/chemrxiv/article-details/60c755bb469df48119f451ae`.

(49) Moret, M.; Friedrich, L.; Grisoni, F.; Merk, D.; Schneider, G. Generative molecular design in low data regimes. *Nat. Mach. Intell.* **2020**, *2*, 171–180.

(50) Brown, N.; Fiscato, M.; Segler, M. H.; Vaucher, A. C. GuacaMol: Benchmarking models for de novo molecular design. *J. Chem. Inf. Model.* **2019**, *59*, 1096–1108.

(51) Polykovskiy, D.; Zhebrak, A.; Sanchez-Lengeling, B.; Golovanov, S.; Tatanov, O.; Belyaev, S.; Kurbanov, R.; Artamonov, A.; Aladinskiy, V.; Veselov, M., et al. Molecular sets (MOSES): A benchmarking platform for molecular generation models. *Front. Pharmacol.* **2020**, *11*.

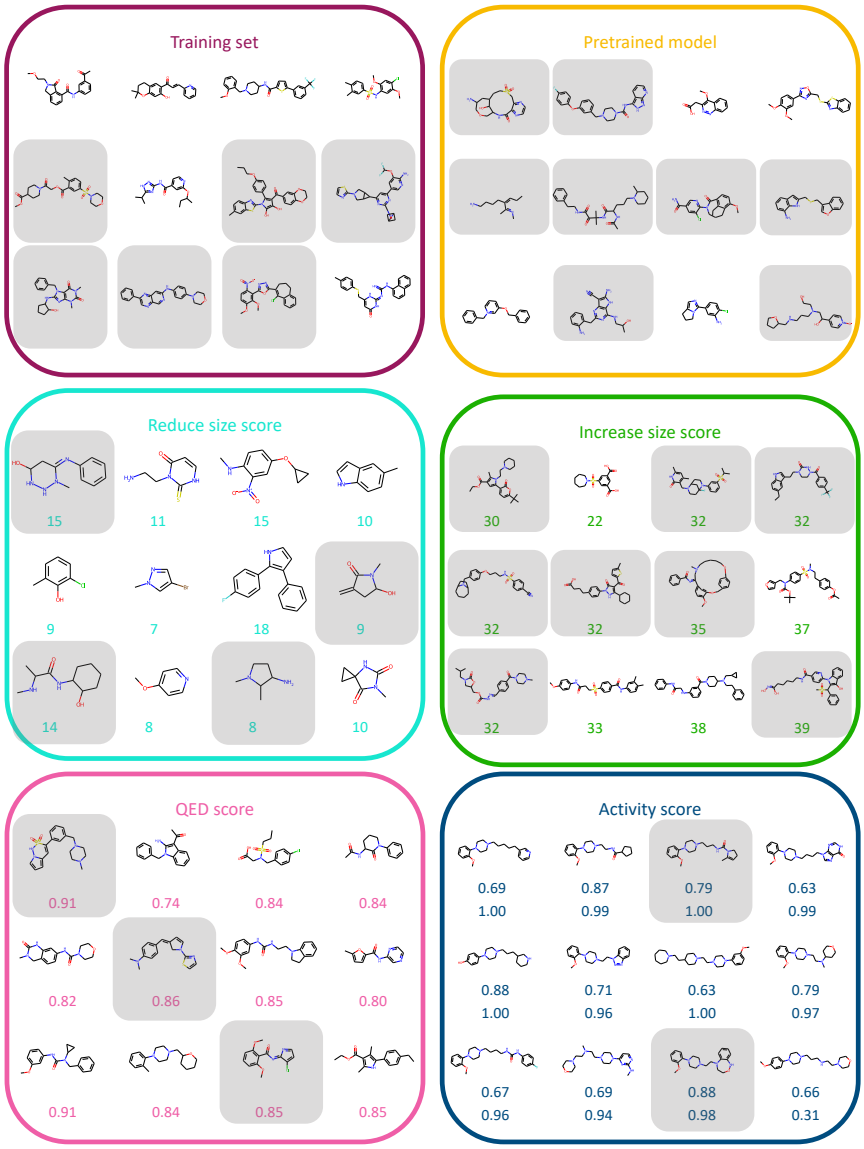
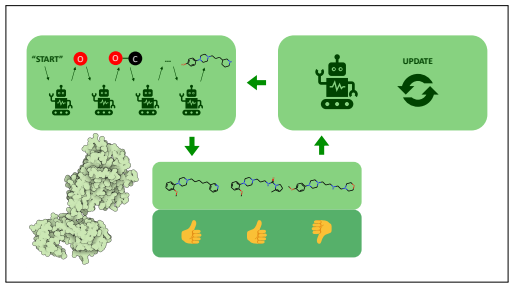


Figure 4: Examples of molecules. Top left: Training set. Top right: Pre-trained model. Centre left: Model fine-tuned to reduce the size of sampled molecules; the value below each molecule corresponds to its number of nodes. Centre right: Model fine-tuned to increase the size of the generated molecules; the value below each molecule corresponds to its number of nodes. Bottom left: Model fine-tuned to promote drug-likeness; the value below each molecule corresponds to its QED. Bottom right: Model fine-tuned to promote the generation of drug-like, DRD2 active molecules; the numbers below each molecule correspond to its QED (top) and activity estimate (bottom). Shaded molecules indicate those predicted to not be synthesizable by AiZynthFinder.

Graphical TOC Entry



Supporting Information for *De Novo* Drug Design Using Reinforcement Learning with Graph-Based Deep Generative Models

Sara Romeo Atance,[*,†,‡,¶] Juan Viguera Diez,[†,‡,¶] Ola Engkvist,[†,‡] Simon Olsson,[‡] and Rocío Mercado[*,†]

†*Molecular AI, Discovery Sciences, R&D, AstraZeneca Gothenburg*

Pepparedsleden 1, 431 50 Mölndal, Sweden

‡*Department of Computer Science and Engineering, Chalmers University of Technology*

Chalmersplatsen 4, 412 96 Gothenburg, Sweden

¶*Contributed equally to this work*

E-mail: sararromeo@gmail.com; rociomer@mit.edu

Abbreviations

- APD : action probability distribution
- BAR : best agent reminder
- DGM : deep generative models
- DRD2 : dopamine receptor D2
- GCN : graph convolutional network
- GCPN : graph convolutional policy network
- GGNN : gated graph neural network
- GNN : graph neural network
- KL : Kullback-Leibler
- ML : machine learning
- PT : properly terminated
- QED : quantitative estimate of drug-likeness
- QSAR : quatitative structure activity relationship
- RL : reinforcement learning
- RNN : recursive neural network
- VAE : variational autoencoder

Hyperparameters

Generative model: GraphINVENT hyperparameters

The GraphINVENT model consists of two main components: the GGNN and the global readout block. The hyperparameter values chosen in both models are those found to work best in the original publication.[1] Taking into account that, in the GGNN, the message size must be equal to the number of hidden node features, the effective parameters to optimise are shown in Table S1.

Table S1: Effective model parameters and their optimal values.

Parameter	Value
Number of hidden node features	100
Graph embedding size	100
Hidden size of MLPs in GGNN	250
Depth of MLPs in GGNN	4
Dropout probability in GGNN	0
Number of message passes	3
Hidden size of MLPs in global readout	500
Depth of MLPs in global readout	4
Dropout probability in global readout	0

The parameter values used in the multi-layer perceptrons (MLPs) of the GGNN are detailed in Table S2. Their weights are initialised from Xavier uniform distributions.[2] The activation functions used are SELUs.[?] The number of message passes in the message passing phase is also shown in Table S2. These parameters relate the message passing and graph readout parameters in the following way: *hidden node features* = *input features*, *message size* = *output features*, and *graph embedding size* = *output features*.

The parameter values chosen in the MLPs of the global readout block are detailed in Table S3. Again, weights are initialised from an Xavier normal distribution and the activation

Table S2: Model hyperparameters used in the MLPs and the message passing phase of the GGNN in GraphINVENT.

Parameter	Value
Input features	100
Hidden features	250
Output features	100
Depth	4
Dropout probability	0
Message passes	3

functions are SELUs. The remaining parameters of the MLPs are chosen as needed to encode all necessary information for the probabilities of adding an atom, connecting two nodes, or terminating a graph. These sizes depend on another hyperparameter, the maximum number of nodes in molecules in a given dataset.

Table S3: Common hyperparameters used in the MLPs of the global readout block in GraphINVENT.

Parameter	Value
Hidden features	500
Depth	4
Dropout probability	0

Other parameters which need to be specified in GraphINVENT are those related to the dataset and its features. We use a 'simple' version of GraphINVENT which ignores aromatic bonds, chirality, and hydrogens (neither explicit nor implicit). Additionally, we use canonical node orderings and allow a maximum number of heavy atoms of 72 (largest molecule in the DRD2 actives set).

Training of GraphINVENT is done using the Adam optimiser[3] with no weight decay and the OneCycleLR[4] learning rate scheduler implemented in PyTorch. In the scheduler, we have used the default parameters but disabled learning rate 'warm-up'. Furthermore, we

take as many steps as epochs and set the fraction of steps for increasing the learning rate to 0.05. Other parameters such as the initial and final learning rates and the batch size must be adjusted for each specific dataset via hyperparameter optimisation. For optimal training, we trained for 30 epochs, used an initial learning rate of 10^{-4}, a final learning rate of 10^{-7}, and a batch size of 1000 sub-graphs.

RL framework hyperparameters

When training the agent, we again used the PyTorch Adam optimiser with no weight decay and the OneCycle learning rate scheduler (same settings as before). We found an initial learning rate of 10^{-4}, together with a final learning rate of 10^{-6}, to work best during RL-based training.

A batch size of 64 *molecules* was used in all RL settings ($64 \times 26 \sim 1164$ *sub-graphs*), except for models in which the scoring function aimed to increase the size of the molecules, where it was necessary to reduce the batch size to 32 molecules (~ 832 sub-graphs) due to memory constraints. However, these batch sizes are comparable to the ones used for training GraphINVENT, where the batch size consists of 1000 sub-graphs.

The hyperparameter values used in the activity scoring function (Eq. 6 in Main text) were $\sigma = 20$ and $\alpha = 0.5$, and were the result of hyperparameter optimisation. $\alpha = 0.25$ and 0.75 were also tried, though $\alpha = 0.5$ was found to work best. For the other scoring functions (Eqs. 4 and 5 in Main text) we used $\sigma = 10$ and $\alpha = 0.5$, although these were not as thoroughly optimised.

GraphINVENT details

Action space

GraphINVENT uses both the node- and graph-level information to predict the action probability distribution, or APD, in the final (global) readout block. The APD specifies how

to grow the input subgraphs, and is made up of three components: f_{add}, f_{conn}, and f_{term}.

f_{add} contains probabilities for *adding* a new node to the graph. f_{conn} contains probabilities for *connecting* the last appended node in the graph to another existing node in the graph. f_{term} is the probability of *terminating* the graph. f_{add} and f_{conn} are multi-dimensional tensors as they must encode for a variety of properties, including which atom to connect to, with which each atom type, the identity of the new atom, etc. However, as the APD is a 'vector' property, f_{add} and f_{conn} are flattened for concatenation with f_{term} before forming the final APD. The shapes/indices of the three (unflattened) APD components are described in detail in the original publication.[1]

As it is a probability distribution, the APD for each graph should sum to 1. Using the learned node and graph embeddings, H^L and g respectively, each APD is computed as follows:

$$
\begin{aligned}
f'_{add} &= \texttt{MLP}^{add,1}\left(H^L\right) \\
f'_{conn} &= \texttt{MLP}^{conn,1}\left(H^L\right) \\
f_{add} &= \texttt{MLP}^{add,2}\left([f'_{add}, g]\right) \\
f_{conn} &= \texttt{MLP}^{conn,2}\left([f'_{conn}, g]\right) \\
f_{term} &= \texttt{MLP}^{term,2}\left(g\right) \\
APD &= \texttt{SOFTMAX}\left([f_{add}, f_{conn}, f_{term}]\right)
\end{aligned}
$$

Note that in practice this is done for a mini-batch of graphs simultaneously on a GPU.

Iterative molecular generation

To demonstrate how the APD is used in GraphINVENT, we show a schematic of the generation loop in Figure S1.

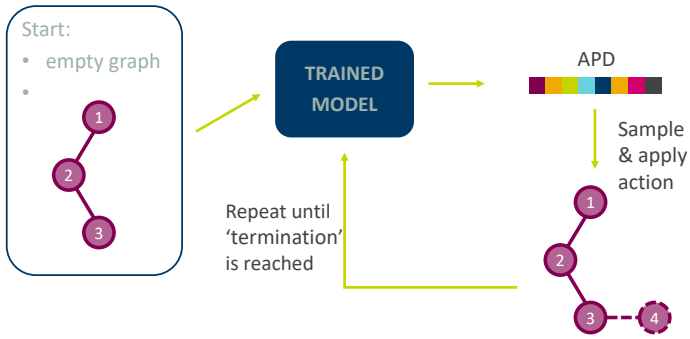


Figure S1: Schematic of the generation loop in GraphINVENT.

Synthesizability analysis with SA and RA scores

We also performed synthesis score analysis using the synthetic accessibility (SA)[5] and retrosynthetic accessibility (RA) score.

SA score is a method for estimating the ease of synthesis of molecular structures based on fragment contribution and a complexity penalty. Fragment contribution is computed based on the analysis of one million representative molecules from PubChem,[6] and complexity takes into account the presence of non-standard structural features, such as large rings, non-standard ring fusions, stereocomplexity, and molecule size. Lower score means better synthesizability, ranging from 1 (easy to make) to 10 (very difficult to make).

RA score is an ML-based surrogate model for AiZynthFinder[7] and is 4500 faster than the former. This score ranges from 0 to 1, with a higher score corresponding to easier-to-make molecules.[8]

We present our results for the SA score in Table S4 and Figure S2, and for the RA score in Table S5 and Figure S3. Independently of the model used to estimate synthesizability, we generally observe the best synthesizability scores for the fine-tuned models promoting QED and activity. This is in agreement with the results obtained using AiZynthFinder, which are presented in the Main text.

However, we also observe some discrepancies between the two methods. While SA scores

Table S4: Average SA scores for molecules in the ChEMBL dataset, sampled from the pre-trained model, and sampled under the different RL scenarios.

	SA score
ChEMBL	2.96
Pre-trained	3.29
Reduce size	2.97
Augment size	3.20
QED	2.47
Activity	2.53

Table S5: Average RA scores for molecules in the ChEMBL dataset, sampled from the pre-trained model, and sampled under the different RL scenarios.

	RA score
ChEMBL	0.758
Pre-trained	0.757
Reduce size	0.921
Augment size	0.783
QED	0.935
Activity	0.952

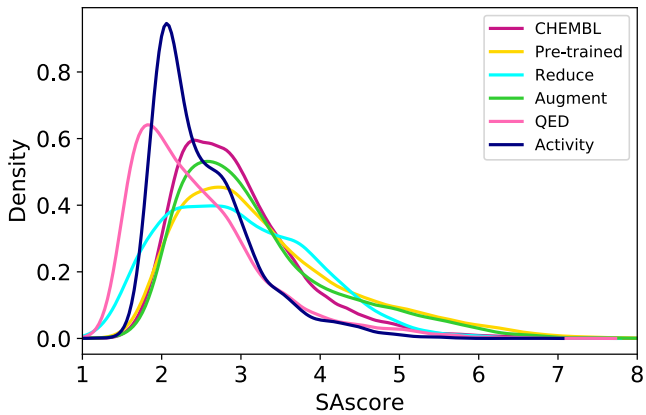


Figure S2: Kernel density estimates of the SA scores calculated for the reference data, molecules sampled from a pre-trained GraphINVENT model, and molecules sampled in the different RL scenarios.

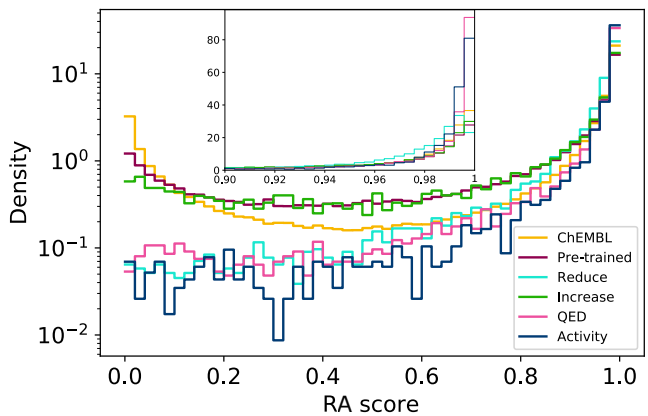


Figure S3: Histogram of the RA scores calculated for the reference data, molecules sampled from a pre-trained GraphINVENT model, and molecules sampled in the different RL scenarios. Insert: zoomed-in estimates for the region with high RA scores

for molecules in ChEMBL suggest that these would be easier to synthesize than the molecules generated by a pre-trained GraphINVENT model (same observation as with AiZynthFinder), the RA score predicts almost identical estimates of retrosynthetic accessibility for both. On the other hand, molecules generated by the size-reduction model are predicted to be more

synthesizable than those in ChEMBL by using both the RA score and AiZynthFinder, in disagreement with the predictions from the SA score. Nonetheless, both scores predict these to be easier to synthesize than molecules sampled from the pre-trained model. Surprisingly, both SA and RA scores predict that molecules generated by the model trained to increase molecular size would be more synthesizable than those sampled from the pre-trained model. Conversely, AiZynthFInder predicts them to be harder to make than both the average molecule in ChEMBL and those sampled from the pre-trained GraphINVENT model.

In the histograms presented in Figures S2 and S3, we observe heavier tails in the harder-to-make region (right for SA, left for RA) for the pre-trained and increasing-size model. Therefore, according to these metrics, it is more likely to sample a "very" difficult-to-make molecule from these models. We observe again here a discrepancy among these two metrics for the size-reduction model, where the SA score is predicted to have a heavier tail to the right for this model compared to ChEMBL, in contrast to the RA score, which predicts a light tail to the left (comparable with QED- and activity-promoting models) for this same scenario.

References

(1) Mercado, R.; Rastemo, T.; Lindelöf, E.; Klambauer, G.; Engkvist, O.; Chen, H.; Bjerrum, E. J. Graph networks for molecular design. *Mach. Learn.: Sci. Technol.* **2021**, *2*, 025023.

(2) Glorot, X.; Bengio, Y. Understanding the difficulty of training deep feedforward neural networks. Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics. Chia Laguna Resort, Sardinia, Italy, 2010; pp 249–256.

(3) Kingma, D. P.; Ba, J. Adam: A Method for Stochastic Optimization. 2014; `https://arxiv.org/abs/1412.6980`.

(4) Smith, L. N.; Topin, N. Super-convergence: Very fast training of neural networks using large learning rates. Artificial Intelligence and Machine Learning for Multi-Domain Operations Applications. 2019; p 1100612.

(5) Ertl, P.; Schuffenhauer, A. Estimation of synthetic accessibility score of drug-like molecules based on molecular complexity and fragment contributions. *J. Cheminf.* **2009**, *1*.

(6) The PubChem Database. `https://pubchem.ncbi.nlm.nih.gov/`.

(7) Genheden, S.; Thakkar, A.; Chadimová, V.; Reymond, J.-L.; Engkvist, O.; Bjerrum, E. AiZynthFinder: a fast, robust and flexible open-source software for retrosynthetic planning. *J. Cheminf.* **2020**, *12*.

(8) Ertl, P.; Lewis, R.; Martin, E.; Polyakov, V. In silico generation of novel, drug-like chemical matter using the LSTM neural network. 2017; `https://arxiv.org/abs/1712.07449`.