# UNIVERSITÀ DEGLI STUDI DI TRIESTE

## XXXVIII CICLO DEL DOTTORATO DI RICERCA IN

## APPLIED DATA SCIENCE AND ARTIFICIAL INTELLIGENCE

Finanziato dall'Unione europea - NextGenerationEU
*Funded by the European Union - NextGenerationEU*

# Advanced simulation algorithms for amorphous materials: from enhanced sampling to structural optimisation

Settore scientifico-disciplinare: FIS/03

DOTTORANDO
**LEONARDO GALLIANO**

COORDINATORE
**PROF. FRANCESCO PAULI**

SUPERVISORE DI TESI
**PROF. DANIELE COSLOVICH** Daniele Coslovich

CO-SUPERVISORE DI TESI
**PROF. LUDOVIC BERTHIER** BERTHIER

**ANNO ACCADEMICO 2024/2025**

Finanziato dall'Unione europea NextGenerationEU
Ministero dell'Università e della Ricerca
Italiadomani Piano Nazionale di Ripresa e Resilienza
Università degli Studi di Trieste

UNIVERSITÀ
DEGLI STUDI
DI TRIESTE

UMR 7083
Gulliver

ESPCI PARIS | PSL

Finanziato
dall'Unione europea
NextGenerationEU

Ministero dell'Università
e della Ricerca

Italiadomani
PIANO NAZIONALE
DI RIPRESA E RESILIENZA

# Abstract

Understanding the structure and dynamics of amorphous materials remains one of the major challenges in statistical physics. These difficulties stem from the complexity of their configurational landscapes, which gives rise to collective phenomena and extremely slow dynamics. In recent years, advances in machine learning and non-equilibrium algorithms have provided promising tools to tackle these challenges. In this thesis, we contribute to this growing effort by developing a set of algorithmic strategies to enhance sampling of complex landscapes and to optimise selected structural properties. We begin by introducing a general framework for adaptive Monte Carlo sampling inspired by policy gradient methods in reinforcement learning. This Policy-guided Monte Carlo approach dynamically adjusts the proposal distribution in the Metropolis–Hastings algorithm to maximise sampling efficiency. We demonstrate its application to paradigmatic models of glass-forming mixtures, where it can accelerate convergence in specific cases. We then explore a more expressive, though less general, approach to sampling built upon flow-based generative models. In particular, we extend the recent stochastic interpolants framework to particle systems, adapting it to their geometric constraints and underlying symmetries. We assess the limitations of current architectures in generating accurate samples for representative models of complex fluids. Moving beyond equilibrium sampling, we employ non-equilibrium algorithms to optimise targeted structural properties of disordered solids. Specifically, we generate hyperuniform configurations and samples exhibiting a high degree of local order. We then assess the stability of these configurations and find that neither feature correlates with better glasses, challenging recent claims in the literature. We believe this work offers a new set of advanced algorithmic tools to tackle key computational challenges related to the high-dimensional landscapes of disordered and complex systems.

**Keywords:** Enhanced sampling, high-dimensional optimisation, disordered systems, adaptive Monte Carlo, generative models.

# Table of Contents

# Introduction

The physical properties of amorphous materials pose deep conceptual challenges for condensed matter physics [1, 2]. At the same time, standard computational approaches struggle with modelling and simulating these systems [3]. Amorphous materials include atomic and molecular supercooled liquids [4–6], structural glasses [3], colloidal suspensions [7, 8], emulsions [9], foams [10], and granular assemblies [11, 12]. Unlike crystalline solids, these systems explore complex, rugged energy landscapes, where local minima are separated by barriers with a broad distribution of heights [13]. While liquids are disordered yet ergodic and can freely explore configuration space, amorphous solids remain trapped within a restricted subset of metastable states for times exceeding the observation time window. The transition between these two physical states is a long-standing open problem in condensed matter physics. Theoretical descriptions are elusive due to the absence of a broken-symmetry phase, while numerical simulations are hindered by the slow physical processes associated with rare collective rearrangements and barrier crossing [1, 2, 13].

This thesis is about *tools* to tackle these issues. It combines theoretically grounded algorithms with practical implementations and examines the physical consequences of their results. The methodological and the phenomenological aspects coexist throughout the manuscript: while some sections are more formal and mathematically dense, others take a more physical perspective, interpreting the results through the lens of statistical mechanics. We believe these two aspects are complementary: rigorous formulations are necessary for stating general results, while physical applications to simplified models of amorphous materials provide a better understanding of the problem at hand.

At the heart of the problem lies the complex energy landscape of amorphous materials. When the system is at equilibrium, this complexity translates into a highly multimodal Boltzmann distribution, which sampling algorithms must explore efficiently to estimate physical observables accurately. Similarly, optimising selected properties by searching for global minima in the same landscape is hindered by the presence of multiple local minima. A broad class of *stochastic algorithms* has been developed to deal with such high-dimensional landscapes [14–16]. Among them, *Monte Carlo methods* aim to compute averages with respect to a target probability distribution by generating representative configurations [17], while gradient-based optimisation algorithms exploit stochasticity to explore the landscape more efficiently [18].

Importance sampling (IS) and Markov chain Monte Carlo (MCMC) are among the most widely used Monte Carlo approaches. In IS, one draws independent samples from an auxiliary distribution and reweights them to recover averages over the target one. Choosing the auxiliary distribution is however far from trivial and can be nearly as difficult as sampling from the target itself: the auxiliary distribution must approximate the target closely enough to ensure that reweighting produces accurate estimates with a reasonable number of samples. MCMC methods, instead, generate a stochastic sequence of configurations whose stationary distribution coincides with the target one [19]. If the chain is ergodic, *i.e.*, able to explore the configuration state space without getting trapped, its samples can be used to compute averages with respect to the target distribution, even though they are not independent. As choosing the auxiliary distribution is crucial in importance sampling, so is designing an appropriate update rule, or *transition kernel*, to ensure convergence to the correct target distribution. The Metropolis-Hastings (MH) algorithm [20, 21] is, by far, the most popular MCMC kernel. It prescribes a specific way to construct an update rule that leads to the correct stationary distribution by splitting each update into a proposal and an acceptance step. The MH algorithm is extremely general, but its efficiency depends crucially on the choice of the proposal distribution: local proposals are simple and computationally cheap in high-dimensional settings, yet they may fail to explore multimodal configuration spaces efficiently, leading to slow convergence and poor sampling. The search for an optimal proposal, potentially adapted on the fly, has long been a central theme in stochastic simulations [22–24].

In recent years, *deep learning* has opened new perspectives [25]. Instead of relying on local moves, one may attempt to learn a global approximation of the target distribution using *generative models*. Among these methods, normalising flows (NFs) are particularly well-suited for the task: they learn an invertible map that transforms a simple base distribution into an approximation of a complex target distribution, while simultaneously providing access to the corresponding probability density [26–28]. However, training NFs is often computationally expensive, which has so far limited their applicability to relatively small physical systems [29]. Recent developments based on *flow-matching* techniques have significantly reduced this cost [30, 31], enabling the training of large models that now form the backbone of modern generative artificial intelligence for images [32], videos [33], and even weather prediction [34]. Physical systems, however, require some special care. Exact sampling of the Boltzmann distribution is required to obtain correct observables, whereas in tasks such as image generation, "good enough" is often sufficient. This key difference explains why generative models still struggle to sample physical systems accurately, despite their remarkable success in other domains. Learning good approximations of the Boltzmann distribution remains extremely challenging, but incorporating prior knowledge, such as symmetries or geometric constraints, has recently led to remarkable progress, allowing NFs to be trained on moderately large physical systems [35–37]. This is a rapidly evolving field, with new architectures and training algorithms being proposed constantly.

Disordered systems and amorphous materials are known to be a very hard benchmark for all these approaches [38–40]. Despite decades of efforts, no general solution exists for efficiently sampling their equilibrium properties, and the problem remains open. In this work, we take a step in this direction by developing general methods with solid theoretical grounds. Their current applicability is still rather limited, but we believe their generality provides a solid foundation for future advances toward systematic approaches to sampling and optimisation of systems with rugged energy landscapes.

The manuscript is organised as follows:

- **Chapter 1** introduces the necessary background on Monte Carlo methods and generative models, presenting the definitions and theoretical results that will be used throughout the thesis.

- **Chapter 2** reviews the phenomenology of glassy behaviour in amorphous materials and introduces the models and numerical techniques we employed to simulate them.

- **Chapter 3** presents a generalisation of the *Policy-guided Monte Carlo* (PGMC) framework, originally introduced by Bojesen [41], to general configuration spaces. PGMC is an adaptive MCMC method inspired by reinforcement learning, in which the proposal distribution within the MH algorithm is adapted on the fly to maximise a chosen reward. After developing the theoretical formulation, we demonstrate its application to glass-forming mixtures, where simple moves are optimised to accelerate convergence. This chapter is based on the published article *Policy-Guided Monte Carlo on General State Spaces: Application to Glass-Forming Mixtures*, written with Riccardo Rende and Daniele Coslovich [42].

- **Chapter 4** introduces *Equivariant Riemannian Stochastic Interpolants* (eRSI), a flow-based generative model specifically designed for multicomponent particle systems with periodic boundary conditions. eRSI combines Riemannian stochastic interpolants [43] with equivariant flows [44] to construct an approximation of the Boltzmann distribution that preserves the relevant symmetries and geometric constraints. This chapter is based on the article *Riemannian Stochastic Interpolants for Amorphous Particle Systems*, written with Louis Grenioux, Giulio Biroli, Ludovic Berthier, and Marylou Gabrié [45], which is currently under review.
- **Chapter 5** explores methods to optimise selected structural quantities in disordered solids [46, 47]. We examine the physical connection between structurally optimised configurations and glass stability, contradicting recent claims [48–51]. The chapter is based on unpublished work.

# Chapter 1

# Preliminaries

We begin by reviewing some fundamental concepts and tools on which this thesis is built. We adopt a general formalism based on measure-theoretic probability and Riemannian geometry. Readers less familiar with these notions may simply interpret expressions such as $P(\mathrm{d}x)$ and $K(x, \mathrm{d}x')$ in the more familiar forms $p(x)\,\mathrm{d}x$ and $k(x'|x)\,\mathrm{d}x'$, and think of the underlying space as the Euclidean domain $\mathbb{R}^d$, where geodesics are straight lines. Appendix A provides a brief overview of this mathematical background. We first introduce Monte Carlo methods [14], which form the foundation of all algorithms discussed in this thesis, following closely the presentation given by Robert and Casella [17]. Specifically, we focus on importance sampling (IS) [52] and Markov chain Monte Carlo (MCMC) [20]. We then discuss the Metropolis–Hastings algorithm [53], a popular MCMC method at the core of both the adaptive Monte Carlo framework developed in Chapter 3 and one of the optimisation algorithms described in Chapter 5. Finally, we introduce continuous normalising flows [54] and stochastic interpolants [31], which we extensively use in Chapter 4 to construct a generative model for particle systems.

## 1.1 Monte Carlo methods

Many problems in statistical physics can be expressed as the calculation of expectations with respect to a probability measure [17]. Let $(\mathsf{X}, \mathcal{E}, P)$ be a probability space and $h : \mathsf{X} \to \mathbb{R}$ be an integrable measurable function, we are interested in quantities of the form

$$\mathbb{E}_P[h] \;=\; \int_{\mathsf{X}} h(x)\,P(\mathrm{d}x) < \infty \,. \tag{1.1}$$

In low dimension, deterministic quadrature methods may approximate such integrals effectively. In high-dimensional settings, or when $P$ is only specified up to a normalising constant, these methods become impractical: the number of function evaluations typically grows exponentially with dimension. The key idea of Monte Carlo methods is to replace the integral by an empirical average built from random samples:

$$\hat{\mu}_n \;=\; \frac{1}{n}\sum_{k=1}^{n} h(X_k), \quad X_k \sim P \; . \tag{1.2}$$

The quantity $\hat{\mu}_n$ is an *unbiased estimator* of $\mathbb{E}_P[h]$, *i.e.* $\mathbb{E}[\hat{\mu}_n] = \mathbb{E}_P[h]$. Moreover, if the samples $X_k$ are independent and identically distributed (i.i.d.), the strong law of large numbers ensures that $\hat{\mu}_n$ converges almost surely to $\mathbb{E}_P[h]$ as $n \to \infty$ (see Definition A.1.17). If $h \in \mathcal{L}^2\left(\mathsf{X},\, \mathcal{E},\, P\right)$, then the central limit theorem states that the fluctuations of $\hat{\mu}_n$ around $\mathbb{E}_P[h]$ are asymptotically normal with variance

$$\mathrm{Var}\left[\hat{\mu}_n\right] \;=\; \frac{\mathrm{Var}\left[h\right]}{n} \;=\; \frac{1}{n}\,\mathbb{E}_P\left[\left(h - \mathbb{E}_P\left[h\right]\right)^2\right] \; . \tag{1.3}$$

An unbiased estimator of $\mathrm{Var}\left[\hat{\mu}_n\right]$ is the empirical variance

$$\hat{\sigma}_n^2 \;=\; \frac{1}{n-1}\sum_{k=1}^{n}\left(h(X_k) - \hat{\mu}_n\right)^2 \; , \tag{1.4}$$

which allows one to easily estimate confidence intervals.

The apparent simplicity of the method relies on the strong assumption that we can draw i.i.d. samples from $P$. In practice, this is rarely the case. There are two widely used strategies to address this issue:

- **Importance sampling** (IS): sample from an auxiliary distribution and reweight to recover expectations under $P$.

- **Markov chain Monte Carlo** (MCMC): construct a Markov chain with invariant distribution $P$ and use its correlated samples to form the estimator.

We discuss these two strategies in more detail in the following sections, as each method introduced in this thesis relies on one of them. More specifically, IS is used to reweight samples from generative models in Chapter 4 and for Umbrella Sampling in Chapter 5, while MCMC forms the basis of the adaptive Monte Carlo framework described in Chapter 3 and one of the optimisation algorithms in Chapter 5.

### 1.1.1 Importance sampling

Suppose that drawing i.i.d. samples from $P$ is not feasible, but that we can sample from another probability distribution $Q$ defined on the same measurable space $(\mathsf{X}, \mathcal{E})$. If $P$ is absolutely continuous with respect to $Q$ (see Definition A.1.8), the Radon–Nikodym derivative $w = \mathrm{d}P/\,\mathrm{d}Q$ is well defined, and we can rewrite (1.1) as

$$\mathbb{E}_P[h] \;=\; \mathbb{E}_Q\left[h\,w\right] \;. \tag{1.5}$$

Provided that $h\,w \in \mathcal{L}^1\left(\mathsf{X},\, \mathcal{E},\, Q\right)$, the *importance sampling estimator*

$$\hat{\mu}_n^{\mathrm{IS}} \;=\; \frac{1}{n}\sum_{k=1}^{n} h(X_k)\,w(X_k), \quad X_k \sim Q \;, \tag{1.6}$$

is an unbiased estimator of $\mathbb{E}_P[h]$ and, if the samples are i.i.d., it converges almost surely to $\mathbb{E}_P[h]$ as $n \to \infty$.

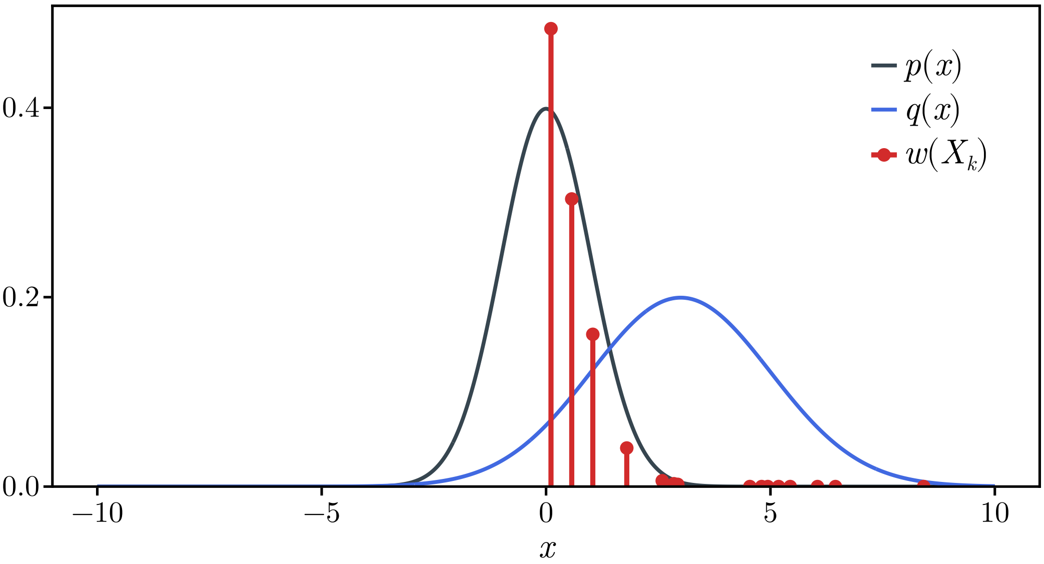

**Figure 1.1:** Importance sampling. The target distribution $P$ is a Gaussian with mean $\mu_P = 0$ and standard deviation $\sigma_P = 1$, whose density $p(x)$ is shown in black. The proposal distribution $Q$ is a Gaussian with mean $\mu_Q = 3$ and standard deviation $\sigma_Q = 2$, with density $q(x)$ shown in blue. Samples $\{X_k\}$ drawn from $Q$ are reweighted by the importance weights $w(x) = p(x)/q(x)$, illustrated as red stems.

Importantly, the variance of $\hat{\mu}_n^{\mathrm{IS}}$ depends on the choice of $Q$ and it is bounded if and only if $h\,w \in \mathcal{L}^2\left(\mathsf{X},\, \mathcal{E},\, Q\right)$. If this condition fails, the variance is infinite and the estimator, though unbiased, is useless: a few samples with large weights can dominate and the sequence $\hat{\mu}_n^{\mathrm{IS}}$ can have uncontrolled jumps [17].

**Example 1.1.1** (Infinite variance)**.** Suppose $P = \mathcal{N}(0{,}1)$, $Q = \mathcal{N}(0, \sigma^2)$ and $h = 1$. Then

$$w(x) \;=\; \frac{\mathrm{d}P}{\mathrm{d}Q}(x) \;=\; \sigma \exp\left(\frac{1-\sigma^2}{2\sigma^2}\, x^2\right) \;,$$

and

$$\mathbb{E}_Q\left[w^2\right] \;=\; \frac{\sigma^2}{\sqrt{2\sigma^2 - 1}} \;,$$

which is finite if and only if $\sigma^2 > 1/2$. That is, even if $Q$ and $P$ are equivalent for any $\sigma > 0$, $Q$ must be "wide enough" to yield a useful estimator.

### Effective sample size

Even when the variance is finite, it may still be large if the importance weights $w$ fluctuate strongly. A convenient way to measure this effect is the *effective sample size* (ESS):

$$\mathrm{ESS}(n) \;=\; \frac{\left(\sum_{k=1}^{n} w\left(X_k\right)\right)^2}{\sum_{k=1}^{n} w^2\left(X_k\right)}, \quad X_k \sim Q \text{ i.i.d.} \tag{1.7}$$

Intuitively, the ESS measures how many equally-weighted, independent samples from $P$ would provide approximately the same variance as the weighted estimator. From the law of large numbers, $\mathrm{ESS}(n) \to n/\mathbb{E}_Q\left[w^2\right]$ almost surely as $n \to \infty$.

**Example 1.1.2** (Poor sampling)**.** Suppose $P = \mathcal{N}(0{,}1)$, $Q = \mathcal{N}(\mu{,}1)$ and $h = 1$. Then

$$w(x) \;=\; \frac{\mathrm{d}P}{\mathrm{d}Q}(x) \;=\; \exp\left(-\mu x + \frac{\mu^2}{2}\right) \;,$$

and

$$\mathbb{E}_Q\left[w^2\right] \;=\; \exp\left(\mu^2\right) \;.$$

So, for large $n$, $\mathrm{ESS}(n) \sim n \exp\left(-\mu^2\right)$, which means that estimating expectations in Eq. (1.1) using samples from $Q$ requires a number of samples that grows exponentially with $\mu^2$.

### Self-normalised importance sampling

Often $P$ is only known up to a constant: $P = Z^{-1}\tilde{P}$ with unknown normalising constant $Z > 0$. In that case importance weights can still be defined as

$$w(x) \;=\; \frac{\mathrm{d}\tilde{P}}{\mathrm{d}Q}\,(x), \tag{1.8}$$

but the corresponding estimator

$$\hat{\mu}_n^{\mathrm{SNIS}} \;=\; \frac{\sum_{k=1}^n h(X_k)\, w(X_k)}{\sum_{k=1}^n w(X_k)} \tag{1.9}$$

is biased for finite $n$, *i.e.* $\mathbb{E}_P\left[\hat{\mu}_n^{\mathrm{SNIS}}\right] \neq \mathbb{E}_P[h]$. However, $\hat{\mu}_n^{\mathrm{SNIS}}$ is *consistent*, in the sense that if $h\,w \in \mathcal{L}^1(\mathsf{X}, \mathcal{E}, Q)$ and $w \in \mathcal{L}^1(\mathsf{X}, \mathcal{E}, Q)$, then $\hat{\mu}_n^{\mathrm{SNIS}} \to \mathbb{E}_P[h]$ almost surely as $n \to \infty$.

### 1.1.2 Markov Chain Monte Carlo

Choosing an effective proposal distribution for IS becomes increasingly difficult in high dimensions: regions of significant target probability may be missed, while importance weights concentrate on a small fraction of samples. In such cases, the effective sample size vanishes and IS becomes impractical. An alternative idea is to build a sequence of dependent random variables, a *Markov chain*, whose long time behaviour follows the target distribution [19]. Although the samples generated in this way are no longer independent, classical limit theorems, such as the law of large numbers and central limit theorem, extend to this setting under specific conditions. This strategy thus bypasses the need to draw i.i.d. samples from a given distribution.

**Definition 1.1.1.** A stochastic process $(X_t)_{t\geq 0}$ on a measurable space $(\mathsf{X}, \mathcal{E})$ is a *Markov chain* with transition kernel $K$ and initial distribution $P_0$ if for any $t \geq 0$ and measurable sets $A_0, \ldots, A_t \in \mathcal{E}$, the joint distribution $P_X$ of $(X_0, \ldots, X_t)$ satisfies

$$P_X\left(A_0, \ldots, A_t\right) = \int_{A_0\times\cdots\times A_t} K\left(x_{t-1}, \mathrm{d}x_t\right) \cdots K\left(x_0, \mathrm{d}x_1\right) P_0\left(\mathrm{d}x_0\right) \;. \tag{1.10}$$

Intuitively, a Markov chain is a random walk through the state space in which each step is determined only by the current position. The transition kernel $K(x, A)$ specifies the local rule that tells how to jump from state $x$ to a new state $x'$ within $A$ (see Definition A.1.12). The *Markov property* formalises this by stating that the conditional distribution of the future depends only on the present and not on the past [19, Proposition 3.4.3]: for any bounded measurable $h$,

$$\mathbb{E}\left[h\left(X_t\right) \mid \left(X_s\right)_{s<t}\right] \;=\; \mathbb{E}\left[h\left(X_t\right) \mid X_{t-1}\right] \;. \tag{1.11}$$

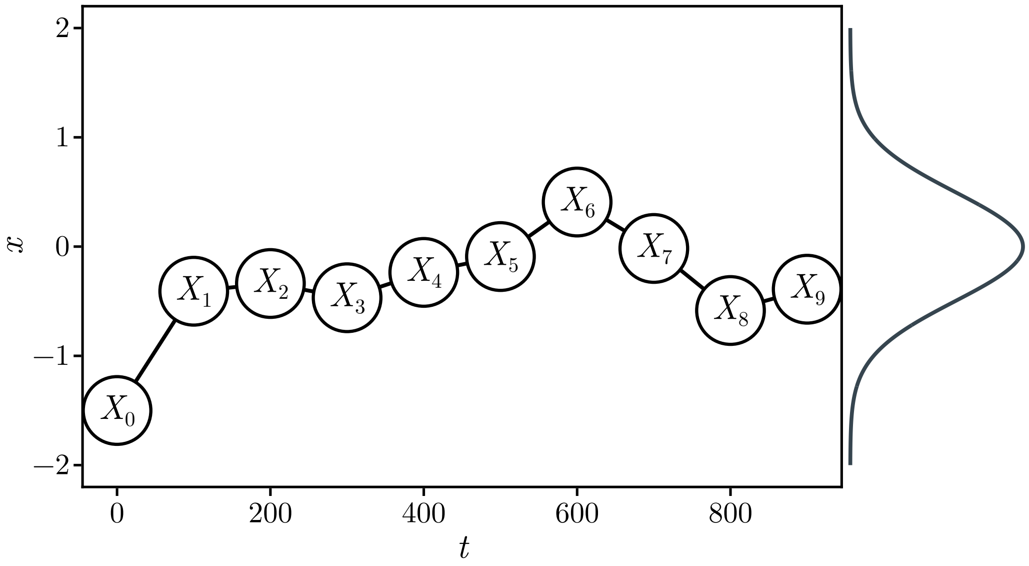

**Figure 1.2:** A Markov chain. Starting from an initial state $X_0$, the chain evolves according to the transition kernel $K$. After an initial transient, the samples follow the target distribution $P$, whose density is shown on the right.

### Irreducibility, recurrence and aperiodicity

To understand the long time behaviour of Markov chains and, most importantly, their convergence to an invariant distribution, it is necessary to introduce a few properties. We will only recall the most essential notions needed to determine whether the chain can reach all the relevant parts of the state space, whether it keeps coming back, and whether it avoids being trapped in a cycle. These ideas are formalised by the notions of irreducibility, recurrence, and aperiodicity, respectively (see [19] for a more complete treatement). We denote by $K^t$ the $t$-th iteration of the kernel $K$, defined recursively as

$$K^t(x, A) \;:=\; \int_{\mathsf{X}} K^{t-1}(y, A)\, K(x, \mathrm{d}y), \quad \forall A \in \mathcal{E}\ . \tag{1.12}$$

We also denote the *law* of $X_t$ by

$$\begin{aligned} P_t(A) &:= \int_{\mathsf{X}} K^t(x, A)\, P_0(\mathrm{d}x) && (1.13)\\ &= \int_{\mathsf{X}} K^{t-1}(x, A)\, P_{t-1}(\mathrm{d}x), \quad \forall A \in \mathcal{E}\ . && (1.14) \end{aligned}$$

A Markov chain $(X_t)$ is *time homogeneous* if the transition kernel $K$ does not depend on $t$. Given a set $A$, we call $\eta_A$ the number of passage of the chain in $A$. Given a measure $\mu$, a Markov chain is said to be *$\mu$-irreducible* if for any $A$ such that $\mu(A) > 0$ and any $x \in \mathsf{X}$, there exists $t > 0$ such that $K^t(x, A) > 0$. A chain is *recurrent* if it is $\mu$-irreducible and, for every $A$ such that $\mu(A) > 0$, $\mathbb{E}\left[\eta_A \mid X_0 = x\right] = \infty$, for every $x \in A$. A chain is *Harris recurrent* if it is $\mu$-irreducible and, for every $A$ such that $\mu(A) > 0$, $\eta_A = \infty$ almost surely. A $\mu$-irreducible chain is aperiodic if there exists a set $A$ with $\mu(A) > 0$ such that for all $x \in A$,

$$\mathrm{gdc}\left\{t > 0 : K^t(x, A) > 0\right\} = 1. \tag{1.15}$$

### Invariant measure, reversibility and detailed balance

A $\sigma$-finite measure is *invariant* for a Markov chain $(X_t)$ with transition kernel $K$ if for any measurable set $A \in \mathcal{E}$,

$$\int_{\mathsf{X}} K(x, A)\, \mu(\mathrm{d}x) \;=\; \mu(A)\,. \tag{1.16}$$

An invariant measure describes the long term behaviour of the chain: if $X_t \sim \mu$, then $X_s \sim \mu$ for any $s > t$. For this reason, a Markov chain with invariant measure $\mu$ is called a *stationary* Markov chain. The goal of MCMC methods is precisely to construct a Markov chain whose invariant measure coincides with the target distribution $P$.

**Theorem 1.1.1** (Existence of invariant measure)**.** *If $(X_t)$ is time homogeneous and recurrent, then there exists an invariant measure $\mu$, which is unique up to a multiplicative constant.*

*Proof.* See [19, Theorem 10.0.1]. □

There is a stronger condition, often easier to verify in practice, which guarantees invariance automatically: a stationary Markov chain is *reversible* with respect to $\mu$ if for any measurable sets $A, B \in \mathcal{E}$,

$$\int_A K(x, B)\, \mu(\mathrm{d}x) \;=\; \int_B K(x, A)\, \mu(\mathrm{d}x)\,. \tag{1.17}$$

This condition is also known as *detailed balance*. Intuitively, it means that the probability flow from $A$ to $B$ is exactly balanced by the reverse flow from $B$ to $A$. For this reason, reversibility is often interpreted as a microscopic definition of equilibrium. Note that reversibility implies stationarity with respect to $\mu$ (take $B = \mathsf{X}$ in (1.17)), but the converse is not true in general. Most classical MCMC algorithms, such as Metropolis–Hastings, are designed to build a transition kernel that satisfies detailed balance with respect to the target distribution $P$.

### Ergodicity and convergence

For building a MCMC algorithm, it is not enough to ensure that the target distribution $P$ is invariant for the chain. We also need to ensure that the distribution of $(X_t)$ *converges* to $P$. It turns out this can be guaranteed under specific conditions.

**Theorem 1.1.2** (Convergence of Markov chains)**.** *If $(X_t)$ is time homogeneous, Harris recurrent and aperiodic with invariant measure $\mu$, then, for any initial distribution $P_0$, the law $P_t$ of $X_t$ converges to $\mu$ in total variation as $t \to \infty$.*

*Proof.* See [19, Theorem 13.3.3]. ☐

A chain that is Harris recurrent and aperiodic is called *ergodic*.

The convergence in total variation (see Definition A.1.15) ensures that the distribution of a single state $X_t$ approaches the invariant measure $\mu$. However, in practice we never observe an entire distribution, only one trajectory $(X_t)$. For Monte Carlo estimation, what matters is the behaviour of *empirical averages* computed along this trajectory. In the i.i.d. case, the law of large numbers guarantees that empirical averages converge almost surely to the expectation, and the central limit theorem describes their fluctuations. These results can be extended to the dependent and not stationary samples of Markov chains under suitable conditions in what are known as *ergodic theorems*: they ensure that the MCMC estimator remains consistent and asymptotically normal despite correlations. These results are the justification for using Markov chains in Monte Carlo sampling. Here, we only state the equivalent of the strong law of large numbers.

**Theorem 1.1.3** (Strong law of large numbers for Markov chains)**.** *Let $(X_t)$ be a time homogeneous Markov chain with invariant probability measure $P$ and $h \in \mathcal{L}^1(\mathsf{X}, \mathcal{E}, P)$. Then*

$$\frac{1}{n}\sum_{k=1}^{n} h(X_k) \to \mathbb{E}_P[h] \tag{1.18}$$

*almost surely, if and only if $(X_t)$ is Harris recurrent.*

*Proof.* See [19, Theorem 17.1.7]. ☐

## 1.1.3 Handling multimodal distributions

While IS and MCMC are powerful in theory, both methods struggle when applied to multimodal distributions. We are particularly interested in such cases, since multimodality naturally arises from the complex energy landscapes of amorphous systems (see Chapter 2). For IS, a simple proposal distribution with support over the entire state space is highly inefficient: most of the sampling effort is wasted in low–probability regions, and in high dimensions this inefficiency grows exponentially. Designing an *ad hoc* proposal that avoids this problem and remains easy to sample from can be just as hard as solving the original problem. For standard MCMC, the difficulty comes from *local* updates: they allow efficient exploration within a single mode, but make it extremely unlikely to cross the barriers that separate different modes. As a result, different chains may become trapped in different regions, and the combined average may fail to reproduce the correct relative weights of the modes.

**Example 1.1.3** (Gaussian mixture)**.** Consider a Gaussian mixture with two separated components, one much taller than the other. An IS scheme with a broad proposal that covers both modes is technically valid, but in practice most samples fall in regions where the target density is negligible, leading to a waste of resources. Conversely, a local MCMC chain initialised near one of the modes will typically remain confined there. Even running several independent chains does not solve the problem: some chains explore the smaller mode, others the larger one, but the relative frequencies will not match the true ones (see Fig. 1.3).

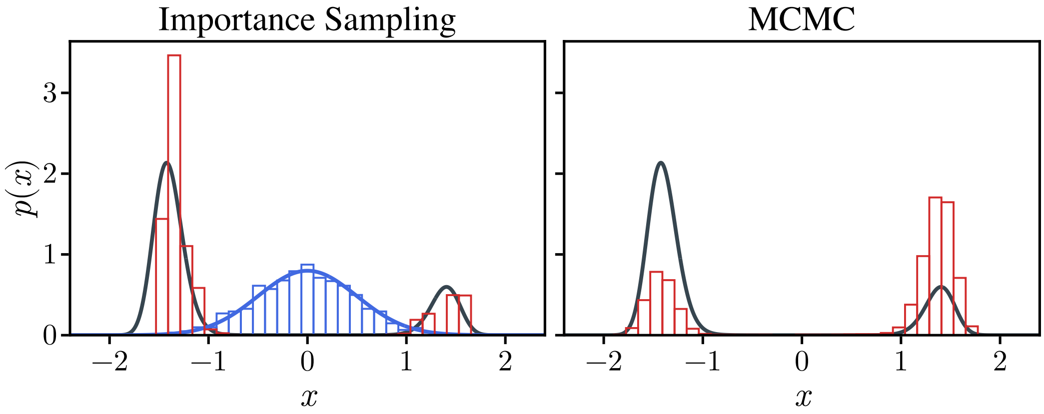

**Figure 1.3:** Sampling a bimodal distribution. Left: IS with a broad proposal (blue) wastes most samples in low-probability regions. The target density is shown in black, and reweighted samples are shown in red. Right: multiple MCMC chains, all initialised at $x = 0$, get trapped in different modes, leading to incorrect relative weights.

This example highlights the limitations of standard IS and MCMC for complex distributions, and motivates the development of more advanced techniques. We will return to this issue in later chapters, where we explore strategies that can, at least in part, address these challenges.

## 1.2 The Metropolis-Hastings algorithm

We now present the most popular and widely used MCMC algorithm: the Metropolis–Hastings (MH) algorithm [20, 21]. Originally introduced in 1953, it has since become a milestone of computational physics and modern statistics, with applications ranging from statistical mechanics to Bayesian inference. Here, we present the algorithm in its most general formulation, which can be applied to discrete, continuous, or mixed probability distributions. This version, due to Green [55], is sometimes referred to as the Metropolis–Hastings–Green algorithm. Green also introduced the idea of state-dependent mixing, which we will not consider here. For a general overview of the algorithm, we refer to Ref. [53, Chapter 1]. Working with the general formulation will be necessary to state the theoretical results of Chapter 3 in full generality.

The basic idea of the MH algorithm is to construct a Markov chain that satisfies detailed balance with respect to the target distribution $P$. To this end, the transition kernel $K$ is built in two steps:

- **Proposal step:** given the current state $x$, propose a new state $x' \sim Q(x, \cdot)$, where $Q$ is the *proposal kernel.*
- **Acceptance step:** accept the proposed state $x'$ with probability

  $$\alpha(x, x') \;=\; \min\left\{1,\, h(x, x')\right\}, \tag{1.19}$$

  where

  $$h(x, x') \;=\; \begin{cases} \dfrac{P\left(\mathrm{d}x'\right) Q\left(x', \mathrm{d}x\right)}{P\left(\mathrm{d}x\right) Q\left(x, \mathrm{d}x'\right)}, & \text{if } P(\mathrm{d}x')Q(x', \mathrm{d}x) \sim P(\mathrm{d}x)Q(x, \mathrm{d}x') \\ 0, & \text{otherwise} \end{cases} \tag{1.20}$$

  is the Hastings ratio.

The resulting transition kernel from a state $x \in \mathsf{X}$ to any state $x' \in \mathrm{d}x'$ is then

$$K(x, \mathrm{d}x') \;\coloneqq\; Q\left(x, \mathrm{d}x'\right) \alpha\left(x, x'\right) + \delta\left(x, \mathrm{d}x'\right) \int_{\mathsf{X}} \left(1 - \alpha\left(x, x''\right)\right) Q\left(x, \mathrm{d}x''\right)\;, \tag{1.21}$$

where $\delta$ is the Dirac kernel (see Example A.1.4). The first term corresponds to the case where the proposal from $x$ to a state $x' \in \mathrm{d}x'$ is accepted, while the second term accounts for the case where $x \in \mathrm{d}x'$ and the proposal is rejected, so that $x$ stays in $\mathrm{d}x'$.

The key property of the MH algorithm is that, by construction, the resulting transition kernel $K$ satisfies detailed balance with respect to the target distribution $P$. Before stating the result, three remarks are in order:

- The acceptance probability (1.19) can be generalised by replacing $P$ with a measure $\tilde{P}$ such that $P = Z^{-1}\tilde{P}$ for some unknown normalising constant $Z > 0$. This is particularly useful when $P$ is only known up to a constant, as often happens in statistical physics and Bayesian statistics. To simplify the notation, we will always assume that $P$ is a probability distribution, but all results extend trivially to this more general case.
- In typical statistical physics applications, the target distribution has the Boltzmann form $P(\mathrm{d}x) \propto \exp\left(-U(x)/(k_B T)\right)\mathrm{d}x$. If, in addition, the proposal kernel $Q$ is symmetric, $Q(x, \mathrm{d}x') = Q(x', \mathrm{d}x)$, the Hastings ratio simplifies to $h(x, x') = \exp\big(-\left(U(x') - U(x)\right)/(k_B T)\big)$, and the acceptance probability reduces to the original Metropolis algorithm [20].
- The Radon–Nikodym derivative in the first case of (1.20) is well defined only if the measure $P(\mathrm{d}x)Q(x, \mathrm{d}x')$ is absolutely continuous with respect to $P(\mathrm{d}x')Q(x', \mathrm{d}x)$. This might not be the case in general. However, it can be shown that there exists a symmetric set $R$ such that these two measures are equivalent (see Definition A.1.8) on $R$, and mutually singular on $R^c$ [56, Proposition 1]. In order for the algorithm to work, $\alpha$ must be set to zero almost everywhere on $R^c$.

### 1.2.1 Reversibility and convergence

**Theorem 1.2.1** (Metropolis-Hastings-Green). *Given a proposal kernel $Q$, the transition kernel* (1.21) *is reversible with respect to $P$.*

*Proof.* See [56, Theorem 2]. □

This result is remarkably general: for any proposal kernel $Q$, the MH algorithm automatically gives a Markov chain that satisfies detailed balance with respect to $P$, which is therefore an invariant distribution of the chain. This (partially) removes the need to design an *ad hoc* transition kernel $K$ tailored to each problem. However, reversibility alone is not enough. It guarantees that $P$ is a stationary distribution of the chain, but not that the chain will actually converge to $P$ from an arbitrary initial condition, nor that convergence will occur within a reasonable time. These properties depend, unsurprisingly, on the specific choice of $Q$.

**Theorem 1.2.2** (Convergence of Metropolis-Hastings)**.** *Let $(X_t)$ be a Markov chain generated by the MH algorithm with proposal kernel $Q$. If $(X_t)$ is time-homogeneous and $P$-irreducible, then* (1.18) *holds almost surely for any $h \in \mathcal{L}^1(\mathsf{X}, \mathcal{E}, P)$. If, in addition, $(X_t)$ is aperiodic, then it is ergodic and the law $P_t$ of $X_t$ converges to $P$ in total variation as $t \to \infty$, for any initial distribution.*

*Proof.* It follows from [17, Lemma 7.3] together with Theorems 1.1.2 and 1.1.3. ☐

The key assumption here is $P$-irreducibility, which is the property of the chain of eventually reaching any region of the state space where $P$ assigns positive probability. A sufficient condition ensuring $P$-irreducibility is that $Q$ can reach a neighbourhood of every point with uniformly positive probability and that its support is symmetric [17, Lemma 7.6]. Under such conditions, the MH chain is both irreducible and aperiodic, and hence ergodic.

While these properties ensure convergence in the asymptotic sense, they still provide no information about the *rate* of convergence. In practice, the efficiency of the MH algorithm crucially depends on the choice of the proposal kernel $Q$. An effective proposal must balance two goals: it should generate moves that are accepted with reasonable probability, yet these moves must be substantial enough to escape local regions of high probability and allow mixing between different modes. The design of such proposals is an art in itself, which lies at the core of many modern extensions of MH.

### 1.2.2 Adaptive versions

The idea of tuning the proposal kernel $Q$ to improve the performance of the Metropolis–Hastings algorithm is almost as old as the algorithm itself [22, 57–59]. Adaptive versions of MH (and, more generally, of MCMC) build on the idea of adjusting the proposal distribution on the fly, based on the past history of the chain, in order to achieve some notion of optimality. This notion can be, for example, the acceptance rate, the effective sample size, the asymptotic variance of the MC estimator. Here we briefly outline the main ideas behind these methods, specifically focusing on adaptive MH algorithms. For a more complete review of adaptive MCMC methods, see the tutorial by Andrieu and Thoms in Ref. [24].

To formalise the concept, consider a family of proposal kernels $\{Q_\theta\}$ parametrised by $\theta \in \mathbb{R}^p$. The key property of the MH algorithm is that, for any fixed $\theta$, the corresponding kernel $K_\theta$ (1.21) satisfies detailed balance with respect to the target distribution $P$, and thus $P$ is an invariant measure of the chain. However, as soon as $\theta$ is allowed to vary with time, the ergodicity properties of the chain may be lost [60].

**Example 1.2.1** (Variance collapse)**.** Consider $\mathsf{X} = \mathbb{R}$, $P = \mathcal{N}(0,1)$, and $Q_\theta(x, \cdot) = \mathcal{N}(x, e^\theta)$, $\theta \in \mathbb{R}$. Suppose that the adaptation rule depends on the average acceptance rate

$$J(\theta) = \mathbb{E}_{\substack{x \sim P \\ x' \sim Q_\theta}} \left[\alpha(x, x')\right] ,$$

and that $\theta_{t+1}$ is updated according to a stochastic approximation rule [15]

$$\theta_{t+1} = \theta_t - \eta_t\big(J(\theta_t) - J^\star\big) ,$$

where $J^\star$ is a target acceptance rate (typically 0.234 for random–walk MH in high dimensions [58]) and $\eta_t > 0$ is a step size. For each fixed $\theta$, $P$ is an invariant distribution and $K_\theta$ satisfies detailed balance. However, for arbitrary values of $\eta_t$, $\theta_t$ may decrease indefinitely, so that the proposal variance shrinks to zero and the chain effectively freezes. The resulting process is then non-ergodic, even though each $K_\theta$ is individually ergodic.

It turns out that, under suitable conditions, the convergence of adaptive MCMC methods can be guaranteed.

**Theorem 1.2.3** (Roberts-Rosenthal)**.** *Let $(K_{\theta_t})_{t \geq 0}$, $\theta_t \in \mathbb{R}^p$, be a family of MH kernels with common invariant distribution $P$. Assume the following conditions hold:*

*i. For any $\varepsilon > 0$, there exists $N$ such that*

$$\sup_{A \in \mathcal{E}} \left\{K_\theta^N (x, A) - P(A)\right\} \leq \varepsilon , \tag{1.22}$$

*for all $x \in \mathsf{X}$ and $\theta \in \mathbb{R}^p$.*

*ii.*

$$\sup_{\substack{x \in \mathsf{X} \\ A \in \mathcal{E}}} \left\{K_{\theta_{t+1}} (x, A) - K_{\theta_t} (x, A)\right\} \xrightarrow[t \to \infty]{} 0 . \tag{1.23}$$

*Then the adaptive chain with transition kernel $K_{\theta_t}$ is ergodic.*

*Proof.* See [23, Theorem 1]. □

These conditions are known as *simultaneous uniform ergodicity* and *diminishing adaptation*, respectively. They essentially require that the algorithm keeps the chain away from "bad" regions of parameter space and that the updates to $\theta$ become slower and slower over time.

### Modern implementations

In the last two decades, adaptive MH methods have evolved from the tuning of a few scalar hyperparameters to the optimisation of proposal kernels based on deep neural networks. Early work focused on adjusting simple statistical quantities. In 2007, Pasarica and Gelman [61] proposed to optimise the variance of a Gaussian proposal kernel by maximising the expected squared jump distance, effectively reducing the correlation between consecutive samples. As already mentioned, Andrieu and Thoms [24] later unified these ideas within the framework of stochastic approximation [15]. More recently, automatic differentiation [62] has made it possible to compute gradients of arbitrarily complex functions, thereby enabling the optimisation of parameters of more expressive proposal distributions. For example, Christiansen *et al.* [63] applied this idea to Hamiltonian Monte Carlo (HMC), optimising the integration step size by differentiating through short molecular-dynamics trajectories.

Another line of work has been the connection between adaptive MCMC and the classical reinforcement learning (RL) problem [64]. In this context, the proposal distribution is interpreted as a *policy* that guides an *agent* in selecting transitions to maximise a *reward* related to sampling efficiency. A particularly relevant example for this thesis is the work of Bojesen [41], who introduced the *Policy-Guided Monte Carlo* (PGMC) framework to learn proposals via policy-gradient methods in discrete systems. Later, Zhao *et al.* [65] applied this framework to spin-ice models, using neural networks to represent more complex proposals. While the original PGMC formulation was limited to discrete spaces and made simplifying assumptions, we will revisit this approach in Chapter 3, presenting a corrected and generalised version that can be applied to arbitrary state spaces. A related contribution by Wang *et al.* [66] maps adaptive MH into a “true” RL problem, where the current state of the chain and the proposal are interpreted as a single coupled state. This formulation makes it possible to apply standard RL algorithms directly.

A complementary line of research has explored the use of *generative models* to learn complex proposal distributions from data. These models are built to generate samples from a distribution that approximates the target one, which can be used to construct efficient proposals [67]. Pioneering examples include the *Boltzmann generators* of Noé *et al.* [29] and the flow-based samplers of Albergo *et al.* [68], later improved by Gabrié *et al.* [69] (see also Section 1.3). These ideas have been recently extented to conditional proposals [70] and to generative models trained with flow matching [71]. We will return to several of these ideas and compare them with our formulation of PGMC in Chapter 3.

## 1.3 Continuous normalising flows

Normalising flows (NFs) are generative models that construct complex probability distributions by transforming a simple base distribution through an invertible map [26, 27]. They can be used not only to *sample* from such distributions but also to *evaluate* their density at any point in the state space. Although NFs were popularised in machine learning for generating images, audio, and text [28], their ability to directly evaluate probability densities makes them particularly suitable in statistical physics and Bayesian inference, where one needs to approximate a target distribution $P$ known only up to a normalising constant [29, 68]. Such approximations can then be employed as proposal distributions in importance sampling or MCMC algorithms. In this section we briefly review the main ideas underlying NFs, focusing on *continuous normalising flows* (CNFs) introduced by Chen *et al.* [54], which will later provide the foundation for our *equivariant Riemannian stochastic interpolants* (eRSI) framework discussed in Chapter 4.

NFs work by transforming a simple base distribution $Q_0$ on a space $\mathsf{X}$ that is easy to sample from into a more complex distribution $Q$ via a diffeomorphism $\Phi : \mathsf{X} \to \mathsf{X}$. The resulting distribution $Q$ is defined as the *pushforward* $\Phi_{\#}Q_0$ of $Q_0$ through $\Phi$ (see Definition A.1.4). Sampling from $Q$ is then straightforward: one simply draws $x_0 \sim Q_0$ and computes $x = \Phi(x_0)$, so that $x \sim Q$ by construction. The distribution $Q$ naturally depends on the choice of the map $\Phi$. In practice, to build a generative model that samples from an approximation of $P$, $\Phi$ is parametrised by a set of parameters $\theta \in \mathbb{R}^p$ to obtain a family of distributions $Q_\theta = \Phi_{\theta\#}Q_0$. The parameters are then optimised to make $Q_\theta$ close to $P$, typically by minimising the Kullback–Leibler divergence between $Q_\theta$ and $P$. The main challenge in this approach is to design a flexible yet tractable family of diffeomorphisms $\Phi_\theta$.

### 1.3.1 Neural ODEs

Continuous normalising flows (CNFs) provide an elegant way to construct such diffeomorphisms as solutions of an ordinary differential equation (ODE). Originally introduced by Chen *et al.* [54] for the Euclidean space $\mathsf{X} = \mathbb{R}^D$ and later extended to Riemannian manifolds by Mathieu *et al.* [72], CNFs define a time-dependent vector field $v : [0,1] \times \mathsf{X} \to T\mathsf{X}$ and consider the ODE

$$\frac{\mathrm{d}x}{\mathrm{d}t} = v\left(t, x\right) \ , \tag{1.24}$$

with initial condition $x(0) = x_0 \sim Q_0$. Since $v$ is a vector tangent to the manifold at each point, the solution $x(t)$ remains on the manifold for all times [73]. Under mild regularity conditions on $v$, this ODE admits a unique solution for all $t \in [0,1]$ [74]. The pushforward distribution $Q$ of $x(1)$ is then the result of transporting $Q_0$ through $\Phi : x_0 \mapsto x(1)$.

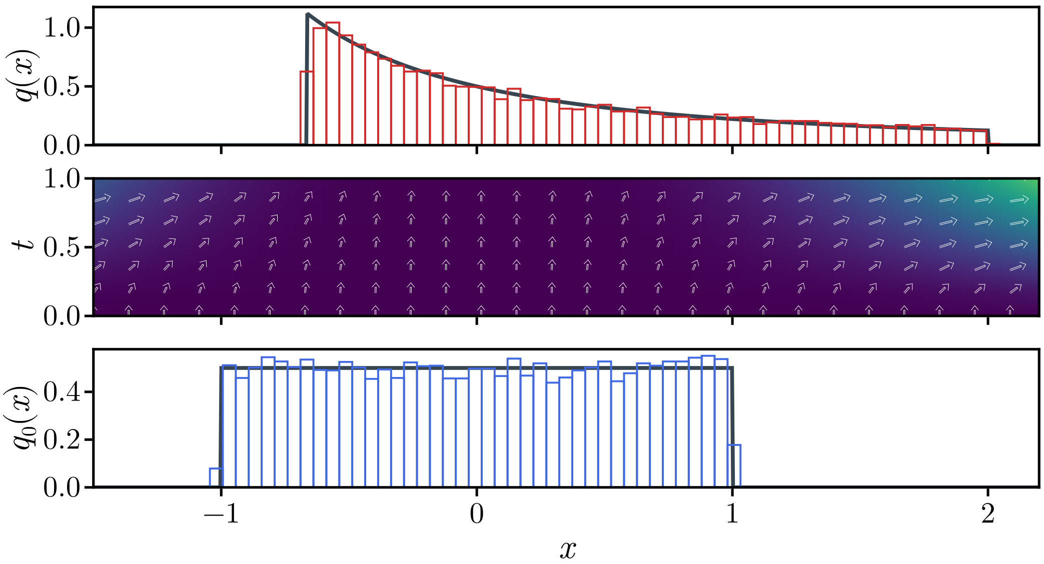

**Figure 1.4:** Continuous normalising flow. Bottom: density $q_0(x)$ of an uniform base distribution $Q_0$ (black), together with samples $x_0 \sim Q_0$ (blue). Middle: time-dependent vector field $v(t, x) = tx^2$ defining the continuous flow $\Phi$ that transports samples from $Q_0$ to the target distribution. Top: density $q(x)$ of the pushforward distribution $Q$ (black), with the corresponding transported samples $x = \Phi(x_0)$ (red).

Note that the time-dependent map $\Phi_t : x_0 \mapsto x(t)$, $t \in [0,1]$, defines a family of diffeomorphisms, with $\Phi_0 = \mathrm{Id}$ and $\Phi_1 = \Phi$. The following theorem shows how the distribution of the samples evolves with time.

**Theorem 1.3.1** (Transport equation). *Let* $Q_t = \Phi_{t\#} Q_0$ *be the distribution of* $x(t)$ *on the Riemannian manifold* $(\mathsf{X}, g)$. *Assume* $Q_t$ *and* $Q_0$ *have densities* $q : [0,1] \times \mathsf{X} \to \mathbb{R}^+$ *and* $q_0 : \mathsf{X} \to \mathbb{R}^+$ *with respect to the volume measure induced by* $g$. *Then* $q$ *satisfies the transport equation*

$$\frac{\partial q}{\partial t} + \mathrm{div}_g\,(v\,q) = 0\,, \tag{1.25}$$

*with initial condition* $q(0, x) = q_0(x)$.

*Proof.* See [72, Proposition 2]. □

The transport equation (1.25) is a continuity equation that describes the conservation of probability mass along the flow generated by the vector field $v$. Note that the divergence operator $\mathrm{div}_g$ depends on the metric $g$ of the manifold $\mathsf{X}$ [72]. The density of the full pushforward $Q$ is then simply given by $q(x) = q(1, x)$.

To generate new samples from $Q$, one simply draws $x_0 \sim Q_0$ and solves the ODE (1.24) forward in time up to $t = 1$. This can be useful when one only needs to sample from $Q$, without computing the corresponding density. As a consequence of Theorem 1.3.1, CNFs also allow one to compute the density of the generated sample $x$ by solving the coupled ODE

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{pmatrix} x \\ \log q \end{pmatrix} = \begin{pmatrix} v\left(t, x(t)\right) \\ -\mathrm{div}_g\big(v\left(t, x(t)\right)\big) \end{pmatrix} , \tag{1.26}$$

with initial condition

$$\begin{pmatrix} x(0) \\ \log q(0, x) \end{pmatrix} = \begin{pmatrix} x_0 \\ \log q_0(x_0) \end{pmatrix} . \tag{1.27}$$

The ODE can also be solved backwards in time, which allows one to compute $x_0$ and $\log q$ from $x$ and therefore to evaluate the density of $Q$ at any point.

In practice, the divergence term $\mathrm{div}_g(v)$ must be evaluated many times during the ODE integration, which can be computationally expensive. To reduce this cost, Grathwohl *et al.* [75] proposed to use Hutchinson's stochastic trace estimator [76].

### 1.3.2 Kullback–Leibler training

Up to this point we only described how to construct a CNF given a vector field $v$, without yet specifying how to choose $v$ to target a specific distribution. To construct a generative model that approximates a given target distribution $P$, one needs to determine a suitable $v$ such that the resulting distribution $Q$ is "close" to $P$, in some sense. A common approach in machine learning is to parametrise $v$ by a set of parameters $\theta \in \mathbb{R}^p$, and to learn these parameters by minimising a loss function that measures the discrepancy between $Q_\theta$ and $P$. Provided that $Q_\theta \ll P$, a natural choice for this loss function is the (direct) Kullback–Leibler (KL) divergence of $P$ with respect to $Q_\theta$,

$$\mathcal{D}_{\mathrm{KL}}\left(P \| Q_\theta\right) := \mathbb{E}_{x \sim P}\left[\log \frac{P(\mathrm{d}x)}{Q_\theta(\mathrm{d}x)}\right] . \tag{1.28}$$

When $P$ and $Q_\theta$ have densities $p$ and $q_\theta$ respectively, the KL divergence can be rewritten as

$$\mathcal{D}_{\mathrm{KL}}\left(P \| Q_\theta\right) = C - \mathbb{E}_{x \sim P}\left[\log q_\theta(x)\right] , \tag{1.29}$$

where $C$ is a constant indepentend of $\theta$. Minimising (1.29) is therefore equivalent to maximising the likelihood of the data under $Q_\theta$, a procedure commonly referred to as *maximum likelihood training*. In practice, the expectation over $P$ can be estimated with Eq. (1.2) using samples from $P$.

This measure is particularly useful when one only has access to data without knowing the explicit form of $P$, which is often the case in generative artificial intelligence for text, audio and images. However, in computational physics and Bayesian inference, the goal is precisely to generate samples from an approximation of $P$ which is known up to a normalising constant but hard to sample. Yet, even in these cases, samples from $P$ may be available from other expensive methods, such as molecular dynamics or Monte Carlo simulations (see Section 2.4). Once properly trained, the NF can then be used to generate new uncorrelated samples from $Q_\theta \approx P$ at a much cheaper cost.

A practical alternative when samples from $P$ are not available is to use the *reverse* KL divergence

$$\mathcal{D}_{\mathrm{KL}}\left(Q_\theta \| P\right) \;\coloneqq\; \mathbb{E}_{x\sim Q_\theta}\left[\log \frac{Q_\theta(\mathrm{d}x)}{P(\mathrm{d}x)}\right] \,, \tag{1.30}$$

provided that $P \ll Q_\theta$ [68]. When $P$ is only known up to a normalising constant, $P = Z^{-1}\tilde{P}$, this expression can be rewritten as

$$\mathcal{D}_{\mathrm{KL}}\left(Q_\theta \| P\right) = \log Z + \mathbb{E}_{x\sim Q_\theta}\left[\log q_\theta(x) - \log \tilde{p}(x)\right] \,. \tag{1.31}$$

Training with reverse KL divergence is sometimes referred to as *energy-based training*. Note that $\log Z$ is a constant independent of $\theta$, so minimising (1.31) is equivalent to minimising the second term, which requires samples from $Q_\theta$ that can be generated by the NF itself. The problem with the expression written in this form is that the expectation depends on $\theta$ through $Q_\theta$. Differentiating with respect to $\theta$ for gradient descent would require to differentiate through the sampling procedure, which is in general not straightforward. We will return to this point in Chapter 3, where the same problem comes up in a different context. Here, we can avoid this issue with the *reparametrisation trick* [77], which allows to rewrite the expectation over $Q_\theta$ as one over the base distribution $Q_0$ using the definition of pushforward:

$$\mathcal{D}_{\mathrm{KL}}\left(Q_\theta \| P\right) = \log Z + \mathbb{E}_{x_0\sim Q_0}[\log q_\theta\left(\Phi(x_0)\right) - \log \tilde{p}\left(\Phi(x_0)\right)] \,. \tag{1.32}$$

Despite its appeal, reverse KL training is known to be *mode-seeking*: it may concentrate on a subset of the modes of $P$ rather than covering its full support, which can limit its effectiveness in complex energy landscapes [78]. When both samples from $P$ and the unnormalised density $\tilde{p}$ are available, one can also consider a hybrid loss function that combines both direct and reverse KL divergences [29].

Although KL training is conceptually simple and has been widely used, its implementation for CNFs can be computationally costly. Each gradient descent step for both Eq. (1.29) and Eq. (1.32) requires evaluating the density $q_\theta$, which in turn involves solving the coupled ODE (1.26) and repeatedly computing the divergence term $\mathrm{div}_g(v)$. This cost grows quickly with the dimension of the state space $\mathsf{X}$, even without considering the number of parameters in $v_\theta$. Moreover, computing gradients of $q_\theta$ requires differentiating through the numerical ODE solver. A naive approach would require storing all intermediate states of the ODE, leading to prohibitive memory costs. A possible solution to this problem is the *adjoint sensitivity method* [79], which computes gradients by integrating an auxiliary ODE backwards in time (see Ref. [54, Appendix B] for a modern derivation). However, this method can suffer from numerical instability and is not always straightforward to implement in practice. These difficulties have motivated the search for alternative training algorithm that do not require explicit evaluation of the model density.

### 1.3.3 Stochastic interpolants

Recently, Lipman *et al.* [30] and Albergo *et al.* [80] independently proposed *flow matching* and *stochastic interpolants* as alternative training methods for CNFs. The two frameworks are closely related and share the same underlying idea: instead of minimising a divergence between $P$ and $Q_\theta$, they directly learn the vector field $v_\theta$ by matching it to a target field $v^\star$ that transports samples from $Q_0$ to $P$. These methods were later generalised to Riemannian manifolds by Chen and Lipman [81] with *Riemannian flow matching*, and by Wu *et al.* [43] with *Riemannian geodesic interpolants*. In what follows, we introduce the main ideas behind these frameworks, focusing on the stochastic interpolants formulation, which we find conceptually more elegant. We focus on the Riemannian setting without the additional noise variable in the interpolation process, which is sufficient for the applications discussed in Chapter 4. For a complete overview of stochastic interpolants (in Euclidean space), we refer the reader to Ref. [31].

The ODE (1.24) defines a *stochastic process* $x(t)$ with law $Q_t$ satisfying the transport equation (1.25). Among all possible velocity fields $v$ that transport $Q_0$ to $Q_1 = Q$, there is an *optimal* one, denoted $v^\star$, that transports $Q_0$ exactly to $Q_1 = P$. The corresponding process $x^\star(t)$, with law $Q^\star_t$, then satisfies $Q^\star_1 = P$. The idea of stochastic interpolants is to construct a *different* stochastic process $X(t)$

whose law $Q_t$ coincides with that of $x^\star(t)$, while being defined in a way that allows expressing the associated velocity field $v^\star$ in Eq. (1.25) as a conditional expectation over the coupling of $Q_0$ and $P$. This additional step allows to approximate $v^\star$ by minimising a simple mean-squared error loss function, without ever evaluating the density of $Q_\theta$. Since the laws of $X(t)$ and $x^\star(t)$ coincide, the learned field $v_\theta$ can then be used in the ODE (1.24) to generate new samples from $Q_\theta \approx P$. We now formalise these ideas.

**Definition 1.3.1.** Given two probability distributions $Q_0$ and $Q_1$ on a Riemannian manifold $(\mathsf{X},\, g)$, a *stochastic interpolant* between $Q_0$ and $Q_1$ is a stochastic process $X : [0{,}1] \times \Omega \to \mathsf{X}$ defined as

$$X(t) = I\left(t, X_0, X_1\right), \quad t \in [0{,}1]\ , \quad (X_0, X_1) \sim \nu\ , \tag{1.33}$$

where $I : [0{,}1] \times \mathsf{X} \times \mathsf{X} \to \mathsf{X}$ is a twice-differentiable measurable map such that $I\left(0, X_0, X_1\right) = X_0$ and $I\left(1, X_0, X_1\right) = X_1$, and where $\nu\left(\mathrm{d}X_0, \mathrm{d}X_1\right)$ is a coupling of $Q_0$ and $Q_1$ satisfying $\nu\left(\mathrm{d}X_0, \mathsf{X}\right) = Q_0\left(\mathrm{d}X_0\right)$ and $\nu\left(\mathsf{X}, \mathrm{d}X_1\right) = Q_1\left(\mathrm{d}X_1\right)$.

In other words, a stochastic interpolant is a process that *bridges* two distributions $Q_0$ and $Q_1$. The pair $(X_0, X_1)$ is drawn from a coupling $\nu$ whose marginals are $Q_0$ and $Q_1$. A simple example is the independent coupling $\nu\left(\mathrm{d}X_0, \mathrm{d}X_1\right) = Q_0\left(\mathrm{d}X_0\right) Q_1\left(\mathrm{d}X_1\right)$. The interpolation path $I$ between $X_0$ and $X_1$ can be chosen arbitrarily. In Euclidean space, a natural choice is the linear interpolation $I\left(t, X_0, X_1\right) = (1-t)X_0 + tX_1$. In practice, it is often beneficial to choose the interpolation path as short as possible [82]. A natural choice on Riemannian manifolds is the geodesic between $X_0$ and $X_1$ (see Section A.2.1). Following Wu *et al.* [43], the corresponding interpolant can be written using the exponential and logarithmic maps (Definition A.2.3) as

$$I\left(t, X_0, X_1\right) = \exp_{X_0}\!\big(t \log_{X_0}(X_1)\big)\ . \tag{1.34}$$

Naturally, the geodesic interpolant reduces to the linear interpolation in Euclidean space.

Note that, unlike the process $x(t)$ defined by Eq. (1.24), the process $X(t)$ in Eq. (1.33) cannot be used directly as a generative model to sample from $Q_1$, since it requires samples from $Q_1$ to be defined in the first place. Nevertheless, the two processes are closely related, as stated by the following theorem.

**Theorem 1.3.2** (Velocity field)**.** *Let $X(t)$ be a stochastic interpolant between $Q_0$ and $Q_1$ as in Definition 1.3.1, and let $Q_t$ denote the law of $X(t)$. Assume that*

*$Q_t$, $Q_0$, and $Q_1$ have densities $q : [0,1] \times \mathsf{X} \to \mathbb{R}^+$, $q_0 : \mathsf{X} \to \mathbb{R}^+$, and $q_1 : \mathsf{X} \to \mathbb{R}^+$ with respect to the volume measure induced by $g$. Then $q$ satisfies the transport equation* (1.25) *with velocity field*

$$v\left(t,x\right) \;=\; \mathbb{E}_{(X_0,X_1)\sim\nu}\left[\frac{\partial I}{\partial t}\left(t,X_0,X_1\right) \mid X(t)=x\right] \,, \tag{1.35}$$

*provided that $v$ is integrable.*

*Proof.* See [43, Theorem 2] □

Intuitively, Eq. (1.35) states that the velocity field at time $t$ and position $x$ is given by the average velocity of all interpolation paths between $X_0$ and $X_1$ that pass through $x$ at time $t$ (see Fig. 1.5).

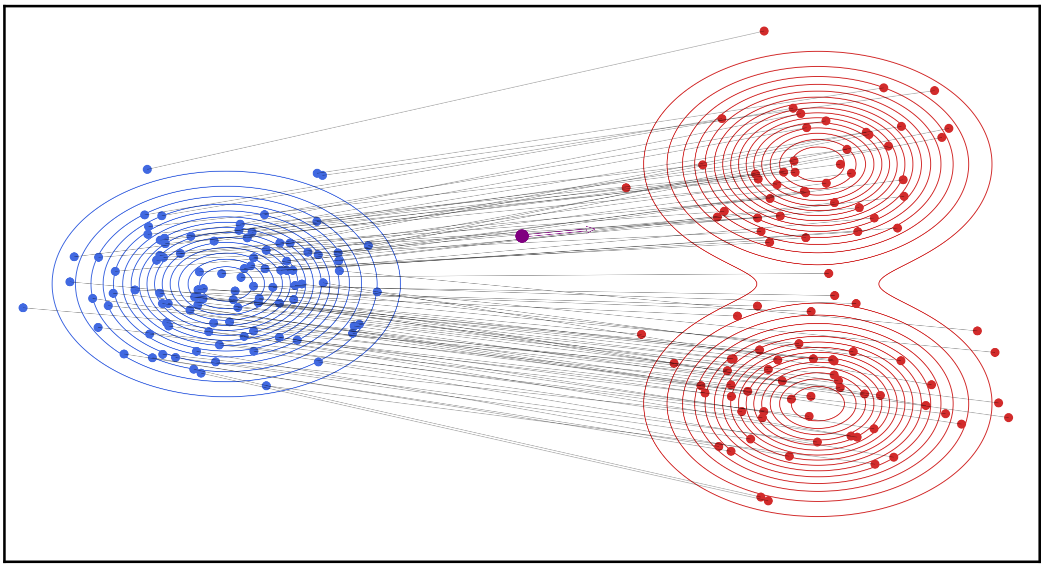

**Figure 1.5:** Stochastic interpolant. Samples $X_0 \sim Q_0$ (blue) are coupled to samples $X_1 \sim Q_1$ (red) and connected by straight interpolation paths $I(t, X_0, X_1)$ (thin black lines). The corresponding velocity field $v(t, x)$ at position $x$ and time $t$ (purple point) is obtained as the average velocity of all interpolation paths passing through $x$ at time $t$ (purple arrow).

Theorem 1.3.2 states that the law of the stochastic interpolant $X(t)$ coincides with that of a CNF $x(t)$ whose velocity field is given by Eq. (1.35). If one has access to samples from $Q_0$ and $Q_1 = P$, the target velocity field $v^\star(t, x)$ can be estimated at any point $x$ and time $t$ by sampling from the coupling $\nu$ and computing the conditional expectation (1.35). The field $v^\star$ can then be used in the ODE (1.24) to generate new samples from $Q_\theta \approx P$. In practice, however, computing the conditional expectation in Eq. (1.35) for every $x$ and $t$ is infeasible. A more practical approach is to learn an approximation $v_\theta$ of $v^\star$ by minimising a mean-squared error loss, as formalised by the following theorem.

**Theorem 1.3.3** (Quadratic loss)**.** *The velocity field $v$ defined in Eq.* (1.35) *is the unique minimiser of the loss function*

$$\mathcal{L}\left[\hat{v}\right] = \int_0^1 \mathbb{E}_{(X_0,X_1)\sim\nu}\left[\frac{1}{2}\left\|\hat{v}(t,x(t))\right\|^2_{x(t)} - g\left(\frac{\partial I}{\partial t}\left(t,X_0,X_1\right),\hat{v}(t,x(t))\right)\right]\mathrm{d}t\,. \tag{1.36}$$

*Proof.* See [31, Theorem 2.7] with the loss function in [43, Equation 12]. □

This loss function can be minimised using stochastic gradient descent with $\hat{v} = v_\theta$, by drawing samples from the coupling $\nu$ and evaluating the time derivative of the interpolant on those samples. Although less direct, this approach avoids any evaluation of the density $q_\theta$ in the training process, making it considerably more efficient than KL training. Once learned, the velocity field $v_\theta$ can be used in the ODE (1.24) to generate new samples from $Q_\theta \approx P$, or in the coupled ODE (1.26) to also evaluate the corresponding density.

### 1.3.4 Incorporating symmetries

Regardless of the training algorithm, CNFs provide a general framework to approximate a target distribution $P$. When the target distribution is complex, however, learning an accurate approximation may require a very expressive vector field $v_\theta$, which corresponds to a large number of parameters and training data. However, this “search” can be made more efficient by incorporating prior knowledge about the target distribution into the CNF architecture. In statistical physics, probability distributions often exhibit *symmetries*, which means that samples related by certain transformations are equally likely. In many systems these symmetries are known *a priori*, and the corresponding transformations that leave $P$ invariant can be explicitly written down. These transformations are usually *isometries* of the underlying space $\mathsf{X}$, *i.e.*, maps that preserve the distance of the corresponding manifold where the system lives. While it is generally not possible to exactly guess the right velocity field $v$ that transports $Q_0$ to $P$, it turns out there is a way to build $v$ so that the resulting CNF distribution $Q$ automatically inherits the symmetries of $P$. This is the idea behind *equivariant flows* developed by Köhler *et al.* [44, 83] and later generalised to Riemannian manifolds by Katsman *et al.* [36]. This framework allows to build CNFs that “search” in a much smaller space of possible distributions, without wasting resources on learning distributions that are surely not compatible with the target one. When applied to physical systems, incorporating symmetries into CNFs can significantly improve both training and sampling efficiency [35, 44, 84]. In this section, we formalise these ideas following Katsman *et al.* [36].

First of all, we need to define what it means for a distribution and a vector field to be "symmetric" with respect to a group of isometries. Let $(\mathsf{X}, g)$ be a Riemannian manifold and let $G$ be a group of isometries acting on $\mathsf{X}$. With a slight abuse of notation, we denote both a group element as $h \in G$ and the action of such element on a point $x \in \mathsf{X}$ as $h(x)$.

**Definition 1.3.2.** A probability measure $P$ on $\mathsf{X}$ is said to be *$G$-invariant* if

$$h_{\#}P = P, \quad \forall\, h \in G\,. \tag{1.37}$$

If $P$ has density $p$ with respect to the volume measure induced by $g$, $P$ is $G$-invariant if $p \circ h = p$ for all $h \in G$.

**Definition 1.3.3.** A diffeomorphism $\Phi : \mathsf{X} \to \mathsf{X}$ is said to be *$G$-equivariant* if $\Phi \circ h = h \circ \Phi$, for all $h \in G$.

**Definition 1.3.4.** A vector field $v : \mathsf{X} \to T\mathsf{X}$ is said to be *$G$-equivariant* if $v \circ h = \mathrm{d}h \circ v$, for all $h \in G$, where $\mathrm{d}h : T\mathsf{X} \to T\mathsf{X}$ is the differential of $h$ (see Definition A.2.4).

In other words, a $G$-invariant measure is unchanged under the action of any group element, while a $G$-equivariant diffeomorphism or vector field commutes with such actions. Using these definitions, the following results show how to construct CNFs that respect the symmetries of a target distribution in two steps.

**Theorem 1.3.4** (Equivariant flow to invariant distribution)**.** *Let $Q_0$ be a probability measure with density $q_0$ on $(\mathsf{X}, g)$ and let $\Phi : \mathsf{X} \to \mathsf{X}$ be a diffeomorphism. If $Q_0$ is $G$-invariant and $\Phi$ is $G$-equivariant, then the pushforward $Q = \Phi_{\#}Q_0$ is also $G$-invariant.*

*Proof.* See [36, Theorem 3]. □

The previous result applies to any normalising flow defined by a diffeomorphism $\Phi$. The next result shows how to construct such an equivariant diffeomorphism for CNFs, which are defined by their time-dependent vector field $v$.

**Theorem 1.3.5** (Equivariant field to equivariant flow)**.** *Let* $v : [0,1] \times \mathsf{X} \to T\mathsf{X}$ *be a time-dependent vector field and let* $\Phi_t : \mathsf{X} \to \mathsf{X}$ *be the corresponding flow defined by the ODE* (1.24)*. If* $v\,(t, \cdot)$ *is* $G$*-equivariant for all* $t \in [0{,}1]$*, then* $\Phi_t$ *is also* $G$*-equivariant for all* $t \in [0{,}1]$*.*

*Proof.* See [36, Theorem 2]. ☐

In practice, to construct a CNF whose distribution $Q$ is $G$-invariant, one needs:

- a $G$-invariant base distribution $Q_0$ (for example, a uniform distribution on $\mathsf{X}$),
- a parameterisation of the velocity field $v_\theta$ that is $G$-equivariant for all $t \in [0{,}1]$.

We will build on these ingredients to develop the equivariant Riemannian stochastic interpolants (eRSI) framework in Chapter 4.

# Chapter 2

# Amorphous materials and glass transition

Amorphous materials are condensed systems that lack the periodic long-range order characteristic of crystals. They are found everywhere in nature and technology, ranging from simple liquids [85] to structural glasses [3], colloids [7, 8], foams [10], and granular materials [11, 12]. These systems are all structurally disordered, yet they can exhibit very different dynamic behaviours. Simple liquids, for instance, are disordered but ergodic: their particles explore configuration space efficiently, and equilibrium statistical mechanics can successfully predict their properties [85]. Amorphous solids, such as structural glasses, are effectively non-ergodic on experimental timescales and display ageing and memory effects, which pose further theoretical challenges [1].

A particularly important class of amorphous materials are *glassy* systems. Broadly speaking, we call a system “glassy” when it combines structural disorder with extremely slow relaxation [86, 87]. When control parameters such as temperature or pressure are varied, the dynamics of glassy systems becomes so slow that equilibration is practically impossible. This behaviour is observed in a wide range of materials in and out of equilibrium, from molecular liquids to colloids and granular media. Despite decades of research, the microscopic origin of this behaviour remains one of the major open problems in condensed matter physics [88].

In this chapter, we review essential concepts from the physics of amorphous materials focusing on the phenomenology of glass-forming liquids and the glass transition [2]. We then discuss how their properties can be studied through computer simulations, introducing some common models used to represent these systems, the physical observables of interest, and the relevant simulation techniques [3].

## 2.1 Phenomenology

When a liquid is cooled from a temperature $T_0$ above its melting point $T_m$, three distinct regimes emerge as the temperature decreases (see Fig. 2.1). For $T > T_m$, the system is an equilibrium liquid, whose thermodynamic properties are described by standard statistical mechanics. Below $T_m$, the thermodynamically stable phase is the crystal. However, if the cooling rate is high enough, crystallisation can often be avoided. In this case, the liquid remains disordered and enters a metastable state known as a *supercooled liquid*. In the supercooled regime, the static thermodynamic quantities, such as energy or enthalpy, follow their equilibrium liquid extrapolation (see Fig. 2.1). The dynamics, however, change dramatically: the viscosity and the structural relaxation time increase by many orders of magnitude upon cooling. In this regime, the liquid is said to exhibit *glassy dynamics*. At sufficiently low temperatures, around the *glass transition temperature* $T_g$, the system's relaxation time becomes so large that equilibrium can no longer be achieved on experimental timescales and the system behaves as an amorphous solid: we have obtained a *glass*. Below $T_g$, the system continues to slowly relax towards equilibrium, and its macroscopic properties depend on its thermal history. This phenomenon is known as *ageing*. Note that $T_g$ does not coincide with a true thermodynamic phase transition: it marks a crossover that depends on the cooling rate, with faster cooling leading to higher $T_g$ values.

### 2.1.1 Kinetic fragility

The dramatic slowing down observed in liquids that can be supercooled without crystallising, commonly referred to as *glass-forming liquids*, can be quantified by measuring their structural relaxation time $\tau_\alpha$ as a function of temperature. Such measurements can be performed, for instance, with light scattering, dielectric spectroscopy, and rheology. A typical way to represent the results is the Angell plot [4, 90], which shows $\tau_\alpha$ versus the inverse temperature $1/T$ normalised by $T_g$ (see Fig. 2.2). In this plot, glass-forming liquids can be broadly classified into two categories. *Strong* liquids exhibit an Arrhenius temperature dependence,

$$\tau_\alpha = \tau_0 \exp\left(\frac{E_a}{k_B T}\right) , \tag{2.1}$$

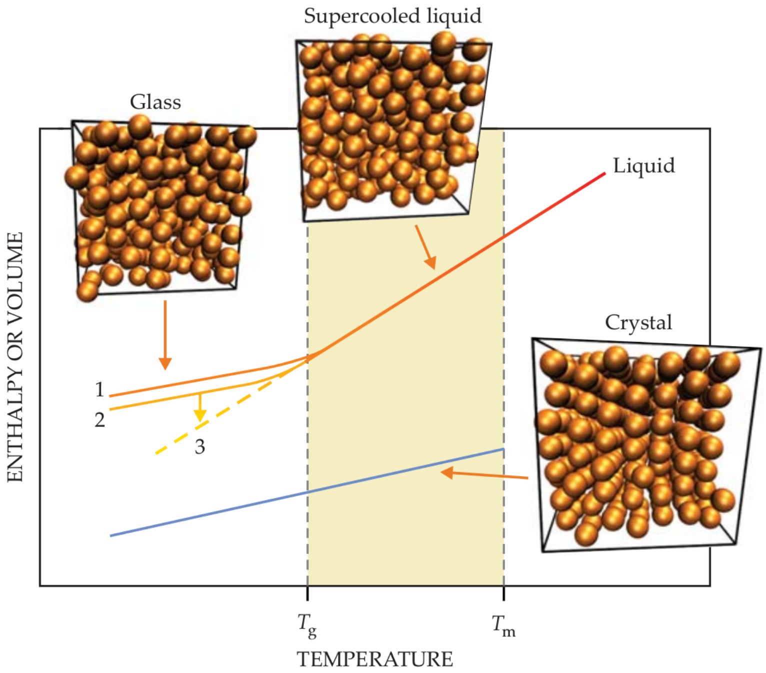

**Figure 2.1:** Enthalpy–temperature diagram of a glass-forming liquid. Above the melting temperature $T_m$, the system is a liquid, while below $T_m$ the crystal is the thermodynamically stable phase. Upon sufficiently fast cooling, crystallisation can be avoided and the liquid enters a metastable supercooled regime (shaded area). With further cooling, the system falls out of equilibrium around the glass transition temperature $T_g$ and becomes an amorphous solid (glass). Slower cooling rates produce glasses with lower enthalpy. Reprinted from Ref. [89].

where $E_a$ is a constant activation energy, $k_B$ the Boltzmann constant, and $\tau_0$ a microscopic timescale. Their slowing down arises from local, thermally activated processes. In contrast, *fragile* liquids display a super-Arrhenius increase of the relaxation time, often described by empirical forms such as the Vogel–Fulcher–Tammann (VFT) law,

$$\tau_\alpha = \tau_0 \exp\left(\frac{DT_0}{T - T_{\mathrm{VFT}}}\right) , \tag{2.2}$$

where $D > 0$ is a dimensionless parameter and $T_{\mathrm{VFT}}$ is the temperature at which the extrapolated relaxation time would diverge. Within this phenomenology, fragile liquids behave as if their effective activation energy increases upon cooling. This observation can be explained by the growing importance of cooperative relaxation processes, although the precise microscopic origin of this behaviour remains the subject of ongoing debate, with several theoretical scenarios having been proposed [1, 91–96]. On a practical level, this enormous increase in relaxation times makes it extremely difficult to probe the equilibrium properties of glass-forming liquids at low temperatures. Direct experimental access to the hypothetical thermodynamic transition is therefore impossible in practice, as the equilibration times exceed any realistic observation time.

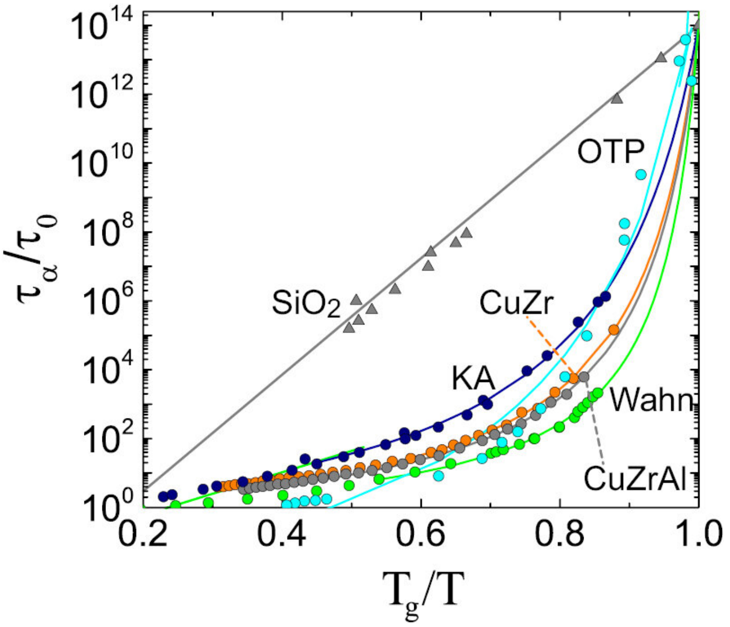

**Figure 2.2:** Angell plot. Structural relaxation time $\tau_\alpha$ versus inverse temperature $1/T$ normalised by $T_g$ for several glass-forming liquids. Strong liquids (*e.g.* $SiO_2$) show Arrhenius behaviour, appearing as straight lines, while fragile liquids (*e.g.* CuZr) exhibit a super-Arrhenius increase of $\tau_\alpha$. Reprinted from Ref. [97].

### 2.1.2 The onset temperature

Up to this point, we have remained somewhat vague about the precise temperature at which a liquid begins to display glassy dynamics. Such temperature is commonly referred to as the *onset temperature* $T_o$ [13]. Below $T_o$, the liquid no longer behaves as a simple fluid: structural relaxation becomes increasingly sluggish, and the relaxation time $\tau_\alpha$ follows a super-Arrhenius temperature dependence in fragile liquids. Naively, one may expect $T_o$ to be around the melting temperature $T_m$. In this section, we address the delicate question of whether the onset of glassy dynamics is actually related to thermodynamic metastability.

Before doing so, a technical clarification is needed. Glass-forming liquids are often multicomponent mixtures containing particles of different types. For such systems, the temperature at which crystallisation begins upon cooling (the *freezing* temperature $T_f$) and the temperature at which melting starts upon heating (the *melting* temperature $T_m$) generally differ, with $T_f \geq T_m$ [98]. In monocomponent systems, these two temperatures coincide. Below $T_f$, the liquid phase is metastable with respect to the crystal (in monocomponent systems), or to coexistence between crystal and liquid (in multicomponent systems). A connection between the onset temperature and metastability has then to be investigated through the relation between $T_o$ and $T_f$. Several theoretical approaches have explored a possible connection between $T_o$ and $T_f$ (often improperly identified with $T_m$) [99, 100]. A correlation between the two would support–although not prove–the idea that the emergence of slow dynamics is somehow tied to the existence of a thermodynamically more stable phase. However, metastability is not a necessary condition to observe glassy dynamics: for instance, in liquid silica one finds $T_o > T_m$ [101]. Nonetheless, in several models and experiments, $T_o$ and $T_m$ appear to follow similar trends [102, 103].

In our recent work [104], we addressed this issue through numerical simulations of a model glass-former [105], starting from a precise definition of $T_o$ (see Section 2.3.2). Our results demonstrated that $T_o$ is not directly related to the freezing temperature $T_f$: a liquid can be metastable without being glassy (see Fig. 2.3). Interestingly, however, our data also show that $T_o$ roughly follows the melting line $T_m$. Whether this correspondence reflects a physical connection or is merely coincidental remains unclear at present. A possible way to clarify this issue would be to investigate the presence of crystalline precursors in the metastable liquid, which remains an open direction for future research.

### 2.1.3 Stability

As discussed above, not all glasses are equal: their properties depend sensitively on their preparation history. Glasses formed by slower cooling reach lower energy (or enthalpy) at fixed temperature and pressure and are typically characterised by a lower fictive temperature, *i.e.* the temperature at which the glass structure would be in equilibrium, often associated with a lower apparent glass transition temperature $T_g$. These glasses are more *stable* in the sense that they are more resistant to melting: when heated, a more stable glass melts at a higher temperature, compared to a less stable one. This enhanced stability has been shown to originate from a density difference between different glasses [107]: denser glasses must expand in order for the growing fluid phase to fill up the space, which slows down the melting process.

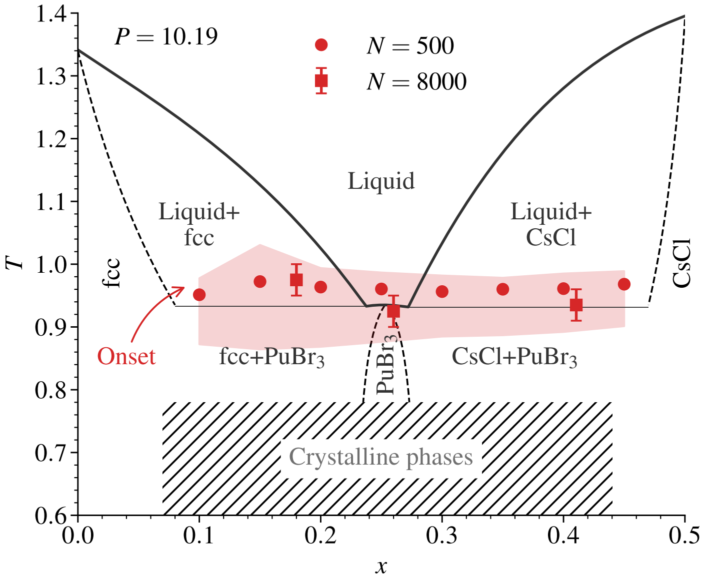

**Figure 2.3:** Temperature–composition diagram of a model glass-former [105]. The onset temperature $T_o$ (red symbols) is compared to the freezing (thick solid line) and melting (thin solid line) temperatures as a function of composition, inferred from Ref. [106]. Reprinted from Ref. [104].

Glass stability can be measured by slowly heating a glass and monitoring its enthalpy (or volume) as it transforms into the liquid state. The relation between enthalpy and temperature in equilibrium is known as the *equation of state* (EOS). Below the glass transition, the system falls out of equilibrium, yet one can still measure an effective relation between enthalpy and temperature, the *glass equation of state*. Unlike the equilibrium EOS, the glass EOS depends on the preparation protocol: better aged glasses lie at lower enthalpies and follow a distinct EOS. When heated, the glass EOS eventually merges with the liquid EOS at a temperature that depends on the glass stability (see Fig. 2.4).

Waiting for long times will eventually produce more stable glasses, but it is a slow and inefficient process. In physical vapour deposition [108], molecules are slowly deposited onto a substrate maintained at constant temperature. The resulting glasses can be extremely stable, with properties equivalent to those expected for glasses aged for thousands of years. For this reason, vapour-deposited glasses are often called *ultrastable* glasses.

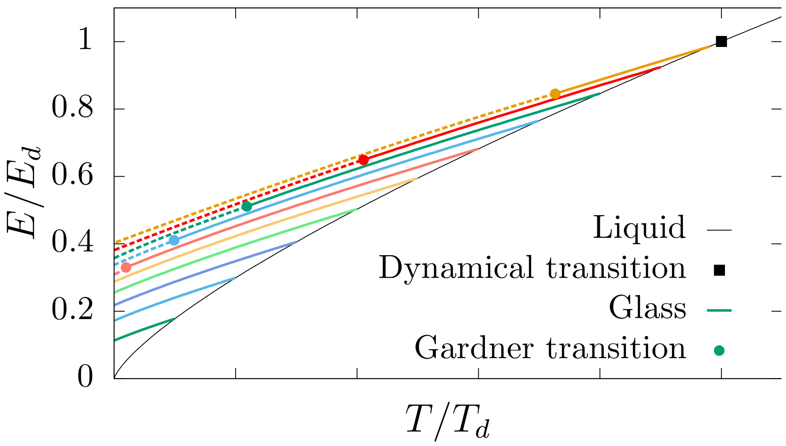

**Figure 2.4:** Mean-field phase diagram of a model glass-former. The black line denotes the equilibrium liquid equation of state, while the black square marks the dynamical glass transition temperature $T_d$. Coloured lines represent glass equations of state associated with different preparation temperatures. Adapted from Ref. [109].

## 2.2 Modelling amorphous materials

Both analytical approaches and computer simulations of amorphous materials rely on *physical models* capturing the essential ingredients of the real systems, while remaining simple enough to handle computationally [85, 110]. Classical models describe these systems as a collection of $N$ interacting particles kept at constant temperature $T$ and constant volume $V$ ($NVT$ ensemble) or constant pressure $P$ ($NPT$ ensemble) by appropriate reservoirs. Knowing the potential energy $U$ of the system, equilibrium statistical mechanics provides a framework to compute its thermodynamic properties. In the $NVT$ ensemble, the probability of a microstate $x$ is given by the *Boltzmann distribution*

$$P\left(\mathrm{d}x\right) = \frac{1}{Z}\exp\left(-\frac{U(x)}{k_B T}\right)\mathrm{d}x\ , \tag{2.3}$$

where $x$ represents the state of the system, $\mathrm{d}x$ is a reference measure, $k_B$ is the Boltzmann constant and $Z$ is the partition function ensuring proper normalisation. In the $NPT$ ensemble, the potential energy is replaced by the enthalpy $H(x,P) = U(x) + PV(x)$, where $V(x)$ is the volume of the system in state $x$. Note that $x$ is an abstract representation of the state, which typically includes the positions $\boldsymbol{r} = (\vec{r}_1, \ldots, \vec{r}_N)$ of the particles, as well as their chemical species $\boldsymbol{s} = (s_1, \ldots, s_N)$. The particle momenta are usually integrated out, so they do not appear explicitly in the configurational description [85]. The model of an amorphous material is therefore fully specified by the choice of the state space $\mathsf{X}$ and the potential energy function $U : \mathsf{X} \to \mathbb{R}$.

### 2.2.1 Periodic boundary conditions

The aim of many particle-based simulations of condensed-matter systems is to describe properties in the thermodynamic limit, corresponding to a very large number of particles, $N \approx 10^{23}$. In practice, however, computational resources restrict simulations to much smaller systems, typically with $N \approx 10^3$–$10^6$ particles. To minimise surface effects and mimic bulk behaviour, simulations commonly employ periodic boundary conditions (PBCs) [111]. Under PBCs, the simulation box is periodically replicated along each Cartesian direction, so that when a particle leaves the box through one face, it re-enters through the opposite face. This approach does not remove finite-size effects completely, but effectively removes surface effects and allows one to estimate bulk properties with a finite number of particles.

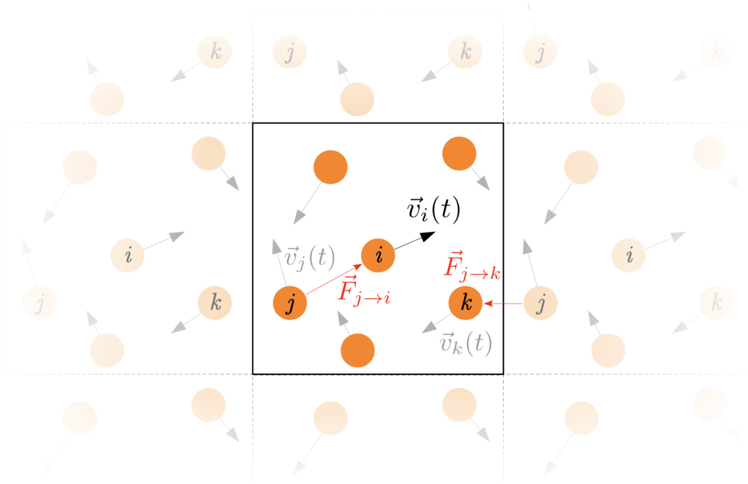

**Figure 2.5:** Periodic boundary conditions. Particles are enclosed in a simulation box that is periodically replicated along each Cartesian direction. Reprinted from Ref. [112].

Mathematically, modelling a system with PBC corresponds to embedding it on a *flat torus*, which is a Riemannian manifold defined as the quotient space $\mathcal{M}_L = \mathbb{R}^d / L\mathbb{Z}^d$, where $L$ is the side of the box and $d$ is the spatial dimension (see Appendix A.2). The corresponding distance between two particles $i$ and $j$ is the *nearest image distance*

$$d_{\mathcal{M}_L}\left(\vec{r}_i, \vec{r}_j\right) = \min_{\vec{k}\in\mathbb{Z}^d} \left\| \vec{r}_i - \vec{r}_j + L\vec{k} \right\| \ . \tag{2.4}$$

When considering $N$ particles in a $d$-dimensional box with PBC, the state space of their positions is then $\mathcal{M}_L^N$.

### 2.2.2 Constraining the composition

Models of amorphous materials often include multiple chemical species to prevent crystallisation and improve glass-forming ability. We represent the set of species as a finite set $\mathcal{S}$, typically $\mathcal{S} = \{1, \ldots, S\}$, where $S$ is the number of species. In practice, $S$ usually takes values between one and three, while for polydisperse systems $S = N$. The full state space of the system is then the *product manifold* $\mathsf{X} = \mathcal{S}^N \times \mathcal{M}_L^N$. We denote a configuration as $x = (\boldsymbol{s}, \boldsymbol{r}) \in \mathsf{X}$, where $\boldsymbol{s} = (s_1, \ldots, s_N) \in \mathcal{S}^N$ specifies the chemical species and $\boldsymbol{r} = (\vec{r}_1, \ldots, \vec{r}_N) \in \mathcal{M}_L^N$ the particle positions.

If the species are allowed to vary freely–say, at fixed total volume–the simulation corresponds to a semi-grand-canonical ensemble, in which the number of particles of each species can fluctuate [110]. In most cases, however, one is interested in systems with a fixed *composition* $c = \{N_s\}_{s\in\mathcal{S}}$, where the number of particles $N_s$ of each species is constrained. Particle species can still change, but the total number of particles of each species remains constant. The corresponding Boltzmann distribution is then

$$P_c\left(\mathrm{d}\boldsymbol{s}, \mathrm{d}\boldsymbol{r}\right) = \frac{1}{Z_c} \exp\left(-\frac{U\left(\boldsymbol{s}, \boldsymbol{r}\right)}{k_B T}\right) \left(\prod_{\alpha\in\mathcal{S}} \delta_{\sum_i \delta_{\alpha(s_i)}}\left(N_\alpha\right)\right) \tau(\mathrm{d}\boldsymbol{s})\, \mathrm{d}\boldsymbol{r} \ , \tag{2.5}$$

where $\tau(\mathrm{d}\boldsymbol{s})$ is the counting measure and the product of Kronecker deltas enforces the composition constraint by fixing the number of particles of each species $\delta_{\sum_i \delta_{\alpha(s_i)}}$. Note that Eq. (2.5) can still be written as proportional to an exponential by taking the logarithm of the product of Kronecker deltas.

### 2.2.3 Interaction models

Once the state space is defined, the next step in building the model is to specify the potential energy function $U$ and the composition $c$. In real materials, the potential energy arises from interactions among all particles, including many-body contributions. Dealing with such interactions in a simulation of glassy dynamics is computationally difficult, so most models approximate the energy by a sum of *pairwise* interactions that depend only on the distance between two particles and their species

$$U(x) = \sum_{i<j} u\left(s_i, s_j, d_{\mathcal{M}_L}\left(\vec{r}_i, \vec{r}_j\right)\right)\ , \tag{2.6}$$

where $u$ is the pair potential between particles $i$ and $j$. Once a pair potential is chosen, the remaining decisions concern the particle composition and the functional form of $u$. Below, we describe several commonly used models for studying amorphous materials and glassy dynamics, focusing on those relevant to this thesis.

### Hard spheres

The hard-sphere model is one of the simplest and most efficient models for simulating simple liquids [85]. Particles are represented as impenetrable spheres of diameter $\sigma_{s_i}$ that interact only via excluded-volume. The diameter $\sigma_{s_i}$ is determined by the species $s_i$ of particle $i$. A common choice is a binary mixture with diameters $\sigma_1$ and $\sigma_2$, and a size ratio $\sigma_2/\sigma_1 = 1.4$. The pair potential at nearest image distance $r$ is

$$u\left(s_i, s_j, r\right) = \begin{cases} \infty, & r < \sigma_{s_i s_j} \\ 0, & r \geq \sigma_{s_i s_j} \end{cases}\ , \tag{2.7}$$

where $\sigma_{s_i s_j} = \left(\sigma_{s_i} + \sigma_{s_j}\right)/2$ is the contact distance between particles $i$ and $j$.

### Harmonic spheres

The harmonic-sphere model is a *soft* generalisation of hard spheres, where particles are allowed to overlap at a finite energy cost [113, 114]. The interaction potential is again defined through the species-dependent diameters $\sigma_{s_i}$ and through a stiffness parameter $\epsilon > 0$:

$$u\left(s_i, s_j, r\right) = \begin{cases} \frac{\epsilon}{2}\left(1 - \frac{r}{\sigma_{s_i s_j}}\right)^2, & r < \sigma_{s_i s_j} \\ 0, & r \geq \sigma_{s_i s_j} \end{cases}\ . \tag{2.8}$$

### Inverse power law

The inverse-power-law (IPL) model is another soft-sphere system, in which particles interact via a purely repulsive potential that decays as a power law [115, 116]. In this model, particles do not have well-defined diameters. Instead, their interactions are governed by a matrix of characteristic length scales $\sigma_{s_i s_j}$, an energy scale $\epsilon > 0$, and an integer exponent $p$. The pair potential reads

$$u\left(s_i, s_j, r\right) = \begin{cases} \epsilon \left(\frac{\sigma_{s_i s_j}}{r}\right)^p + C_0, & r < r^c_{s_i s_j} \\ 0, & r \geq r^c_{s_i s_j} \end{cases}, \tag{2.9}$$

where $r^c_{s_i s_j}$ is a species-dependent cutoff distance, and $C_0$ is a shift constant ensuring continuity at $r^c$. A common choice is $p = 12$, which reproduces the repulsive part of the Lennard-Jones potential.

### Kob-Andersen

The Kob–Andersen (KA) model extends the standard Lennard-Jones potential to binary mixtures and was designed to suppress crystallisation [105]. The potential depends on the matrices of energy scales $\epsilon_{s_i s_j}$ and length scales $\sigma_{s_i s_j}$. The pair potential is

$$u\left(s_i, s_j, r\right) = \begin{cases} \epsilon_{s_i s_j} \left(\left(\frac{\sigma_{s_i s_j}}{r}\right)^{12} - \left(\frac{\sigma_{s_i s_j}}{r}\right)^6\right) + C_0, & r < r^c_{s_i s_j} \\ 0, & r \geq r^c_{s_i s_j} \end{cases}, \tag{2.10}$$

where the cutoff distance $r^c_{s_i s_j}$ is typically set to $2.5\sigma_{s_i s_j}$, and $C_0$ is again a shift constant ensuring continuity at $r^c$. In some implementations, additional terms are introduced to make the first and second derivatives continuous at the cutoff.

## 2.2.4 Symmetries

Since the potential energy in Eq. (2.6) depends only on the particle species and interparticle distances, the Boltzmann distribution in Eq. (2.5) is *invariant* under isometries, *i.e.* transformations that preserve the distance $d_{\mathcal{M}_L}$ (see Definition 1.3.2). In other words, certain transformations of the system leave the Hamiltonian–and therefore the distribution–unchanged. As a consequence, configurations related by such transformations have the same probability. Identifying the *symmetry group* of a given distribution is particularly important when designing algorithms to approximate it efficiently (see Section 1.3.4). For instance, as discussed in Section 1.3.4, normalising flows can be designed to respect the symmetries of the target distribution, which reduces the space of possible distributions and significantly improves efficiency. In Appendix C.2, we show that both distributions in Eq. (2.3) and Eq. (2.5) are invariant under the following transformations:

- **Particle permutations**, which simultaneously permute species and positions.

- **Translations**, which shift all coordinates by the same vector, wrapping positions back into the simulation box.
- **Box symmetries**, which combine axis permutations and sign flips along the coordinate axes.

These invariances are illustrated in Fig. 2.6. Note that while particle systems in Euclidean space, such as single molecules or confined clusters, may exhibit full rotational invariance, systems under PBCs keep only the discrete symmetries of the simulation box. For sufficiently large systems with interaction cutoff $r^{c} < L/2$, however, rotational invariance is effectively recovered at the level of the Hamiltonian, since particle pairs cannot feel the box boundaries.

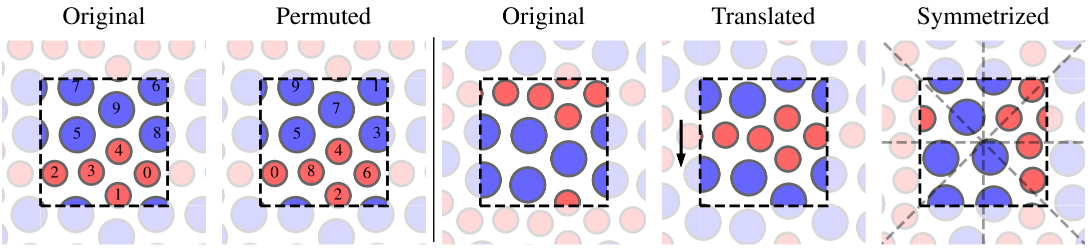

**Figure 2.6:** Symmetries of particle systems under periodic boundary conditions. Reprinted from Ref. [45].

## 2.3 Relevant observables

We now introduce several physical quantities that are commonly used to characterise amorphous materials [85, 110, 111]. These observables will also serve as benchmarks to assess the performance and physical accuracy of the methods discussed in this thesis. Broadly speaking, we distinguish between *static* and *dynamical* quantities. Static observables depend on ensemble averages over individual configurations, whereas dynamical observables measure correlations between configurations separated by a time interval. As a result, dynamical observables generally depend on the choice of microscopic dynamics used to generate the trajectories.

### 2.3.1 Static quantities

Static quantities describe the *structure* of amorphous materials. They provide information on the spatial organisation of particles and can also serve as metrics to compare configurations generated by different algorithms with reference data.

### Radial distribution function

The radial distribution function $g(r)$ quantifies correlations of the density profile

$$\rho(\vec{r}) \;\coloneqq\; \sum_i \delta\left(\vec{r} - \vec{r}_i\right) . \tag{2.11}$$

For a system of $N$ particles in $d$ dimensions and number density $\rho$, it is defined as

$$g(r) \;\coloneqq\; \frac{1}{N\rho S_{d-1}(r)}\, \mathbb{E}\left[\sum_i \sum_{j\neq i} \delta\big(r - d_{\mathcal{M}_L}(\vec{r}_i, \vec{r}_j)\big)\right] , \tag{2.12}$$

where $S_{d-1}(r)$ is the surface area of the $(d-1)$-dimensional sphere of radius $r$, and $d(\vec{r}_i, \vec{r}_j)$ denotes the appropriate distance between particles $i$ and $j$. In amorphous systems, $g(r)$ exhibits damped oscillations that converge to 1 at large distances, reflecting the absence of long-range order (see Fig. 2.7(a)). In crystals, $g(r)$ shows sharp peaks at lattice spacings, before converging to 1 at large distances due to spherical averaging.

For discrete multicomponent mixtures, it is useful to introduce the partial radial distribution function $g_{\alpha\beta}$ which measures correlations between particles of species $\alpha$ and $\beta$:

$$g_{\alpha\beta}(r) \;\coloneqq\; \frac{1}{N\rho x_\alpha x_\beta S_{d-1}(r)}\, \mathbb{E}\left[\sum_{i\in\alpha} \sum_{\substack{j\in\beta \\ j\neq i}} \delta\left(r - d_{\mathcal{M}_L}\left(\vec{r}_i, \vec{r}_j\right)\right)\right] , \tag{2.13}$$

where $x_\alpha$ denotes the fraction of particles of species $\alpha$.

### Structure factors and hyperuniformity

The structure factor $S(q)$ contains essentially the same information as $g(r)$, but expressed in Fourier space. It is directly accessible in experiments by measuring the scattering of probe particles in the sample. $S(q)$ quantifies density correlations at a given wavevector of magnitude $q$ of the Fourier components of the microscopic density field

$$\rho(\vec{q}) \;\coloneqq\; \sum_i \exp\left(-\mathrm{i}\, \vec{q} \cdot \vec{r}_i\right) . \tag{2.14}$$

It is defined as

$$S(q) \coloneqq \frac{1}{N}\, \mathbb{E}\left[\rho\left(\vec{q}\right) \rho\left(-\vec{q}\right)\right] \tag{2.15}$$

$$= \frac{1}{N}\, \mathbb{E}\left[\sum_i \sum_j \exp\big(\mathrm{i}\, \vec{q} \cdot \left(\vec{r}_i - \vec{r}_j\right)\big)\right] , \tag{2.16}$$

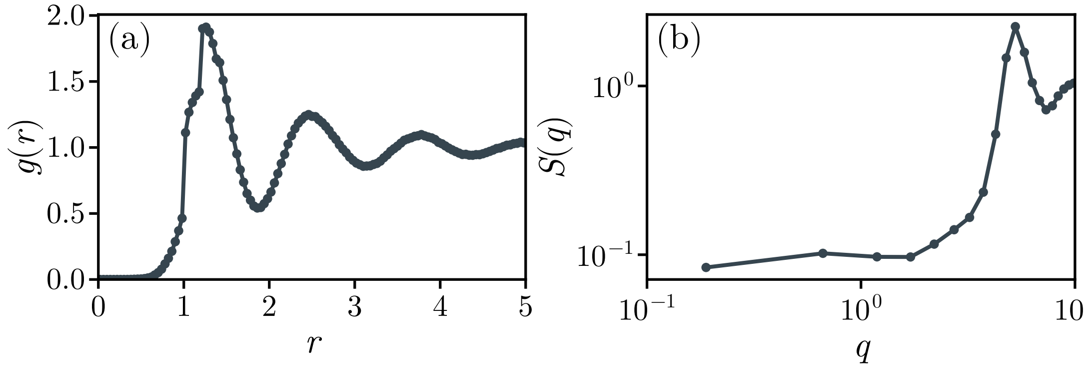

**Figure 2.7:** Structural characterisation of a two-dimensional multicomponent particle system. (a) Radial distribution function $g(r)$, showing the probability of finding a particle at a distance $r$ from a reference particle. (b) Structure factor $S(q)$, reflecting the Fourier-space correlations of particle positions.

and it is a real number by symmetry of the sum.

The structure factor is useful to characterise *hyperuniformity* [46, 117], a structural property characterised by an anomalous suppression of density fluctuations at large scales [46, 117]. For equilibrium systems, the variance of the number of particles in a spherical observation window of radius $R$ scales as the volume of the window, $\mathrm{Var}\left(N_R\right) \sim R^d$. This directly translates into a positive constant structure factor at small wavevectors $q \to 0$. In hyperuniform systems, density fluctuations are suppressed, leading to a scaling of the variance slower than the volume, $\mathrm{Var}\left(N_R\right) \sim R^{d-\alpha}$, and to a structure factor that vanishes at small wavevectors as $S(q) \sim q^\alpha$, with $\alpha > 0$ (see Fig. 2.8). Hyperuniformity is observed in a variety of systems. The simplest example is the perfect crystal, in which density fluctuations scale with the boundary of the spherical region, *i.e.* $\alpha = 1$ [46]. Other examples include jammed packings [46, 118, 119], biased ensembles [120], driven particle models [121–126], and biological systems [127].

As with $g(r)$, one can define partial structure factors $S_{\alpha\beta}(q)$ for multicomponent systems:

$$S_{\alpha\beta}(q) \ \coloneqq \ \frac{1}{N}\,\mathbb{E}\left[\sum_{i\in\alpha}\sum_{j\in\beta}\exp\left(\mathrm{i}\,\vec{q}\cdot(\vec{r}_i-\vec{r}_j)\right)\right] \ . \tag{2.17}$$

In kinetically arrested systems, *i.e.* systems in which structural relaxation is effectively frozen on the timescales of observation, such as crystals and glasses, density fluctuations can be decomposed into $\rho\left(\vec{q}\right) = \mathbb{E}\left[\rho\left(\vec{q}\right)\right] - \delta\rho\left(\vec{q}\right)$, *i.e.* a "backbone" contribution coming from the average density profile, and a fluctuating part $\delta\rho\left(\vec{q}\right)$ around the average profile. The structure factor can then be decomposed

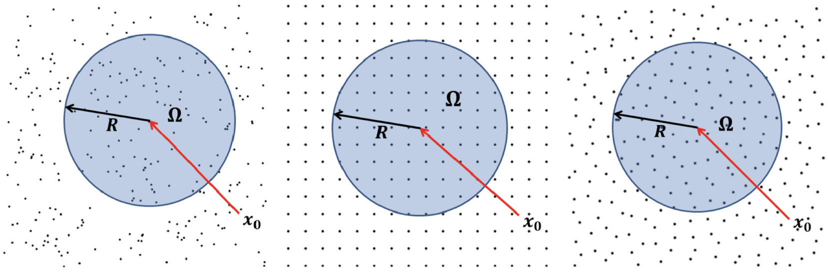

**Figure 2.8:** Scaling of number fluctuations within a disk of increasing radius. For an ideal gas (left), the variance of the number of points scales proportionally to the disk area. In hyperuniform systems (right), number fluctuations grow more slowly than the area. Crystalline systems (centre) are a particular case of hyperuniform systems where fluctuations scale with the disk perimeter. Reprinted from Ref. [117].

as $S(q) = S_0(q) + S_\delta(q)$ [128], where

$$S_0(q) \coloneqq \frac{1}{N}\,\mathbb{E}\left[\rho\left(\vec{q}\right)\right]\mathbb{E}\left[\rho\left(-\vec{q}\right)\right] \tag{2.18}$$

$$S_\delta(q) \coloneqq \frac{1}{N}\,\mathbb{E}\left[\delta\rho\left(\vec{q}\right)\delta\rho\left(-\vec{q}\right)\right] \; . \tag{2.19}$$

In liquids, $S_0(q) = 0$ by translational invariance, while in arrested states it captures the fluctuations of the averaged particle positions. In crystals, $S_0$ has Bragg peaks at reciprocal lattice vectors $\vec{G}$ [129], while in glasses it produces an amorphous background [130].

For binary mixtures, it is often useful to analyse concentration fluctuations separately from total density fluctuations. This is captured by the *concentration–concentration structure factor* [131],

$$S_{CC}(q) \; \coloneqq \; x_{\mathrm{B}}^2\, S_{\mathrm{AA}}(q) + x_{\mathrm{A}}^2\, S_{\mathrm{BB}}(q) - 2x_{\mathrm{A}}x_{\mathrm{B}}\, S_{\mathrm{AB}}(q) \; , \tag{2.20}$$

which isolates spatial variations in the local composition. While $S(q)$ probes density correlations, $S_{CC}(q)$ measures how the relative abundance of the two species fluctuates across different length scales, making it a sensitive indicator of emerging chemical ordering and phase separation.

In multicomponent systems, the total structure factor does not distinguish by itself between number density and composition fluctuations. To isolate fluctuations of the total density field it is convenient to combine the partial structure factors into a single quantity known as the *q-dependent compressibility* [119]. In the simple case of a binary mixture with species A and B, it reads

$$\chi(q) \;=\; \frac{S_{\mathrm{AA}}(q)S_{\mathrm{BB}}(q) - S^2_{\mathrm{AB}}(q)}{x^2_{\mathrm{A}}S_{\mathrm{BB}}(q) + x^2_{\mathrm{B}}S_{\mathrm{AA}}(q) - 2x_{\mathrm{A}}x_{\mathrm{B}}S_{\mathrm{AB}}(q)}\;. \tag{2.21}$$

At small $q$, $\chi(q)$ measures large-scale density fluctuations and is directly related to the isothermal compressibility in the limit $q \to 0$. It is particularly useful for identifying hyperuniformity in multicomponent systems, where density fluctuations vanish at large length scales as $\chi(q) \sim q^{\alpha}$, while $S(q)$ remains finite due to composition fluctuations [119]. As with $S(q)$, one can decompose $\chi$ into a backbone and a fluctuating part as $\chi(q) = \chi_0(q) + \chi_\delta(q)$.

### Local order parameter

Tong and Tanaka [132] introduced a local order parameter $\Theta$ to quantify the degree of local structural order in glass-forming liquids. It quantifies the deviation of local configurations from an ideal, densely packed arrangement. The construction relies on a Voronoi tessellation to identify neighbours of each particle $o$. For each pair of neighbouring particles $(i, j)$ of $o$ that are also neighbours of each other, the triplet $(o, i, j)$ defines a Delaunay triangle. Let $\theta^1_{ij}$ be the angle at vertex $o$ between $\vec{r}_{oi}$ and $\vec{r}_{oj}$ in the actual configuration (see Fig. 2.9(a)), and $\theta^2_{ij}$ the corresponding angle in an ideal packing of hard spheres (see Fig. 2.9(b)). The per-particle $\Theta_o$ is defined as

$$\Theta_o \;:=\; \frac{1}{N_o}\sum_{\langle i,j\rangle} \left|\theta^1_{ij} - \theta^2_{ij}\right| \;, \tag{2.22}$$

where $\langle i, j\rangle$ denotes neighbouring pairs of neighbours of $o$, and $N_o$ their total number. The global order parameter is obtained by averaging over all particles:

$$\Theta \;:=\; \frac{1}{N}\sum_{o} \Theta_o \;. \tag{2.23}$$

A smaller $\Theta$ indicates more locally ordered packings.

## 2.3.2 Dynamical quantities

Dynamical observables probe the evolution of configurations over time and are particularly useful to characterise the performance of an algorithm. They are expressed as *two-time* correlation functions that involve pairs of configurations separated by a time interval $t$. Unlike static quantities, their behaviour depends in general on the underlying dynamics. Although one may think that the only

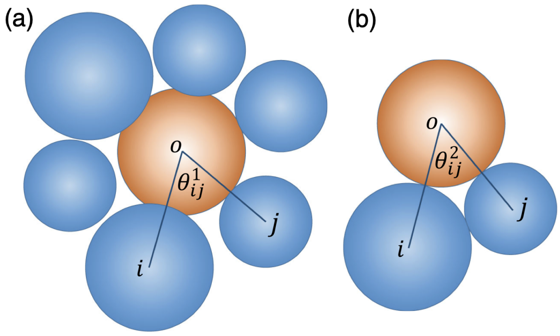

**Figure 2.9:** Definition of the local order parameter $\Theta$. (a) Local configuration of a central particle $o$ (orange) and its neighbours (blue). (b) Ideal configuration with three particles in contact. Reprinted from Ref. [132].

physical dynamics are those derived by Newton's equations (see Section 2.4.1), one can still define correlation functions for other sampling method such as Monte Carlo, even if the corresponding dynamics are unphysical (see Section 2.4.2). If an algorithm generates configurations that decorrelate faster according to two-time correlation functions, it explores configuration space more efficiently.

### Mean square displacement

The simplest and most common dynamical observable is the mean-square displacement (MSD), which quantifies how far particles move, on average, over a time interval $t$. It is defined as

$$\Delta^2(t) \coloneqq \frac{1}{N}\,\mathbb{E}\left[\sum_i \|\vec{r}_i(t) - \vec{r}_i(0)\|^2\right] . \tag{2.24}$$

Note that the distance is Euclidean rather than the nearest-image distance, since the goal is to measure the distance travelled by each particle. The behaviour of the MSD as a function of time reveals important information about the system's dynamics. For Newtonian dynamics, particles move ballistically at very short times, leading to a quadratic growth $\Delta^2(t) \sim t^2$. At long times, diffusion dominates, corresponding to $\Delta^2(t) \sim t$. In glassy systems, an intermediate plateau appears as particles become temporarily trapped in cages formed by their neighbours. The plateau grows as the system approaches the glass transition and, at $T_g$, it becomes as long as the accessible observation timescale (see Fig. 2.10(a)).

### Self-intermediate scattering function

Another widely used dynamical observable is the self-intermediate scattering function,

$$F_s(q,t) \;:=\; \frac{1}{N}\,\mathbb{E}\left[\sum_i \exp\left(\mathrm{i}\,\vec{q}\cdot(\vec{r}_i(t)-\vec{r}_i(0))\right)\right]\,. \tag{2.25}$$

which probes the motion of individual particles at a characteristic length scale $2\pi/q$. It is usually measured at a wavevector $q^\star$ corresponding to the first peak of the structure factor, which roughly corresponds to the typical interparticle distance.

In liquids, $F_s(q,t)$ decays exponentially from 1 to 0 as particles lose memory of their initial positions. In glassy systems, the decay becomes non-exponential and exhibits a two-step relaxation. At short times, $F_s$ rapidly decays to a plateau due to ballistic motion, a process known as *$\beta$-relaxation*. The plateau corresponds to the caging regime observed in the MSD, where particles are temporarily trapped by their neighbours. The final decay from the plateau to zero is called *$\alpha$-relaxation* and corresponds to the structural rearrangements due to particles exiting their cages (see Fig. 2.10(b)). For this reason, the $\alpha$-relaxation time $\tau_\alpha$ introduced in Section 2.1 is typically defined by $F_s(q^\star,\tau_\alpha)=1/\mathrm{e}$.

The change in behaviour of $F_s(q,t)$ from a single exponential decay in liquids to a two-step decay in glassy systems can be used to precisely determine the onset temperature $T_o$ introduced in Section 2.1.2. Indeed, it can be identified with the temperature below which $F_s(q,t)$ develops a rising inflection point. This is precisely the definition we propose in Ref. [104].

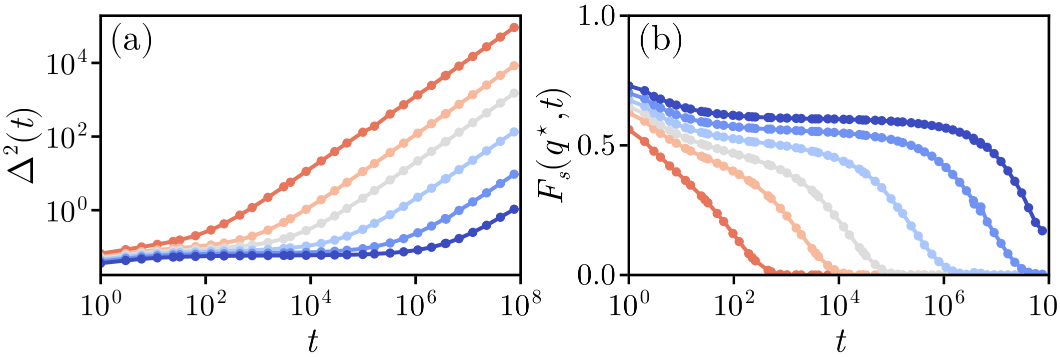

**Figure 2.10:** Two-times correlation functions. (a) Mean square displacement $\Delta^2(t)$ at different temperatures. (b) Corresponding self-intermediate scattering function $F_s(q^\star,t)$.

## 2.4 Simulation methods

Developing accurate theoretical descriptions of amorphous materials remains one of the open challenges in condensed-matter physics. While mean-field models in the limit of infinite dimension can be solved exactly [133], in finite dimensions the Boltzmann distribution is intractable and the partition function cannot be evaluated analytically. As a result, most theoretical descriptions of the glass transition in finite dimensions remain phenomenological [1]. For these reasons, numerical simulations play a crucial role in advancing our understanding of amorphous materials [3]. Their strength lies in providing microscopic information at machine precision in a controlled environment, which is difficult to achieve in experiments.

Simulations of interacting particle systems typically consists in generating a sequence of configurations $\{x^{(k)}\}_{k=1}^{K}$ distributed according to the Boltzmann measure (2.5). These configurations are then used to estimate ensemble averages of physical observables via Monte Carlo integration (see Chapter 1.1). A wide variety of algorithms have been developed for this purpose, including nonequilibrium sampling schemes [134–136]. Other approaches, such as simulations of vapour deposition [137], aim instead to reproduce specific experimental preparation protocols and do not sample the Boltzmann distribution directly. However, the two most widely used approaches remain molecular dynamics (MD) and Markov chain Monte Carlo (MCMC) [110, 111]. In this section, we briefly review both methods in the context of amorphous materials, as they provide the starting point for the computational methods developed in this thesis.

### 2.4.1 Molecular dynamics

In molecular dynamics (MD), the trajectories of all particles are obtained by numerically integrating Newton's equations of motion for all particles subject to the potential energy function $U$. This corresponds to solving the system of coupled differential equations

$$m_i \frac{\mathrm{d}^2 \vec{r}_i}{\mathrm{d}t^2} = -\nabla_{\vec{r}_i} U(\boldsymbol{s}, \boldsymbol{r}) \ , \tag{2.26}$$

starting from initial conditions $(\boldsymbol{s}, \boldsymbol{r}(0))$ and $(\boldsymbol{s}, \boldsymbol{v}(0))$, where $\boldsymbol{v} = \mathrm{d}\boldsymbol{r}/\,\mathrm{d}t$ are the particle velocities and $m_i$ the masses. These equations are typically integrated using finite-difference methods such as the velocity-Verlet algorithm [110].

MD may feel like the most natural simulation method, as it directly mimics a realistic microscopic dynamics. It can therefore be used to compute dynamical observables that can be compared directly with experimental data.

Since Newton's equations conserve the total energy, MD samples the microcanonical ($NVE$) ensemble rather than the canonical ($NVT$) ensemble defined by the Boltzmann distribution (2.5). To sample the canonical ensemble, one introduces a *thermostat* to mimic the effect of a thermal bath. Similarly, coupling the system to a *barostat* allows one to sample the isothermal–isobaric ($NPT$) ensemble.

## 2.4.2 MCMC for particle systems

Markov chain Monte Carlo methods are an alternative way to sample configurations from the Boltzmann distribution. We have already introduced their general principles in Chapter 1.1.2. Here we focus on specific applications to particle systems and on the most common transition kernels used in this context. In the physics literature, these simulations are often referred to simply as "Monte Carlo simulations".

The most common MCMC method for particle systems is the *Metropolis–Hastings* algorithm introduced in Section 1.2. It is defined by a proposal kernel $Q(x, \mathrm{d}x')$, which in turns determines the transition kernel $K(x, \mathrm{d}x')$ of the Markov chain through the acceptance step (1.19). In simulations of physical systems, individual proposals are usually referred to as *moves*, as they represent local or global modifications of the current configuration, *e.g.* by flipping a spin or displacing a particle. Below we review two of the most widely used moves in MCMC simulations of particle systems.

### Single-particle displacements

The most common move in MCMC for particle systems is the *single-particle displacement.* A particle $i$ is chosen uniformly at random and displaced by a small random vector sampled uniformly within a cube of side $\sigma$ centred at the origin. While this move is simple to describe in words and to implement, writing its proposal kernel is not completely trivial. Indeed, it cannot be expressed as a density with respect to the Lebesgue measure, as it involves a discrete choice of the particle to move and updates only one variable at a time. Formally, the proposal kernel reads

$$Q\left(x, \mathrm{d}x'\right) = \frac{1}{N\,\sigma^d} \sum_i \mathbf{1}_{\Delta_\sigma(\vec{r}_i)}(\vec{r}_i{}')\, \mathrm{d}\vec{r}_i{}' \left( \prod_{j \neq i} \delta\left(\vec{r}_j, \mathrm{d}\vec{r}_j{}'\right) \right) \left( \prod_j \delta\left(s_j, \mathrm{d}s_j'\right) \right) , \qquad (2.27)$$

where $\Delta_\sigma\left(\vec{r}_i\right)$ is hypercube of side $\sigma$ centered in $\vec{r}_i$. In words, the kernel selects one particle uniformly, displaces it within a cube of side $\sigma$, and leaves all other particles and species unchanged.

A useful variant is the *Gaussian displacement*, where the new position of particle $i$ is sampled from a $d$-dimensional normal distribution centred at $\vec{r}_i$ with covariance matrix $\sigma^2 \mathrm{I}_d$. The kernel for this move is

$$Q\left(x, \mathrm{d}x'\right) = \frac{1}{N\left(2\pi\sigma^2\right)^{d/2}} \sum_i \exp\left(-\frac{d_{\mathcal{M}_L}\left(\vec{r}_i', \vec{r}_i\right)^2}{2\sigma^2}\right) \mathrm{d}\vec{r}_i' \times \\ \times \left(\prod_{j\neq i} \delta\left(\vec{r}_j, \mathrm{d}\vec{r}_j'\right)\right) \left(\prod_j \delta\left(s_j, \mathrm{d}s_j'\right)\right) \ , \quad (2.28)$$

where $d_{\mathcal{M}_L}$ is the nearest image distance. We will use this version in Chapter 3, where we will need transition kernels with unbounded support. Single-particle displacements are straightforward to implement and computationally inexpensive, since only the interaction terms involving the moved particle need to be recomputed when evaluating the acceptance probability. However, they become inefficient at high densities, where large moves are frequently rejected due to particle overlaps. Interestingly, their glassy dynamics resembles that of molecular dynamics in the diffusive regime [138], with particles exploring configuration space through an accumulation of small, local displacements.

### Swap moves

In multicomponent systems, one can also introduce *swap moves*, where the species of two particles $i$ and $j$ are exchanged while keeping their positions fixed [139]. The proposal kernel for a swap move between species $\alpha$ and $\beta$ is

$$Q\left(x, \mathrm{d}x'\right) = \sum_{i\in\alpha} \sum_{j\in\beta} \frac{1}{N_\alpha N_\beta} \delta\left(s_j, \mathrm{d}s_i'\right) \delta\left(s_i, \mathrm{d}s_j'\right) \times \\ \times \left(\prod_{k\neq i,j} \delta\left(s_k, \mathrm{d}s_k'\right)\right) \left(\prod_k \delta\left(\vec{r}_k, \mathrm{d}\vec{r}_k'\right)\right) \ . \quad (2.29)$$

This proposal also involves discrete choices and cannot be expressed as a continuous density.

The idea behind the swap is to introduce non-local updates that can help particles escape local cages and accelerate structural relaxation. For swap moves to be effective, particle sizes must be sufficiently similar to give a reasonable acceptance rate, while the model itself must remain robust against crystallisation when sampling becomes more efficient. In a breakthrough study, Ninarello *et al.* [140] showed that by combining suitably designed models with swap moves, equilibration times could be reduced by many orders of magnitude. The relaxation time at the estimated glass transition temperature for MD and conventional displacement moves can be as large as $10^{10}$ times longer than with swap moves. This work made it possible to equilibrate glass-forming liquids at temperatures well below the experimental glass transition, which lead to significant advances in our understanding of these systems over the past decade [5, 141–143]. Recent work [144] indicates that the combination of swap moves with other enhanced sampling methods may enable the equilibration of glasses down to zero temperature in two dimensions. Even more recently, swap moves have been generalised to molecular glasses via the so-called *flip move*, enabling equilibration of molecular systems below $T_g$ as well [145].

Despite their success, swap-based algorithms are only applicable to a limited class of models. In particular, they perform poorly in systems such as the Kob–Andersen mixture, where the swap acceptance is practically zero. This limitation motivates the development of more general methods capable of accelerating sampling in a broader range of models. In the following chapters, we aim to contribute to this effort.

# Chapter 3

# Policy-guided Monte Carlo on general state spaces

*Yes, it does respect detailed balance.*

Policy-guided Monte Carlo (PGMC) is an adaptive version of the Metropolis-Hastings (MH) algorithm built on a formalism inspired by reinforcement learning. Originally introduced by Bojesen [41] for discrete systems, it showed an improvement in sampling efficiency on the Kagome lattice Ising model, and was later applied to spin-ice systems [65], where it successfully learned non-local updates. In Ref. [42], we extended the method to arbitrary state spaces using a fully measure-theoretic formalism, allowing to deal with discrete, continuous, and mixed variables. This formulation also clarified and formalised several aspects of the original algorithm, for instance the expression of the transition kernel and the nature of chain policies. Our initial motivation was to apply the method to glass-forming mixtures, where the state space combines continuous (particle positions) and discrete (species) degrees of freedom. The process led us to develop a general and rigorous framework applicable to a broad class of systems, including statistical inference problems and algorithms involving dimensional changing kernels [55]. After establishing the formalism, we applied PGMC to models of glass-forming mixtures, using simple, physically motivated moves depending on a few parameters. The method was capable of efficiently learning optimal parameters for these moves which led to improved sampling performance up to two orders of magnitude.

In this chapter, we present the work of Ref. [42] and further extend the formulation of the algorithm to arbitrary reward functions. This extension is an original contribution of this thesis. We start by introducing the general formulation of PGMC, building on its analogy with the classical reinforcement learning problem, and then move to our main theoretical result, which generalises the policy-gradient theorem [64]. We then present the results obtained for the additive soft-sphere and the Kob–Andersen mixtures, assessing the performance and limitations of the designed moves through various metrics. We conclude with a high-level discussion of the method, including the differences with the original formulation and possible directions for future research. As part of this work, in collaboration with Romain Simon, we developed `Arianna.jl`, a general-purpose Julia implementation of PGMC released as an open-source package [146] (see Appendix E.1). Although its use has so far been limited to proof-of-concept studies, we believe that PGMC, possibly with the support of `Arianna.jl`, holds the potential to become a general tool for enhancing sampling efficiency across a wide range of applications.

This chapter is based on the article *Policy-Guided Monte Carlo on General State Spaces: Application to Glass-Forming Mixtures*, written in collaboration with Riccardo Rende and Daniele Coslovich [42].

## 3.1 Theoretical framework

As discussed in Section 1.2, provided that the proposal distribution $Q$ guarantees ergodicity, the condition of detailed balance ensures that the MH algorithm eventually samples the desired distribution $P$. This leaves great freedom in the choice of the specific form of the proposal distribution. The general goal of adaptive MC methods, including PGMC, is to find an optimal proposal distribution that maximises some measure of the sampling efficiency of the Markov chain. As noted by Bojesen, Reinforcement learning (RL) provides a natural language to formalise this optimisation problem [64]. In RL, an *agent* interacts with an *environment* by performing *actions* that modify its *state*. The agent follows a *policy*, a probability distribution over actions conditioned on the current state, and receives a *reward* quantifying the quality of each transition. The objective is to learn a policy that maximises the expected cumulative reward. This setup defines a *Markov decision process* (MDP) characterised by the tuple ($\mathsf{X}$, $\mathcal{A}$, $p$, $r$), where $\mathsf{X}$ denotes the state space, $\mathcal{A}$ the space of possible actions, $p(x'|x, a)$ is a transition probability, and $r(x, a, x')$ the reward function. Usually, the policy is denoted by $\pi(a|x)$, and the goal is to find the optimal policy $\pi^*$ that maximises the expected reward $J[\pi]$.

Unlike in supervised learning, the agent does not rely on labelled data but learns directly from experience, adjusting its policy according to the observed rewards. Most RL formulations consider *episodic tasks*, in which the agent repeatedly interacts with the environment over finite trajectories. However, a more natural setup for Markov chains is that of *continuing tasks*, where the process has no terminal state and evolves indefinitely. In this case, the objective is to maximise the *expected reward* averaged over the stationary distribution of the process, or equivalently, the time averaged reward along an infinite trajectory. This corresponds precisely to the adaptive MH algorithms described in Section 1.2.2, where the goal is to find a proposal distribution that maximises the sampling efficiency of the Markov chain in steady state.

### 3.1.1 Defining the problem

PGMC builds upon the analogy with RL by viewing a proposal step as an action taken by an agent to modify the configuration of a system. The policy corresponds to the proposal distribution $Q(x, \mathrm{d}x')$, while the acceptance term in Eq. (1.19) plays the role of the dynamical rule $p\left(x'|x, a\right)$ determining the next state once the action has been chosen. In PGMC, we identify an action $a$ with a move $x \to x'$, aligning our notation more closely with that of traditional MCMC simulations. Each proposed transition $x \to x'$ can then be associated with a reward $r(x, x')$ that quantifies its performance. A natural choice is to set the reward to a measure of inverse correlation between states $x$ and $x'$, but other choices depending on the type of move are also possible.

Assuming $r : \mathsf{X} \times \mathsf{X} \to \mathbb{R}$ is measurable and integrable, the *objective function* of PGMC is defined as the expected reward per step in the stationary regime of the Markov chain:

$$J\left[Q\right] \;\coloneqq\; \mathbb{E}_{\substack{x\sim P\\ x'\sim K}}\left[r\left(x, x'\right)\right] \;. \tag{3.1}$$

The objective depends on the policy $Q$ through the transition kernel $K$ defined in Eq. (1.21). Note that estimating $J[Q]$ requires sampling from both the target distribution $P$ and the transition kernel $K$, which is not straightforward due to the diagonal term in Eq. (1.21). In Proposition B.1.1, we provide an alternative expression for $J[Q]$ that only involves sampling from $P$ and $Q$:

$$J\left[Q\right] \;=\; \mathbb{E}_{\substack{x\sim P\\ x'\sim Q}}\left[\big(r\left(x, x'\right) - r\left(x, x\right)\big)\alpha\left(x, x'\right)\right] + \mathbb{E}_{x\sim P}\left[r\left(x, x\right)\right] \;. \tag{3.2}$$

This result generalises our previous finding in Ref. [42], which was limited to reward functions such that $r(x, x) = 0$ for all $x \in \mathsf{X}$.

Our goal is to find a policy $Q^\star$ that maximises the objective in Eq. (3.1). Since searching for an optimal policy in the space of all possible proposal distributions is intractable, we restrict our search to a parametric family of policies $\{Q_\theta\}$, with $\theta \in \Theta \subseteq \mathbb{R}^p$, as is common in adaptive MCMC methods. The optimisation problem then reduces to finding the parameters $\theta^\star$ that maximise the objective

$$J(\theta) \;=\; \mathbb{E}_{\substack{x\sim P\\ x'\sim Q_\theta}} \left[\big(r(x,x') - r(x,x)\big)\alpha_\theta(x,x')\right] \;, \tag{3.3}$$

where we ignored the second term in Eq. (3.2), since it is independent of $\theta$.

There are various ways to search for the optimal parameters $\theta^\star$. A naive approach is to perform a grid search over the parameter space, evaluating the performance of each candidate policy by running a full MCMC simulation waiting for convergence to the stationary distribution. This method is clearly inefficient, as it requires a large number of costly simulations. A more effective strategy is to use gradient-based optimisation methods, which iteratively update the parameters in the direction of the gradient of the objective function $\nabla J_\theta$. In practice, starting from an initial guess $\theta_0$, we generate a Markov chain using the policy $Q_{\theta_0}$, estimate the gradient $\nabla_\theta J$ from the samples collected along the trajectory, and update the parameters according to the *stochastic gradient ascent* procedure

$$\theta_{t+1} \;=\; \theta_t + \eta_t \widehat{\nabla_\theta J}(\theta_t) \;, \tag{3.4}$$

where $\eta_t > 0$ is the learning rate at iteration $t$, and $\widehat{\nabla_\theta J}(\theta)$ is an estimator of the true gradient $\nabla_\theta J(\theta)$. Then, we repeat the process using the updated parameters $\theta_{t+1}$, and so on, until convergence. In order to use Eq. (3.4), we need to estimate the gradient of $J$. Note that while Eq. 3.2 allows to express $J$ as an average over $Q_\theta$ and $P$, which can be easily estimated with Monte Carlo, estimating $\nabla_\theta J$ is not straightforward. This is precisely the idea behind the policy-gradient theorem in RL [64], which provides an expression for $\nabla_\theta J$ that can be estimated from samples generated by the current policy. In the next section, we present our main theoretical result, which generalises the policy-gradient theorem to PGMC, where the transition kernel includes the acceptance step of the MH algorithm which depends on the policy itself.

### 3.1.2 Estimating $\nabla_\theta J$

Why is estimating the gradient of the objective function $J$ challenging? There are two main reasons. First, the integration measure $Q_\theta$ in Eq. (3.3) depends on the parameters themselves, which prevents us from differentiating the integral naively.

Second, even if we introduce a density $q_\theta(x'|x)$ with respect to a fixed reference measure $\mu$ and bring the derivative inside the integral, we would obtain terms involving $\nabla_\theta q_\theta$ that cannot be sampled directly using $Q_\theta$. These difficulties are familiar in RL and appear as well in other machine learning contexts, such as energy-based training of normalising flows, variational autoencoders, or variational quantum Monte Carlo [29, 77, 147, 148].

If we aim at deriving a general expression for the gradient of $J$ that can be estimated using only samples drawn from the current policy $Q_\theta$ we need to address these two problems. The first difficulty is resolved by Lemma B.2.1, which allows us to rewrite $J$ as an expectation with respect to a fixed measure $Q_{\theta_0}$. In a PGMC simulation, this measure can be interpreted as the proposal distribution at the *current* parameter values $\theta = \theta_t$, which remains fixed during the estimation of the gradient. Once the measure no longer depends on $\theta$, the derivative may be brought inside the integral under mild regularity assumptions. This leads to the following Theorem, which is our main theoretical result for PGMC.

**Theorem 3.1.1** (Policy-guided Monte Carlo gradient estimation)**.** *Let $J$ be the objective function defined in Eq.* (3.3)*, $\theta_0 \in \Theta$, and $\nu_{\theta_0} : \mathcal{E} \otimes \mathcal{E} \to \mathbb{R}$ be the measure defined as $\nu_{\theta_0}(\mathrm{d}x, \mathrm{d}x') = Q_{\theta_0}(x, \mathrm{d}x')\, P(\mathrm{d}x)$. Assume*

*i. There exists a neighbourhood $U_{\theta_0} \subseteq \Theta$ of $\theta_0$ such that $Q_\theta \ll Q_{\theta_0}$ for all $\theta \in U_{\theta_0}$. Let $L_\theta : U_{\theta_0} \times \mathsf{X}^2 \to \mathbb{R}$ be the map defined as*

$$L_\theta(x, x') := \frac{Q_\theta(x, \mathrm{d}x')}{Q_{\theta_0}(x, \mathrm{d}x')} \,. \tag{3.5}$$

*ii. For $\nu_{\theta_0}$-almost every $(x, x')$, the map $\theta \mapsto \alpha_\theta(x, x')\, L_\theta(x, x')$ is differentiable for all $\theta \in U_{\theta_0}$.*

*iii. There exists $g \in \mathcal{L}^1(\mathsf{X} \times \mathsf{X},\, \mathcal{E} \otimes \mathcal{E},\, \nu_{\theta_0})$ such that*

$$\sup_{\theta \in U_{\theta_0}} \left\| \nabla_\theta \big( \alpha_\theta(x, x')\, L_\theta(x, x') \big) \right\| \le g(x, x') \quad \nu_{\theta_0}\text{-almost everywhere.}$$

*Then, $J$ is differentiable in $U_{\theta_0}$, and its gradient is*

$$\begin{aligned} \nabla_\theta J(\theta) \;=\; & \mathbb{E}_{\substack{x \sim P \\ x' \sim Q_{\theta_0}}} \Big[ \big( r(x, x') - r(x, x) \big) \alpha_{\theta_0}(x, x') \times \\ & \times \big( H(h_{\theta_0}(x, x') - 1)\, \nabla_\theta L_\theta(x, x') + H(1 - h_{\theta_0}(x, x'))\, \nabla_\theta L_\theta(x', x) \big) \Big] \,, \end{aligned} \tag{3.6}$$

*where $h$ is the Hastings ratio defined in Eq.* (1.20) *and $H$ is the Heaviside function.*

*Proof.* See Appendix B.2 □

The main assumption is that the proposal distributions $Q_\theta$ are mutually absolutely continuous in a neighbourhood of the current parameters $\theta_0$, which allows us to express $Q_\theta$ in terms of $Q_{\theta_0}$ through the Radon-Nikodym derivative $L_\theta$ and to perform importance sampling of $Q_\theta$ using $Q_{\theta_0}$ as a proposal. In practice, this condition can be thought of as requiring that the support of $Q_\theta$ does not change abruptly when $\theta$ varies around $\theta_0$, for instance by changing the range of a uniform proposal. The other assumptions are standard regularity conditions ensuring the differentiability of the relevant functions and allowing to exchange differentiation and integration. Moreover, Corollary B.2.2 states that if $Q_\theta$ admits a density $q_\theta$ with respect to a common dominating measure $\mu$, then the gradient of $J$ at $\theta = \theta_0$ can be expressed as

$$\begin{aligned}\nabla_\theta J\left(\theta_0\right) \;=\; & \mathbb{E}_{\substack{x\sim P\\ x'\sim Q_{\theta_0}}} \Big[\big(r(x,x') - r(x,x)\big)\alpha_{\theta_0}(x,x') \times \\ & \times\big(H(h_{\theta_0}(x,x') - 1)\, s_{\theta_0}(x,x') + H(1 - h_{\theta_0}(x,x'))\, s_{\theta_0}(x',x)\,\big)\Big] \;, \quad (3.7)\end{aligned}$$

where $s_\theta\left(x,x'\right) \coloneqq \nabla_\theta \log q_\theta\left(x'|x\right)$ is the so-called *score function.*

The proof in Appendix B.2 relies on standard likelihood-ratio arguments [149] widely used in RL and related fields, followed by an application of the score-function identity [147]. The result may be viewed as a measure-theoretic generalisation of the policy-gradient theorem for continuing tasks, extended to the case where the transition probability (usually written as $p(x'|x,a)$) depends explicitly on $\theta$. Importantly, Theorem 3.1.1 applies to any proposal distribution $Q_\theta$, without the need to assume the existence of a density with respect to a fixed reference measure. It also extends and further formalises the result of Ref. [42], which was limited to reward functions such that $r(x,x) = 0$.

### 3.1.3 PGMC in practice

We now have all the ingredients required to implement PGMC in a practical simulation. Let us summarise what we have achieved so far. Starting from a parametric family of proposal distributions (policies) $Q_\theta$ and a reward function $r(x,x')$, we introduced in Eq.(3.3) the objective function $J(\theta)$, which represents the expected reward per step of the Markov chain generated by the Metropolis–Hastings algorithm using $Q_\theta$ as proposal distribution. Since the goal is to adapt $Q_\theta$ in the direction that increases $J(\theta)$, according to the stochastic gradient ascent rule in

Eq.(3.4), we derived in Eq. (3.7) an expression for $\nabla_\theta J(\theta)$ that can be estimated directly from samples generated using the current policy $Q_{\theta_0}$.

Starting from a standard MH algorithm, incorporating PGMC is conceptually simple. In a conventional MH simulation, when designing a move, one must specify a method for sampling new states, given a *fixed* policy $Q$. The system then evolves by drawing successive states from $Q$ and accepting them with probability $\alpha$. The difference that sets PGMC apart from MH is the integration of an optimisation step at regular time intervals. In practice, at each optimisation step, a set of $M$ configurations is drawn from the standard MH procedure given by a fixed policy $Q_{\theta_0}$ and the corresponding acceptance step. These can correspond to the last $M$ states of the Markov chain or to samples from independent chains, or a combination of both. The latter strategy is often preferred to reduce correlations between samples, but it requires running multiple synchronised chains in parallel. Note that, as long as the Markov chain has not converged to the stationary distribution, the samples used to estimate the gradient will not be exactly distributed according to $P$. However, as long as the chain is sufficiently close to equilibrium, the bias introduced by this approximation should be small and the estimate should get better and better as the optimisation proceeds [41]. Then, for each of these $M$ configurations, a set of $M'$ new configurations is proposed according to the current policy $Q_{\theta_0}$ (see Fig. 3.1). The resulting $MM'$ pairs $(x_m, x'_n)$ are then used to compute a Monte Carlo estimate of the gradient $\nabla_\theta J(\theta_0)$ in Eq. (3.7), which is given by

$$\begin{aligned}\widehat{\nabla_\theta J}(\theta_0) \;=\; & \frac{1}{MM'}\sum_{m=1}^{M}\sum_{n=1}^{M'}\big(r(x_m, x'_n) - r(x_m, x_m)\big)\alpha_{\theta_0}(x_m, x'_n)\times \\ & \times\big(H(h_{\theta_0}(x_m, x'_n) - 1)\, s_{\theta_0}(x_m, x'_n) + H(1 - h_{\theta_0}(x_m, x'_n))\, s_{\theta_0}(x'_n, x_m)\big)\;. \qquad (3.8)\end{aligned}$$

Using this estimate in Eq. (3.4), we update the parameters $\theta$ of the policy $Q_\theta$. The updated policy is then used to continue the MH simulation until the next optimisation step, and so on.

This procedure, currently limited to the case where $r(x, x) = 0$, is the one implemented in `Arianna.jl` [146], our open-source Julia package for Monte Carlo simulations. The implementation is modular and flexible, allowing one to easily define custom policies, reward functions, and optimisation methods (see Appendix E.1 for details).

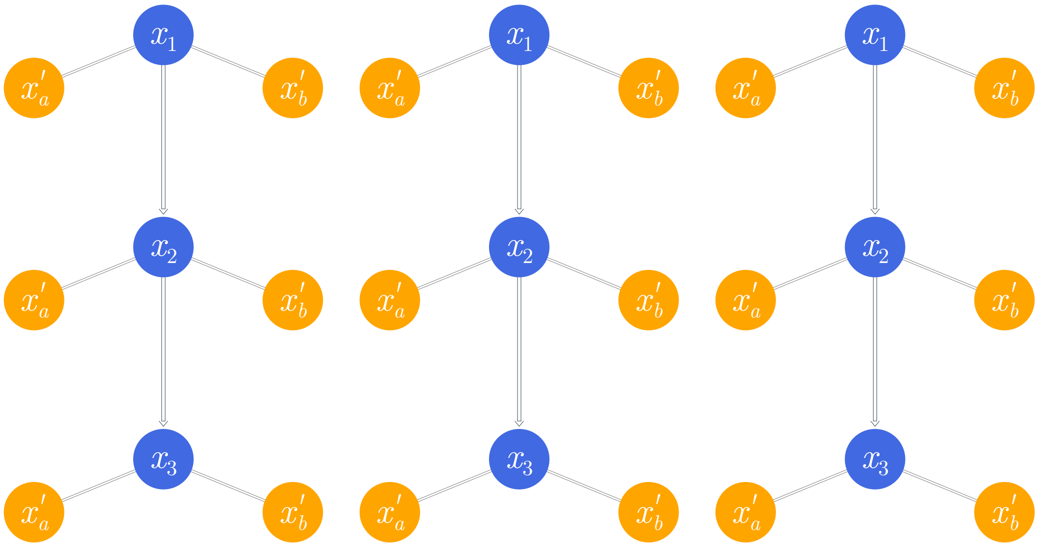

**Figure 3.1:** Schematic representation of the sampling procedure in PGMC. Three independent Markov chains are run in parallel using the MH kernel $K_{\theta_0}$ associated with the current policy $Q_{\theta_0}$ (blue circles). To estimate the gradient of the objective function $J(\theta)$ at $\theta = \theta_0$, for each of the $M = 9$ configurations $x_m$, a set of $M' = 2$ new configurations $x'_n$ is proposed according to $Q_{\theta_0}$ (orange circles). The resulting $MM'$ pairs $(x_m, x'_n)$ are then used to compute a Monte Carlo estimate of the gradient in Eq. (3.8).

### 3.1.4 Reversibility and convergence

A remark is in order regarding the effect of adaptation on the Markov chain. The fact that the proposal distribution $Q_\theta$ gets adapted during the simulation makes the Markov chain non-homogeneous, but this does not affect reversibility and the existence of $P$ as a stationary distribution. Indeed, configurations sampled directly from $Q_\theta$ to estimate the gradient are then discarded and do not contribute to the actual sampling of the target distribution. Sampling is performed instead through the standard MH algorithm, where even though the proposal distribution changes over time, the acceptance probability depends on the current policy, ensuring that detailed balance with respect to $P$ is always satisfied (see Theorem 1.2.1). This guarantees that if the chain starts from a configuration drawn from $P$, it will remain distributed according to $P$ at all times.

In practice, of course, the chain is initialised from an arbitrary configuration and must be equilibrated before the samples are representative of $P$. For a fixed proposal distribution $Q_\theta$, Theorem 1.2.2 guarantees convergence to $P$ provided that the chain is ergodic. Since the chain in PGMC is nonhomogeneous, the same theorem does not apply directly and convergence to $P$ is not guaranteed a priori. For instance, the pathological behaviour seen in Example 1.2.1 can be reproduced in PGMC by setting $r(x, x') = 1 - \delta(x - x')$, which could lead to a continuous decrease of the proposal variance violating the simultaneous uniform ergodicity condition (see Theorem 1.2.3).

Such pathologies, however, can be avoided by designing reward functions and policy families that respect the conditions of Theorem 1.2.3. When this is the case, convergence to $P$ is guaranteed despite adaptation. In practice, once the parameters $\theta$ have converged to an approximate optimum $\theta^\star$, continuing to adapt them is unnecessary and would only increase the computational cost of the simulation. A natural strategy is therefore to consider the initial part of the simulation as a *training phase*, after which adaptation is switched off and the simulation proceeds with a standard MH chain using the learned proposal $Q_{\theta^\star}$, which is expected to exhibit improved efficiency compared to the initial policy.

### 3.1.5 Optimisation methods

We now return to the update rule in Eq.(3.4) and discuss how to choose the learning rate $\eta_t$ [15, 18]. A constant learning rate, $\eta_t = \eta$, leads to what is known in reinforcement learning as *vanilla policy gradient* (VPG). This simple choice often works surprisingly well, but more sophisticated methods have been developed to improve stability and convergence. We review here three common families of strategies to improve the basic update rule in Eq. (3.4), all of which can be combined in different ways. All these methods are currently implemented in `Arianna.jl`.

#### Baseline policy gradient

Stochastic gradient estimates such as Eq.(3.8) often suffer from high variance, which may slow down convergence or lead to unstable parameter updates [150]. A classical remedy is to subtract a *baseline* from the gradient estimate, *i.e.* a zero-mean term that reduces variance without introducing bias. In RL, these approaches are known as *baseline policy gradients* (BLPG) [64]. Since fluctuations usually come from the score term, a common baseline is a term proportional to the score function itself.

To apply this idea to PGMC, we define the baseline $b_\theta : \mathsf{X} \times \mathsf{X} \to \mathbb{R}^p$ as

$$b_\theta(x, x') \coloneqq J(\theta)\, s_\theta(x, x') \; . \tag{3.9}$$

Since

$$\mathbb{E}_{\substack{x\sim P\\ x'\sim Q_{\theta_0}}}\left[b_{\theta_0}\left(x,x'\right)\right]=0 \tag{3.10}$$

(see Proposition B.3.2), then

$$\widehat{\nabla_\theta J}^{\mathrm{BL}}(\theta_0)\;=\;\widehat{\nabla_\theta J}(\theta_0)-\frac{\widehat{J}}{MM'}\sum_{m=1}^{M}\sum_{n=1}^{M'}s_{\theta_0}(x_m,x'_n) \tag{3.11}$$

is an unbiased estimator of $\nabla_\theta J\left(\theta_0\right)$, which is expected to exhibit lower variance than the original estimate. In BLPG, we simply replace $\widehat{\nabla_\theta J}$ with $\widehat{\nabla_\theta J}^{\mathrm{BL}}$ in Eq. (3.4). The formulation BLPG for PGMC is an original contribution of this thesis.

### Natural policy gradient

In VPG, the update rule (3.4) can be interpreted as maximising a linear approximation of the average reward $J$ centred around the current parameter values $\theta_t$, while simultaneously penalising large steps in parameter space. In fact, the update rule in Eq. (3.4) can be seen as the solution of the constrained optimisation problem where we look for the parameters $\theta_{t+1}$ that maximise the first-order Taylor expansion of $J$ around $\theta_t$ under the constraint that $\left\|\theta_{t+1}-\theta_t\right\|^2$ stays small [18]. Although this idea is simple and effective, it does not take into account the geometry of the policy space, which may lead to sub-optimal convergence rates [151–153]. These limitations led to the development of *natural policy gradient* (NPG) methods, which replace the Euclidean distance with an approximation of the Kullback-Leibler divergence between the old and new policies [152]. This means that the size of the steps is not taken uniformly in all directions of the parameter space, but rather according to how much the *policy* changes. In Proposition B.3.3, we apply a similar procedure to that of [152] to show that the KL divergence between two policies $Q_{\theta_0}$ and $Q_\theta$ averaged over $P$ can be approximated to second order as

$$\begin{aligned} D\left(\theta\right) &\coloneqq \mathbb{E}_{x\sim P}\left[\mathcal{D}_{\mathrm{KL}}\left(Q_{\theta_0}\|Q_\theta\right)\right] && (3.12)\\ &=\frac{1}{2}\left(\theta-\theta_0\right)^T F\left(\theta_0\right)\left(\theta-\theta_0\right)+o\left(\left\|\theta-\theta_0\right\|^2\right)\ , && (3.13)\end{aligned}$$

where

$$F\left(\theta\right)\;=\;\mathbb{E}_{\substack{x\sim P\\ x'\sim Q_\theta}}\left[s_\theta\left(x,x'\right)s_\theta\left(x,x'\right)^T\right] \tag{3.14}$$

is the *Fisher information matrix* of the policy $Q_\theta$. Since the score is already required by PGMC, the Fisher information matrix can be estimated from the same samples used to estimate $\nabla_\theta J$ at a minimal computational cost as

$$\widehat{F}\,(\theta) \;=\; \frac{1}{MM'} \sum_{m=1}^{M} \sum_{n=1}^{M'} s_\theta(x_m, x'_n)\, s_\theta(x_m, x'_n)^T \ . \tag{3.15}$$

In practice, estimating the Fisher information matrix reliably generally requires more samples than estimating $\nabla_\theta J$ alone, and the benefits of natural gradients are most apparent when $M$ and $M'$ are sufficiently large for $\widehat{F}$ to be reasonably well conditioned (see Section 3.2.4). NPG consists in replacing the Euclidean distance $\|\theta - \theta_0\|$ of VPG with the *Fisher metric* $(\theta - \theta_0)^T\, F\,(\theta_0)\,(\theta - \theta_0)$, which is the natural information-geometric metric in the space of probability distributions [154]. Solving the resulting constrained optimisation problem and fixing a constant learning rate leads to the update rule

$$\theta_{t+1} \;=\; \theta_t + \eta\, \widehat{F}^{-1}(\theta_t)\, \widehat{\nabla_\theta J}(\theta_t) \ . \tag{3.16}$$

Note that NPG can be combined with BLPG by replacing $\widehat{\nabla_\theta J}$ with $\widehat{\nabla_\theta J}^{\mathrm{BL}}$ in the above equation.

### Adaptive stepsize

Another common issue with stochastic gradient methods is the choice of the learning rate scheduling $\eta_t$. If $\eta_t$ is too large, the updates may overshoot the optimum and lead to divergence, while if it is too small, convergence may be very slow. To address this problem, various adaptive learning rate methods have been developed [155]. A relatively simple approach is that of chosing the learning rate at each step so that the considered metric (either Euclidean or Fisher) between the old and new parameters is kept below a certain threshold $\epsilon > 0$ [18, 153]. For a generic metric defined by a positive definite matrix $G\,(\theta_t)$, this leads to the update rule

$$\theta_{t+1} \;=\; \theta_t + \sqrt{\frac{\epsilon}{\widehat{\nabla_\theta J}(\theta_t)^T\, G^{-1}(\theta_t)\, \widehat{\nabla_\theta J}(\theta_t)}}\; G^{-1}(\theta_t)\, \widehat{\nabla_\theta J}(\theta_t) \ . \tag{3.17}$$

For VPG, $G$ is simply the identity matrix, while for NPG it is the Fisher information matrix $\widehat{F}$. We call these versions of VPG and NPG with adaptive learning rate *adaptive policy gradient* (APG) and *adaptive natural policy gradient* (ANPG), respectively.

## 3.2 Application to glass-forming mixtures

The original motivation for extending PGMC to general state spaces was to apply it to glass-forming mixtures, where the state space combines continuous (particle positions) and discrete (species) degrees of freedom. As discussed in Chapter 2, most conventional simulation methods struggle to efficiently sample these systems at low temperatures. While modern implementations of adaptive MCMC methods and other related machine learning methods [29, 156] have shown very good results for different physical systems, even in the proximity of phase transitions [157], it is now acknowledged that glassy systems remain a major challenge that provides a hard benchmark for enhanced sampling techniques [38, 158]. In the following, we will focus on the simple models of glass-forming liquids described in Section 2.2.3, in which particles interact through a pairwise potential and a suitable mismatch between particle sizes frustrates the nucleation of the stable crystalline phase. We mostly focus on systems for which the swap move (see Section 2.4.2) does not provide any speedup due to the large size difference between species, which dramatically reduces the acceptance rate. As described in the previous sections, the PGMC method is very general and allows us to optimise virtually any parametrised move. The challenge lies in designing suitable rewards and policies for efficient sampling of configuration space. Below, we present the results obtained from simple moves based on single-particle displacements and particle swaps, whose number of parameters remains small (in the range 1–6).

### 3.2.1 Simulation details

We consider three-dimensional systems composed of $N = 216$ particles in a cubic box with periodic boundary conditions at constant temperature $T$ in the canonical ($NVT$) ensemble. As discussed in Section 2.2, we specify the state of the system as $x = (\boldsymbol{s}, \boldsymbol{r})$, where $\boldsymbol{s} \in \mathcal{S}^N$ are the particle species and $\boldsymbol{r} \in \mathcal{M}_L^N$ their positions embedded in the $d$-dimensional manifold $\mathcal{M}_L$ specified by the simulation box. We focus on three discrete mixture models based on the interaction models described in Section 2.2.3: two different binary mixtures [105, 115] composed of $N_\mathrm{A}$ particles of type A and $N_\mathrm{B}$ particles of type B, as well as a ternary mixture [159] that includes $N_\mathrm{C}$ particles of type C. All these glass-forming mixtures are also fairly robust against crystallization, at least in the temperature range accessible to conventional simulations. In each of the three models, we truncate and shift the potential at a cutoff distance $r^c_{\alpha\beta} = 2.5\sigma_{\alpha\beta}$, with $\alpha, \beta \in \{\mathrm{A, B, C}\}$.

The soft sphere model [115] is a 50:50 mixture with $N_\mathrm{A} = N_\mathrm{B} = 108$. The pair potential is given by the inverse power law in Eq. (2.9), with $\epsilon = 1.0$, $\sigma_\mathrm{AA} = 1.0$, $\sigma_\mathrm{AB} = \sigma_\mathrm{BA} = 1.2$, $\sigma_\mathrm{BB} = 1.4$, and exponent $n = 12$. The number density is fixed to $\rho = 0.5342$. We found that the onset of glassy dynamics occurs around $T \approx 0.25$ for this model. Note that the size ratio of this mixture is larger than the one used in earlier simulation studies employing the swap MC method [139, 140].

We then considered the binary Kob–Andersen mixture [105] composed of $N_{\mathrm{A}} = 173$ and $N_{\mathrm{B}} = 43$ particles at density $\rho = 1.2$. The interaction is given by the Lennard-Jones potential in Eq. (2.10) with parameters $\epsilon_{\mathrm{AA}} = 1.0$, $\epsilon_{\mathrm{AB}} = \epsilon_{\mathrm{BA}} = 1.5$, $\epsilon_{\mathrm{BB}} = 0.5$, $\sigma_{\mathrm{AA}} = 1.0$, $\sigma_{\mathrm{AB}} = \sigma_{\mathrm{BA}} = 0.8$, and $\sigma_{\mathrm{BB}} = 0.88$. The onset of glassy dynamics occurs around $T \approx 1.0$ [13].

Finally, we also study a variant of the Kob–Andersen mixture [159] characterized by three chemical species with $N_{\mathrm{A}} = 144$, and $N_{\mathrm{B}} = N_{\mathrm{C}} = 36$ at density $\rho = 1.2$. The interaction parameters of the additional C species interpolate between those of the A and B species of the original Kob–Andersen mixture. Namely, particles interact again with the Lennard-Jones potential in Eq. (2.10), where the interaction parameters for particles A and B are the same as in the original Kob–Andersen model, while $\epsilon_{\mathrm{AC}} = \epsilon_{\mathrm{CA}} = 1.25$, $\epsilon_{\mathrm{BC}} = \epsilon_{\mathrm{CB}} = 1.0$, $\epsilon_{\mathrm{CC}} = 0.75$, $\sigma_{\mathrm{AC}} = \sigma_{\mathrm{CA}} = 0.9$, $\sigma_{\mathrm{BC}} = \sigma_{\mathrm{CB}} = 0.84$ and $\sigma_{\mathrm{CC}} = 0.94$. Thanks to this modification, the model can be simulated efficiently using the swap MC algorithm even below the so-called mode-coupling crossover temperature, which is around 0.34 [159]. However, swap MC simulations struggle to equilibrate the system below $T \approx 0.3$.

We perform simulations that combine several moves in the same run, such as the displacement of a single particle and the swap of two particles of different species. Each move is independently associated with a reward $r$ and a policy $Q_\theta$, which corresponds to a specific parameterization of the proposal distribution. We also assign to each move a weight, which quantifies the probability of performing that type of move at each MC step. For the sake of simplicity, we fix these weights at the beginning of the simulation, and we optimise each policy independently. In a single MH step, once a type of move has been chosen, a transition $x \to x'$ is proposed according to the policy $Q_\theta$, and it is accepted with probability $\alpha_\theta$.

We perform a PGMC optimisation step for each move at every MH sweep (corresponding to $N$ MH steps), which defines our time unit. In order to estimate $\nabla_\theta J$ at each optimisation step, we employ the sampling strategy presented in Section 3.1.3. For this application, instead of running multiple chains in parallels or accumulating samples over time, we start from configuration $x_t$ and we draw $M$ samples from $K_\theta\left(x_t, \cdot\right)$ by performing $M$ independent MH sweeps. Although this approach is less efficient than running multiple chains in parallel because the samples are discarded in the end, it is straightforward to implement. Then, for each of these $M$ samples $\{x_m\}$, we draw $M'$ samples $\{x_n\}$ from $Q_{\theta_t}\left(x_m, \cdot\right)$. This process yields a total of $MM'$ samples for estimating $\nabla_\theta J$. The choice of $M$ and $M'$ depends on the temperature and on the specific optimisation algorithm used to update the parameters. In practice, we tuned the values of $M$, $M'$, and the learning rate $\eta$ so that the parameters converged in roughly the same amount of time as a standard MC simulation for data production. Typically $M, M' \in [5,\ 15]$ for VPG and $M, M' \in [20,\ 50]$ for NPG. Once convergence is achieved, we perform standard MC simulations to compute the relevant structural and dynamical quantities, performing standard consistency checks; see Section 3.2.4. The code used for these simulations was as an early prototype of the `Arianna.jl` and `ParticlesMC.jl` packages described in Appendix E. Analysis of the simulation data was carried out with `atooms-postprocessing` available at `https://framagit.org/atooms/postprocessing`. The data supporting this study are available on Zenodo at `https://zenodo.org/records/11396665`.

### 3.2.2 Designing moves

In order to apply PGMC to glass-forming mixtures, we need to define suitable moves, policies and rewards. In practice, as in the standard MH algorithm, this amounts to defining how to sample the proposal distribution $Q_\theta$ starting from the current state, and how to evaluate its density $q_\theta$ with respect to a reference measure that can be used to compute the acceptance probability $\alpha_\theta$ and the score function $s_\theta$. The only additional ingredient required by PGMC is the reward function $r(x, x')$, which quantifies the quality of each proposed move.

#### Gaussian displacements

To illustrate PGMC for glass-forming mixtures in the simplest possible case, consider the displacement of a single particle $i$ as described in Section 2.4.2. Because of assumption *i.* in Theorem 3.1.1, which requires local absolute continuity of the proposal distributions, we consider the Gaussian proposal defined in Eq. (2.28), which displaces particle $i$ from its current position $\vec{r}_i$ to a new position $\vec{r}_i'$ according to a normal distribution with variance $\sigma^2$. To optimise this move, we consider the family of policies $\{Q_\sigma\}$ defined by Eq. (2.28), where the standard deviation $\sigma > 0$ is the parameter to be optimised. Since the proposal kernel $Q_\sigma$ updates only the position of a single particle $i$, it does not have a density with respect to the Lebesgue measure on the full configuration space $\mathsf{X}$. However, since $Q_\sigma$ is a sum of $N$ mutually singular measures (one for each particle), we can define a density $q_\sigma^i$ on the marginal distribution of the position of particle $i$ as

$$q_\sigma^i\left(\vec{r}_i'|x\right) \;=\; \frac{1}{N\left(2\pi\sigma^2\right)^{3/2}} \exp\left(-\frac{d_{\mathcal{M}_L}\left(\vec{r}_i', \vec{r}_i\right)^2}{2\sigma^2}\right) . \tag{3.18}$$

In practice, this corresponds exactly to the standard way in which single-particle displacements are proposed in MCMC simulations of particle systems, even if the underlying measure-theoretic structure is not usually made explicit [110].

We set the reward to

$$r(x, x') = \sum_i d_{\mathcal{M}_L}(\vec{r}_i', \vec{r}_i)^2 \ , \tag{3.19}$$

so that small displacements (which are always accepted but inefficient) are discouraged. This is equivalent to minimising the lag-one autocorrelation among configurations [61, 63]. Figure 3.2 illustrates how the algorithm converges to the optimal value of $\sigma$ for the soft sphere mixture. In particular, panel (a) shows the evolution of $\sigma$ during optimisations at different temperatures, ranging from the normal liquid ($T = 5$) to the moderately glassy regime ($T = 0.2$). The converged value $\sigma^\star$ corresponds to a maximum of the expected reward $\widehat{J}$; see panel (b). We note, however, that minimising short-time correlations does not ensure a faster exploration of configuration space at long times. To quantify the latter, we compute instead the mean square displacement $\Delta^2$ defined in Eq. (2.24), and the corresponding diffusion coefficient

$$D \ \coloneqq \ \lim_{t\to\infty} \frac{\Delta^2(t)}{6t} \ . \tag{3.20}$$

As a reference, we obtained $D$ from several independent MC simulations using fixed values of $\sigma$. In Fig. 3.2(c), we see that the maximum of the expected reward is indeed very close to the maximum of the diffusion coefficient $D(\sigma)$. This shows, quite strikingly, that optimising the efficiency of the individual MH step also maximises the long time diffusion, possibly thanks to a coupling between short and long-time dynamics.

### Biased displacements

Displacement moves can be improved by encoding information on the local environment around a particle directly into the policy. This is a generalization of the so-called biased moves, well-known in standard MC methods [110]. As a natural extension of Eq. (3.18), we thus consider the following biased displacement: the proposed position $\vec{r}_i'$ is drawn from a Gaussian distribution whose mean includes a bias term aligned with the local force $\vec{f}_i(x)$ acting on particle $i$,

$$q^i_{\sigma,\lambda}(\vec{r}_i'|x) \ = \ \frac{1}{N\left(2\pi\sigma^2\right)^{3/2}} \exp\left(-\frac{d_{\mathcal{M}_L}\left(\vec{r}_i', \vec{r}_i - \lambda\sigma^2\vec{f}_i(x)/T\right)^2}{2\sigma^2}\right) \ . \tag{3.21}$$

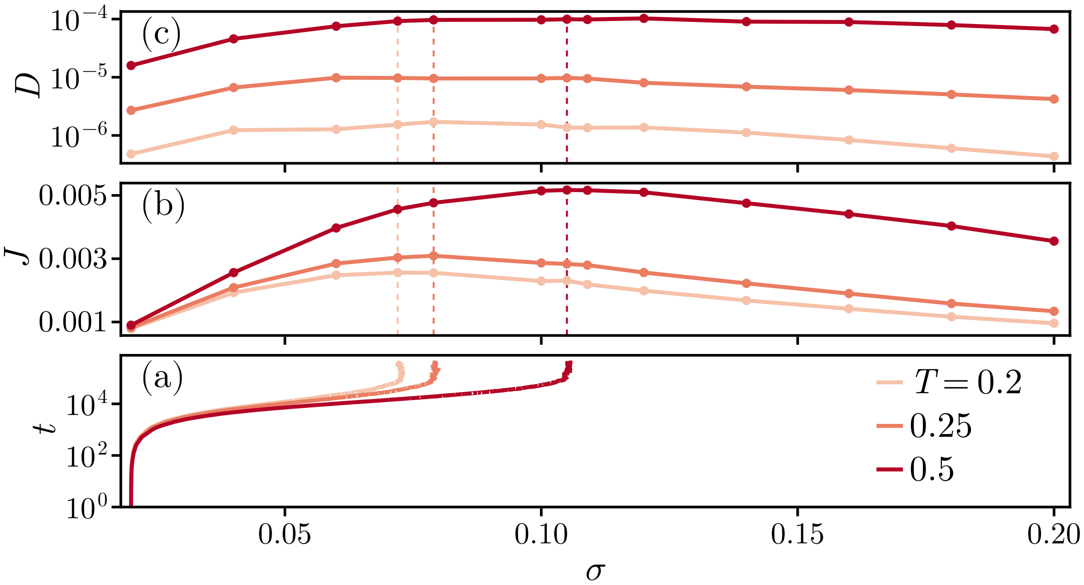

**Figure 3.2:** Parameter optimisation for Gaussian displacements. (a) Time evolution of $\sigma$ during a PGMC simulation for three selected temperatures. (b) Expected reward $J$ as a function of $\sigma$ obtained from standard MC simulations. (c) Diffusion coefficient $D$ as a function of $\sigma$ obtained from standard MC simulations. The dashed lines in panels (b) and (c) indicate the optimal parameters identified by PGMC in panel (a). Adapted from Ref. [42].

This is a reformulation in the PGMC framework of the “smart Monte Carlo” algorithm proposed by Rossky *et al.* [160], the only difference being the inclusion of the parameter $\lambda$ to adjust the amount of bias. Such a policy incorporates more detailed information about the current state $x$ of the system and leads to better performance compared to simple displacements. This is illustrated in the top panels of Fig. 3.3, for both the soft sphere and Kob–Andersen models for temperatures below the onset of glassy dynamics. The expected rewards of biased displacements are about a factor 2 larger than those of simple displacements, irrespective of temperature.

### Biased swaps

Despite their remarkable effectiveness in equilibrating polydisperse systems [140], swap moves are inadequate for most binary mixtures. As already mentioned, the crux is that binary mixtures with a size ratio close to unity can be simulated efficiently with swap moves [139], but also have a strong tendency to crystallise

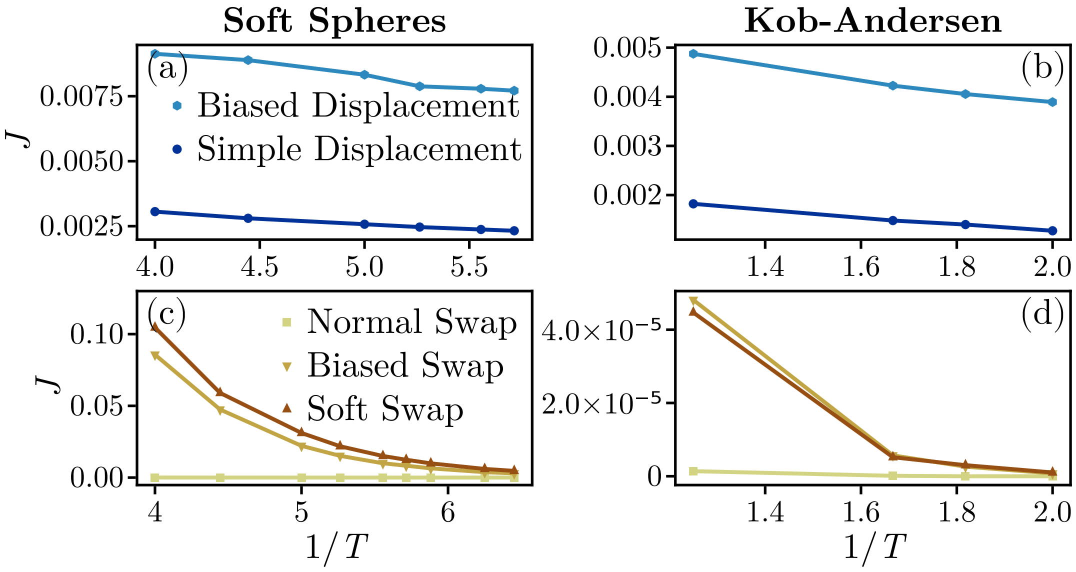

**Figure 3.3:** Reward comparison. Average rewards $J$ as a function of $1/T$ for displacement moves (panels (a) and (b)) and swap moves (panels (c) and (d)). Left and right panels display results for the soft sphere and Kob-Andersen models, respectively. Adapted from Ref. [42].

around the mode-coupling crossover temperature. By contrast, mixtures with a larger size ratio or non-additive interactions, such as the ones used in this work, are more robust against crystallisation, but the energy cost of enlarging the smaller particle dramatically suppresses the acceptance of swap moves; see Section 3.2.4. In the standard implementation of the swap move, the particle pairs are uniformly sampled from all the possible $N_\mathrm{A}N_\mathrm{B}/2$ pairs; see Eq. (2.29).

One possible strategy to improve the method is to incorporate information about the local structure into the pair selection, identifying those pairs whose swap is more likely to be accepted. A sketch of such a biased swap move is shown in Fig. 3.4(a). As a simple test of this idea, we check whether the local energy can be used to identify the most favorable pairs to swap. Figure 3.5 shows a scatter plot of pairs of local potential energies $(u_i(x), u_j(x))$, with particles $i$ and $j$ of species A and B, respectively. For each model, we consider equilibrium samples from MC simulations close to the respective onset temperature. We colour-code each pair according to the acceptance of the swap in the absence of bias, *i.e.*, $\log \alpha = \min \{0, -\left(U\left(x'\right) - U\left(x\right)\right)/T\}$. In the soft sphere mixture, we observe a systematic increase in acceptance as $e_A$ decreases: pairs with low local energy of the small particle have a higher probability of being swapped. In the Kob–Andersen mixture, the typical acceptance is lower by several orders of magnitude, and, crucially, it is not easy to identify the pairs with a larger $\log \alpha$ just by looking at their potential energies: the pairs with the highest probability of being swapped do not clearly lie at the boundaries of the energy distribution. These results indicate that introducing a simple bias on the local energy could make a more effective swap move for the soft sphere, much less so for the Kob–Andersen mixture.

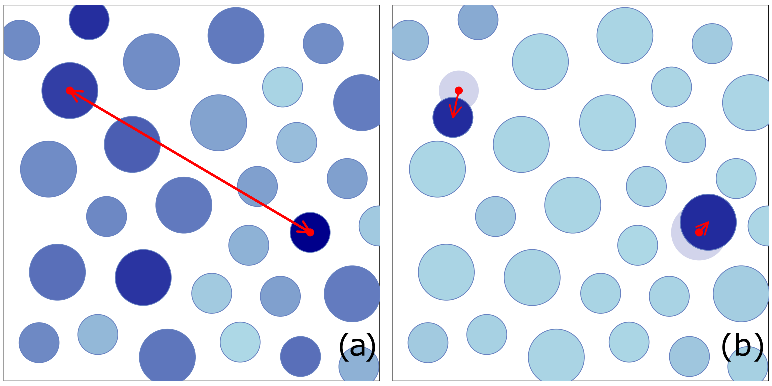

**Figure 3.4:** Sketches of (a) biased and (b) soft swap moves. In panel (a), the colour code indicates the probability of selecting the particle, with darker blue indicating higher probability; the arrow connects the particles whose swap is attempted. In panel (b), we show the additional biased displacements, indicated by the arrows, attempted after the swap. Reprinted from Ref. [42].

The PGMC method provides a natural framework for designing a move that exploits this information. We thus introduce a biased swap, in which particles $i \in \mathrm{A}$ and $j \in \mathrm{B}$ are chosen from a categorical distribution with a bias linked to their respective local energies. Given the sensitivity of the acceptance to local energy, a natural parametrisation for the proposal density of such a move restricted to the marginal distribution of species for particles $i$ and $j$ is

$$q^{ij}_{\theta_\mathrm{A},\theta_\mathrm{B}}\left(s_i, s_j|x\right) = \frac{\exp\left(\theta_\mathrm{A} u_i(x)\right)}{\sum_{k\in\mathrm{A}} \exp\left(\theta_\mathrm{A} u_k(x)\right)} \frac{\exp\left(\theta_\mathrm{B} u_j(x)\right)}{\sum_{k\in\mathrm{B}} \exp\left(\theta_\mathrm{B} u_k(x)\right)} \,. \tag{3.22}$$

The two parameters $\theta_\mathrm{A}$ and $\theta_\mathrm{B}$ control the sign and strength of the bias. To promote those swaps that are more likely to be accepted, we maximise the acceptance of the move and set the reward to $r\left(x, x'\right) = 1 - \delta\left(x - x'\right)$. Note that the computational complexity of the move is $O\left(N\right)$ unless the swap is restricted to a finite subset of (possibly neighboring) pairs. This is due to the fact that sampling a categorical distribution with $N_c$ categories takes $O\left(\log N_c\right)$ with binary search, but evaluating the normalisation factor in Eq. (3.21) requires $O\left(N_c\right)$ operations. In principle, this additional overhead could be mitigated by implementing a cache to store the normalisation factor. However, with each policy update, such a cache would have to be completely recalculated.

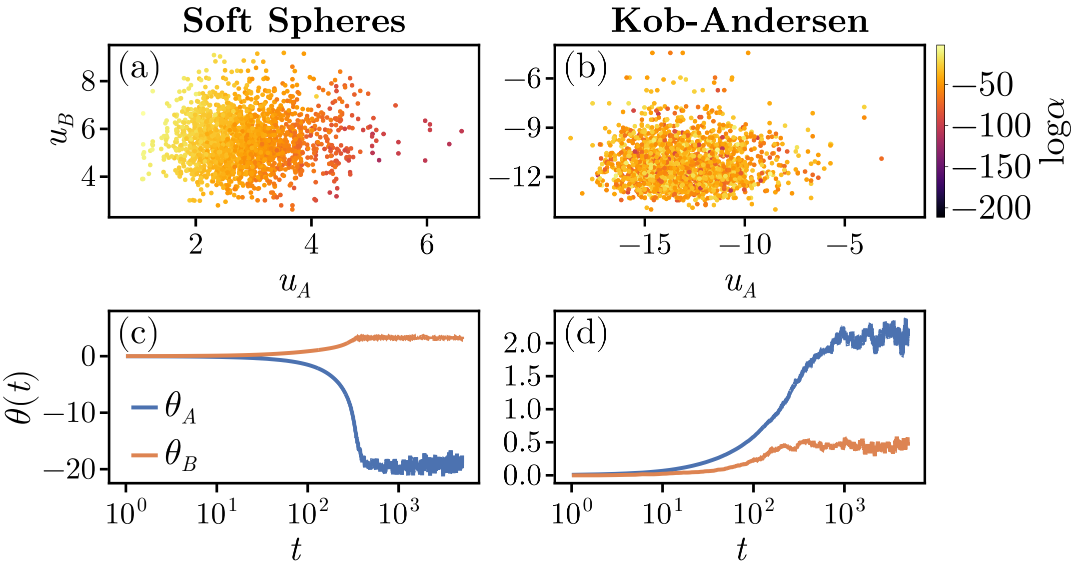

**Figure 3.5:** Biased swaps. Upper panels: scatter plot of the local energies $(e_\mathrm{A}, e_\mathrm{B})$ of pairs of particles before a normal swap from representative configurations at $T = 0.3$ and $T = 0.8$ in (a) the soft sphere and (b) Kob-Andersen models, respectively. The points are colour-coded by the acceptance $\log\alpha$ of the attempted normal swap. In the soft sphere mixture, we observe a systematic increase in acceptance as $e_A$ decreases: pairs with low local energy of the small particle have a higher probability of being swapped. In the Kob–Andersen mixture, the typical acceptance is lower by several orders of magnitude, and, crucially, it is not easy to identify the pairs with a larger $\log\alpha$ just by looking at their potential energies: the pairs with the highest probability of being swapped do not clearly lie at the boundaries of the energy distribution. These results indicate that introducing a simple bias on the local energy could make a more effective swap move for the soft sphere, much less so for the Kob–Andersen mixture. Lower panels: time evolution of $\theta_\mathrm{A}$ and $\theta_\mathrm{B}$ during a PGMC simulation for (c) soft sphere and (d) Kob-Andersen models. Adapted from Ref. [42].

The results for the optimisation of the parameters $\theta_\mathrm{A}$ and $\theta_\mathrm{B}$ are shown in Figs. 3.5(c) and 3.5(d). At least for the soft sphere model, the values of optimal parameters confirm our physical intuition: the gradient of $\log\alpha$ in Fig. 3.5(a) indicates that it is favorable to swap pairs in which the A-particle has low energy, and indeed we find $\theta_\mathrm{A} < 0$ and $|\theta_\mathrm{A}| > |\theta_\mathrm{B}|$. Note that it would be inefficient for the algorithm to just make $\theta_\mathrm{A}$ as small as possible; the optimal value results from the balance between the weights of the forward and inverse moves within Eq. 1.19. The optimised parameters for the Kob–Andersen mixture are instead both positive, which probably reflects the difficulty of finding an effective bias for this model. As we shall see below, in fact, biased swap moves are inefficient for this model; see Section 3.2.4. More sophisticated approaches may be required to enhance swap moves in the Kob–Andersen mixture.

#### Soft swaps

The primary source of energy cost in a swap move is due to the enlargement of the smaller particle; see Section 3.2.4. A potential strategy to mitigate this issue is to lower the local energy of such a particle, thereby reducing the energy difference between the two states. We thus introduce a composite MC move, where the two swapped particles undergo a biased displacement right after the swap. These displacements occur prior to the acceptance evaluation and potentially allow the system to tunnel between energy barriers. To sample such a composite move, particles $i \in \mathrm{A}$ and $j \in \mathrm{B}$ are again chosen from a categorical distribution, as in the biased swap move. Then, the two particles are independently shifted to new positions $\vec{r}_i'$ and $\vec{r}_j'$ as in the biased displacement move. A sketch of the additional step involved in such a soft swap move is shown in Fig. 3.4(b). The expected rewards $\widehat{J}$ obtained for soft swaps are shown in Fig. 3.3(d). Compared to biased swaps, soft swaps provide no improvement for the Kob–Andersen mixture. We observe instead a marginal increase in $\widehat{J}$ for the soft sphere mixture with respect to biased swaps. Such a difference may be deemed negligible when simulating glassy liquids, where large speed-ups are needed in order to achieve equilibrium at low temperatures. Nevertheless, this result shows that the PGMC framework can be used to implement complex moves in a constructive way, building on physical intuition as well.

### 3.2.3 Performance assessment

To compare the performance of the policies described in the previous sections, we consider standard time-dependent correlation functions used to quantify structural relaxation in glass-forming liquids [161] (see Section 2.3.2). In particular, we compute the self-intermediate scattering function $F_s(q^\star, t)$ defined in Eq. (2.25) at a wavevector $q^\star$ corresponding to the first peak of the static structure factor $S(q)$ defined in Eq (2.16). From $F_s(q,t)$, we extract the structural relaxation time $\tau_\alpha$ as $F_s(q^\star, \tau_\alpha) = 1/\mathrm{e}$.

In Fig. 3.6, we present $F_s(q,t)$ obtained from MH simulations of the soft sphere and Kob–Andersen mixture using simple displacements, simple displacements combined with swaps, and simple displacements combined with biased swaps. When combining displacements and swap moves, we set the probability of selecting a swap among the possible moves to $p_{\text{swap}} = 0.1$. Each simulation is performed using the optimal parameters obtained in a preliminary PGMC simulation. From the figure, we recognise the typical features of glassy dynamics: below the onset temperature $T_o$, the correlation functions develop a plateau due to the cage effect of the neighboring particles. The results for the soft sphere mixture illustrate well the effectiveness of the PGMC framework. At the lowest temperature, $T = 0.155$, neither simple displacements nor normal swaps can equilibrate this model. This can be appreciated by the fact that the plateau in $F_s\left(q^\star,t\right)$ stretches beyond the observation time scale of our simulations. However, simulations with soft swaps (and biased swaps, not shown) allow the correlation function to relax below 1/e even at this temperature.

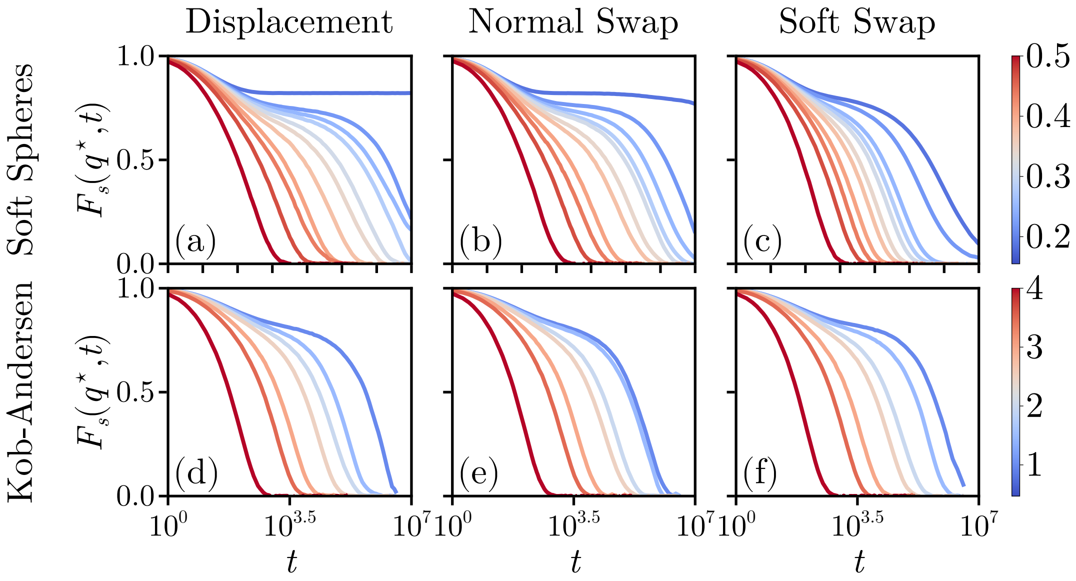

**Figure 3.6:** Glassy dynamics. Comparison of self intermediate scattering functions $F_s(q^\star,t)$ for selected temperatures in the soft sphere (top panels) and Kob-Andersen models (bottom panels). Adapted from Ref. [42].

Biased and soft swaps are, by contrast, of little use for the Kob–Andersen mixture. As shown in the bottom panels of Fig. 3.6, the decay of $F_s(q^\star,t)$ occurs on similar time scales irrespective of the simulation method. In the Kob–Andersen mixture, in fact, swap moves have negligible acceptance probabilities, and neither biased nor soft swaps can promote their acceptance. For this mixture, a small performance improvement can be achieved by introducing biased displacements; see below.

To provide a more quantitative evaluation of the performance, we show in Fig. 3.7 the temperature dependence of the structural relaxation times $\tau_\alpha$ for all the policies. For the soft sphere model, the performance of biased and soft swaps is nearly identical. Below the onset temperature $T_o$, these improved swap moves provide a substantial reduction in structural relaxation times compared to displacement moves and normal swaps. At the temperatures at which all these methods can achieve equilibrium, we estimate the speed-up of biased and soft swaps to be about two orders of magnitude. To estimate the speed-up at lower temperatures, we extrapolate the $\tau_\alpha$ obtained from displacements and normal swaps using the parabolic law [162]

$$\tau_\alpha(T) \;=\; \tau_0 \exp\left[J^2\left(\frac{1}{T}-\frac{1}{T_o}\right)^2\right] \,, \tag{3.23}$$

fitted in the range of temperatures below the onset temperature. At the lowest temperature at which biased and soft swaps can fully equilibrate the system, *i.e.*, $T = 0.16$, we estimate that the speed-up is still about 2 orders of magnitude. While this result is less spectacular than the one achieved by normal swaps in polydisperse particle models close to the laboratory glass transition temperature [140, 163] it is nonetheless a significant achievement. These configurations could be used, for instance, as starting points to study the equilibrium dynamics of this mixture below the mode-coupling crossover temperature [164]. The results for the Kob–Andersen mixture in panel (b) show instead that none of the swap moves brings any performance improvement over simple displacements. For this model, biased displacements nonetheless provide a speed-up of about a factor of 2, irrespective of temperature.

The physical interpretation of the above results is that biased swaps are effective in mixtures for which the acceptance of normal swaps is non-negligible, at least close to the onset temperature. In such systems, swap MC will eventually become inefficient upon cooling [140], as the standard Metropolis acceptance rate drops with decreasing temperature. Introducing an energy bias raises acceptance and shifts the numerical glass transition to a lower temperature. To illustrate these ideas and corroborate our results, we consider a ternary variant of the Kob–Andersen mixture, which has been equilibrated successfully using normal swap MC even below the

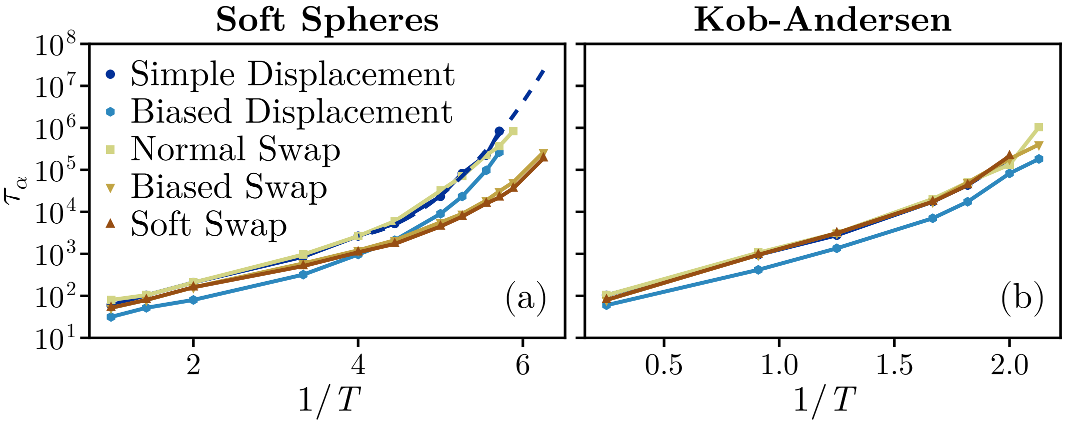

**Figure 3.7:** Assessing Performance. Structural relaxation time $\tau_\alpha$ for all policies in (a) soft sphere and (b) Kob-Andersen models. The dashed line in panel (a) indicates a fit to Eq. (3.23). In panel (a), the symbols for simple and biased displacements lie almost on top of each other, as do those for biased and soft swaps. In panel (b), the symbols for all moves except bias displacements lie almost on top of each other. Adapted from Ref. [42].

mode-coupling crossover temperature. We simulate the model around the lowest temperature at which normal swap MC achieves equilibration, $T = 0.28$. In Fig. 3.8, we show $F_s(q^\star, t)$ obtained for this model with all the available policies. We see that biased swaps reduce the relaxation times by almost an order of magnitude around this temperature. Biased swaps are thus a potential candidate to push simulations of this model close to experimental time scales.

Finally, to take into account the different computational costs of the policies, we also evaluated the typical wall time necessary to carry out a single move. Although the timing depends, in general, on implementation details and hardware, the *ratios* between wall times of different moves are fairly robust and can be estimated from the number of single-particle energy evaluations. We found that the overhead required by complex policies does not severely impact the performance of our PGMC code: biased displacements take about 1.15 times as long as simple displacements, while biased swaps take ≈1.4 times as long as normal swaps. Soft swaps are more costly as they take about twice as long as biased swaps. Focusing on the soft sphere model at $T = 0.16$, we found that biased (soft) swaps achieve an effective speed-up in the range of 90–100 (60–70), compared to normal swaps.

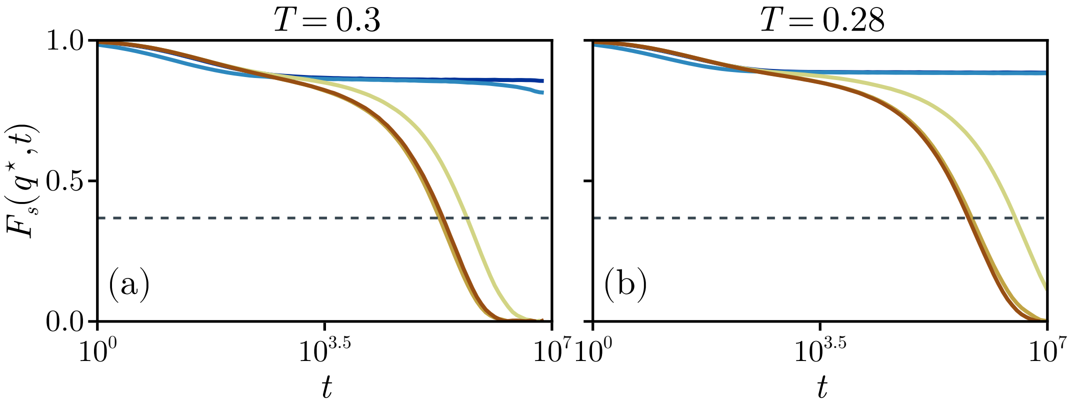

**Figure 3.8:** Ternary Kob-Andersen. Self intermediate scattering functions $F_s(q^\star, t)$ obtained from MH simulations for all policies in the ternary variant of the Kob-Andersen mixture at (a) $T = 0.3$ and (b) $T = 0.28$. The horizontal dashed line is drawn at 1/e. In both panels, the lines for simple and biased displacements lie almost on top of each other, as do those for biased and soft swaps. Adapted from Ref. [42].

### 3.2.4 Additional results

We conclude this section by presenting some additional results that further illustrate the properties of the PGMC method when applied to glass-forming mixtures. Specifically, we compare the performances of some of the optimisation algorithms described in Section 3.1.5, and we present various consistency checks to validate our implementation. Finally, to better understand the difference in performance of swap moves in the two models, we carry out a mean field analysis of their energy cost.

#### Comparison of optimisation algorithms

We compare the VPG, NPG and ANPG optimisation methods described in Section 3.1.5 for the soft sphere model in the normal liquid regime ($T = 5$) and slightly below the onset temperature ($T = 0.225$) in Fig. 3.9. Since ANPG requires tuning a different parameter $\epsilon$ compared to the learning rate $\eta$ of the other two methods, we adjusted $\epsilon$ so that ANPG and NPG had similar parameter variance once convergence was achieved at $T = 0.5$. For VPG and NPG, we set $\eta = 0.1$, while for ANPG, we used $\epsilon = 10^{-4}$. It is clear that ANPG leads to significantly faster convergence at low temperatures compared to VPG and NPG. Nevertheless, we point out that estimating $F(\theta)$ requires a higher number of samples compared to that needed for estimating $\nabla_\theta J(\theta)$. We used $M$, $M' \in [5, 15]$ for VPG and $M$, $M' \in [20, 50]$ for NPG and APG.

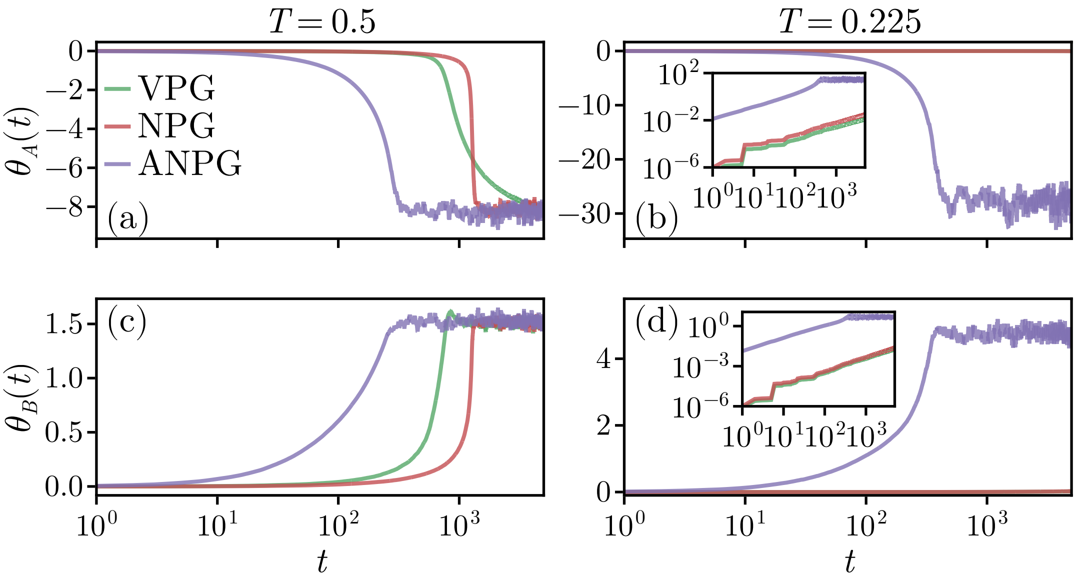

**Figure 3.9:** Optimisation methods. Overview of VPG (green), NPG (red), and ANPG (violet) at high temperature (left panels) and around the onset temperature (right panels) for the soft sphere model using biased swaps. Panels (a) and (b) show the time evolution of the parameter $\theta_{\mathrm{A}}$, while panels (c) and (d) the one of $\theta_{\mathrm{B}}$. The insets in panels (b) and (d) display the absolute value of the two parameters in log-log scale. Adapted from Ref. [42].

### Consistency checks

As a consistency check, we verify that detailed balance is preserved in our MC simulations by comparing in Fig. 3.10 the potential energy distributions for all the policies. Any discrepancies in these distributions would suggest an incorrect implementation of certain moves. All the policies yield overlapping energy distribution histograms in both models, indicating that each move samples the correct distribution.

Finally, we also made sure that the systems did not crystallise or phase separate at low temperatures by inspecting the relevant structure factors. In particular, we compute the total structure factor $S(q)$ in Eq. (2.16) and the average concentration–concentration structure factoe $S_{CC}(q)$ in Eq. (2.20) for trajectories. In Fig. 3.11, we show representative results for the soft sphere model at the lowest

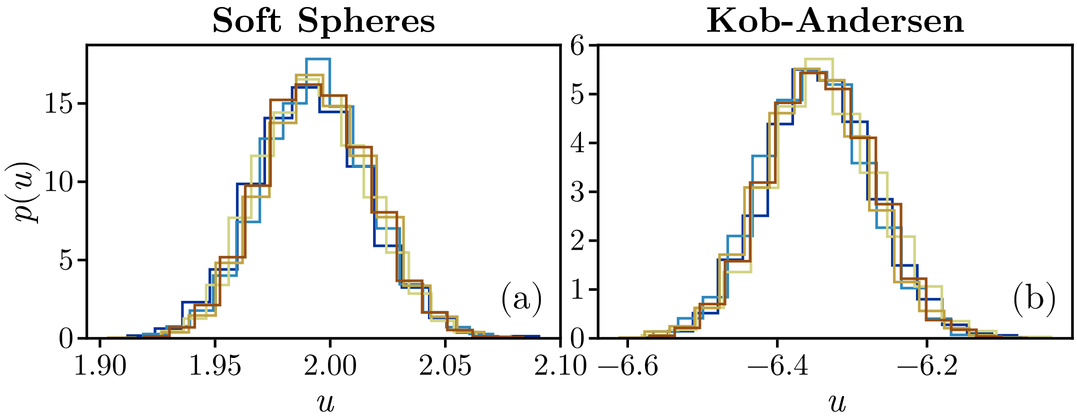

**Figure 3.10:** Potential energy distributions. (a) Soft sphere model at $T = 0.25$. (b) Kob-Andersen model at $T = 0.8$. Adapted from Ref. [42].

temperature, $T = 0.155$: there are no signs of crystallisation or phase separation in the shape of $S_{CC}(q)$ and $S(q)$.

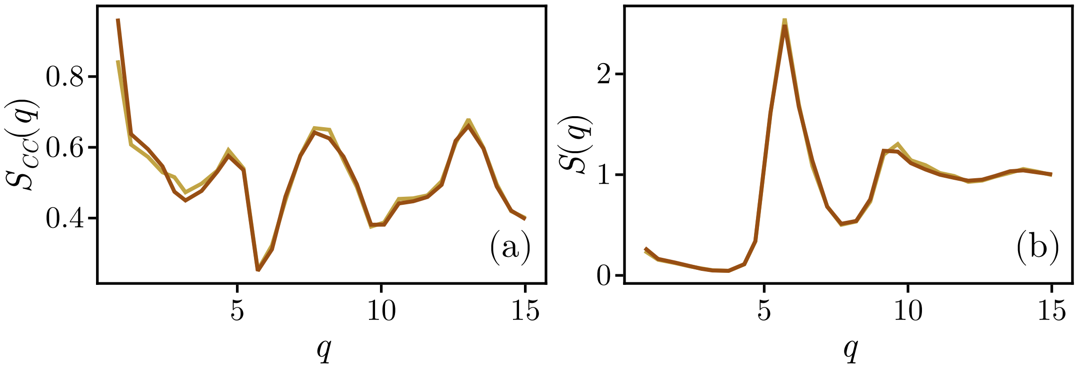

**Figure 3.11:** Structure factors. (a) Concentration-concentration structure factor $S_{CC}(q)$ and (b) total structure factor $S(q)$ at $T = 0.155$ in the soft sphere model for Biased Swaps and Soft Swaps. Adapted from Ref. [42].

### Energy cost of swap moves

To better understand the difference of swap acceptance rates between the soft spheres and Kob-Andersen models, we carry out a spatially resolved analysis of the energy cost associated with a swap move. We consider the swap between a particle of species A and a particle of species B. To simplify the analysis, we consider only the *average* energy change due to such a swap. Before the swap, the corresponding average single-particle energies of the two particles are $u_\mathrm{A}$ and $u_\mathrm{B}$, respectively, with

$$u_\alpha = \sum_\beta 4\pi\rho x_\alpha x_\beta \int_0^\infty r^2 g_{\alpha\beta}(r) u_{\alpha\beta}(r)\,\mathrm{d}r \ , \tag{3.24}$$

where $g_{\alpha\beta}(r)$ are the partial pair correlation function for particles of species $\alpha$ and $\beta$ defined in Eq. (2.13).

The corresponding average single-particle energies after the swap are denoted by $u'_\mathrm{A}$ and $u'_\mathrm{B}$, and the average energy difference due to the swap is $\Delta u_\alpha = u'_\alpha - u_\alpha$, with $\alpha = \mathrm{A}, \mathrm{B}$. Since the particle positions remain the same when attempting the swap, we can write the average energy change for the particle of species $\alpha$ as

$$\Delta u_\alpha = \sum_\beta 4\pi\rho \int_0^\infty x_\alpha x_\beta g_{\alpha\beta}(r) \big(u_{\alpha'\beta}(r) - u_{\alpha'\alpha}(r)\big)\,\mathrm{d}r \ , \tag{3.25}$$

where $\alpha$, $\alpha'$ are the species before and after the swap, respectively. The total average energy cost due to a swap is then as $\Delta u = \Delta u_\mathrm{A} + \Delta u_\mathrm{B}$.

To get some insight into the contributions to $\Delta u$, we inspect the integrands $\delta u_\alpha(r)$ in Eq. (3.25) as a function of $r$ using the actual $g_{\alpha\beta}(r)$ obtained from equilibrium configurations. In Fig. 3.12 we plot the two terms $\delta u_\mathrm{A}$ and $\delta u_\mathrm{B}$, normalised by the temperature, for our two benchmark models close to the onset temperature. From the different sign of the integrands, we recognise that, as expected, what hinders particle swaps is the energy cost of expanding the small particle (A for soft spheres, B for the Kob-Andersen model). In the soft sphere model, the energy change due to shrinking a particle is always negative and has largest contributions from particles at a distance of about 1.4, which corresponds to the most favorable distance between A and B species. By contrast, the energy change of the A-particles in the Kob-Andersen mixture has both positive and negative contributions, and the latter are negligible compared to large energy cost of $\delta u_\mathrm{B}$. The difference in acceptance rates between the two models likely reflect the discrepancies seen in Fig. 3.12. Our analysis also suggests that it may be possible to build a more efficient biased swap by targeting those small particles whose local environments have an excess of large particles in correspondence to the minima of $\delta u_\alpha$, *i.e.*, $r \approx 1.4$ for the soft sphere model and $r \approx 1$ for the Kob-Andersen mixture. However, we expect the improvement to be negligible in this latter model, due to the overall much lower acceptance rates.

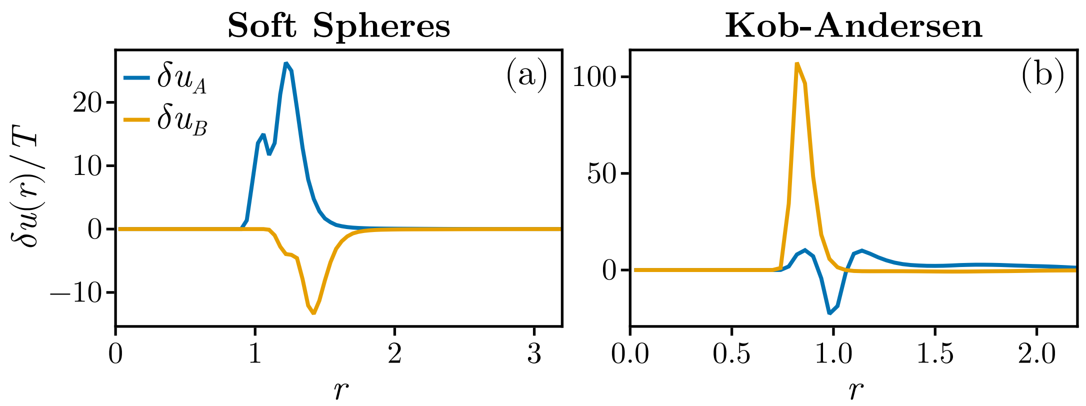

**Figure 3.12:** Energy cost of swap moves. Integrands $\delta u_\alpha(r)$ of Eq. (3.25) divided by $T$ for (a) the soft sphere model at $T = 0.3$ and (b) the Kob-Andersen model at $T = 0.8$. Adapted from Ref. [42].

## 3.3 Discussion and perspectives

In this chapter, we have formally extended the PGMC method [41] to deal with a general state space, comprising, for instance, both continuous and discrete degrees of freedom. By recasting the adaptive MH as a reinforcement learning problem and working directly with the measure-theoretic formulation of MH, PGMC provides a general and flexible framework for constructing parametrised moves. The generalisation of the policy-gradient theorem [64] in Theorem 3.1.1, together with the explicit gradient estimators derived from it, establishes a solid theoretical foundation for the method.

Compared to the original formulation of PGMC [41], our generalisation allows one to design moves acting on both continuous and discrete variables, and it corrects ambiguities in the treatment of the MH kernel that become problematic when $r(x, x) \neq 0$. The original work by Bojesen also introduced the notion of "chain policies", where multiple moves are combined into a single one. However, determining the proposal density for these composite moves is not trivial due to the many pathways through which the system could transition between two states when multiple random moves are combined. This issue can, at least in principle, be resolved by imposing path-wise detailed balance [165, 166], but it remains unclear how to construct chain policies that simultaneously modify different degrees of freedom—*e.g* positions and spins—in a consistent way. For this reason, while the original PGMC algorithm relied on short trajectories to estimate gradients, our version of PGMC employs single transitions $x \to x'$. Note that collective moves remain possible in our approach, but only when the transition probability between consecutive states can be written explicitly, independently of the specific path.

Adaptive MCMC methods generally fall into two categories. On the one hand, classical methods focus on optimising a handful of parameters of simple, general-purpose proposal distributions such as Gaussian random walks [24]. On the other hand, modern machine learning approaches based on generative models aim at generating entirely new configurations from scratch using deep neural networks trained to approximate the target distribution [29]. PGMC lies in between these two paradigms. While compatible with both viewpoints, it is particularly well suited for optimising user-designed moves built on physical intuition that would be difficult to tune by hand. The efficiency of this approach depends on the ability to desing *ad hoc* parametric moves whose performance can be improved through optimisation.

Our applications to glass-forming mixtures illustrate both the potential and the limitations of this approach. We have shown that a simple set of biased moves based on elementary particle swaps increase the sampling efficiency by two orders of magnitude in a mixture of additive soft spheres, for which simple displacement and swap moves become soon inadequate when lowering the temperature. By contrast, the same moves achieved essentially no speed-up in the well-studied Kob–Andersen mixture. In this model, we found a $\times 2$ speed-up with biased displacements, which is, however, insufficient in the context of glass transition studies [3].

Many variants of the biased moves used here can be adapted to other glass-forming mixtures. For instance, one could design a biased swap move for polydisperse systems that discourages swaps when the particle radii are too close. In systems where local energies are not relevant, such as hard spheres, alternative local descriptors could be used [132]. In principle, one could consider complex parameterisations relating the moves' parameters to the local structure, *e.g.*, through neural networks that learn a better local descriptor. Nevertheless, we believe it is unlikely that these more complex parameterisations could significantly speed up simulations: when moves are fundamentally local, performance is largerly determined by the their design.

Extensions to other types of moves are also natural. PGMC can be used to optimise "flips" in molecular liquids [145], bond-switching moves in network glasses [167], or any proposal for which a tractable density can be written down. The modularity of the framework also makes it suitable for problems beyond glassy liquids. Potential applications include lattice models, Bayesian inference on high-dimensional spaces, and simulations involving dimensional-changing kernels. The implementation provided through `Arianna.jl` offers a flexible platform to explore these possibilities in full generality, bridging conventional MCMC methods and modern differentiable programming.

Of course, these ideas can be pushed in more ambitious directions. The PGMC framework allows to design composite moves that combine multiple elementary updates, such as the soft swap described in this chapter. A promising direction involves implementing more complex collective moves in which multiple degrees of freedom are updated simultaneously [136, 168, 169]. PGMC could provide a tool to build such collective moves autonomously using more flexible policies with a large number of parameters. Generative models already provide powerful building blocks for such proposals [29, 68, 69, 156]. Auto-regressive networks or normalising flows, in particular, could provide arbitrarily complex policies that can be directly sampled. The PGMC reward naturally guides their training toward conditional updates on selected variables, which can be especially beneficial for acceptance rates. Implementing these ideas in the PGMC framework represents an interesting extension to be explored in future work.

# Chapter 4

# Equivariant Riemannian Stochastic Interpolants

> *"I believe this project is somehow cursed."*

Equivariant Riemannian Stochastic Interpolants (eRSI) are flow-based generative models specifically designed to sample multicomponent particle systems with periodic boundary conditions (PBCs). This approach is in some sense orthogonal to the one presented in Chapter 3, where we improved a MCMC algorithm that, in the infinite-time limit, samples the *exact* Boltzmann distribution. Generative models, by contrast, sample *approximately* the target distribution. For this approximation to be useful in statistical physics, the models must provide exact *likelihoods* for the generated configurations, so that unbiased estimators of thermodynamic observables can be obtained through importance sampling or by using the generative model as a proposal mechanism in MCMC algorithms [29, 69]. Instead of relying on a dynamical process to reach equilibrium, these methods learn a direct transformation from a simple reference measure to an approximation of the equilibrium distribution. In doing so, they produce *independent* samples, which is particularly relevant for systems with slow dynamics where MCMC samples are highly correlated.

Generative models have recently shown great promise in the context of molecular simulations [29, 68, 156], ranging from biomolecules [170–172] to crystals [37, 173–175]. Much less work, however, has targeted glassy systems. Continuous normalising flows (CNFs) trained with maximum likelihood can in principle model the equilibrium distribution exactly, but their training cost grows rapidly with system size [158]. Diffusion-based models, by contrast, efficiently generate visually realistic amorphous configurations but lack tractable likelihoods [176]. As a consequence, they cannot be used to compute unbiased thermodynamic observables.

The eRSI framework combines the scalability of Riemannian flow matching [43, 81] with the symmetry-preserving properties of equivariant flows [35, 36, 44]. Combining these ideas so that they can be applied to particle systems is not straightforward and requires some care. First, under PBCs, the configuration space has a toroidal structure (see Section 2.2.1), so we must design interpolation paths and velocity fields that respect these geometric constraints. Second, since the Boltzmann distribution in Eq. (2.5) has characteristic symmetries, we must ensure that the generative model shares these symmetries to avoid inefficiencies in learning unphysical distributions (see Section 1.3.4). By doing so, we provide an original theoretical contribution: unlike prior works applying very similar methods to crystalline systems [37, 174], we formally state and prove the conditions on the interpolation path that guarantee equivariance of the optimal velocity field and invariance of the optimal density at all times. Third, the parametric velocity field used to approximate the optimal one must be designed to satisfy the required equivariance properties while remaining expressive and efficiently learnable. We achieve this by adapting the graph neural network of [84] so that it respects the full set of symmetries relevant for multicomponent particle systems under PBCs.

We emphasise that a crucial feature of our method is the availability of exact likelihoods. This requirement is often irrelevant in generative modelling for images and text, and is sometimes overlooked in physical applications. Yet, it is essential for computing unbiased thermodynamic observables. Notably, diffusion-based models on the toroidal manifolds [173, 177–179] do not provide exact likelihoods.

To validate our framework, we apply it to representative amorphous systems in two dimensions and benchmark it against symmetry- and geometry-agnostic baselines. Our approach yields higher quality generations than these baselines, both at the level of individual configurations and of averaged physical observables, while also exhibiting better scalability with system size

This chapter is based on the article *Riemannian Stochastic Interpolants for Amorphous Particle Systems*, written in collaboration with Louis Grenioux, Giulio Biroli, Ludovic Berthier, and Marylou Gabrié [45].

## 4.1 Theoretical framework

We begin by presenting the theoretical framework underlying eRSI. It builds directly on stochastic interpolants (SI) (see Section 1.3.3) and equivariant flows (see Section 1.3.4). The main difference is the requirement that the interpolation paths are equivariant under the symmetries of the target distribution. The motivation for this constraint will become clear shortly.

Recall that the idea of SI is to learn a velocity field that transports samples from a simple reference distribution to a target distribution using observed data [31, 80]. Once learned, this velocity field defines a continuous normalising flow (CNF) that can be used to generate new samples from an approximation of the target. The quality of this approximation naturally depends on how accurately the velocity field is learned.

Our goal is to use the SI framework to sample from the Boltzmann distribution in Eq. (2.5) for multicomponent particle systems with PBCs. This requires addressing three challenges. First, as discussed in Section 1.3.4, incorporating symmetries in generative models of physical sysmtes is crucial to improve their efficiency. Although symmetries have been included in various families of normalising flows [44, 83, 84, 180, 181], even in Riemannian settings [36], their role within the SI framework (and flow-matching approaches more broadly) has not been systematically addressed. This is the origin of the "equivariant" aspect of eRSI. Second, the configuration space of these systems is a non-Euclidean manifold due to the PBCs (Section 2.2.1). Interpolation paths and velocity fields must therefore remain compatible with this geometric structure [43, 81]. This is the "Riemannian" part of eRSI. Third, we must design a parametric architecture that approximates the optimal velocity field while respecting the required equivariance properties. In this section, we address these three points in turn.

### 4.1.1 Symmetries in stochastic interpolants

As noted in Section 1.3.4, imposing invariances in CNFs reduces to ensuring that the parametrised velocity field $\hat{v}_\theta$ is equivariant under the symmetries of the target distribution. In the SI framework, however, there is an additional requirement: $\hat{v}_\theta$ is trained to approximate the optimal velocity field $v^\star$ defined by the interpolation path (see Theorem 1.3.2). Hence, for the mean-squared loss in Eq. (1.36) to be meaningful, the optimal velocity field itself must be equivariant at all times. This ensures that $\hat{v}_\theta$ is regressed onto a target sharing the same symmetry. It turns out that this property can be guaranteed by choosing an appropriate interpolation path.

**Definition 4.1.1.** An interpolation path $I : [0,1] \times \mathsf{X} \times \mathsf{X} \to \mathsf{X}$ is said to be *G-equivariant* with respect to a group $G$ if for all $g \in G$, $t \in [0,1]$, and $x_0, x_1 \in \mathsf{X}$ it holds that $I\left(t, g\left(x_0\right), g\left(x_1\right)\right) = g\left(I\left(t, x_0, x_1\right)\right)$.

**Theorem 4.1.1.** *Let $G$ be an isometry group of $\mathsf{X}$, and let $x(t)$ be the stochastic interpolant defined by the interpolation path $I : [0,1] \times \mathsf{X} \times \mathsf{X} \to \mathsf{X}$ and the coupling $\nu$ between $Q_0$ and $Q_1$. If $\nu$ is $G$-invariant and $I$ is a $G$-equivariant, then the optimal velocity field $v^\star$ defined in Eq.* (1.35) *is $G$-equivariant for all $t \in [0,1]$.*

*Proof.* See Appendix C.1 □

We emphasise that this result holds for *any* isometry group $G$. To our knowledge, the equivariance of the optimal velocity field in SI—and, more generally, in flow-matching frameworks—had not been discussed before. Theorem 4.1.1 thus extends the theory of equivariant flows developed by Köhler *et al.* [44] and Katsman *et al.* [36] to the SI framework. A direct consequence is that the marginal distribution $q_t$ of the stochastic interpolant is $G$-invariant for all $t \in [0,1]$ (see Proposition C.1.2). Since the law of the optimal CNF coincides with that of the stochastic interpolant (see Theorem 1.3.2), the density of the optimal CNF is likewise $G$-invariant at all times.

In practice, to ensure that $v^\star$ is equivariant, one must construct an equivariant interpolation path and a $G$-invariant base distribution $Q_0$. The target distribution $Q_1$ is invariant by construction, since the symmetries are precisely those we aim to model. When using a non-trivial coupling $\nu$ one must likewise ensure that $\nu$ is invariant. Interestngly, when $\nu$ is chosen as the *optimal transport coupling* [82] between two invariant distributions $Q_0$ and $Q_1$, then Ref. [35, Theorem 1] ensures that $\nu$ is itself invariant.

### 4.1.2 Defininining the interpolant

The first step in eRSI is to define a suitable stochastic interpolant between a base distribution $Q_0$ and a target distribution $Q_1$ (see Definition 1.3.1). Since the goal is to sample from the Boltzmann distribution in Eq. (2.5), we obviously set $Q_1 = P$. In practice, samples from $P$ are provided by equilibrium configurations obtained with other simulation methods (see Section 4.2). For the base distribution $Q_0$, we choose the uniform distribution on the configuration space $\mathsf{X}$, implemented by drawing a random permutation of the species and sampling positions uniformly in the simulation box. For now, we keep $\nu$ as a general coupling between $Q_0$ and $Q_1$.

The next step is to specify the interpolation path $I$ between samples from $Q_0$ and $Q_1$. Since the configuration space $\mathsf{X}$ of amorphous systems is a Riemannian manifold, paths must remain entirely within the simulation box. These paths should also be as short as possible to reduce the complexity of the learning task [35, 43, 81, 82]. Following Ref. [43] and Section 1.3.3, a natural choice is the geodesic connecting $x_0$ and $x_1$ on $\mathsf{X}$.

Recall from Section 2.2 that $\mathsf{X}$ decomposes as $\mathsf{X} = \mathcal{S}^N \times \mathcal{M}_L^N$, *i.e.* a product manifold of the species space $\mathcal{S}^N$ and the position space $\mathcal{M}_L^N$. The species space is Euclidean, so the geodesic path is the linear interpolation between $\boldsymbol{s}_0$ and $\boldsymbol{s}_1$. The position space $\mathcal{M}_L^N$, instead, is the product of $N$ flat tori of side $L$, where geodesics are given by the exponential and logarithmic maps of Example A.2.2. The geodesic interpolation path is therefore

$$I\left(t, x_0, x_1\right) \; = \; \begin{pmatrix} \left(1-t\right) \boldsymbol{s}_0 + t \boldsymbol{s}_1 \\ \left(\boldsymbol{r}_0 + t\left(\left(\boldsymbol{r}_1 - \boldsymbol{r}_0 + \dfrac{L}{2}\right) \% \, L - \dfrac{L}{2}\right)\right) \% \, L \end{pmatrix} , \qquad (4.1)$$

where $\%L$ denotes the modulo operation with respect to the box side $L$ applied component-wise. This path remains on $\mathsf{X}$ by construction and is twice differentiable with respect to its arguments almost everywhere under $\nu$, thereby satisfying the requirements of Definition 1.3.1. In Proposition C.3.1 we show that this interpolation path is equivariant with respect to the symmetries of the Boltzmann distribution, satisfying the conditions of Theorem 4.1.1. Combined with the invariance of $Q_0$, this implies that the optimal velocity field $v^\star$ is equivariant for all times. Proposition C.3.2 gives the explicit form of $v^\star$:

$$v^\star(t, x) = \mathbb{E}_{(x_0,x_1)\sim\nu}\left[ \begin{matrix} \boldsymbol{s}_0 - \boldsymbol{s}_1 \\ \left(\left(\boldsymbol{r}_1 - \boldsymbol{r}_0 + \dfrac{L}{2}\right) \% \, L - \dfrac{L}{2}\right) \% \, L \end{matrix} \middle| \, x(t) = I(t, x_0, x_1) \right] . \qquad (4.2)$$

This is the target onto which the parametrised velocity field $\hat{v}_\theta$ is regressed during training.

### Fixing the species

While the species are elements of a discrete set (with $|\mathcal{S}| = N$ for continuously polydisperse systems), we can still define continuous interpolation paths in the embedding Euclidean space. The important point is that the interpolation path must start and end at valid species vectors, *i.e.* $\boldsymbol{s}_0, \boldsymbol{s}_1 \in \mathcal{S}^N$. A simple choice that we adopt here is to keep the species *fixed* along the interpolation path, *i.e.* $\boldsymbol{s}_0 = \boldsymbol{s}_1 = \tilde{\boldsymbol{s}}$. This choice is particularly convenient when combined with an optimal transport coupling (see next paragraph), as it allows to match particles only within the same species, thereby reducing transport distances significantly. In practice, we order $\boldsymbol{s}$ such that particles of the same species are contiguous, and we match particles only within each group. Since the position distribution is uniform, this ordering does not affect the positional part of the base distribution $Q_0$.

#### Optimal transport coupling

A simple choice of coupling $\nu$ is the independent coupling, where $X_0 \sim Q_0$ and $X_1 \sim Q_1$ independently. However, as noted in Ref. [31, 82], a better option is to construct an approximation of the optimal transport (OT) coupling between $Q_0$ and $Q_1$. Following Refs. [35, 182–184], we can obtain such approximation by solving a linear assignment problem between samples from $Q_0$ and samples from $Q_1$. Given two configurations $x_0 = (\tilde{\boldsymbol{s}}, \boldsymbol{r}_0)$ and $x_1 = (\tilde{\boldsymbol{s}}, \boldsymbol{r}_1)$ with identical species $\tilde{\boldsymbol{s}}$, each is partitioned by species, and within each group the Hungarian algorithm [185] is applied using the nearest image distance (A.20) as cost. In practice, this procedure finds the optimal permutation of particle indices in $x_1$ that minimises the total distance to the particles in $x_0$, effectively mapping each initial particle to its closest counterpart in the target configuration. This OT coupling, consistent with the permutation invariance, shortens particle transport paths significantly. Note that Ref. [35] also accounts for other invariances in their solution of the OT problem, such as rotations and translations. We do not consider these here, although doing so would be possible in principle.

### 4.1.3 Architecture

In order to model $v^\star$, which is $G_{\mathsf{X}}$-equivariant by Theorem 4.1.1, we need to design a parametric velocity field $\hat{v}_\theta : [0,1] \times \mathsf{X} \to T\mathsf{X}$ that is also $G_{\mathsf{X}}$-equivariant. As commonly done in the literature of modern generative models of particle systems [37, 84, 177] we parametrise $\hat{v}_\theta$ with an equivariant graph neural network (GNN). The idea of GNNs is that a particle configuration can be seen as a graph whose nodes represent particles and whose edges carry information such as relative displacements or distances from other particles. A GNN processes this graph by iteratively passing "messages" along edges and updating each particle's latent state based on information from its neighbours. GNNs can be designed to be equivariant to translations, rotations, and permutations of particles by carefully defining how messages are computed and aggregated [84]. Here, we adapt the $E(n)$-equivariant GNN architecture of Satorras *et al.* [84] to instead respect the symmetry group $G_{\mathsf{X}}$ of multicomponent particle systems with PBCs.

First, note that $\hat{v}_\theta$ can be decomposed into a species part and a positional part as $\hat{v}_\theta = (\hat{v}_\theta^{\boldsymbol{s}},\, \hat{v}_\theta^{\boldsymbol{r}})$. Since the species are fixed along the interpolation path, we set $\hat{v}_\theta^{\boldsymbol{s}} = \mathbf{0}_N$. For the positional part $\hat{v}_\theta^{\boldsymbol{r}}$, we adapt the GNN of [84] by replacing the update of particle positions at each layer with an update that is aware of PBCs. In practice, the network constructs $K+1$ layers of a fully-connected graph connecting all particles. At each layer $k$, each node has a *position* variable $\vec{R}_i^k \in \mathcal{M}_L$ and a *feature* variable $H_i^k \in \mathbb{R}^2$ encoding additional information. At layer $k=0$, the position variables are initialised to the input positions, *i.e.* $\vec{R}_i^0 = \vec{r}_i$, while the feature variables are initialised to embed time and particle species, *i.e* $H_i^0 = (t,\, s_i)$. The GNN architecture then iterates from layer $k$ to layer $k+1$ as follows

$$M_{ij}^k = \hat{\phi}_e\left(H_i^k, H_j^k, d_{\mathcal{M}_L}\left(\vec{r}_i, \vec{r}_j\right)^2\right) \tag{4.3}$$

$$P_i^k = \sum_{j\neq i} M_{ij}^k \odot \hat{\phi}_m\left(M_{ij}^k\right) \tag{4.4}$$

$$H_i^{k+1} = \hat{\phi}_h\left(H_i^k, P_i^k\right) \tag{4.5}$$

$$\vec{R}_i^{k+1} = \left(\vec{R}_i^k + \sum_{j\neq i} \frac{\left(\left(\vec{R}_i^k - \vec{R}_j^k + \frac{L}{2}\right) \% \, L - \frac{L}{2}\right) \% \, L}{d_{\mathcal{M}_L}\left(\vec{r}_i, \vec{r}_j\right) + 1} \hat{\vec{\phi}}_d\left(M_{ij}^k\right)\right) \% \, L \,, \tag{4.6}$$

where $\hat{\phi}_e : \mathbb{R}^5 \to \mathbb{R}^n$, $\hat{\phi}_m : \mathbb{R}^n \to \mathbb{R}^n$, $\hat{\phi}_h : \mathbb{R}^{n+2} \to \mathbb{R}^2$, $\hat{\vec{\phi}}_d : \mathbb{R}^n \to \mathbb{R}^d$ are neural networks, and $n$ is the *embedding dimension* of the GNN. Basically, Eq. (4.6) updates the position variables by aggregating contributions from neighbouring particles, weighted by a function of their pairwise distance. After $K$ layers, we set $\boldsymbol{R}_\theta^K := \left(\vec{R}_1^K, \ldots, \vec{R}_N^K\right)$ and define the positional part of the velocity field as

$$\hat{v}_\theta^{\boldsymbol{r}}(t, x) = \left(\left(\boldsymbol{R}_\theta^K - \boldsymbol{r} + \frac{L}{2}\right) \% \, L - \frac{L}{2}\right) \% \, L \,. \tag{4.7}$$

The full velocity field is then $\hat{v}_\theta(t, x) = (\mathbf{0}_N,\, \hat{v}_\theta^{\boldsymbol{r}}(t, x))$. The parameters $\theta$ of the model are all the weights of the neural networks $\hat{\phi}_e$, $\hat{\phi}_m$, $\hat{\phi}_h$, and $\hat{\vec{\phi}}_d$.

In Proposition C.3.3, we show that this architecture is indeed $G_{\mathsf{X}}$-equivariant. This can be visually seen in Fig. 4.1, where we show that applying a box-symmetry transformation $g \in G_{\mathsf{X}}$ to the input configuration results in the same transformation applied to the output velocity field.

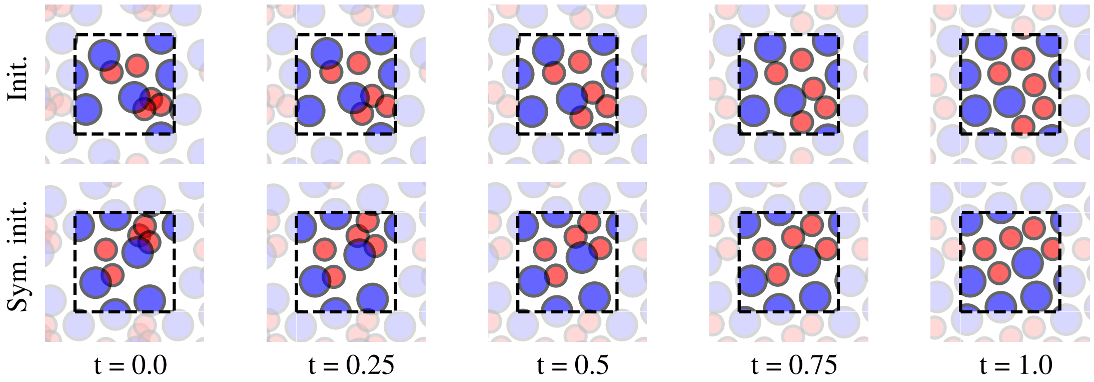

**Figure 4.1:** Equivariance of $\hat{v}_\theta$. Two trajectories of particle configurations generated by the CNF defined by the velocity field $\hat{v}_\theta$ in Eq. (4.7). The intermediate configurations are related by the same box-symmetry transformation applied to the initial configuration. Reprinted from Ref. [45].

### 4.1.4 eRSI in practice

We now have all the elements to *train* a generative model within the eRSI framework. First of all, note that any SI method requires existing data from the target distribution $Q_1$ to learn the velocity field. This may seem counter-intuitive, as the goal is precisely that of generating new samples from the Boltzmann distribution. However, the idea is that generating new samples from a trained generative model is much cheaper than using traditional simulation methods such as MCMC or molecular dynamics.

Training the eRSI model consists in learning the velocity field $v^\star$ in Eq. (4.2) using the loss function in Eq. (1.36). Since species remain fixed along the interpolation path, only the positional component of the velocity field must be learned. In our setting, since the Riemannian metric for the flat torus coincides with the Euclidean one, the loss can be rewritten as

$$\mathcal{L}\left(\theta\right) = \int_0^1 \mathbb{E}_{(x_0,x_1)\sim\nu}\left[\left\| \hat{v}^r_\theta(t, x(t)) - \left(\left(\boldsymbol{r}_1 - \boldsymbol{r}_0 + \frac{L}{2}\right) \% \, L - \frac{L}{2}\right) \% \, L \right\|^2\right] \mathrm{d}t \; . \quad (4.8)$$

Assuming we have access to samples from $Q_0$ (cheap) and from $Q_1$ (expensive), training proceeds as follows. We iteratively draw batches of $M$ pairs $(x_0, x_1)$ from the coupling $\nu$ between $Q_0$ and $Q_1$. Next, we sample a batch of $M$ time steps $t$ uniformly in [0,1] and estimate the loss in Eq. (4.8) over the batch via Monte Carlo integration. We then compute the gradient of the loss with respect to the parameters $\theta$ by automatic differentiation, and update the parameters via stochastic gradient descent. The process is repeated until convergence, leading to a trained velocity field $\hat{v}_\theta^r$. Note that since $\hat{v}_\theta$ is equivariant by construction and since the optimal velocity field $v^\star$ is equivariant by Theorem 4.1.1, we regress an equivariant model onto an equivariant target, something that was not guaranteed in prior works [37, 174].

Once the model is trained, as explained in Section 1.3.3, we can use the learned velocity field $\hat{v}_\theta^r$ to define a CNF that approximately transports samples from $Q_0$ to $Q_1$. This will be our generative model for the Boltzmann distribution (2.5). Since the learned velocity field is equivariant, the resulting CNF will also be equivariant, thereby ensuring that the generated distribution $\hat{Q}_1$ respects the relevant symmetries. The resulting distribution, however, remains an approximation and would lead to biased estimates of thermodynamic observables. To correct for this, we evaluate the exact density of generated samples by integrating the coupled ODE (1.26). This enables exact importance sampling (see Section 1.1.1), which provides unbiased estimates. While density evaluation substantially increases the computational cost of sample generation, it is not merely a numerical refinement but a necessary prerequisite for the correct evaluation of thermodynamic observables. Note that since the loss in Eq. (4.8) does not require computing likelihoods, the training cost remains very low. Evaluating densities for all the samples in the training set would instead be prohibitive, while only evaluating it for generated samples is manageable, at least for moderate system sizes.

## 4.2 Application to glass-forming mixtures

We now apply the eRSI framework to two representative glass-forming mixtures: a binary inverse power-law mixture [186] and a ternary version of the Kob–Andersen mixture [144, 158, 187]. By comparing eRSI against both symmetry- and geometry-agnostic baselines, we demonstrate that incorporating symmetries and geometric constraints significantly improves the quality of the generated configurations, both at the level of individual realisations and of ensemble-averaged observables.

### 4.2.1 Simulation details

We consider two-dimensional systems composed of $N = 10$ and $N = 44$ particles in a square box with PBCs at constant temperature $T$ in the canonical ($NVT$) ensemble.

The first model (IPL) [186] is a 50:50 mixture interacting via the inverse power law (2.9), with $\epsilon = 1.0$, $\sigma_{\mathrm{AA}} = 1.0$, $\sigma_{\mathrm{AB}} = \sigma_{\mathrm{BA}} = 1.2$, $\sigma_{\mathrm{BB}} = 1.4$, and exponent $n = 12$. The potential is truncated and shifted at a cutoff distance $r^c_{\alpha\beta} = 2.5\sigma_{\alpha\beta}$, with $\alpha, \beta \in \{\mathrm{A, B}\}$. The number density is fixed to $\rho = 0.5$ and the temperature to $T = 0.1$, corresponding to a dense fluid state.

The second model (KA) [144] is a (5/11, 3/11, 3/11) ternary version of the Kob-Andersen mixture introduced for efficient simulations with the swap MC algorithm (see Section 2.4.2). The interaction is given by the Lennard-Jones potential in Eq. (2.10), with $\epsilon_{\mathrm{AA}} = 1.0$, $\epsilon_{\mathrm{AB}} = \epsilon_{\mathrm{BA}} = 1.5$, $\epsilon_{\mathrm{AC}} = \epsilon_{\mathrm{CA}} = 0.75$, $\epsilon_{\mathrm{BB}} = 0.5$, $\epsilon_{\mathrm{BC}} = \epsilon_{\mathrm{CB}} = 1.5$, $\epsilon_{\mathrm{CC}} = 0.75$, $\sigma_{\mathrm{AA}} = 1.0$, $\sigma_{\mathrm{AB}} = \sigma_{\mathrm{BA}} = 0.8$, $\sigma_{\mathrm{AC}} = \sigma_{\mathrm{CA}} = 0.9$, $\sigma_{\mathrm{BB}} = 0.88$, $\sigma_{\mathrm{BC}} = \sigma_{\mathrm{CB}} = 0.8$, and $\sigma_{\mathrm{CC}} = 0.94$. The potential is adjusted to ensure continuity and differentiability up to second order at a cutoff distance $r^c_{\alpha\beta} = 2.5\sigma_{\alpha\beta}$, with $\alpha, \beta \in \{\mathrm{A, B, C}\}$. The number density is fixed at $\rho = 1.192075$, and we consider two temperatures, $T = 1.0$ (fluid) and $T = 0.32$ (moderately glassy).

We generate the training datasets with the Metropolis–Hastings (MH) algorithm (see Section 1.2) implemented in `ParticlesMC.jl` (see Appendix E.2). Starting from particles uniformly distributed in the box, we perform Gaussian displacements (see Section 2.4.2) with standard deviation 0.065, obtained using Policy-guided Monte Carlo (see Chapter 3) with the reward in Eq. (3.19). We define a MH sweep (our unit of time) as $N$ attempted moves. For the IPL model and for the KA model at temperature $T = 1.0$, we run 100 independent chains, each initialised from a different random configuration, for $10^4$ MH sweeps to reach equilibrium. In this time scale, the potential energy rapidly relaxes to a steady value, and the self-intermediate scattering function (2.25) decays to zero, confirming that the system is equilibrated. From these equilibrated configurations, each chain is then propagated for an additional $10^7$ MH sweeps, storing one configuration every $10^4$ steps. For the KA model at temperature $T = 0.32$, we also include swap moves (see Section 2.4.2) with weight $p_{\mathrm{swap}} = 0.2$ to accelerate equilibration. We run 100 independent chains for $5 \times 10^4$ MH sweeps to reach equilibrium, followed by $5 \times 10^7$ MH sweeps for data collection, storing one configuration every $5 \times 10^4$ steps. Both procedures yield a total of $10^5$ uncorrelated equilibrium configurations, which we use as training data. The datasets are available on Zenodo at `https://zenodo.org/records/17966995`.

For the IPL model, we compare the eRSI framework introduced with the two following ablations: (i) equivariant flow matching (eFM) [35], which applies standard flow matching with an $E(n)$-equivariant GNN [84] respecting permutation-translation-rotation symmetries in $\mathbb{R}^{Nd}$ but ignoring PBCs; and (ii) Riemannian stochastic interpolants (RSI), which accounts for torus geometry but lacks the invariances of Section 2.2.4 using a simple multi-layer perceptron to learn the velocity field. For a a fair comparison, the number of trainable parameters of all architectures is taken equal and all models are trained for the same number of epochs.

In our implementation of the velocity field in Eq. (4.7), $\hat{\phi}_e$ and $\hat{\phi}_h$ are $K$-layer multi-layer perceptrons (MLPs) with width $n$ and SiLU activations, while $\hat{\phi}_m$ is a 2-layer MLP with width $n$ and SiLU activations. Species are embedded using one-hot encoding. For $K = 3$, $n = 32$, the network has a total of approximately 22k trainable parameters. For eFM, we adopt the GNN architecture of Ref. [84] using the same hyperparameters as in eRSI. For RSI, the configuration is fed directly into a 3-layer MLP with width 64, yielding a comparable parameter count to the GNN models. To account for PBCs, inputs are projected onto the manifold following Ref. [81, Equation 26].

We train the models for 1000 epochs on datasets of size $10^5$, with batch sizes of 2048 samples for $N = 10$ and 1024 for $N = 44$. We employ the Adam optimiser [188], with hyperparameters selected via validation loss among learning rates $\{10^{-5}, 5 \times 10^{-5}, 10^{-4}, 5 \times 10^{-4}, 10^{-3}, 5 \times 10^{-3}\}$ and with or without gradient clipping at 2.0 [189]. We report the hyperparameters in Table 4.1. For the non-equivariant RSI baseline, we tested additional data augmentation using random actions of $G_{\mathsf{X}}$, but we found no improvement.

After training, we generate new configurations by solving the coupled ODE (1.26) with the order 5 Dormand-Prince-Shampine-Runge-Kutta [190] method (`dopri5`) from `torchdiffeq` [191], with absolute and relative tolerances set to $10^{-5}$. We evaluate divergences in likelihood computations exactly via automatic differentiation. The code for these simulations is part of the `learndiffeq` library available at `https://github.com/h2o64/learndiffeq`.

### 4.2.2 Framework comparison

We begin by comparing RSI, eFM and eRSI on the IPL system with $N = 10$ and $N = 44$ particles. We consider (i) the average potential energy $U = \mathbb{E}\left[U(x)\right]$, (ii) the specific heat $c_V = \left(\mathbb{E}\left[U(x)^2\right] - \mathbb{E}\left[U(x)\right]^2\right)/(NT^2)$, probing energy fluctuations and known to be harder to recover than average energies [158, 192], and (iii) the radial distribution function $g(r)$ defined in Eq. (2.12). We evaluate these metrics on the generated configurations ($x \sim \hat{Q}_1$) and, in the case of the IPL model, on configurations reweighted via importance sampling ($x \sim P$) using Eq. (1.9). For comparison, we also report reference values extracted from the training dataset generated with MCMC simulations.

Figure 4.2 presents results for the small system with $N = 10$ particles. RSI fails

**Table 4.1:** Learning rates (LR) and gradient clipping (GC) for each model and system size. Architectures are described by the number of GNN layers $K$ and width $n$.

| $N$ | Model | $K$ | $n$ | LR | GC |
|---|---|---|---|---|---|
| 10 | eRSI | 3 | 32 | $1 \times 10^{-4}$ | 2.0 |
| | eRSI | 4 | 64 | $5 \times 10^{-5}$ | – |
| | eRSI | 5 | 128 | $1 \times 10^{-5}$ | 2.0 |
| | eFM | 3 | 32 | $5 \times 10^{-3}$ | 2.0 |
| | RSI | 3 | 64 | $5 \times 10^{-4}$ | 2.0 |
| 44 | eRSI | 3 | 32 | $5 \times 10^{-3}$ | 2.0 |
| | eRSI | 4 | 64 | $1 \times 10^{-3}$ | 2.0 |
| | eRSI | 5 | 128 | $1 \times 10^{-3}$ | 2.0 |
| | eFM | 3 | 32 | $1 \times 10^{-3}$ | 2.0 |
| | RSI | 3 | 64 | $5 \times 10^{-5}$ | 2.0 |

to capture meaningful structure, as indicated by the flat $g(r)$. In contrast, eFM performs better: $U$ and $c_V$ can be recovered through reweighting, along with a noisy estimate of $g(r)$. eRSI achieves even stronger performance: $U$ and $c_V$ converge to the target with fewer samples, and the estimate of $g(r)$ is more accurate.

We repeat the analysis for $N = 44$, with results shown in Fig. 4.3. Here, eFM fails: both $c_V$ and $U$ estimators stabilise away from the ground truth (sampling was stopped at $R = 8 \times 10^3$ due to limited computational resources). By contrast, eRSI produces $U$ and $c_V$ that converge rapidly to the target, together with an accurate $g(r)$. For $g(r)$, we also include the poor estimates obtained without reweighting (dotted lines), highlighting both the importance of reweighting and the need for models that can handle it accurately.

Examples of configurations generated by the different models for $N = 44$ are shown in Fig. 4.4, alongside typical configurations from $P$. RSI fails to move particles, producing configurations that remain close to the uniform base distribution, which is consistent with its poor performance. We attempted augmenting the training data with random actions from $G_{\mathsf{X}}$ to help the RSI model learn the invariances, but this did not improve performance. eFM generates more realistic configurations, but many particle overlaps occur near the box edges because PBCs are ignored. These overlaps cause three issues: (i) averages obtained without reweigthing display unphysical characteristics, as a much higher value of $g(r < 1)$, (ii) many configurations are discarded during reweighting, as configurations with close particle pairs receive zero weight in the IS estimator (thus requiring a large

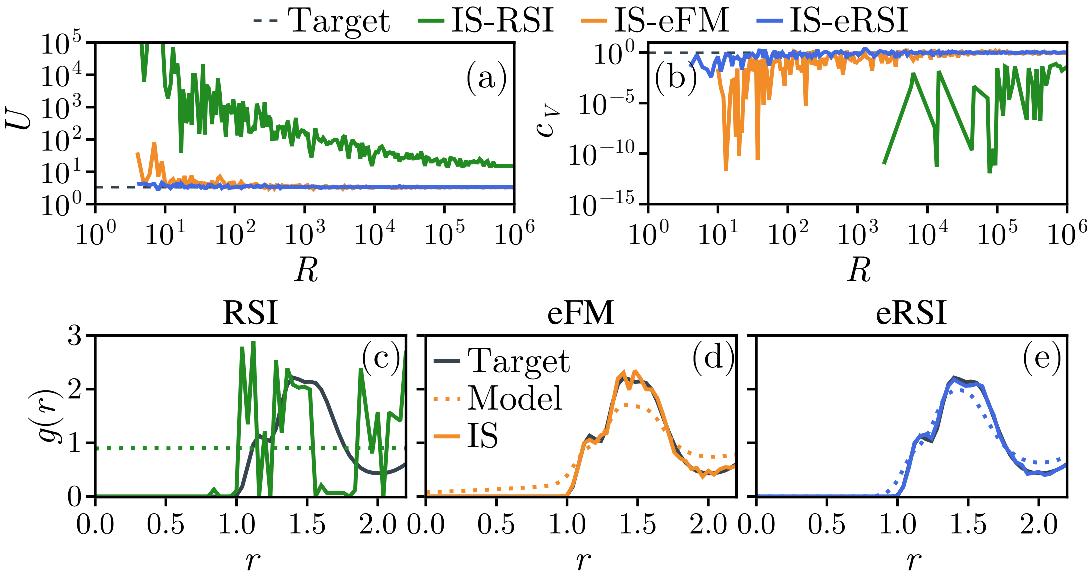

**Figure 4.2:** Results for $N = 10$ particles. (a) Mean potential energy $U$ and (b) specific heat $c_V$ as a function of the number of generated samples $R$ for RSI, eFM, and eRSI. (c–e) Radial distribution function $g(r)$ for the three models, showing target, direct model, and reweighted estimates. RSI fails completely, eFM partially recovers observables, and eRSI performs best. Adapted from Ref. [45].

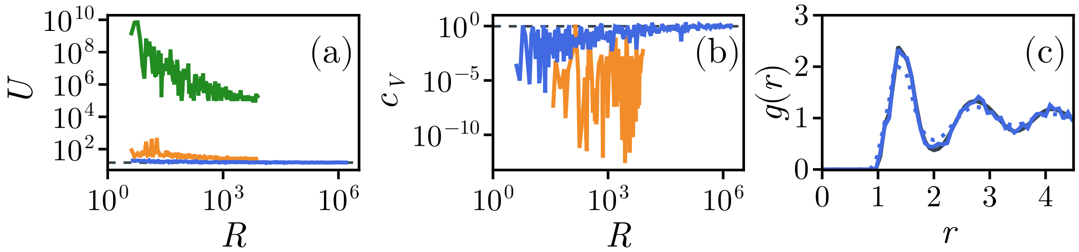

**Figure 4.3:** Results for $N = 44$ particles. (a) Mean potential energy $U$ and (b) specific heat $c_V$ as a function of the number of generated samples $R$ for RSI, eFM, and eRSI. (c) Radial distribution function $g(r)$ for eRSI averaged over $1.8 \times 10^6$ samples. RSI and eFM deviate, due to collapsed states (RSI) or boundary overlaps (eFM). RSI samples are of too poor quality to produce $c_V$ estimates. Only eRSI remains consistent with the target distribution. Adapted from Ref. [45].

number of samples to obtain accurate estimation), and (iii) the likelihood of overlaps at the boundaries increases with system size, limiting the scalability of the eFM model. By contrast, configurations produced by eRSI correctly incorporate periodic boundary conditions and do not suffer from this problem. We include additional samples from the training set and generated by eRSI in Fig. 4.5.

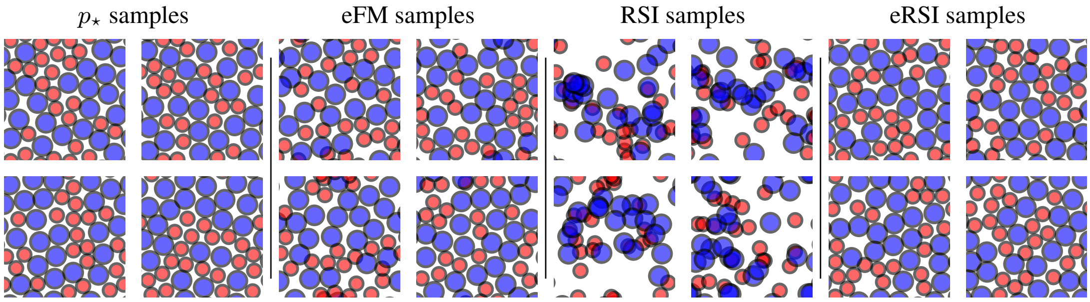

**Figure 4.4:** Samples from different frameworks. Configurations generated from $P$ are compared to the ones obtained from different generative models on the IPL system with $N = 44$. eFM generates unphysical particle overlaps near boundaries, RSI produces random configurations, while eRSI generates visually plausible samples. Reprinted from Ref. [45].

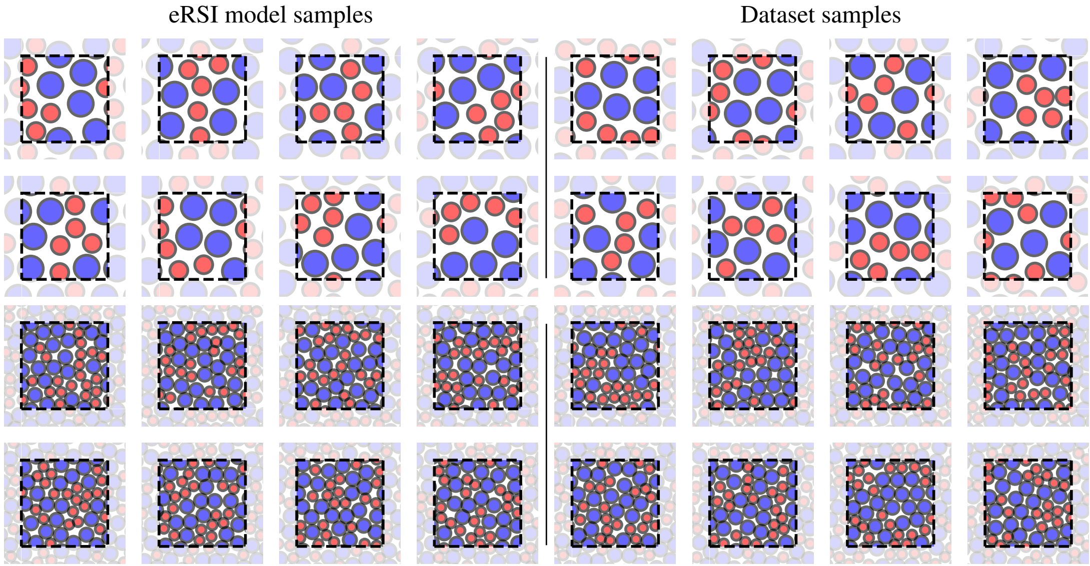

**Figure 4.5:** Additional samples for the IPL system with $N = 10$ (top rows) and $N = 44$ (bottom rows). Adapted from Ref. [45].

### 4.2.3 Effective sample size

A central difficulty in reweighting is that the effective sample size (ESS) defined in Eq. (1.7) can be extremely low in high dimensions. In our simulations, we observe a behaviour akin to Example 1.1.2, where the ESS decays approximately as $1/R$ over a broad range of sample sizes $R$, before eventually stabilising to a finite plateau only at large $R$. This behaviour implies that, at small $R$, the ESS is essentially uninformative: estimating it reliably requires a number of samples that may itself exceed the plateau value.

For $N = 10$, Fig. 4.6 shows that the crossover scale $\bar{R}$ at which the ESS departs from the $1/R$ trend is strongly model-dependent. For RSI, no such crossover is observed: the ESS remains proportional to $1/R$ throughout. For eFM, $\bar{R} \approx 10^4$, while for eRSI it is significantly smaller, $\bar{R} \approx 10^3$. To explain this phenomenon, we compare the energy distributions of samples from the three models with that of the target, shown in the bottom panels of Fig. 4.6. For clarity, we discard configurations with energies larger than twice the maximum observed in the target dataset. The discarded fraction is 100% for RSI, about 84% for eFM, and about 3% for eRSI. For small $R$, the two densities $p(U)$ (target) and $q(U)$ (model) overlap poorly: $q(U)$ is concentrated at higher energies and only begins to penetrate the region of significant $p(U)$ weight from the right tail. As $R$ increases, rare samples from this region appear with very small $q(U)$ but relatively large $p(U)$, producing very large reweighting factors. A few such configurations dominate the averages, driving the $1/R$ behaviour of the ESS. Once $R$ exceeds $\bar{R}$, $q(U)$ has infiltrated sufficiently deep into the bulk of $p(U)$ so that ratios $p(U)/q(U)$ are less extreme, and the ESS stabilises to a plateau.

We repeat the analysis for $N = 44$ in Fig. 4.7, using up to $R = 8 \times 10^3$ samples for RSI and eFM and up to $R = 10^6$ samples for eRSI. The results are qualitatively the same, although eFM samples are significantly worse due to overlaps at the boundaries (see Fig. 4.4).

### 4.2.4 Additional results

We conclude this section by examining how network size, training set size, and temperature affect eRSI performance. While limited computational resources prevented us from performing an exhaustive hyperparameter search, the results presented below provide useful insights into the behaviour of eRSI under different conditions and leave room for further improvement.

#### Network size dependence

Figure 4.8 shows the effect of network size on the IPL system with $N = 44$. Larger models (≈580k parameters vs. 22k for the smallest) achieve substantially higher fidelity and are expected to reweight observables with far fewer samples. This

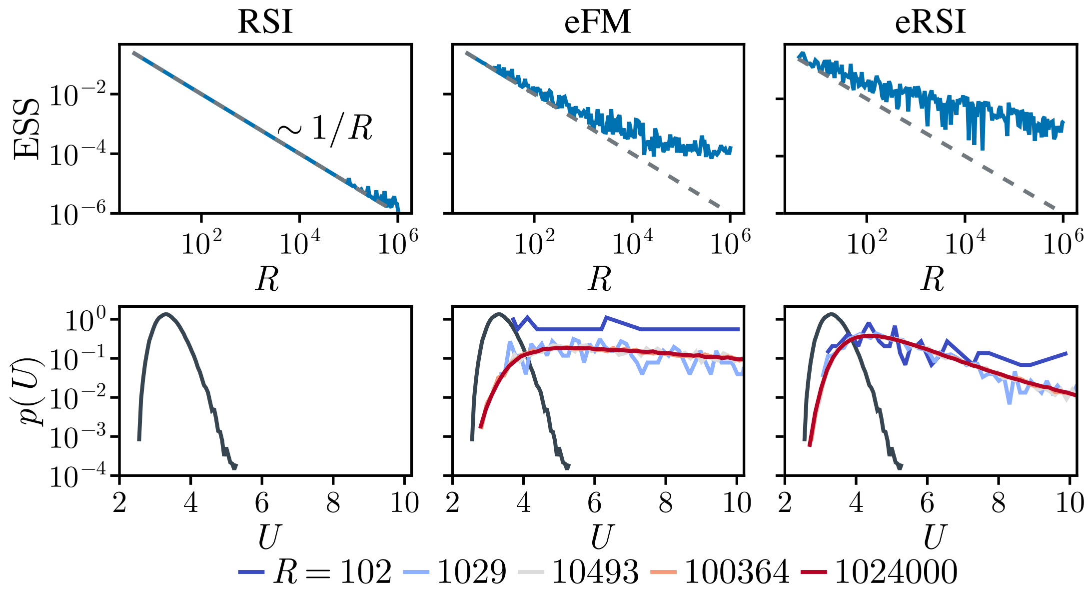

**Figure 4.6:** Effective sample size for $N = 10$ particles. The top panels show the normalised ESS as a function of the number of samples $R$. This metric reflects the proportion of samples effectively contributing to expectation estimates and provides an upper bound on the estimator variance. The bottom panels compare histograms of the target energy $U$ for samples generated by each model (coloured lines) and for training samples (grey line). Adapted from Ref. [45].

gain, however, comes at the cost of a ≈ 2.5× slower training and a ≈ 10× slower sampling on an NVIDIA A100 GPU.

### Dataset size dependence

Figure 4.9 shows that increasing the training dataset size consistently improves model quality, with larger datasets yielding energy and structural statistics closer to the target distribution. Generating these datasets, especially for larger systems at low temperatures, can be computationally expensive, so the trade-off between dataset size and model performance needs to be balanced based on available resources.

### Temperature dependence

Finally, Fig. 4.10 shows results for the KA model at temperatures $T = 1.0$ and $T = 0.32$ with $N = 44$ particles. At $T = 0.34$, the system is in the glassy regime and sampling with molecular dynamics or standard MCMC methods becomes

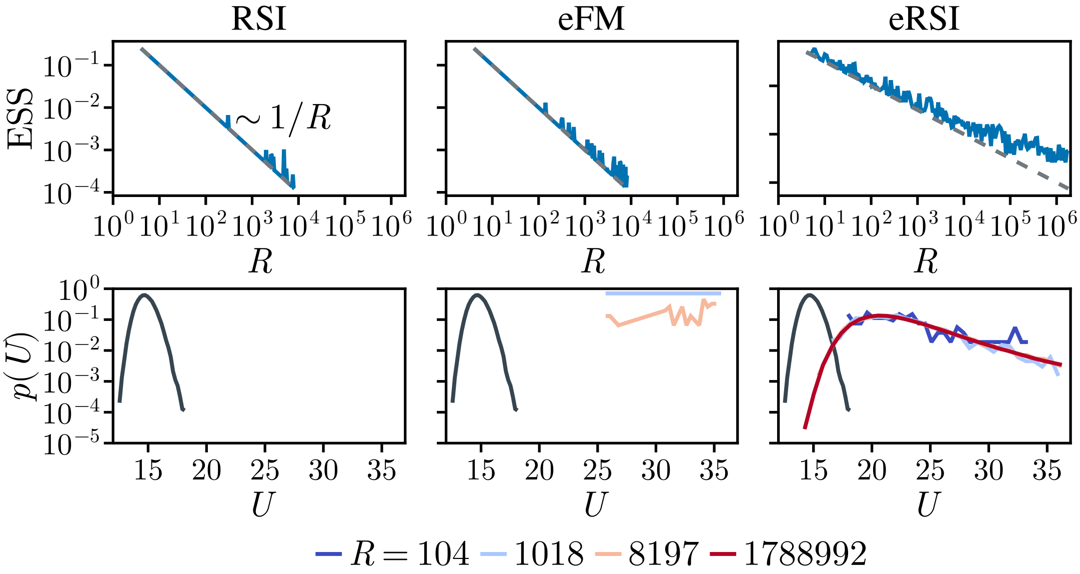

**Figure 4.7:** Effective sample size and target energy for $N = 44$ particles. Adapted from Ref. [45].

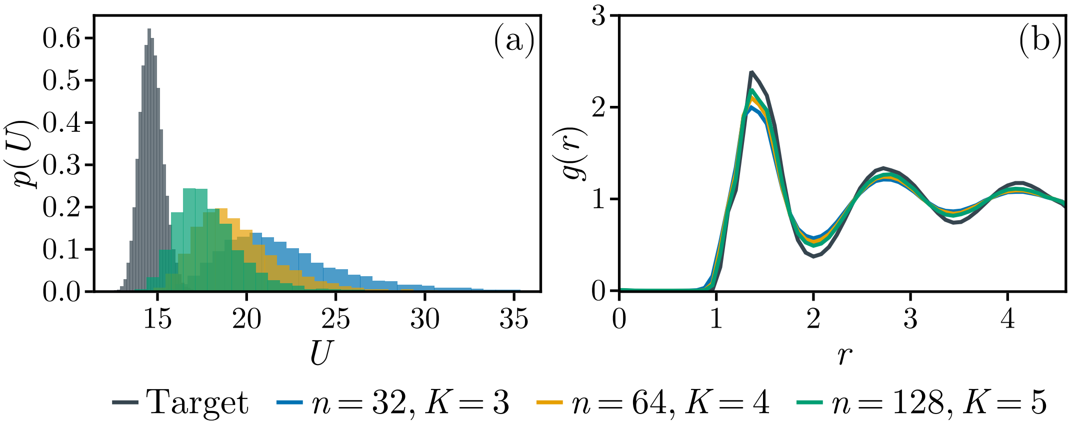

**Figure 4.8:** Network size dependence. Variants of the EGNN architecture in Eq. (4.7) with $K \in \{3, 4, 5\}$ and hidden feature size $n \in \{32{,}64{,}128\}$. (a) Energy distribution from 8192 generated samples per model. (b) Corresponding radial distribution functions. Larger architectures yield improved fidelity to the target distribution. Adapted from Ref. [45].

challenging [158]. A reference training set can still be generated using swap MC (see Section 2.4.2).

Example configurations generated by eRSI at both temperatures are shown

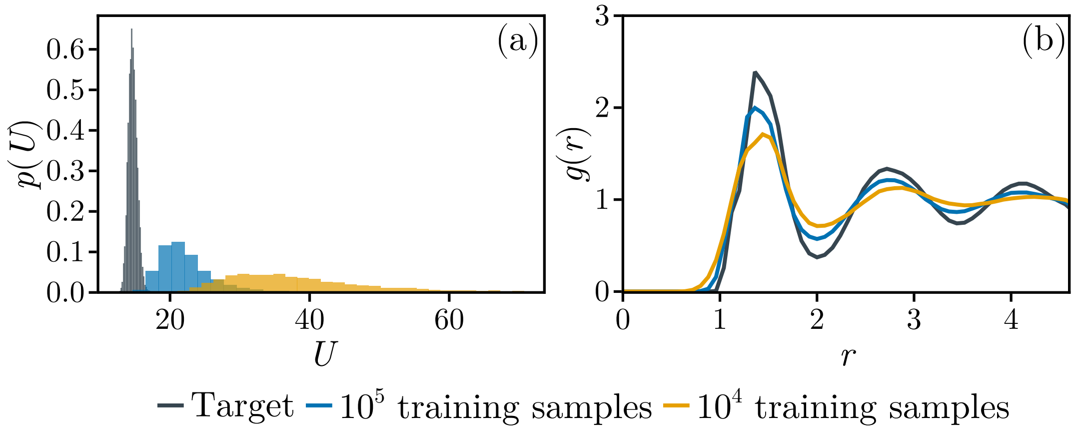

**Figure 4.9:** Dataset size dependence. Comparison between eRSI model trained on $10^4$ and $10^5$ samples. (a) Energy distribution of 8192 generated samples. (b): corresponding radial distribution functions. Larger datasets yield notably improved fidelity in both metrics. Adapted from Ref. [45].

in Fig. 4.11. eRSI successfully generates visually convincing configurations and captures meaningful structural features at both temperatures, indicating that reweighting could be effective even in the glassy regime.

## 4.3 Discussion and perspectives

In this chapter, we introduced eRSI, a framework that formally combines stochastic interpolants, equivariant generative models, and Riemannian geometry. Our main theoretical contribution is the identification of precise conditions on the interpolation path that guarantee equivariance of the optimal velocity field and invariance of the marginal densities at all times. This result, proved in Theorem 4.1.1, extends the theory of equivariant flows [35, 36, 44] to stochastic interpolants on general geometries and provides a practical recipe to train symmetry-preserving generative models with a flow matching objective.

Building on this foundation, we designed an interpolation path, an optimal transport coupling, and an equivariant graph neural network architecture tailored to multicomponent particle systems with periodic boundary conditions, respecting both geometric constraints and physical symmetries. Similar work has been recently done on crystalline materials [37, 174], but these methods focus on generative modelling from samples drawn from an unknown distribution. In that context, success is typically assessed by visual or structural fidelity to the training

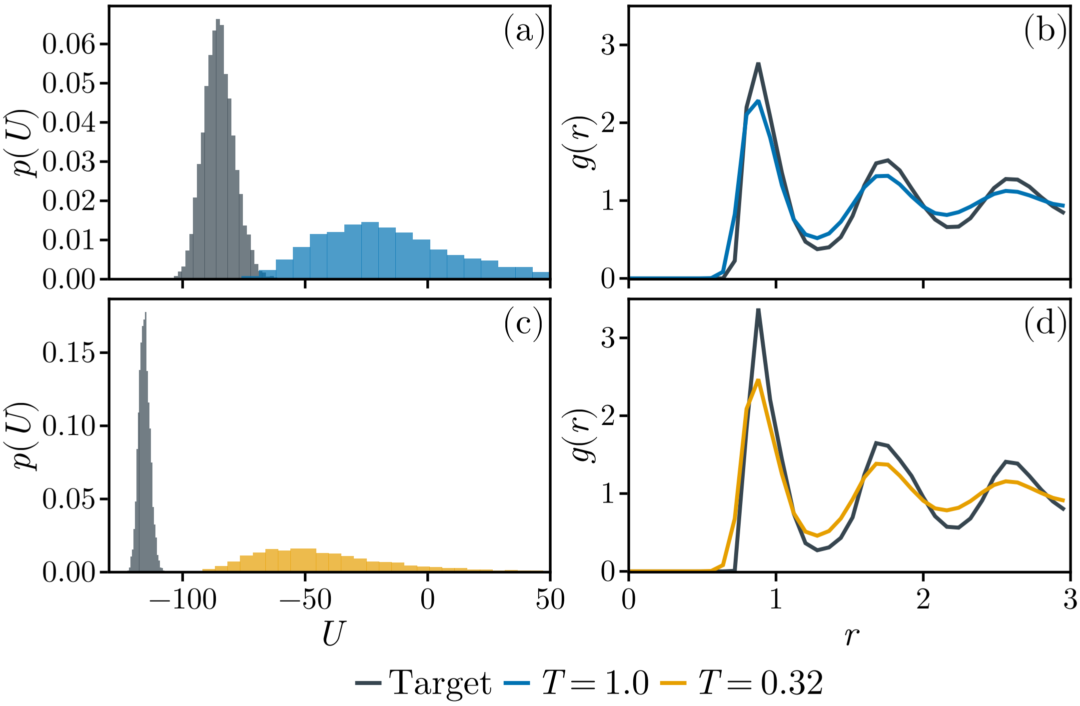

**Figure 4.10:** Temperature dependence in the KA model. The left panels show histograms of the potential energy evaluated on 8192 samples from the model, while the right panels display the corresponding radial distribution functions. Result are shown at $T = 1.0$ (top row) and $T = 0.32$ (bottom row). Adapted from Ref. [45].

configurations, making the task substantially easier than sampling exactly from a target Boltzmann distribution. Crucially, compared to similar diffusion-based methods [173, 176–179], our framework allows one to compute exact likelihoods of generated samples, enabling unbiased importance sampling of thermodynamic observables. On the other hand, compared to previous work with the same goal on amorphous materials [158], our training objective does not require computing likelihoods during training, greatly reducing computational cost and allowing for more expressive architectures.

Our numerical results on two representative glass-forming mixtures show the benefits of incorporating geometry and symmetry into generative modelling. Compared to ablations that ignore either geometry or symmetry, eRSI generates configurations of high visual and structural fidelity, while providing accurate estimates of both average observables and higher-order fluctuations when combined with importance sampling. Importantly, by exposing how inaccuracies manifest through the energy distribution and the effective sample size, our simulations also clarify the role of

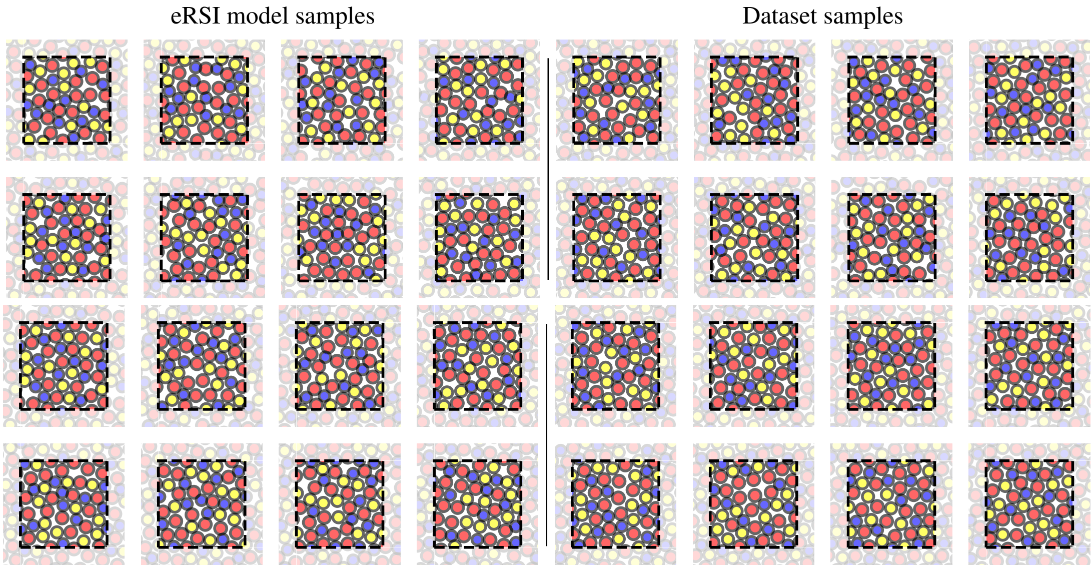

**Figure 4.11:** Samples for the KA system with $N = 44$. Configurations generated by the eRSI model (left) and reference dataset samples (right), shown at $T = 1.0$ (top rows) and $T = 0.32$ (bottom rows). Adapted from Ref. [45].

reweighting: visually plausible samples do not guarantee accurate physical averages, and reliable likelihood estimation remains essential.

The framework opens several promising research directions. The main long-term goal is to lift the reliance on pre-existing equilibrium samples and train the generative model directly from knowledge of the potential energy function. Recent advances on energy-based training for diffusion models [193–195] suggest that this is feasible, but they do not address how to debias or reweight samples from the learned flow. Establishing that reweighting continues to work for stochastic interpolants in sample-free regimes would provide a foundation for implementing adaptive strategies previously used for normalising flows and autoregressive samplers [69, 156, 196, 197].

A second direction concerns scalability. The present implementation of eRSI focuses on generating full configurations, which limits its applicability to large systems. While generating samples from the trained model is relatively cheap, evaluating likelihoods remains costly due to the need to compute the divergence term in the coupled ODE (1.26). A recent work by Rheman *et al.* [198] proposes an efficient method to tackle this problem. From a complementary perspective, a natural next step is to move towards *conditional* updates, where the generative model updates local patches of the system given their environment. Such approaches promise significantly improved scaling with system size and integrate well with adaptive training [70, 199–201].

# Chapter 5

# How to control the stability of glasses

“*Equilibrium is so 20th century.*”

In recent years, advances in experimental techniques and computer simulations have made it possible to produce ultrastable glasses [89, 108, 202], *i.e.* amorphous solids whose kinetic stability exceeds the one obtained through conventional cooling protocols (see Section 2.1.3). Ultrastability manifests in “better” physical properties such as higher density [203], lower enthalpy [204], and an enhanced resistance to crystallisation [205] and melting [202, 206, 207].

Different approaches have been developed to produce ultrastable glasses in simulations. A natural route is to simulate the physical vapour deposition process employed in experiments directly [137, 208]. However, despite its conceptual appeal, simulated vapour deposition remains severely limited by accessible timescales. A major breakthrough came with the introduction of the swap Monte Carlo algorithm (see Section 2.4.2), which allows the simulation of ultrastable glasses in thermal equilibrium [107, 140, 159]. Swap moves accelerate equilibration by several orders of magnitude, making it possible to reach extremely low temperatures while still sampling the canonical ensemble within the familiar framework of standard MCMC simulations.

More recently, several heuristic nonequilibrium protocols have been shown to generate exceptionally stable configurations [48–50, 209–211]. These methods differ fundamentally from swap MC and the other algorithms described in the previous chapters: rather than sampling the Boltzmann distribution at low temperature, they aim at directly generating configurations with target structural properties that are believed to correlate with glass stability. For instance, Refs. [48, 51] focused on optimising hyperuniformity, while Ref. [49] targeted local structural order, and Ref. [50] aimed at reducing local stress. In all these cases, the resulting configurations showed remarkable kinetic stability, which seems to suggest a link between structure and stability.

However, a closer look reveals a subtle but crucial feature shared by all these nonequilibrum algorithms: they rely on dynamic adjustments of the particle diameters, a mechanism closely related to swap MC and known to enhance stability on its own. This raises a natural question: is the exceptional stability of these optimised glasses truly *caused* by the targeted structural properties, or is it simply a consequence of diameter dynamics that closely mimics the effect of swap moves?

To disentangle these interpretations, we introduce computational methods that optimise physical quantities while keeping the particle diameters fixed. Specifically, we develop stochastic algorithms that generate hard disk configurations with enhanced hyperuniformity (as in Refs. [48, 51]) or enhanced local order (as in Ref. [49]), and show that they produce configurations whose structural properties are strongly optimised relative to their bulk equilibrium values. We then evaluate the kinetic stability of these configurations by measuring their glass equation of state (see Section 2.1.3). We find that, despite being optimised, these "ideal" configurations are not more stable than conventional glassy states. This demonstrates that these structural quantities are not causally responsible for stability, challenging the interpretations recently proposed in the literature.

## 5.1 Optimisation algorithms

To generate configurations with optimised structural properties while keeping the diameters fixed, we develop algorithms that directly target the quantities of interest. Following previous works, we focus on two structural properties that have been proposed to correlate with glass stability: hyperuniformity and local order parameter (see Section 2.3.1) In contrast with the sampling methods discussed earlier in the thesis, the stochastic algorithms developed here aim only at optimising these structural quantities, without targeting the Boltzmann distribution. The resulting configurations are thus sampled from nonequilibrium ensembles whose underlying distribution can be unknown.

The two optimisation procedures we use rely on very different mechanisms. We enhance hyperuniformity (see Section 2.3.1) by using nonequilibrium dynamics with an absorbing transition, belonging to the class of random organisation models [121–124, 212–214]. For local order, instead, we employ a biased Monte Carlo scheme [110, 134, 215–218] typically used in umbrella sampling [219, 220], where a biasing potential favours configurations with high local order parameter $\Theta$ (see Eq. (2.22)). We describe the two approaches in the following sections.

### 5.1.1 Conserved biased random organisation

Random organisation models describe particle systems evolving through discrete nonequilibrium dynamics driven by random displacements of overlapping particles [122–124, 212, 213] (see Appendix D.1). Originally introduced to model periodically sheared suspensions [212, 221], these models are known to generate hyperuniform configurations in different physical regimes, depending on the specific details of the dynamics [122, 123, 213, 214, 222, 223]. This makes them natural candidates for generating "more hyperuniform" amorphous configurations than those found in equilibrium. We quantify the degree hyperuniformity by measuring the structure factor at small wavevectors, $S(q \to 0)$, and aim to reduced its value relative to that of bulk equilibrium configurations [48].

To this end, we employ the conserved biased random organisation (CBRO) model [123, 124], which has been shown to generate hyperuniform configurations in both dilute [123] and crystalline [124] systems. The dynamics proceed as follows. Starting from an initial configuration $x$, at each discrete time step $t$, any pair of overlapping particles is displaced along the line connecting the two centers by opposite vectors $\vec{\delta}$ and $-\vec{\delta}$ of the *same* amplitude $\delta$ drawn uniformly in the interval $[0, \epsilon]$ (see Fig. 5.1). If a particle overlaps with multiple neighbours, its total displacement is the sum of the individual contributions. By construction, this dynamical rule conserves the centre of mass [123, 223].

As long as the density and maximal kick's size $\epsilon$ are sufficiently large, these dynamics do not reach an absorbing configuration and instead converge to a nonequilibrium steady state. In this regime, centre-of-mass conservation ensures that the *fluctuating* component of the structure factor $S_\delta(q)$ (see Eq. (2.19)), vanishes as $q^2$ at small $q$ (see Appendix D.1.4). In liquids, the backbone contribution $S_0(q)$ (see Eq. (2.18)) vanishes at low $q$ by translational invariance, while in crystals it vanishes everywhere except at Bragg peaks. Thus in both cases the total structure factor follows $S(q) \sim q^2$, corresponding to hyperuniform scaling with exponent $\alpha = 2$ [123, 124].

However, in amorphous solids, $S_0(q)$ remains finite at small $q$, preventing the

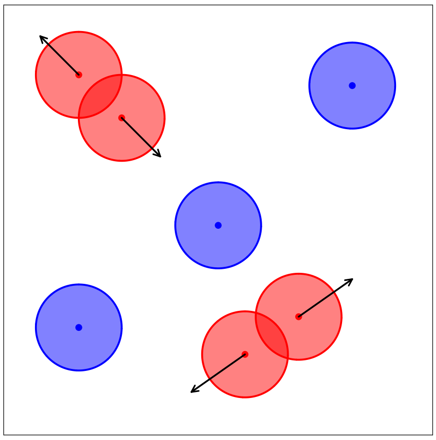

**Figure 5.1:** CBRO dynamics. At each time step, pairs of overlapping particles (red) are displaced in opposite directions by the same random amount.

total structure factor $S(q)$ from exhibiting true hyperuniform scaling, even if $S_\delta(q)$ does [128]. When applying CBRO to an amorphous configuration, we thus expect $S_\delta(q)$ to vanish at small $q$, while the total structure factor saturates at a finite (yet smaller) value determined by $S_0(q \to 0)$ (see Section 5.2.3). Since CBRO removes the fluctuating contribution, it is expected to produce configurations with enhanced hyperuniformity compared to the initial equilibrium state. We will use this protocol to generate such configurations in Section 5.2.

### 5.1.2 Biased Monte Carlo

In order to produce configurations with enhanced local order, we adopt a more direct approach. A common strategy to explore configurations associated with rare or atypical values of a given observable is to alter the statistical weight so that those regions of phase space become accessible. This is the principle behind *biased Monte Carlo* (BMC) [110, 217] and *umbrella sampling* [219]. In these approaches, the usual Boltzmann distribution $P(\mathrm{d}x)$ (Eq. (2.3)) is "tilted" with an additional potential $V(O(x))$ that depends on the observable of interest $O(x)$. Sampling is then performed according to the modified measure

$$\widetilde{P}(\mathrm{d}x) \;:=\; \frac{1}{\widetilde{Z}} \exp\left(-\frac{U(x) + V\left(O(x)\right)}{k_B T}\right) . \tag{5.1}$$

Since this biased measure is known explicitly (up to normalisation), unbiased estimates of equilbirium averages can be reconstructed via importance sampling (see Section 1.1.1). This is precisely the essence of umbrella sampling [219, 220].

Although recovering equilibrium averages is not our main objective here, the reweighting procedure provides a quantitative measure of how atypical the generated configurations are. Note that this contrasts with the CBRO protocol, where the underlying sampling distribution is unknown. To favour configurations with high local order, we choose the biasing potential $V_\lambda(\Theta(x)) = \lambda\Theta^2(x)$, where $\Theta(x)$ is the local order parameter defined in Eq. (2.22), and $\lambda \geq 0$ controls the strength of the bias.

In principle, we could sample from $\widetilde{P}$ using any algorithm described in Section 1.1, for instance a standard MH algorithm based on single particle displacements (see Section 2.4.2). However, this would require recalculating $\Theta(x)$ at each step, which is computationally expensive due to the Voronoi tessellation [132, 224]. To overcome this issue, following Refs. [215, 218], we adopt the following procedure. Starting from a configuration $x$, we first generate a trial configuration $x'$ by running a short MH trajectory targeting the *unbiased* Boltzmann distribution $P$. Then, we accept the trial configuration $x'$ with probability

$$\alpha\left(x, x'\right) \;=\; \min\left\{1,\, \exp\left(-\frac{V_\lambda\left(\Theta(x')\right) - V_\lambda\left(\Theta(x)\right)}{k_B T}\right)\right\}\;. \tag{5.2}$$

In Corollary D.2.2, we show that the resulting transition kernel satisfies detailed balance with respect to the biased distribution $\widetilde{P}$. This result is general: it provides a recipe for sampling any target distribution $\widetilde{P}$ when the proposal kernel is reversible with respect to another distribution $P$. In the BMC case, this strategy allows us to sample from $\widetilde{P}(\mathrm{d}x)$ while efficiently exploring configuration space and without the need to recalculate $\Theta(x)$ at each step. Once the Markov chain targeting $\widetilde{P}(\mathrm{d}x)$ has converged, it can sample configurations corresponding to atypical equilibrium states that would be essentially inaccessible without the bias.

## 5.2 Application to glass-forming mixtures

We now test the optimisation algorithms developed in the previous sections on a standard model glass former. Our goal is to show concretely that it is possible to generate configurations with strongly enhanced structural properties without obtaining glasses that are kinetically more stable. To this end, we apply the optimisation algorithms for hyperuniformity and local order to a two-dimensional hard disk mixture for which swap MC is inefficient. We show that both methods successfully produce configurations with enhanced structural properties compared to bulk equilibrium states, reaching values that would be essentially inaccessible in conventional equilibrium simulations. Finally, we measure their glass equation of state (see Section 2.1.3) and compare it to that of glasses prepared through standard equilibrium protocols, showing that the optimised configurations exhibit no signs of enhanced kinetic stability.

### 5.2.1 Simulation details

We consider a two-dimensional system of $N = 200$ and $N = 1000$ particles in a square box of side $L$, with periodic boundary conditions. The model is a $65 : 35$ binary mixture of hard disks with diameter ratio $\sigma_{\mathrm{B}}/\sigma_{\mathrm{A}} = 1.4$ (see Eq. (2.7)). This composition prevents crystallisation, while the large size ratio renders swap Monte Carlo inefficient [225, 226].

We first generate equilibrium configurations using the MH algorithm targeting the isothermal–isobaric ($NPT$) ensemble (see Section 2.2). At each pressure $P$, we measure the packing fraction $\phi = \left(\pi \sum_i \sigma_{s_i}^2\right) / \left(4L^2\right)$, which is a dimensionless quantity proportional to the density. It is also convenient to express the pressure in dimensionless form through the compressibility factor $Z = P/\left(\rho k_B T\right)$. Throughout this section, we work with $k_B T = 1$ and $\sigma_{\mathrm{A}} = 1$. For each value of $Z$, we run MH simulations until steady state is reached and record both static and dynamical observables to characterise the resulting configurations.

Starting from these equilibrium configurations, we apply the optimisation algorithms introduced in Section 5.1 to produce configurations with enhanced hyperuniformity and local order. For hyperuniformity, we take three equilibrium configurations with $N = 1000$ at different values of $Z$ and use them as initial conditions for the CBRO dynamics described in Section 5.1.1. Since CBRO requires at least one overlap to start, we begin by giving each particle a small random displacement of maximum amplitude 0.1. We then evolve the system at fixed $\epsilon$ until a steady state is reached. To keep rearrangements minimal, we choose $\epsilon \in [0.18, 0.23]$ and verify that the mean square displacement (see Eq. (2.24)) at the end of the simulation remains below 0.5. The resulting configurations in steady state contain particle overlaps. To restore a valid hard disk configuration without significantly altering the hyperuniform structure obtained during the dynamics, we quench $\epsilon$ to a small value $\epsilon = 0.02$, which rapidly drives the system into an absorbing state with no overlaps. We performed these simulations with the code available at `https://framagit.org/leonardogalliano/random-organisation`, originally developed for Ref. [124].

To generate configurations with enhanced local order, we perform biased Monte Carlo (BMC) simulations as described in Section 5.1.2. We take three equilibrium configurations with $N = 200$ at different values of $Z$ as initial conditions, and apply a biasing potential with strength up to $\lambda = 5 \times 10^5$. To sample from the biased distribution (Eq. (5.1)), we use the two-step procedure described in Section 5.1.2, where trial configurations are generated through short $NVT$ MH trajectories of $n$ MH sweeps targeting the unbiased Boltzmann distribution. We run the BMC simulation for the largest $\lambda$ until $\Theta$ has nearly converged. We performed these simulations using a custom branch of the `ParticlesMC.jl` package (see Section E.2), specifically extended to support biased sampling in the $NPT$ ensemble.

After generating bulk, hyperuniform, and locally ordered configurations, we compare their glass equation of state (EOS) along isobars (see Section 2.1.3). Each configuration is quenched to high compressibility factors $Z$ using $NPT$ MH simulations and evolved until steady state. We then record the average packing fraction $\mathbb{E}[\phi]$ as a function of $Z$. As discussed in Section 2.1.3, the glass EOS can be used to assess kinetic stability, since more stable glasses show systematically higher packing fractions at fixed pressure [107, 225].

## 5.2.2 Bulk properties

We begin by characterising the dynamics of the hard disk mixture at different values of the compressibility factor $Z$ using standard $NPT$ simulations. Figure 5.2(a) shows the evolution of the self-intermediate scattering function $F_s(q^\star, t)$ (see Eq. (2.25)) at the wavevector $q^\star$ corresponding to the first peak of the structure factor $S(q)$. As $Z$ increases, $F_s(q^\star, t)$ develops the familiar two-step relaxation characteristic of glassy dynamics, with an increasingly pronounced plateau at intermediate times.

We extract the structural relaxation time from $F_s(q^\star, \tau_\alpha) = 1/\mathrm{e}$ and show its evolution in Fig. 5.2(b). Since $Z \sim P/T$, this representation is analogous to an Arrhenius plot of $\tau_\alpha$ versus inverse temperature for systems with soft interactions (see Section 2.1). As expected for a glass-forming system, $\tau_\alpha$ grows rapidly with $Z$, spanning several orders of magnitude in the considered range. The onset of slow dynamics occurs around $Z_o \approx 20$, while equilibration becomes increasingly difficult above $Z_g \approx 27$. This narrow window is where we can reliably generate equilibrium glassy configurations that we use as initial conditions for the optimisation algorithms.

To provide a reference for the optimised properties, we measure the degree of hyperuniformity and local order of these bulk equilibrium configurations. Figure 5.3(a) shows the $q$-dependent compressibility $\chi(q)$ (see Eq. (2.21)) for $N = 1000$ at different values of $Z$. As discussed in Section 2.3.1, $\chi(q)$ is the relevant measure

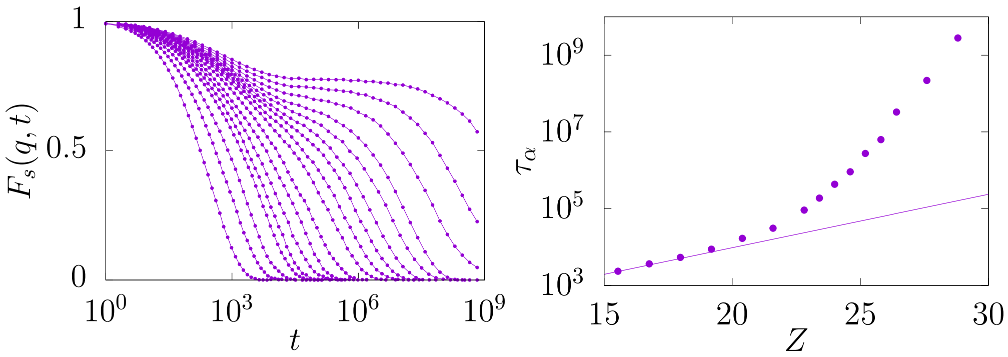

**Figure 5.2:** Glassy dynamics. (a) Self-intermediate scattering function $F_s(q^\star, t)$ at different values of compressibility factor $Z$ measured from $NPT$ simulations. (b) Relaxation time $\tau_\alpha$ as a function of $Z$. Both quantities display the typical signatures of glassy dynamics as $Z$ increases.

of density fluctuations in a multicomponent system. In equilibrium, $\chi(q)$ reaches a finite limit as $q \to 0$, with only a weak dependence on $Z$ over the relevant range. This limiting value provides a reference level of density fluctuations that we seek to reduce using the CBRO dynamics.

Figure 5.3(b) shows the average local order parameter $\mathbb{E}[\Theta]$ (see Eq. (2.22)) and its standard deviation as functions of $Z$ for $N = 200$. We find that $\mathbb{E}[\Theta]$ slightly decreases with $Z$, indicating that local order varies only weakly with compression in this regime. In particular, while the structural relaxation time $\tau_\alpha$ grows by roughly four orders of magnitude over this range (see Fig. 5.2(b)), $\mathbb{E}[\Theta]$ decreases by only about 20%. These equilibrium values will serve as a benchmark to assess the effectiveness of the BMC algorithm in producing configurations with enhanced local order.

Finally, Fig. 5.4 shows the equation of state (EOS) $Z = Z(\phi)$ along isobars. Following Ref. [107], we plot the dimensionless "volume" $v = 1/\mathbb{E}[\phi]$ as a function of $1/Z \sim T/P$, in analogy with the commonly used representation of the average potential energy $\mathbb{E}[U]$ versus temperature in systems with continuous interactions (see Section 2.1.3). In the regime where the system can be equilibrated ($Z < Z_g$), the curve corresponds to the liquid EOS and is independent of the preparation protocol. Above $Z_g$, the system falls out of equilibrium, but we can nonetheless follow the EOS by quenching configurations to higher pressures. This produces the glass EOS [107], which depends on the state *before* the quench. This behaviour is illustrated in the left region of Fig. 5.4, where we report glass EOS curves obtained by quenching equilibrium configurations prepared at different initial $Z$. Configurations equilibrated at larger $Z$ yield glasses with systematically higher

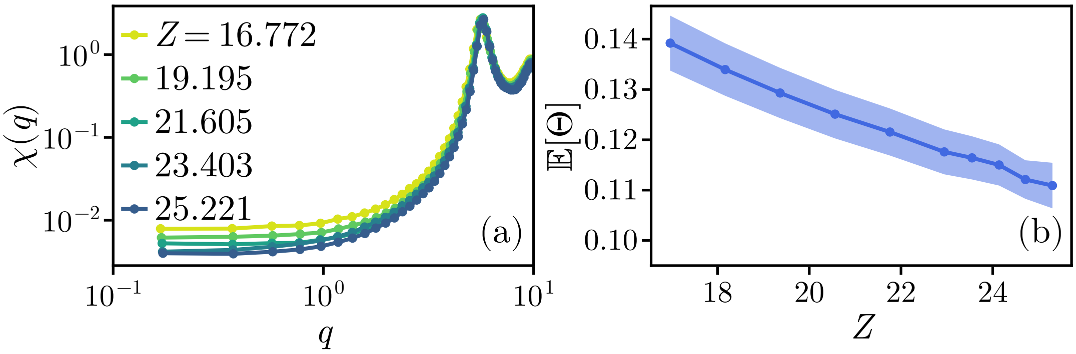

**Figure 5.3:** Bulk properties. (a) $q$-dependent compressibility $\chi(q)$ at different values of compressibility factor $Z$ in an equilibrium system of $N = 1000$ particles. (b) Average local order parameter $\mathbb{E}[\Theta]$ as functions of $Z$ in an equilibrium system of $N = 200$ particles. The shaded area indicates the standard deviation of $\Theta$ across different configurations.

packing fractions at fixed pressure, reflecting their enhanced kinetic stability [107, 225].

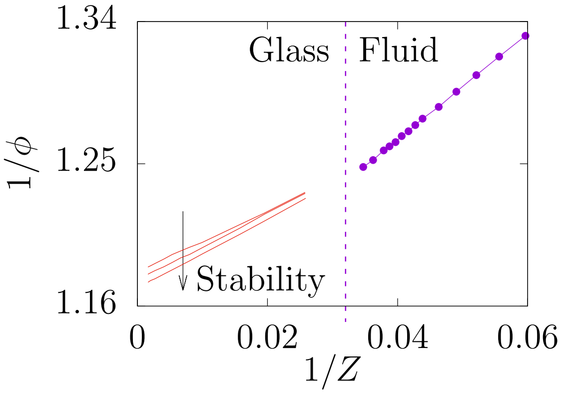

**Figure 5.4:** Equation of state in the $1/\phi$ versus $1/Z$ representation. The equilibrium equation of state for the fluid ends near the computational glass transition $Z_g \approx 27$ (vertical line). We show three equations of state three glasses of different stabilities.

### 5.2.3 Optimised structural properties

We now apply the optimisation procedures introduced in Section 5.1 to selected equilibrium configurations of the hard-disk mixture. Starting from three initial states at different compressibility factors $Z$, we generate configurations with enhanced hyperuniformity using the CBRO dynamics, and with enhanced local order using the BMC algorithm.

### Hyperuniformity

Figure 5.5(a) shows the $q$-dependent compressibility $\chi(q)$ measured in the steady state of the CBRO dynamics starting from the three initial configurations. At small wavevectors, $\chi(q)$ is strongly reduced relative to its equilibrium value, showing that long-wavelength density fluctuations are efficiently suppressed by the dynamics. As expected, the curves eventually deviate from the ideal $q^2$ scaling because of the backbone contribution $\chi_0(q \to 0)$ (see Appendix D.1.4). The smallest accessible wavevector $q_{\rm min}$ provides a practical measure of the achieved hyperuniformity. In our system, we find $\chi(q_{\rm min}) \approx 5 \times 10^{-4}$, which is far below the equilibrium values (see Fig. 5.5(b))

A naive extrapolation of the equilibrium data suggests that reaching comparable $\chi$ values through conventional $NPT$ simualtions would require compressibility factors $Z > 35$, corresponding to relaxation times many orders of magnitude larger than those that can be equilibrated in practice (see Fig. 5.2(b)). The configurations produced by CBRO are therefore atypical and significantly more hyperuniform than any accessible equilibrium configuration.

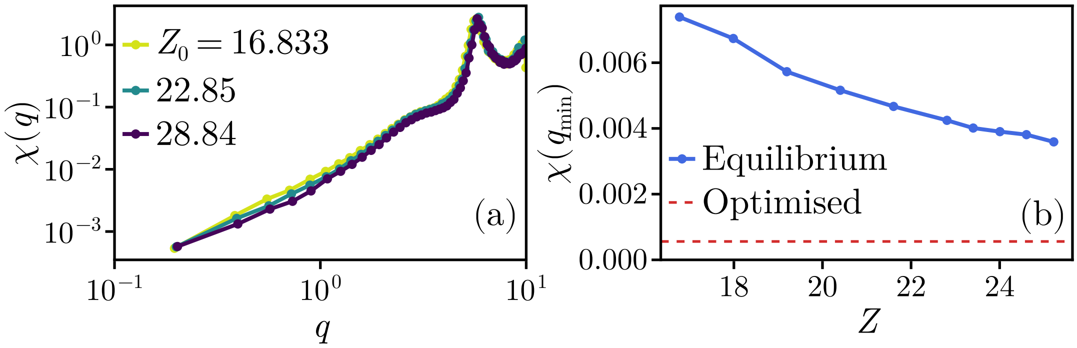

**Figure 5.5:** Optimising hyperuniformity. (a) $q$-dependent compressibility $\chi(q)$ measured in the steady state of the CBRO dynamics starting from three equilibrium configurations at different compressibility factors $Z$. (b) Comparison of the minimum value of $\chi(q)$ for optimised (red dashed line) and equilibrium (blue) configurations. The minimum value of $\chi(q)$ achieved at small $q$ is significantly lower than the equilibrium values.

### Local order parameter

Figure 5.6(a) shows the evolution of the local order parameter $\Theta$ during a BMC simulation with a bias strength $\lambda = 5 \times 10^5$, using one MH sweep per trial move. All the three trajectories display a significant decrease in $\Theta$ relative to their initial equilibrium values, confirming that BMC effectively favours configurations with high local order.

A comparison with the equilibrium curve in Fig. 5.6(b) again indicates that reaching similar values of $\Theta$ by equilibrium compression alone would require $Z > 35$, well beyond our equilibration window. Although the absolute variations in $\Theta$ remain moderate compared with the values reported in Ref. [49], where optimisation is driven by diameter dynamics, a direct quantitative comparison is difficult because the systems and reference states differ. Nevertheless, the configurations produced here remain clearly atypical with respect to equilibrium sampling (see Section 5.2.4).

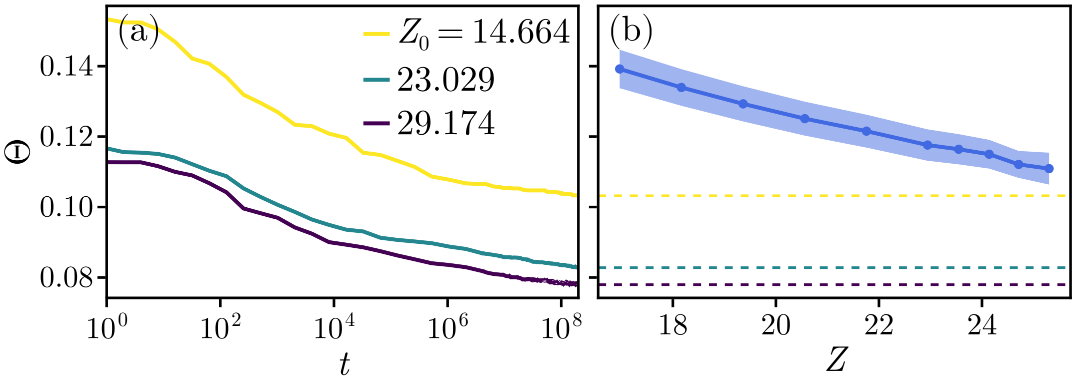

**Figure 5.6:** Optimising local order. (a) Evolution of the local order parameter $\Theta$ during BMC simulations with bias strength $\lambda = 5 \times 10^5$, starting from three equilibrium configurations at different compressibility factors $Z$. (b) Comparison of the final values of $\Theta$ (dashed lines) with the equilibrium values.

### Assessing stability

We now turn to our main objective and test whether the structural optimisation achieved by CBRO or BMC has any impact on kinetic stability. Figure 5.7 shows the glass EOS obtained by quenching both the initial equilibrium configurations and the optimised configurations to high compressibility factors $Z$.

In all cases, the glass EOS of the optimised configurations coincides with that of the corresponding equilibrium states from which they were generated. Configurations prepared at the same initial $Z$ therefore lie in the same basin of attraction in the density landscape, independently of the optimisation protocol. These results demonstrate that enhancing hyperuniformity or local order alone does not lead to increased kinetic stability, challenging the proposed causal links between these structural properties and glass stability.

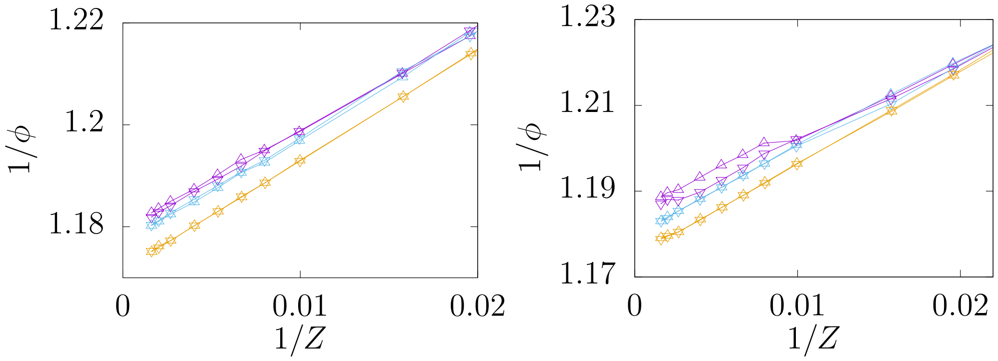

**Figure 5.7:** Glass equation of state of optimised configurations. The curves obtained from quenching equilibrium configurations are compared with (a) configurations with enhanced hyperuniformity and (b) configurations with enhanced local order. Hyperuniform and locally ordered configurations are not more stable than bulk ones.

### 5.2.4 Additional results

We conclude this section with two complementary analyses related to the optimisation of local order. First, we take advantage of the fact that the biased distribution in Eq. (5.1) is known explicitly and use umbrella sampling to reconstruct equilibrium averages. Then, we show that a simple gradient-based algorithm can also generate configurations with reduced $\Theta$, provided that overlaps are carefully considered.

#### Umbrella sampling

To assess the validity of the BMC algorithm for enhancing local order, we measure $\Theta$ in the steady state of biased simulations for several values of the bias strength $\lambda$. Figure 5.8(a) shows the corresponding histograms. As $\lambda$ increases, the distribution shifts towards lower $\Theta$ and becomes narrower, indicating that the bias efficiently suppresses both the mean and fluctuations of the order parameter.

Since the sampling distribution $\widetilde{P}$ is known (up to normalisation), these simulations can be used in umbrella sampling to recover equilibrium averages. Applying the reweighting procedure described in Section 1.1.1, we reconstruct the equilibrium distribution of $\Theta$ by combining samples drawn at different values of $\lambda$, with weights $w(x) = \exp\left(\lambda\Theta^2(x)\right)$. Figure 5.8(b) shows the resulting equilibrium histogram, which follows the one obtained from unbiased simulations. This agreement confirms that the reweighting procedure is reliable and that the BMC simulations correctly sample the biased ensembles.

The equilibrium histogram also quantifies how atypical the optimised configurations are. Already for $\lambda = 7500$, which is far below the values used in the optimisation runs in Section 5.2.3, the reweighted equilibrium density extends down to values of order $10^{-20}$, showing that such configurations lie deep in the tails of the equilibrium distribution and are exceptionally rare.

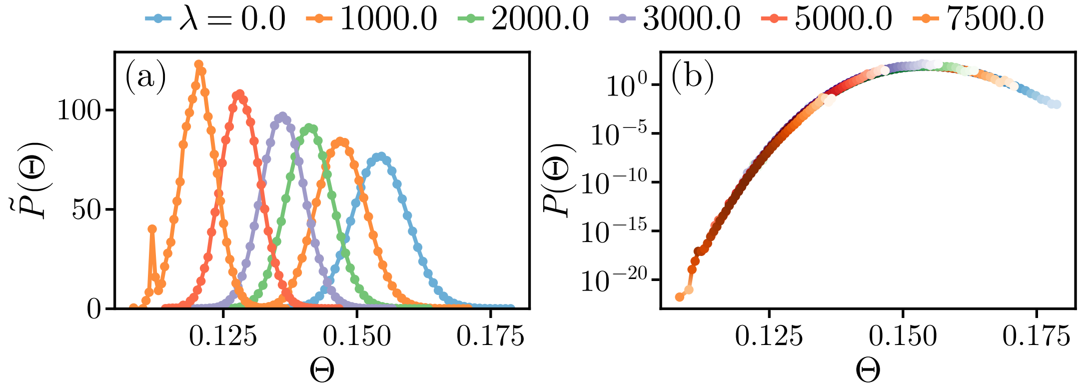

**Figure 5.8:** Umbrella sampling. (a) Histograms of the local order parameter $\Theta$ measured in biased simulations at different values of the bias strength $\lambda$. (b) Reconstructed equilibrium histogram of $\Theta$ obtained by reweighting the biased samples using umbrella sampling.

### Gradient-based optimisation

Finally, we show that local order can also be enhanced using a direct optimisation approach based on gradient descent. Note that a naive gradient descent on $\Theta$ immediately creates particle overlaps, which correspond to invalid hard disk configurations. Attempting to remove the overlaps at the end of the optimisation simply restores $\Theta$ to its typical equilibrium value.

To overcome this issue, we employ a cyclic procedure that alternates between small gradient descent steps on $\Theta$ [18] and relaxation steps that minimise the harmonic potential in Eq. (2.8) using the FIRE algorithm [227], thereby removing overlaps. The tricky part is choosing the step sizes and the relative frequency of the two protocols. Too large gradient steps would create many overlaps that cannot be removed easily, while too small steps would make the optimisation inefficient.

Figure 5.9(a) shows the evolution of $\Theta(t)$ during the optimisation cycle, and Fig. 5.9(b) reports the corresponding fraction $f(t)$ of overlapping particles. As expected, $\Theta$ decreases during the descent steps while $f$ increases. Then, the FIRE relaxation effectively eliminates overlaps, bringing $f$ back to zero, but also slightly increasing $\Theta$. Iterating this process yields hard disk configurations with values of $\Theta$ comparable to those obtained with the BMC algorithm, although the latter remains more general and easier to control.

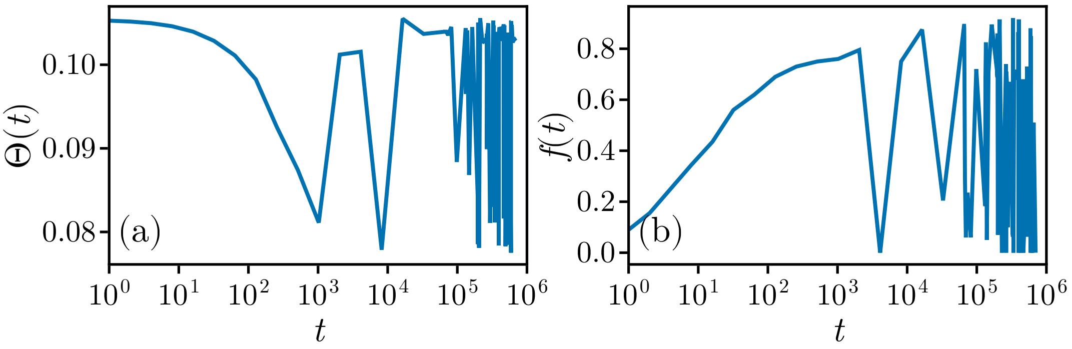

**Figure 5.9:** Gradient-based optimisation of $\Theta$. (a) Evolution of $\Theta$ during optimisation cycles alternating gradient descent steps and FIRE relaxations to remove overlaps. (b) Corresponding fraction of overlapping particles $f$. Valid hard disk configurations corresponds to $f = 0$.

## 5.3 Discussion and perspectives

The results of this chapter clearly show that optimising structural quantities such as hyperuniformity and local order without diameter dynamics does not lead to enhanced kinetic stability. Therefore, these quantities cannot be considered causally responsible for glass stability. The two optimisation algorithms used in this work successfully produce configurations with highly atypical structural properties, as confirmed either by naive extrapolation or through umbrella sampling for local order. Yet, despite these strongly optimised structural signatures, the glass equation of state of the resulting configurations remains indistinguishable from that of the initial equilibrium states. As in conventional glass preparations, the stability is determined entirely by initial state and shows no dependence on the optimisation protocol applied afterwards.

The crucial point is that both algorithms are designed to produce optimised configurations while keeping them in the same basin of attraction as their equilibrium starting points, with minimal rearrangements. This is fundamentally different from previous works based on diameters dynamics [48–51], which is known to greatly facilitate exploration of configuration space and access to deeper minima associated with enhanced stability [5, 140, 159, 228–230]. Our work thus confirms that diameter dynamics is solely responsible for the improved stability observed in these previous studies, and that their results simply stem from the statistical correlation between target quantities and stability. In other words, while more stable glasses display improved structural properties, the reverse implication does not hold: optimising such properties alone does not necessarily produce more stable glasses.

While these algorithms are well suited for isolating the role of specific structural quantities, they remain inherently local and are therefore constrained by the rugged structure of the configuration landscape. A natural extension is the development of general optimisation frameworks capable of targeting arbitrary structural features across different models. For instance, reinforcement learning approaches [64] similar to the ones discussed in Chapter 3 could be adapted to learn policies that generate configurations with desired properties. A different route is to move towards evolutionary strategies, where ensembles of configurations evolve under selection rules designed to favour specific properties [16, 231, 232]. More broadly, these ideas naturally lead to the field of *inverse design*, where the objective is to identify interactions and preparation protocols that yield configurations with prescribed properties [233, 234]. From these perspectives, the challenge is not only to optimise a given structural quantity, but to identify which features of the energy landscape can be modified to produce more stable amorphous configurations.

# Conclusion

This thesis presents a series of algorithmic tools for enhanced sampling and structural optimisation in the complex energy landscapes of amorphous materials. Throughout the manuscript, we explore both the theoretical foundations and the practical implementation of these methods, with the goal of clarifying their potential, their generality, and their intrinsic limitations. Addressing theory and practice in parallel is crucial, as their interplay provides a comprehensive understanding of the algorithms' capabilities and of their applicability to physically relevant systems.

Chapters 1 and 2 established the necessary background by reviewing Monte Carlo methods, generative models, and the phenomenology of glassy systems. This provided a unified language for sampling and optimisation in high-dimensional configuration spaces, where equilibrium properties, dynamics, and optimisation tasks are governed by the same underlying landscape.

In Chapter 3, we generalised the Policy-guided Monte Carlo (PGMC) framework [41] to arbitrary state spaces [42]. By reformulating the adaptation problem within the Metropolis–Hastings algorithm from a measure-theoretic and reinforcement-learning perspective, we derived a policy-gradient theorem valid for general transition kernels and constructed explicit gradient estimators to optimise proposal distributions on the fly. Applications to glass-forming mixtures illustrated both the promise and the intrinsic limitations of optimising local, physically motivated moves: while substantial speed-ups can be achieved in some cases, performance ultimately remains constrained by the locality of the updates and by the structure of the energy landscape itself. The associated implementations in `Arianna.jl` and `ParticlesMC.jl` provide flexible platforms for extending these ideas to other models and fields.

Chapter 4 addressed a complementary strategy based on generative modelling. We introduced Equivariant Riemannian Stochastic Interpolants (eRSI), a framework combining equivariant flows [83] and Riemannian flow matching [81] to approximate Boltzmann distributions for multicomponent particle systems with periodic boundary conditions. Our main theoretical result identified sufficient conditions under which equivariance is preserved along the interpolation path, extending existing results on equivariant flows to the stochastic interpolant framework. Simulations showed that incorporating geometry and symmetry is essential to obtain physically meaningful samples and accurate thermodynamic observables. At the same time, our results highlighted the role of likelihood estimation and reweighting: visually plausible configurations alone are not sufficient, and debiasing remains a key challenge for generative approaches in statistical mechanics.

In Chapter 5, we shifted the focus from sampling to optimisation of target structural properties. We introduced specific algorithms to enhance hyperuniformity [46] and local order [132] in particle configurations without relying on diameter dynamics, thereby isolating the effect of these properties in the optimisation process. Crucially, we demonstrated that these features do not causally enhance glass stability: although the resulting configurations display strongly atypical values of the target properties, their kinetic stability remains unchanged compared to the initial configurations. This analysis clarifies recent claims in the literature [48–51] and supports the conclusion that enhanced stability arises from the ability to explore deeper regions of configuration space, rather than from the optimisation of specific structural metrics alone. More broadly, the chapter highlights the limitations of local optimisation schemes and points toward more general inverse-design strategies.

Taken together, the results of this thesis show that no single algorithmic paradigm provides a universal solution to the challenges posed by amorphous materials. Local adaptive updates, global generative models, and nonequilibrium optimisation methods each capture different aspects of the problem, and their effectiveness is ultimately constrained by the same underlying complexity. The main contribution of this work is to formalise, extend, and combine these approaches within a theoretically grounded framework, while providing concrete implementations that demonstrate their performance in realistic settings under computational and physical constraints. Although many challenges remain and universal solutions are currently out of reach, the frameworks developed here provide a solid foundation for future efforts to design algorithms that are both rigorous and practical, paving the way toward a systematic understanding and exploration of complex energy landscapes.

# Acknowledgements

A PhD is a demanding journey, and certainly not a solitary one. I am deeply grateful to the many people who, in different ways and at different moments, made this achievement possible scientifically, professionally, and personally.

I would first like to thank my collaborators Riccardo Rende, Lorenzo Costigliola, Raphaël Maire, Andrea Plati, Louis Grenioux, Marylou Gabrié, and Giulio Biroli. The articles we wrote together form the backbone of this thesis, and working with all of you has been both stimulating and formative. I have learned a great deal from our exchanges, not only about physics, but also about how to approach research problems with rigour and creativity.

I am also very thankful to Robert Jack and François Landes for carefully reading the first version of this manuscript and for their kind and constructive feedback. Their comments improved the clarity and quality of the final document.

Many other colleagues contributed through discussions, questions, meetings, and informal exchanges across conferences, workshops, and daily research life. I would like to thank Antonio Celani, Henrik Christiansen, Johannes Zierenberg, Shiva Darshan, Luca Manzoni, Troels Bojesen, Francesco Turci, Luca Maria Del Bono, Samuel Tamagnone, Roi Holtzman, Ebo Peerbooms, Misaki Ozawa, Gerhard Jung, Juliane Klamser, Luca Tocchio, Leticia Cugliandolo, Michael Schindler, Anthony Maggs, François Villemot, and Patrick Charbonneau. Our conversations helped me refine ideas, identify weaknesses, and learn how to better present my results.

I am also grateful to the people who made the practical side of this journey considerably easier: Ottavia Umani in Trieste, and Jeldy Cuba Hernandez and Feé Sorrentino at Gulliver, whose help behind the scenes made many things simply work. I would also like to thank Olivier Dauchot for welcoming me to the lively Gulliver group.

Trieste welcomed me with magic sunsets and wonderful people. I was lucky to be “adopted” by the kindest Calabrian trio ever. Luciano, one day, when you are famous, I will proudly say that I shared an office with you. Thank you for

all our discussions on science and for your loyal friendship. Rici, thank you for patiently guiding me at the very beginning of this PhD: in many ways, everything started from your Master's project. I wish you the best in your new favourite city. And Roberta, of course, the true pillar of the group. You are genuinely one of the strongest people I know. A special thanks to Marco, with whom I shared this PhD path, and who patiently tolerated my overly theoretical approach to everything. I can easily imagine a future where your palettes become the matplotlib standard. Thank you, Laria, for your presence and support, especially during the most difficult moments. Thank you Giovanni "il Maestro" and Anirudh for the many chats and the jam sessions in odd time signatures. We will finish that song one day. Thanks to Michele Vi(n)sky for your astonishing operas: easily among the funniest things I have seen during this PhD. Davide Bidoggia, I am very glad to have played a part in your secret network. You always seemed to know how to fix any situation. I also thank Davide Piccioni (now truly a grown-up!) for the company during our trips to SISSA, and Davide Bason, my fellow compatriot: I hope you find what you were looking for in the Far East. Thank you, Ollie, for being such a kind office mate and for helping keep the atmosphere light even during intense days. Thank you, Giorgia, for the small "neglia" (is that how you write it?), which I always keep with me. I would also like to thank all the PhD students and colleagues at ICTP: Michele M., Giovanni N., Ario, Nicholas, Giulio, Francesco, Valter, Mitra, and Laura, for the lunches, coffees, discussions, and the small moments that made daily research life enjoyable.

I had the great opportunity to spend a year in Paris, which is probably one of the best cities in the world. It turned out to be much easier to live in than I expected when I first arrived. Somehow, it felt like home. I suspect that this had much to do with the people I met there. Romain, you have truly been a PhD brother to me. I am glad we met and shared part of this experience together. The time we spent working on Arianna remains one of my favourite parts of this journey. Tianze, thank you for being such a patient and fun office mate. I will never forget our romantic sushi dinner overlooking Marseille. Thank you also to Davide and Milo for your energy, your kindness, and for bringing a bit of Italy into the lab. I would also like to thank the whole Gulliver students group: Danqi, Alice, Yorgos, Keivan, Leo-Paul, Vasily, Victoria, Angelo, Kévin, Marion, Rocio, Natalie, Paul, Pierre, Qingju, Nicolas, Ryan, Sandra, Tristan, Zeynep, and Mats, for making the environment lively and welcoming. Raphaël, thank you for sharing your electrifying productivity and for working together on our interface project. That paper was a real achievement. Louis, you are as kind as you are brilliant (and

you are very brilliant), and working with you on “the cursed project” and filling whiteboards at ENS has been one of the most enjoyable intellectual experiences of this PhD. It also made me realise that this is how I would like to continue doing research. Christoph, thank you for your friendship, your honesty, and your uniquely personal perspective on everything. Federico, thank you for welcoming me during my first days in Paris and for the guitar that has since become one of my favourites. Riccardo, for the time shared in Villejuif and the late-night discussions in front of two moelleux au chocolat. Camille, your advice during a difficult period helped me clarify my direction, and our discussion played an important role in my decision to move to ENS. Myriam, thank you for being there all this time. Knowing that we share something that goes far back has always meant something special to me. I am truly glad that I got to know you better over these years. A very special thank you to my adoptive family in Paris: Bibi, Dami, Chloé, and Fanny. Having you there was very reassuring, and our dinners together helped fill the small family-shaped space that comes with moving to a new city.

None of this would have been possible without the constant support of the people closest to me. Giulio, you have been much more than a friend over these years, a brother in arms after everything we have been through. When I think back to our first days at university, it was hard to imagine where we would end up. I am really glad to have had you by my side throughout these years. Clara, of course you are here. Thank you for always believing in me and for pushing me to give my best. I know you will accomplish something great some day. Thank you Peter for showing me that academic life can also be fun, and that research, despite all its challenges, can be one of the most beautiful jobs out there. Thanks to all my friends in Torino: Amos, Freddie, Sofia, Enzo, Matheo, Bea, Alessia, Luke, Fede, Elisa, Luca, and Simona, who probably still have only a vague idea of what I do for a living, but stand by me nonetheless and appreciate me for who I am. Magò, thank you for sailing alongside me through the storms that brought me here. Your presence has always been reassuring, and I am grateful for the calm and steadiness you bring to troubled waters.

I am deeply grateful to my family for their constant support and encouragement throughout this journey. Grazie Mamma, grazie Papà, for always believing in me and giving me the confidence to keep going. Grazie Niki, your affection has always been one of the most important sources of strength.

Daniele, I’ve learned so much from you that it’s hard to put into words. Not just physics, but also practical methods, little tips and tricks, ways to manage research projects efficiently, and, importantly, how to think about the ethics of science. I

hope you've been happy to work with me, despite me being your "black sheep" with my diabolical tools and questionable operating system :) I will always value the freedom you gave me to pursue research my own way, and the fact that you recognised and valued my contributions and technicalities, even when they were different from the kind of work you usually do. I truly felt trusted and considered.

Ludovic, what can I say? Thank you for believing in me and pointing me toward this path back in Montpellier. Working with you, though challenging at times, has always been incredibly rewarding. I have often felt that we were tackling problems at the very top level of science, and I feel very lucky to have had that chance. I felt pushed to do my best but also safe and supported, as if I were in the best hands possible. Thank you for guiding me when I needed it most.

And finally, thank you, dear reader, for making it this far. I hope you came across something useful in these pages, even a small idea or insight that might help you along your own path in science.

# Appendix A

# Mathematical background

In this appendix we provide a brief overview of basic mathematical notions needed to understand the content of the thesis. We decided to include them here as they may sound unfamiliar to some reader.

## A.1 Measure-theoretic probability

Measure-theoretic probability provides a rigorous foundation for probability theory. Unlike the "elementary" approach, it does not require separating discrete and continuous random variables, but instead treats everything at an abstract level. Our aim in this section is to give all the ingredients needed to understand the language used in the thesis, without overwhelming the reader. We will also try to draw parallels with the more familiar language of probability used in statistical physics. This is summarised in Table A.1. This section is largely based on Refs. [235, 236].

### A.1.1 Some measure theory

Measure theory assigns a consistent notion of "size" to sets. This idea extends naturally from lengths of intervals, to areas and volumes, and eventually to the probabilities of events. To get there, we first need to introduce the basic language.

**Definition A.1.1.** A *measurable space* is a pair $(\mathsf{X}, \Sigma)$, where $\mathsf{X}$ is a set and $\Sigma$ is a $\sigma$-algebra on $\mathsf{X}$, *i.e.* a collection of subsets of $\mathsf{X}$ that satisfies:

(i) $\mathsf{X} \in \Sigma$,

(ii) if $A \in \Sigma$, then $A^c \in \Sigma$,

(iii) if $A_1, A_2, \ldots \in \Sigma$, then $\bigcup_{i=1}^{\infty} A_i \in \Sigma$.

Elements of $\Sigma$ are called *measurable sets*. These are the sets to which we will later assign a probability. Importantly, not every subset of $\mathsf{X}$ needs to be measurable, but one can usually construct a $\sigma$-algebra large enough to contain the sets of interest. Given a collection $\mathcal{C}$ of subsets of $\mathsf{X}$, we denote by $\sigma(\mathcal{C})$ the $\sigma$-algebra *generated* by $\mathcal{C}$, *i.e.* the smallest $\sigma$-algebra containing $\mathcal{C}$.

**Definition A.1.2.** A *measure* $\mu$ on a measurable space $(\mathsf{X}, \Sigma)$ is a function $\mu : \Sigma \to [0, \infty]$ such that if $A_1, A_2, \ldots \in \Sigma$ are pairwise disjoint, then

$$\mu\left(\bigcup_{i=1}^{\infty} A_i\right) = \sum_{i=1}^{\infty} \mu\left(A_i\right) \ . \tag{A.1}$$

The triple $(\mathsf{X}, \Sigma, \mu)$ is called a *measure space*.

A measure is called finite if $\mu(\mathsf{X}) < \infty$, and $\sigma$-finite if $\mathsf{X}$ can be written as a countable union of sets of finite measure. Measures generalise the intuitive notion of *size*: depending on the context, $\mu(A)$ may represent the length of an interval, the volume of a region, or the probability of an event. It is important to remember that a measure is a set function: it assigns sizes to measurable sets, not to individual points. This distinction is often a source of confusion in probability, where one informally reasons in terms of probability density functions.

**Example A.1.1.** Consider the measurable space $(\mathbb{N}, 2^{\mathbb{N}})$. A common measure on this space is the *counting measure* $\tau : 2^{\mathbb{N}} \to [0, \infty]$, that assigns to each subset $A$ of $\mathbb{N}$ its cardinality, *i.e.* $\tau(A) = |A|$. For real numbers $\mathbb{R}$, it can be shown that there exists a $\sigma$-algebra $\Sigma_\lambda \subset 2^{\mathbb{R}}$ of measurable sets and a measure $\lambda : \Sigma_\lambda \to [0, \infty]$ called *Lebesgue measure* that assigns to each interval $(a, b]$ its length, *i.e.* $\lambda\left((a, b]\right) = b - a$. However, not all subsets of $\mathbb{R}$ are measurable [237]. Sometimes we denote the Lebesgue measure simply as $\mathrm{d}x$. Given a measurable set $(\mathsf{X}, \Sigma)$ and a point $x_0 \in \mathsf{X}$, another common measure is the *Dirac kernel* $\delta\left(x_0, \cdot\right) : \Sigma \to \{0,1\}$ defined as $\delta\left(x_0, A\right) = \mathbf{1}_A\left(x_0\right)$.

### Measurable functions

Measures allow us to assign sizes to sets. The next step is to extend this notion to functions, which will lead us to integration.

**Definition A.1.3.** Given two measurable spaces $(\mathsf{X}_1, \Sigma_1)$ and $(\mathsf{X}_2, \Sigma_2)$, a function $h : \mathsf{X}_1 \to \mathsf{X}_2$ is *measurable* if $h^{-1}(B) \in \Sigma_1$ for every $B \in \Sigma_2$.

**Example A.1.2.** For a set $A \subset \mathsf{X}$, the indicator function $\mathbf{1}_A : \mathsf{X} \to \{0,1\}$ is measurable if and only if $A$ is measurable. On the real line, consider $(\mathbb{R}, \mathcal{B})$, where $\mathcal{B}$ is the Borel $\sigma$-algebra (the one generated by all open sets). Every continuous function $f : \mathbb{R} \to \mathbb{R}$ is measurable, since the preimage of an open set under $f$ is open.

Given a measurable function $h : \mathsf{X}_1 \to \mathsf{X}_2$, we call $\sigma(h)$ the $\sigma$-algebra generated by $h$, that is the smallest $\sigma$-algebra on $\mathsf{X}_1$ that makes $h$ measurable.

**Definition A.1.4.** The *pushforward* of $\mu$ by $h$ is the measure $h_{\#}\mu : \Sigma_2 \to [0, \infty]$ defined as

$$h_{\#}\mu(B) = \mu\left(h^{-1}(B)\right), \quad \forall B \in \Sigma_2 \ . \tag{A.2}$$

The pushforward $h_{\#}\mu$ describes how the mass distributed according to $\mu$ on $\mathsf{X}_1$ is *transported* to $\mathsf{X}_2$ by the map $h$. The pushforward can be used to equip $(\mathsf{X}_2, \Sigma_2)$ with its own measure.

### Abstract integration

Measure theory provides an abstract notion of integration that generalises sums and Riemann integrals. We build this definition step by step.

**Definition A.1.5.** A function $s : \mathsf{X} \to \mathbb{R}$ is a *nonnegative simple function* if it can be written as

$$s(x) = \sum_{i=1}^{n} a_i \mathbf{1}_{A_i}(x) \ , \tag{A.3}$$

where $a_i \geq 0$ and $A_i \in \Sigma$. The set of all nonnegative simple functions is denoted by $\mathcal{S}^+$.

**Definition A.1.6.** Let $\mu$ be a finite measure on $(\mathsf{X}, \Sigma)$ and $s \in \mathcal{S}^+$. The *Lebesgue integral* of $s$ with respect to $\mu$ is defined as

$$\int_{\mathsf{X}} s(x)\, \mu(\mathrm{d}x) := \sum_{i=1}^{n} a_i \mu(A_i) \ . \tag{A.4}$$

**Definition A.1.7.** Let $\mu$ be a finite measure on $(\mathsf{X}, \Sigma)$ and $h : \mathsf{X} \to [0, \infty)$ be a nonnegative measurable function. The *Lebesgue integral* of $h$ with respect to $\mu$ is defined as

$$\int_{\mathsf{X}} h(x)\, \mu(\mathrm{d}x) := \sup_{s \in \mathcal{S}^+,\, s \leq h} \left\{ \int_{\mathsf{X}} s(x)\, \mu(\mathrm{d}x) \right\} \ . \tag{A.5}$$

The Lebesgue integral can be infinite. We say that $h$ is *p-integrable*, and we write $h \in \mathcal{L}^p\left(\mathsf{X},\, \Sigma,\, \mu\right)$, if the Lebesbgue integral of $|h|^p$ with respect to $\mu$ is finite.

If $h$ can take both positive and negative values, we write $h = h^+ - h^-$, where $h^+$ and $h^-$ are nonnegative, and define

$$\int_{\mathsf{X}} h(x)\, \mu(\mathrm{d}x) := \int_{\mathsf{X}} h^+(x)\, \mu(\mathrm{d}x) - \int_{\mathsf{X}} h^-(x)\, \mu(\mathrm{d}x)\ , \tag{A.6}$$

provided both terms on the right-hand side are finite.

Integrals can be localised to subsets $A \subset \mathsf{X}$ via

$$\int_A h(x)\, \mu(\mathrm{d}x) := \int_{\mathsf{X}} h(x) \mathbf{1}_A(x)\, \mu(\mathrm{d}x)\ . \tag{A.7}$$

The change of variables formula extends naturally to the Lebesgue integral: given a measurable function $g : \mathsf{X}_1 \to \mathsf{X}_2$ and a measurable function $h : \mathsf{X}_2 \to \mathbb{R}$,

$$\int_{\mathsf{X}_1} h\big(g\left(x_1\right)\big)\, \mu\left(\mathrm{d}x_1\right) = \int_{\mathsf{X}_2} h\left(x_2\right)\, \left(g_{\#}\mu\right)\left(\mathrm{d}x_2\right)\ . \tag{A.8}$$

### Derivative of a measure

So far, measures have looked independent. In practice, many measures are related, and it is useful to describe one in terms of another.

**Definition A.1.8.** Let $\mu$ and $\nu$ be two measures on the measurable space $(\mathsf{X},\, \Sigma)$. We say $\nu$ is *absolutely continuous* with respect to $\mu$, written $\nu \ll \mu$, if $\mu(A) = 0$ implies $\nu(A) = 0$. If both $\nu \ll \mu$ and $\mu \ll \nu$, we say $\mu$ and $\nu$ are *equivalent* and we write $\mu \sim \nu$.

**Definition A.1.9.** Two measures $\mu$ and $\nu$ on $(\mathsf{X},\, \Sigma)$ are *mutually singular*, written $\mu \perp \nu$, if there exists a measurable set $A$ such that $\mu(A) = 0$ and $\nu(A^c) = 0$.

**Theorem A.1.1** (Radon–Nikodym)**.** *Let $\mu$ and $\nu$ be two $\sigma$-finite measures on $(\mathsf{X},\, \Sigma)$. There exists a measurable function $f : \mathsf{X} \to [0, \infty)$ such that*

$$\nu(A) = \int_A f(x)\, \mu(\mathrm{d}x), \quad \forall A \in \Sigma\ , \tag{A.9}$$

*if and only if $\nu \ll \mu$.*

*Proof.* See [237, Theorem 6.10]. □

The function $f$ is unique up to a $\mu$-null set and is called the *Radon–Nikodym derivative* of $\nu$ with respect to $\mu$, denoted $f = \frac{\mathrm{d}\nu}{\mathrm{d}\mu}$.

Every $\sigma$-finite measure $\nu$ can be decomposed relative to $\mu$ as $\nu = \nu_a + \nu_s$, where $\nu_a \ll \mu$ and $\nu_s \perp \mu$. This is the *Lebesgue decomposition* of $\nu$ with respect to $\mu$, and it is unique.

### A.1.2 Probability language

We will see that all the work we did above in defining concepts in measure theory can be directly translated into more familiar probability terms.

**Definition A.1.10.** Given a measurable space $(\Omega,\ \mathcal{F})$, a measure $P$ such that $P(\Omega) = 1$ is called a *probability measure* and the triple $(\Omega,\ \mathcal{F},\ P)$ is called a *probability space.*

We change notation from $(\mathsf{X},\ \Sigma,\ \mu)$ to $(\Omega,\ \mathcal{F},\ P)$ to stress the difference. In standard probability, the set $\Omega$ is usually called the *outcome space*, while elements of $\mathcal{F}$ are called *events*.

**Example A.1.3** (Coin toss)**.** Consider the toss of two coins. In this case the outcome space is given by all possible combinations $\Omega = \{\text{HH, HT, TH, TT}\}$ and we can take the $\sigma$-algebra to be the power set $\mathcal{F} = 2^{\Omega}$. A reasonable choice of $P$ is that of a function $P : \mathcal{F} \to [0{,}1]$ that maps each singleton $\{\text{xx}\}$ into $1/4$. The other combinations follow from the definition of measure as $P\left(A_1 \cup A_2\right) = P\left(A_1\right) + P\left(A_2\right)$, for all $A_1, A_2 \in \mathcal{F}$.

**Example A.1.4** (Particle systems)**.** Consider a particle system in thermal equilibrium. In this case the outcame space $\Omega$ is the high-dimensional *phase space* of all possible configurations. An event $A \in \mathcal{F}$ can be a region of such phase space, *i.e.* a set of specific configurations. A suitable probability measure is the *Boltzmann measure*, that assigns to each region $A$ the probability of finding the system within $A$.

In probability language a measurable function $X : \Omega \to E$ between $(\Omega,\ \mathcal{F})$ and $(E,\ \mathcal{E})$ is called a *random variable.* Intuitively, the random variable $X$ maps the abstract outcomes $\omega \in \Omega$ to more "concrete" values in $E$ (usually $E = \mathbb{R}^n$). The pushforward $X_{\#}P : \mathcal{E} \to [0, \infty)$ of a probability measure $P$ on $(\Omega,\ \mathcal{F})$ by $X$ is called *probability distribution* of $X$ and it is usually denoted as $P_X$. We then say that $X$ is *distributed* according to $P_X$ and write $X \sim P_X$. In statistical physics, this notation has an operational interpretation: the distribution $P_X$ specifies the ensemble, and sampling from it corresponds to realising typical configurations of the physical system according to its statistical weights. At this level it is important to distinguish between the underlying probability measure $P$ on $(\Omega,\ \mathcal{F})$ and the induced distribution $P_X$ on $(E,\ \mathcal{E})$. In practice, however, one typically works directly with $P_X$, since it is the object that encodes the statistics of the *observable* values of $X$.

**Definition A.1.11.** Given a probability space $(\Omega,\, \mathcal{F},\, P)$ and an integrable random variable $X : \Omega \to \mathbb{R}$, the *expectation* of $X$ is defined as

$$\mathbb{E}[X] \coloneqq \int_\Omega X(\omega)\, P(\mathrm{d}\omega)\ . \tag{A.10}$$

It is common to make explicit the distribution of $\omega$ in the expectation notation: $\mathbb{E}_{\omega\sim P}[X] \coloneqq \mathbb{E}[X]$. This is particularly convenient when dealing with multiple random variables.

In many applications, it is convenient to describe *conditional distributions* or stochastic transitions directly.

**Definition A.1.12.** A *transition kernel* is a function $K : \Omega \times \mathcal{F} \to [0{,}1]$ such that for each fixed $\omega \in \Omega$, $K(\omega, \cdot)$ is a probability measure on $(\Omega,\, \mathcal{F})$, and for each fixed $A \in \mathcal{F}$, $K(\cdot, A)$ is measurable.

Intuitively, $K(\omega, A)$ specifies a rule for moving from a state $\omega$ to any new state $\omega' \in A$.

**Definition A.1.13.** Given a probability space $(\Omega,\, \mathcal{F},\, P)$, a measurable space $(E,\, \mathcal{E})$ and a random variable $X : \Omega \to E$, let $h : E \to \mathbb{R}$ be an integrable random variable from $(E,\, \mathcal{E})$ to $(\mathbb{R},\, \mathcal{B})$. The *conditional expectation* of $h$ with respect to $X$ is a random variable $G : E \to \mathbb{R}$ such that, for every bounded measurable function $\varphi : E \to \mathbb{R}$,

$$\int_E \varphi(x)\, G(x)\, P_X\,(\mathrm{d}x) = \int_\Omega \varphi(X(\omega))\, h(\omega)\, P\,(\mathrm{d}\omega)\ \ . \tag{A.11}$$

We write $\mathbb{E}\,[h \mid X] \coloneqq G$.

We can also define the conditional expectation of $h$ with respect to an event $A \in \mathcal{F}$ by taking $X : \Omega \to \{0{,}1\}$ defined as $X(\omega) = \mathbf{1}_A(\omega)$. Then we set

$$\mathbb{E}\,[h \mid X \in A] \coloneqq G(1) = \frac{1}{P(A)} \int_A h(\omega)\, P\,(\mathrm{d}\omega)\ \ . \tag{A.12}$$

Finally, we can define probability densities, which are nothing but Radon-Nikodym derivatives of a probability measure. Given a measurable space $(\Omega,\, \mathcal{F},\, P)$ and a measure $\mu$ that dominates $P$, we say that $p = \frac{\mathrm{d}P}{\mathrm{d}\mu}$ is the *density* of $P$ with respect to $\mu$. The most common situation is when $\mu$ is a reference measure such as the Lebesgue measure $\lambda$ on $\mathbb{R}^n$ or the counting measure $\tau$ on a finite or countable set. In these cases, the density $p$ is the familiar *probability density function* or *probability mass function*, respectively.

As promised, we summarise the correspondence between measure theory and probability terminology in Table A.1.

| **Measure theory** | **Probability** |
|---|---|
| Abstract space $\mathsf{X}$ | Outcome space $\Omega$ |
| Sigma-algebra $\Sigma$ | Set of events $\mathcal{F}$ |
| Measure $\mu$ | Probability $P$ |
| Measurable function $h$ | Random variable $X$ |
| Integral $\int_{\mathsf{X}} h(x)\,\mu(\mathrm{d}x)$ | Expectation $\mathbb{E}[X]$ |
| Radon-Nikodym derivative $\frac{d\mu}{d\nu}$ | Probability density $p$ |

**Table A.1:** Correspondence between measure theory and probability terminology.

### A.1.3 Equivalence and convergence

We conclude this section by defining some useful notions for comparing measures and random variables. At the very end we also state two important theorems about exchanging limits and integrals.

**Definition A.1.14.** Given a measure space $(\mathsf{X}, \Sigma, \mu)$, we say that a property holds *$\mu$-almost everywhere* if the set of points where it fails has measure zero.

**Definition A.1.15.** Let $(\mathsf{X}, \Sigma)$ be a measurable metric space and $\mu_n, \mu$ finite measures on it. We say that

- $\mu_n$ *converges weakly* to $\mu$ if
$$\int_{\mathsf{X}} h(x)\,\mu_n(\mathrm{d}x) \to \int_{\mathsf{X}} h(x)\,\mu(\mathrm{d}x) \ ,$$
for all bounded continuous functions $h : \mathsf{X} \to \mathbb{R}$.

- $\mu_n$ *converges setwise* to $\mu$ if
$$\mu_n(A) \to \mu(A), \quad \forall A \in \Sigma \ .$$

- $\mu_n$ *converges in total variation* to $\mu$ if
$$\sup_{A\in\Sigma} |\mu_n(A) - \mu(A)| \to 0 \ .$$

Convergence in total variation implies convergence setwise, which in turn implies weak convergence.

**Definition A.1.16.** Given a probability space $(\Omega,\, \mathcal{F},\, P)$ and a measurable space $(E,\, \mathcal{E})$, we say that two random variables $X_1, X_2 : \Omega \to E$ are

- *Equal in distribution* if $P_{X_1} = P_{X_2}$.
- *Equal almost surely* if $X_1 = X_2$ $P$-almost everywhere.
- *Pointwise equal* if $X_1(\omega) = X_2(\omega)$ for all $\omega \in \Omega$.

**Definition A.1.17.** Given a probability space $(\Omega,\, \mathcal{F},\, P)$ and a measurable space $(E,\, \mathcal{E})$ equipped with a distance $d : E \times E \to [0, \infty]$, let $(X_n)_{n=1}^{\infty}$ and $X$ be random variables from $(\Omega,\, \mathcal{F})$ to $(E,\, \mathcal{E})$. We say that

- $X_n$ *converges in distribution* (or *weakly*, or *in law*) to $X$ if $P_{X_n}$ converges weakly to $P_X$.
- $X_n$ *converges in probability* to $X$ if, for all $\varepsilon > 0$,

$$\lim_{n\to\infty} P\left(d\left(X_n, X\right) > \varepsilon\right) = 0 \; .$$

- $X_n$ *converges almost surely* to $X$ if $X_n \to X$ $P$-almost everywhere.

Almost sure convergence implies convergence in probability, which in turn implies convergence in distribution.

**Theorem A.1.2** (Monotone convergence)**.** *Let $\mu$ be a $\sigma$-finite measure on $(\mathsf{X},\, \Sigma)$ and $(h_n)_{n=1}^{\infty}$ be a sequence of nonnegative measurable functions such that $h_{n+1} \geq h_n$ $\mu$-almost everywhere for all $n$. Let $h = \lim_{n\to\infty} h_n$. Then $h$ is measurable and*

$$\int_{\mathsf{X}} h_n(x)\, \mu(\mathrm{d}x) \to \int_{\mathsf{X}} h(x)\, \mu(\mathrm{d}x) \; . \tag{A.13}$$

*Proof.* See [237, Theorem 1.26]. ☐

**Theorem A.1.3** (Dominated convergence)**.** *Let $\mu$ be a $\sigma$-finite measure on $(\mathsf{X},\, \Sigma)$, and let $g : \mathsf{X} \to \mathbb{R}$ be integrable. Suppose $(h_n)_{n=1}^{\infty}$ is a sequence of measurable functions such that $|h_n(x)| \leq g(x)$ $\mu$-almost everywhere for all $n$. If $h = \lim_{n\to\infty} h_n$ exists pointwise, then $h$ is integrable and*

$$\int_{\mathsf{X}} h_n(x)\, \mu(\mathrm{d}x) \to \int_{\mathsf{X}} h(x)\, \mu(\mathrm{d}x) \; . \tag{A.14}$$

*Proof.* See [237, Theorem 1.34]. ☐

**Corollary A.1.4** (Differentiation under integral sign)**.** *Let* $\mu$ *be a* $\sigma$*-finite measure on* $(\mathsf{X}, \Sigma)$*, and let* $I \subset \mathbb{R}$ *be an open interval. Suppose* $h : \mathsf{X} \times I \to \mathbb{R}$ *is such that for all* $t \in I$*, the function* $h(\cdot, t)$ *is measurable and for all* $x \in \mathsf{X}$*, the function* $h(x, \cdot)$ *is differentiable. If there exists an integrable function* $g : \mathsf{X} \to [0, \infty)$ *such that*

$$\left| \frac{\partial}{\partial t} h(x,t) \right| \leq g(x) \ , \tag{A.15}$$

*for all* $x \in \mathsf{X}$ *and* $t \in I$*, then the function* $H : I \to \mathbb{R}$ *defined as*

$$H(t) = \int_{\mathsf{X}} h(x,t) \, \mu(\mathrm{d}x) \ , \tag{A.16}$$

*is differentiable and*

$$\frac{\mathrm{d}}{\mathrm{d}t} H(t) = \int_{\mathsf{X}} \frac{\partial}{\partial t} h(x,t) \, \mu(\mathrm{d}x) \ . \tag{A.17}$$

*Proof.* See [238, Theorem 2.27]. □

## A.2 Riemannian geometry

When dealing with particle systems, one rarely works in Euclidean space. Instead, these systems are usually simulated with *periodic boundary conditions* (PBCs), to reduce boundary effects. This means that the true configuration space is not $\mathbb{R}^d$ but a flat (hyper)torus, obtained by identifying opposite sides of a box of length $L$. Geometry then matters: distances, interpolation paths, or even the simple notion of moving a particle must be defined in a way that respects the underlying manifold.

To handle such situations in a rigorous way, we need a few concepts from *Riemannian geometry*, which studies properties of non Euclidean spaces [239]. The purpose of this section is to introduce the essential notions needed in this thesis: manifolds and tangent spaces, metrics, geodesics, exponential and logarithmic maps, differential operators and probability measures on manifolds. These notions will allow us to generalise familiar operations from Euclidean space to spaces with more complex geometry.

### A.2.1 Basic concepts

**Definition A.2.1.** A *smooth d-dimensional manifold* $\mathcal{M}$ is a topological space such that for every point $x \in M$ there exists an open neighbourhood $U \subset \mathcal{M}$ and a homeomorphism (called a *chart*) $\varphi : U \to V \subset \mathbb{R}^d$. An *atlas* is a collection of charts that cover $\mathcal{M}$, and it is said to be smooth if the transition maps between overlapping charts are smooth.

In other words, a manifold is a space that looks *locally* like $\mathbb{R}^d$, even if its *global* structure may differ. Charts are a way to assign coordinates to points in $\mathcal{M}$, and an atlas ensures that these coordinates are smoothly compatible across overlapping regions.

At each point $x \in \mathcal{M}$, the *tangent space* $T_x\mathcal{M}$ is the $d$-dimensional vector space that contains all possible directions in which one can move from $x$. The disjoint union of all tangent spaces is called the *tangent bundle*, denoted $T\mathcal{M} = \bigsqcup_{x\in\mathcal{M}} T_x\mathcal{M}$. For the flat torus, the tangent space $T_x\mathcal{M}$ is simply $\mathbb{R}^d$.

**Definition A.2.2.** A *Riemannian metric* on $\mathcal{M}$ is a smoothly varying inner product $g_x : T_x\mathcal{M} \times T_x\mathcal{M} \to \mathbb{R}$ defined on each tangent space. The pair $(\mathcal{M}, g)$ is called a *Riemannian manifold.*

The Riemannian metric is essential to measure angles, lengths and volumes on a manifold. It induces a norm $\|v\|_x = \sqrt{g_x(v,v)}$ for $v \in T_x\mathcal{M}$ and a volume element $\mathrm{dVol}(x) = \sqrt{|G(x)|}\,\lambda(\mathrm{d}x)$, where $G(x)$ is the matrix representation of $g_x$ in coordinates and $\lambda$ is the Lebesbgue measure. On the flat torus with side length $L$, the metric is the same as the Euclidean metric, so distances and volumes are computed exactly as in $\mathbb{R}^d$, but with coordinates wrapped modulo $L$.

With a metric, we can define the notion of length of a curve $\gamma : [0,1] \to \mathcal{M}$ as

$$L(\gamma) = \int_0^1 \|\gamma'(t)\|_{\gamma(t)}\,\mathrm{d}t\,. \tag{A.18}$$

The shortest curves between two points are called *geodesics.* In Euclidean space these are straight lines, while on the torus they are straight lines "wrapped around" periodically. Since geodesics, they induce a global distance $d_{\mathcal{M}}(x, y)$ on the manifold.

**Definition A.2.3.** Given $x \in \mathcal{M}$ and $v \in T_x\mathcal{M}$, the *exponential map* $\exp_x : T_x\mathcal{M} \to \mathcal{M}$ is defined by $\exp_x(v) = \gamma(1)$, where $\gamma$ is the unique geodesic starting at $x$ with initial velocity $v$. Conversely, for $y \in \mathcal{M}$ sufficiently close to $x$, the *logarithmic map* $\log_x : \mathcal{M} \to T_x\mathcal{M}$ gives the tangent vector $v$ such that $\exp_x(v) = y$.

Using these maps, the geodesic connecting $x$ to $y$ can be written explicitly as

$$\gamma(t) = \exp_x\big(t\,\log_x(y)\big), \quad t \in [0,1]\,. \tag{A.19}$$

The exponential map tells how to move on a "straight" line at constant velocity $v$, while the logarithmic map gives the initial velocity needed to reach $y$ from $x$ along a geodesic.

**Proposition A.2.1.** *Let $(\mathcal{M}, g)$ be a Riemannian manifold, and let $x, y \in \mathcal{M}$. Then the following properties hold* *(i)* $d_{\mathcal{M}}(x, y) = \|\log_x(y)\|_x$,

*(ii)* $\log_x(y) = -\frac{1}{2}\, \nabla_x d^2_{\mathcal{M}}(x, y)$,

*(iii) There exists a neighborhood* $U_x \subset \mathcal{M}$ *of* $x$ *such that for all* $y \in U_x$ *and all* $v \in T_x\mathcal{M}$ *sufficiently small,* $\exp_x(\log_x(y)) = y$ *and* $\log_x(\exp_x(v)) = v$.

*Proof.* See [239, Chapter 5]. ☐

**Example A.2.1** (Euclidean space)**.** In Euclidean space $\mathbb{R}^d$, geodesics are straight lines, so the global distance is $d_{\mathcal{M}}(x, y) = \|x - y\|$ and the exponential map is simply vector addition: $\exp_x(v) = x + v$. That is, starting at $x$ and moving in the direction $v$ for unit "time" lands us at $x + v$. The logarithmic map is just the difference: given two points $x, y \in \mathbb{R}^d$, the unique tangent vector $v$ such that $x + v = y$ is $\log_x(y) = y - x$. The geodesic connecting $x$ to $y$ is then $\gamma(t) = (1 - t)x + ty$, which is the equation of a straight line from $x$ to $y$.

**Example A.2.2** (Flat torus)**.** On the flat torus, the situation is similar but moving along a straight line may wrap around the domain. The global distance is the *nearest image distance*, given by

$$d_{\mathcal{M}}(x, y) = \min_{k \in \mathbb{Z}^d} \|(y + kL) - x\| \ . \tag{A.20}$$

The exponential map on the torus is still given by addition, but followed by wrapping the result back into the fundamental domain, that is

$$\exp_x(v) = (x + v) \,\%\, L \ . \tag{A.21}$$

The logarithmic map selects the shortest displacement vector from $x$ to $y$, accounting for all possible "wrapped" paths due to periodicity. This means that, among all vectors connecting $x$ to periodic images of $y$, we pick the one with minimal norm. This logarithmic map can be written as

$$\log_x(y) = \left(y - x + \frac{L}{2}\right) \% \, L - \frac{L}{2} \ . \tag{A.22}$$

### A.2.2 Differential operators

Before defining operators such as the gradient or divergence, we first need a notion of differentiation that works intrinsically on a manifold, without relying on coordinates. In Euclidean space, all tangent spaces can be identified with $\mathbb{R}^d$, but this is not the case on a general manifold.

**Definition A.2.4.** Let $f : \mathcal{M} \to N$ be a smooth map. The *differential* of $f$ at $x \in \mathcal{M}$, is the linear map $\mathrm{d}f_x : T_x\mathcal{M} \to T_{f(x)}\mathcal{N}$ defined by

$$\mathrm{d}f_x(v) = \frac{\mathrm{d}}{\mathrm{d}t} f(\gamma(t))\bigg|_{t=0} \ , \tag{A.23}$$

where $\gamma$ is any smooth curve with $\gamma(0) = x$ and $\gamma'(0) = v$.

Intuitively, $\mathrm{d}f_x$ is the best linear approximation of $f$ near $x$, mapping tangent vectors at $x$ to tangent vectors at $f(x)$. In local coordinates, if $f$ is represented by its components $f^\alpha(x^1, \dots, x^d)$, then

$$(\mathrm{d}f_x)^\alpha{}_i = \frac{\partial f^\alpha}{\partial x^i}(x) \ , \tag{A.24}$$

so that $\mathrm{d}f_x$ coincides with the Jacobian matrix of $f$. Applying the differential to a tangent vector $v = T_x\mathcal{M}$ is then simply matrix-vector multiplication:

$$\mathrm{d}f_x(v)^\alpha = \frac{\partial f^\alpha}{\partial x^i}(x)\, v^i \ . \tag{A.25}$$

For a function $f : \mathbb{R} \to \mathbb{R}$ in Euclidean space, applying the differential $\mathrm{d}f_x$ to a vector $v$ coincides with the usual directional derivative $v \cdot \nabla f$. The differential can be used to define the *gradient* of a smooth function $f : \mathcal{M} \to \mathbb{R}$ is the vector $\nabla f(x) \in T_x\mathcal{M}$ satisfying

$$g_x(\nabla f(x),\, v) = \mathrm{d}f_x(v), \qquad \forall v \in T_x\mathcal{M} \ . \tag{A.26}$$

To differentiate a vector field $v : \mathcal{M} \to T\mathcal{M}$, one cannot take a simple difference between $v(x)$ and $v(y)$, because they belong to different tangent spaces. To compare them, we first need a way to “transport” vectors from one tangent space to another in a consistent manner. This is the *Levi–Civita connection*, usually denoted by $\nabla$. It tells us how to differentiate vector fields along curves on the manifold. In local coordinates $\{x^i\}$ with metric tensor $G = (g_{ij})$, we introduce the *Christoffel symbols* of the metric,

$$\Gamma^i_{jk} \coloneqq \tfrac{1}{2}\, g^{i\ell}\big(\partial_j g_{k\ell} + \partial_k g_{j\ell} - \partial_\ell g_{jk}\big) \ , \tag{A.27}$$

which describe how the basis vectors change from point to point on the manifold.

**Definition A.2.5.** Given two vector fields $v, w \in T\mathcal{M}$ the *covariant derivative* $\nabla_v w$ is defined as

$$(\nabla_v w)^i = v^j \partial_j w^i + \Gamma^i_{jk} v^j w^k \ . \tag{A.28}$$

The covariant derivative $\nabla_v w$ describes the rate of change of $w$ in the direction $v$. We can use this notion to define the *divergence* of a vector field as the trace of the covariant derivative. In local coordinates

$$\mathrm{div}_g(v) = \frac{1}{\sqrt{|G|}}\,\partial_i\Big(\sqrt{|G|}\,v^i\Big)\ . \tag{A.29}$$

This reduces to the usual Euclidean divergence when $G$ is the identity. On the flat torus, where the metric is constant, we recover $\mathrm{div}_g(v) = \nabla \cdot v$.

### A.2.3 Probability measures on manifolds

Finally, we need to generalise probability to manifolds [240]. The Riemannian metric induces a canonical measure Vol on $\mathcal{M}$, called the *Riemannian volume measure*, with local density $\sqrt{|G(x)|}$ relative to the Lebesgue measure in coordinates. A probability distribution $P$ on $\mathcal{M}$ is then a probability measure absolutely continuous with respect to Vol, with density $p = \frac{\mathrm{d}P}{\mathrm{dVol}}$.

On the flat torus, the Riemannian volume is just the uniform measure obtained by wrapping Lebesgue measure on the fundamental domain. Thus, probability densities on the torus look very similar to those in $\mathbb{R}^d$, but are defined modulo $L$.

# Appendix B

# Supplementary material for Chapter 3

## B.1 Objective function

**Proposition B.1.1.** *The objective function $J$ defined in Eq.* (3.1) *can be expressed as*

$$J\left[Q\right] \;=\; \mathbb{E}_{\substack{x\sim P\\ x'\sim Q}}\left[\big(r\left(x,x'\right)-r\left(x,x\right)\big)\alpha\left(x,x'\right)\right]+\mathbb{E}_{x\sim P}\left[r\left(x,x\right)\right]\ . \tag{B.1}$$

*Proof.*

$$\begin{aligned}
J\left[Q\right] &= \mathbb{E}_{\substack{x\sim P\\ x'\sim K}}\left[r\left(x,x'\right)\right]\\
&= \int_{\mathsf{X}^2} r\left(x,x'\right)K\left(x,\mathrm{d}x'\right)P\left(\mathrm{d}x\right)\\
&= \int_{\mathsf{X}^2} r\left(x,x'\right)\alpha\left(x,x'\right)Q\left(x,\mathrm{d}x'\right)P\left(\mathrm{d}x\right)+\\
&\quad+\int_{\mathsf{X}^3} r\left(x,x'\right)\delta\left(x,\mathrm{d}x'\right)\left(1-\alpha\left(x,x''\right)\right)Q\left(x,\mathrm{d}x''\right)P\left(\mathrm{d}x\right)\ ,
\end{aligned}$$

where we used the definition of the transition kernel $K$ in Eq. (1.21). Let us denote the two terms above by $J_1[Q]$ and $J_2[Q]$ respectively. The second term $J_2$ can be simplified using the properties of the Dirac kernel:

$$\begin{aligned} J_2[Q] &= \int_{\mathsf{X}^3} r\left(x, x'\right) \delta\left(x, \mathrm{d}x'\right) \left(1 - \alpha\left(x, x''\right)\right) Q\left(x, \mathrm{d}x''\right) P\left(\mathrm{d}x\right) \\ &= \int_{\mathsf{X}^2} r\left(x, x\right) \left(1 - \alpha\left(x, x''\right)\right) Q\left(x, \mathrm{d}x''\right) P\left(\mathrm{d}x\right) \\ &= \int_{\mathsf{X}^2} r\left(x, x\right) Q\left(x, \mathrm{d}x''\right) P\left(\mathrm{d}x\right) - \int_{\mathsf{X}^2} r\left(x, x\right) \alpha\left(x, x''\right) Q\left(x, \mathrm{d}x''\right) P\left(\mathrm{d}x\right) \\ &= \int_{\mathsf{X}} r\left(x, x\right) P\left(\mathrm{d}x\right) - \int_{\mathsf{X}^2} r\left(x, x\right) \alpha\left(x, x''\right) Q\left(x, \mathrm{d}x''\right) P\left(\mathrm{d}x\right) \ , \end{aligned}$$

where we used the normalisation of $Q$ in the last step. Renaming the dummy variable $x''$ as $x'$ and combining the two terms, we obtain

$$\begin{aligned} J\left[Q\right] &= J_1[Q] + J_2[Q] \\ &= \int_{\mathsf{X}^2} \big(r\left(x, x'\right) - r\left(x, x\right)\big) \alpha\left(x, x'\right) Q\left(x, \mathrm{d}x'\right) P\left(\mathrm{d}x\right) + \int_{\mathsf{X}} r\left(x, x\right) P\left(\mathrm{d}x\right) \\ &= \mathbb{E}_{\substack{x \sim P \\ x' \sim Q}} \left[\big(r\left(x, x'\right) - r\left(x, x\right)\big) \alpha\left(x, x'\right)\right] + \mathbb{E}_{x \sim P}\left[r\left(x, x\right)\right] \ . \qquad \square \end{aligned}$$

Note that if the reward function satisfies $r(x, x) = 0$ for all $x \in \mathsf{X}$, then Proposition B.1.1 reduces to the expression derived in Ref. [42, Equation 9].

## B.2 Gradient derivation

**Lemma B.2.1.** *Let $J$ be the objective function defined in Eq.* (3.3)*. Assume there exists a neighbourhood $U_{\theta_0} \subseteq \Theta$ of $\theta_0$ such that $Q_\theta \ll Q_{\theta_0}$ for all $\theta \in U_{\theta_0}$ and let*

$$L_\theta\left(x, x'\right) \coloneqq \frac{Q_\theta\left(x, \mathrm{d}x'\right)}{Q_{\theta_0}\left(x, \mathrm{d}x'\right)} \ . \tag{B.2}$$

*Then, $J$ can be expressed as*

$$J\left(\theta\right) \ = \ \mathbb{E}_{\substack{x \sim P \\ x' \sim Q_{\theta_0}}} \left[\big(r\left(x, x'\right) - r\left(x, x\right)\big) \alpha_\theta\left(x, x'\right) L_\theta\left(x, x'\right)\right] \ , \tag{B.3}$$

*for all $\theta \in U_{\theta_0}$.*

*Proof.* Let $\theta \in U_{\theta_0}$. By importance sampling,

$$\begin{aligned} J\left(\theta\right) &= \mathbb{E}_{\substack{x \sim P \\ x' \sim Q_\theta}} \left[\big(r\left(x, x'\right) - r\left(x, x\right)\big) \alpha_\theta\left(x, x'\right)\right] \\ &= \int_{\mathsf{X}^2} \big(r\left(x, x'\right) - r\left(x, x\right)\big) \alpha_\theta\left(x, x'\right) Q_\theta\left(x, \mathrm{d}x'\right) P\left(\mathrm{d}x\right) \\ &= \int_{\mathsf{X}^2} \big(r\left(x, x'\right) - r\left(x, x\right)\big) \alpha_\theta\left(x, x'\right) L_\theta\left(x, x'\right) Q_{\theta_0}\left(x, \mathrm{d}x'\right) P\left(\mathrm{d}x\right) \\ &= \mathbb{E}_{\substack{x \sim P \\ x' \sim Q_{\theta_0}}} \left[\big(r\left(x, x'\right) - r\left(x, x\right)\big) \alpha_\theta\left(x, x'\right) L_\theta\left(x, x'\right)\right] \ . \qquad \square \end{aligned}$$

**Theorem 3.1.1** (Policy-guided Monte Carlo gradient estimation)**.** *Let $J$ be the objective function defined in Eq.* (3.3)*, $\theta_0 \in \Theta$, and $\nu_{\theta_0} : \mathcal{E} \otimes \mathcal{E} \to \mathbb{R}$ be the measure defined as $\nu_{\theta_0}(\mathrm{d}x, \mathrm{d}x') = Q_{\theta_0}(x, \mathrm{d}x')\, P(\mathrm{d}x)$. Assume*

*i. There exists a neighbourhood $U_{\theta_0} \subseteq \Theta$ of $\theta_0$ such that $Q_\theta \ll Q_{\theta_0}$ for all $\theta \in U_{\theta_0}$. Let $L_\theta : U_{\theta_0} \times \mathsf{X}^2 \to \mathbb{R}$ be the map defined as*

$$L_\theta(x, x') \coloneqq \frac{Q_\theta(x, \mathrm{d}x')}{Q_{\theta_0}(x, \mathrm{d}x')} \ . \tag{3.5}$$

*ii. For $\nu_{\theta_0}$-almost every $(x, x')$, the map $\theta \mapsto \alpha_\theta(x, x')\, L_\theta(x, x')$ is differentiable for all $\theta \in U_{\theta_0}$.*

*iii. There exists $g \in \mathcal{L}^1(\mathsf{X} \times \mathsf{X},\, \mathcal{E} \otimes \mathcal{E},\, \nu_{\theta_0})$ such that*

$$\sup_{\theta \in U_{\theta_0}} \left\| \nabla_\theta \big( \alpha_\theta(x, x')\, L_\theta(x, x') \big) \right\| \leq g(x, x') \quad \nu_{\theta_0}\text{-almost everywhere.}$$

*Then, $J$ is differentiable in $U_{\theta_0}$, and its gradient is*

$$\begin{aligned} \nabla_\theta J(\theta) \ &= \ \mathbb{E}_{\substack{x \sim P \\ x' \sim Q_{\theta_0}}} \Big[ \big( r(x, x') - r(x, x) \big) \alpha_{\theta_0}(x, x') \times \\ &\times \big( H(h_{\theta_0}(x, x') - 1)\, \nabla_\theta L_\theta(x, x') + H(1 - h_{\theta_0}(x, x'))\, \nabla_\theta L_\theta(x', x) \big) \Big] \ , \end{aligned} \tag{3.6}$$

*where $h$ is the Hastings ratio defined in Eq.* (1.20) *and $H$ is the Heaviside function.*

*Proof.* By Lemma B.2.1, for all $\theta \in U_{\theta_0}$,

$$J(\theta) \ = \ \mathbb{E}_{\substack{x \sim P \\ x' \sim Q_{\theta_0}}} \Big[ \big( r(x, x') - r(x, x) \big) \alpha_\theta(x, x')\, L_\theta(x, x') \Big] \ .$$

By assumptions *ii.-iii.* and Corollary A.1.4, we can exchange the derivative and the integral to obtain

$$\nabla_\theta J(\theta) \ = \ \mathbb{E}_{\substack{x \sim P \\ x' \sim Q_{\theta_0}}} \Big[ \big( r(x, x') - r(x, x) \big) \nabla_\theta \big( \alpha_\theta(x, x')\, L_\theta(x, x') \big) \Big] \ .$$

On the symmetric set $R \subseteq \mathsf{X}^2$ where $P(\mathrm{d}x')\, Q_\theta(x', \mathrm{d}x) \sim P(\mathrm{d}x)\, Q_\theta(x, \mathrm{d}x')$, the Hastings ratio (1.20) can be rewritten as

$$\begin{aligned} h_\theta(x, x') &= \frac{P(\mathrm{d}x')\, Q_\theta(x', \mathrm{d}x)}{P(\mathrm{d}x)\, Q_\theta(x, \mathrm{d}x')} \\ &= \frac{P(\mathrm{d}x')\, \frac{Q_\theta(x', \mathrm{d}x)}{Q_{\theta_0}(x', \mathrm{d}x)} Q_{\theta_0}(x', \mathrm{d}x)}{P(\mathrm{d}x)\, \frac{Q_\theta(x, \mathrm{d}x')}{Q_{\theta_0}(x, \mathrm{d}x')} Q_{\theta_0}(x, \mathrm{d}x')} \\ &= \frac{P(\mathrm{d}x')\, L_\theta(x', x)\, Q_{\theta_0}(x', \mathrm{d}x)}{P(\mathrm{d}x)\, L_\theta(x, x')\, Q_{\theta_0}(x, \mathrm{d}x')} \\ &= \frac{L_\theta(x', x)}{L_\theta(x, x')}\, h_{\theta_0}(x, x') \ . \end{aligned}$$

Since $h_\theta(x, x') = 0$ for all $(x, x') \in R^c$, the acceptance probability $\alpha_\theta$ can be expressed as

$$\begin{aligned}\alpha_\theta(x, x') &= \min\{1,\ h_\theta(x, x')\}\\ &= \begin{cases}1, & h_\theta(x, x') \geq 1\\ h_\theta(x, x'), & h_\theta(x, x') < 1\end{cases}\\ &= \begin{cases}1, & h_\theta(x, x') \geq 1\\ \dfrac{L_\theta(x', x)}{L_\theta(x, x')} h_{\theta_0}(x, x'), & h_\theta(x, x') < 1\end{cases}.\end{aligned}$$

Then,

$$\begin{aligned}\nabla_\theta\big(\alpha_\theta(x, x') L_\theta(x, x')\big) &= \begin{cases}\nabla_\theta L_\theta(x, x'), & h_\theta(x, x') \geq 1\\ h_{\theta_0}(x, x') \nabla_\theta L_\theta(x, x'), & h_\theta(x, x') < 1\end{cases}\\ &= \begin{cases}\alpha_{\theta_0}(x, x') \nabla_\theta L_\theta(x, x'), & h_\theta(x, x') \geq 1\\ \alpha_{\theta_0}(x, x') \nabla_\theta L_\theta(x, x'), & h_\theta(x, x') < 1\end{cases}\\ &= \alpha_{\theta_0}(x, x')\big(H(h_{\theta_0}(x, x') - 1)\, \nabla_\theta L_\theta(x, x') +\\ &\quad + H(1 - h_{\theta_0}(x, x'))\, \nabla_\theta L_\theta(x', x)\big)\ .\end{aligned}$$

Inserting this expression into $\nabla_\theta J$ gives Eq. (3.6). ☐

**Corollary B.2.2.** *In the settings of Theorem 3.1.1, if $Q_\theta$ admits a density $q_\theta$ with respect to a common dominating measure $\mu$ for all $\theta \in U_{\theta_0}$, let*

$$s_\theta(x, x') := \nabla_\theta \log q_\theta(x'|x)\ .$$

*Then, the gradient of $J$ at $\theta = \theta_0$ can be expressed as*

$$\begin{aligned}\nabla_\theta J(\theta_0) \ &= \ \mathbb{E}_{\substack{x\sim P\\ x'\sim Q_{\theta_0}}} \Big[\big(r(x, x') - r(x, x)\big)\alpha_{\theta_0}(x, x') \times\\ &\quad \times\big(H(h_{\theta_0}(x, x') - 1)\, s_{\theta_0}(x, x') + H(1 - h_{\theta_0}(x, x'))\, s_{\theta_0}(x', x)\big)\Big]\ . \quad \text{(B.4)}\end{aligned}$$

*Proof.* Since $Q_\theta$ admits a density $q_\theta$, for all $\theta \in U_{\theta_0}$,

$$L_\theta(x, x') = \frac{q_\theta(x'|x)}{q_{\theta_0}(x'|x)}\ .$$

Thus,

$$\nabla_\theta L_\theta(x, x') = \frac{\nabla_\theta q_\theta(x'|x)}{q_{\theta_0}(x'|x)}\ ,$$

and evaluating at $\theta = \theta_0$ gives

$$\begin{aligned} \nabla_\theta L_{\theta_0}\left(x, x'\right) &= \frac{\nabla_\theta q_{\theta_0}\left(x'|x\right)}{q_{\theta_0}\left(x'|x\right)} \\ &= s_{\theta_0}\left(x, x'\right) \ . \end{aligned}$$

Similarly,

$$\nabla_\theta L_{\theta_0}\left(x', x\right) = s_{\theta_0}\left(x', x\right) \ .$$

Inserting these expressions into Eq. (3.6) for $\theta = \theta_0$ gives Eq. (B.4). ☐

## B.3 Optimisation methods

**Lemma B.3.1.** *Assume the map* $\theta \mapsto L_\theta \in C^n\left(U_{\theta_0}\right)$*,* $\nu_0$*-almost everywhere, and that there exists* $g_n \in \mathcal{L}^1\left(\mathsf{X} \times \mathsf{X},\, \mathcal{E} \otimes \mathcal{E},\, \nu_{\theta_0}\right)$ *such that*

$$\sup_{\theta \in U_{\theta_0}} \left\|\partial_\theta^n L_\theta\left(x, x'\right)\right\| \leq g_n\left(x, x'\right) \quad \nu_{\theta_0}\text{-almost everywhere.}$$

*Then,*

$$\mathbb{E}_{\substack{x \sim P \\ x' \sim Q_{\theta_0}}}\left[\partial_\theta^n L_\theta\left(x, x'\right)\right] \ = \ 0 \ . \tag{B.5}$$

*Proof.* By corollary A.1.4, we can exchange the derivative and the integral to obtain

$$\begin{aligned} \mathbb{E}_{\substack{x \sim P \\ x' \sim Q_{\theta_0}}}\left[\partial_\theta^n L_\theta\left(x, x'\right)\right] &= \int_{\mathsf{X}^2} \nabla_\theta L_\theta\left(x, x'\right) Q_{\theta_0}\left(x, \mathrm{d}x'\right) P\left(\mathrm{d}x\right) \\ &= \partial_\theta^n \int_{\mathsf{X}^2} L_\theta\left(x, x'\right) Q_{\theta_0}\left(x, \mathrm{d}x'\right) P\left(\mathrm{d}x\right) \\ &= \partial_\theta^n \int_{\mathsf{X}^2} Q_\theta\left(x, \mathrm{d}x'\right) P\left(\mathrm{d}x\right) \\ &= \partial_\theta^n 1 \\ &= 0 \ . \end{aligned}$$

☐

**Proposition B.3.2.** *Assume* $L_\theta$ *satisfies the assumptions of Lemma B.3.1 with* $n = 1$*. Let*

$$b_\theta\left(x, x'\right) \ := \ J\left(\theta\right) \ s_\theta\left(x, x'\right) \ . \tag{B.6}$$

*Then,*

$$\mathbb{E}_{\substack{x \sim P \\ x' \sim Q_{\theta_0}}}\left[b_{\theta_0}\left(x, x'\right)\right] \ = \ 0 \ . \tag{B.7}$$

*Proof.* Since $s_{\theta_0}(x,x') = \nabla_\theta L_{\theta_0}(x,x')$ (see the proof of Corollary B.2.2), by Lemma B.3.1 with $n=1$,

$$\begin{aligned}\mathbb{E}_{\substack{x\sim P\\ x'\sim Q_{\theta_0}}}\left[b_{\theta_0}(x,x')\right] &= J(\theta_0)\,\mathbb{E}_{\substack{x\sim P\\ x'\sim Q_{\theta_0}}}\left[s_{\theta_0}(x,x')\right]\\ &= J(\theta_0)\,\mathbb{E}_{\substack{x\sim P\\ x'\sim Q_{\theta_0}}}\left[\nabla_\theta L_{\theta_0}(x,x')\right]\\ &= 0\ .\end{aligned}$$

□

**Proposition B.3.3.** *Let*

$$D(\theta) \ \coloneqq\ \mathbb{E}_{x\sim P}\left[\mathcal{D}_{KL}\left(Q_{\theta_0}\|Q_\theta\right)\right] \tag{B.8}$$

*be the average KL divergence between the policy $Q_{\theta_0}$ and $Q_\theta$. Assume*

*i. There exists a neighbourhood $U_{\theta_0}\subseteq\Theta$ of $\theta_0$ such that $Q_\theta\sim Q_{\theta_0}$ for all $\theta\in U_{\theta_0}$.*

*ii. $L_\theta$ satisfies the assumptions of Lemma B.3.1 with $n=2$.*

*Then*

$$D(\theta) \ =\ \frac{1}{2}(\theta-\theta_0)^T F(\theta_0)(\theta-\theta_0) + o\left(\|\theta-\theta_0\|^2\right)\ , \tag{B.9}$$

*where*

$$F(\theta_0) \ =\ \mathbb{E}_{\substack{x\sim P\\ x'\sim Q_{\theta_0}}}\left[s_{\theta_0}(x,x')\,s_{\theta_0}(x,x')^T\right]\ . \tag{B.10}$$

*Proof.* By assumption *i.*, for all $\theta\in U_{\theta_0}$,

$$\begin{aligned}D(\theta) &= \mathbb{E}_{\substack{x\sim P\\ x'\sim Q_{\theta_0}}}\left[\log\frac{Q_{\theta_0}(x,\mathrm{d}x')}{Q_\theta(x,\mathrm{d}x')}\right]\\ &= \mathbb{E}_{\substack{x\sim P\\ x'\sim Q_{\theta_0}}}\left[-\log L_\theta(x,x')\right]\ .\end{aligned}$$

Since $L_{\theta_0}(x,x')=1$ for $\nu_{\theta_0}$-almost every $(x,x')$, then $D(\theta_0)=0$. By assumptions *ii.* and Corollary A.1.4, we can exchange the derivative and the integral to obtain

$$\begin{aligned}\nabla_\theta D(\theta) &= -\mathbb{E}_{\substack{x\sim P\\ x'\sim Q_{\theta_0}}}\left[\nabla_\theta\log L_\theta(x,x')\right]\\ &= -\mathbb{E}_{\substack{x\sim P\\ x'\sim Q_{\theta_0}}}\left[\frac{\nabla_\theta L_\theta(x,x')}{L_\theta(x,x')}\right]\ .\end{aligned}$$

By Lemma B.3.1 with $n=1$, $\nabla_\theta D(\theta_0)=0$. By assumptions *ii.* and Corollary A.1.4,

$$\begin{aligned}\partial^2_{\theta_i\theta_j}D(\theta) &= -\mathbb{E}_{\substack{x\sim P\\ x'\sim Q_{\theta_0}}}\left[\partial^2_{\theta_i\theta_j}\log L_\theta(x,x')\right]\\ &= -\mathbb{E}_{\substack{x\sim P\\ x'\sim Q_{\theta_0}}}\left[\frac{\partial^2_{\theta_i\theta_j}L_\theta(x,x')}{L_\theta(x,x')} - \frac{\partial_{\theta_i}L_\theta(x,x')\,\partial_{\theta_j}L_\theta(x,x')}{L^2_\theta(x,x')}\right]\ .\end{aligned}$$

By Lemma B.3.1 with $n = 2$, the first term vanishes at $\theta = \theta_0$, and evaluating at $\theta = \theta_0$ gives

$$\begin{aligned}
\partial^2_{\theta_i \theta_j} D\left(\theta_0\right) &= \mathbb{E}_{\substack{x \sim P \\ x' \sim Q_{\theta_0}}} \left[\partial_{\theta_i} L_{\theta_0}\left(x, x'\right) \partial_{\theta_j} L_{\theta_0}\left(x, x'\right)\right] \\
&= \mathbb{E}_{\substack{x \sim P \\ x' \sim Q_{\theta_0}}} \left[s_{\theta_0, i}\left(x, x'\right) s_{\theta_0, j}\left(x, x'\right)\right] \\
&= F_{ij}\left(\theta_0\right) \ .
\end{aligned}$$

Inserting $D\left(\theta_0\right)$, $\nabla_\theta D\left(\theta_0\right)$, and $\partial^2_{\theta_i \theta_j} D\left(\theta_0\right)$ into the Taylor expansion of $D\left(\theta\right)$ around $\theta_0$ gives (B.9). ☐

# Appendix C

# Supplementary material for Chapter 4

## C.1 Equivariance of the optimal velocity field

**Lemma C.1.1.** *Let $G$ be an isometry group acting on $\mathsf{X}$. For any $g \in G$, any $y \in \mathsf{X}$ and any test function $\varphi \in C^\infty(\mathsf{X})$,*

$$\int_{\mathsf{X}} \varphi(x)\, \delta_{g(y)}\left(g(x)\right) \mathrm{d}x \;=\; \int_{\mathsf{X}} \varphi(x)\, \delta_y(x)\, \mathrm{d}x\ . \tag{C.1}$$

*Proof.* Make the change of variable $u = g(x)$, since $g$ is a group element,

$$x = g^{-1}(u), \quad \mathrm{d}x = \left|\det\left(\mathrm{d}g_u^{-1}\right)\right| \mathrm{d}u\ .$$

Since $g$ is an isometry, the differential $\mathrm{d}g_u$ is orthogonal for every $u \in \mathsf{X}$. Consequently $\mathrm{d}g_u^{-1}$ is also orthogonal and $|\det(\mathrm{d}g_u^{-1})| = 1$ for every $u \in \mathsf{X}$. Thus,

$$\begin{aligned}\int_{\mathsf{X}} \varphi(x)\, \delta_{g(y)}\left(g(x)\right) \mathrm{d}x &= \int_{\mathsf{X}} \varphi\left(g^{-1}(u)\right)\, \delta_{g(y)}\left(u\right) \mathrm{d}u \\ &= \varphi\left(g^{-1}\left(g(y)\right)\right) \\ &= \varphi(y) \\ &= \int_{\mathsf{X}} \varphi(x)\, \delta_y\left(x\right) \mathrm{d}x. \qquad \square\end{aligned}$$

**Proposition C.1.2.** *Let $G$ be an isometry group of $\mathsf{X}$, and let $x(t)$ be the stochastic interpolant defined by the interpolation path $I : [0,1] \times \mathsf{X} \times \mathsf{X} \to \mathsf{X}$ and the coupling $\nu$ between $Q_0$ and $Q_1$. If $\nu$ is $G$-invariant and $I$ is a $G$-equivariant, then the marginal densities $q_t = q\left(t, \cdot\right)$ of $x(t)$ are $G$-invariant for all $t \in [0,1]$.*

*Proof.* The marginal density $q_t$ can be expressed as

$$q_t(x) = \int_{\mathsf{X}^2} \delta_{I(t,x_0,x_1)}(x)\, \nu\left(\mathrm{d}x_0, \mathrm{d}x_1\right) \ .$$

Let $g \in G$,

$$q_t\left(g(x)\right) = \int_{\mathsf{X}^2} \delta_{I(t,x_0,x_1)}\left(g(x)\right)\, \nu\left(\mathrm{d}x_0, \mathrm{d}x_1\right) \ .$$

Make the change of variable $x_0 = g\left(u_0\right)$, $x_1 = g\left(u_1\right)$. Using Eq. (A.8), the $G$-invariance of $\nu$ and the $G$-equivariance of $I$,

$$\begin{aligned} q_t\left(g(x)\right) &= \int_{\mathsf{X}^2} \delta_{I(t,g(u_0),g(u_1))}\left(g(x)\right)\, \left(g \times g\right)_{\#}\nu\left(\mathrm{d}u_0, \mathrm{d}u_1\right) \\ &= \int_{\mathsf{X}^2} \delta_{I(t,g(u_0),g(u_1))}\left(g(x)\right)\, \nu\left(\mathrm{d}u_0, \mathrm{d}u_1\right) \\ &= \int_{\mathsf{X}^2} \delta_{g(I(t,u_0,u_1))}\left(g(x)\right)\, \nu\left(\mathrm{d}u_0, \mathrm{d}u_1\right) \ . \end{aligned}$$

Finally, using Lemma C.1.1

$$\begin{aligned} q_t\left(g(x)\right) &= \int_{\mathsf{X}^2} \delta_{I(t,u_0,u_1)}\left(x\right)\, \nu\left(\mathrm{d}u_0, \mathrm{d}u_1\right) \\ &= q_t(x) \ . \end{aligned}$$

☐

**Lemma C.1.3.** *Let $I : [0,1] \times \mathsf{X} \times \mathsf{X} \to \mathsf{X}$ be a time-differentiable interpolation path. If $I$ is $G$-equivariant, then for any $g \in G$, any $t \in [0,1]$ and any $x_0, x_1 \in \mathsf{X}$,*

$$\frac{\partial I}{\partial t}\left(t, g\left(x_0\right), g\left(x_1\right)\right) = \mathrm{d}g_{I(t,x_0,x_1)}\left(\frac{\partial I}{\partial t}\left(t, x_0, x_1\right)\right) \ . \tag{C.2}$$

*Proof.* Let $g \in G$, $t \in [0,1]$ and $x_0, x_1 \in \mathsf{X}$. Using the equivariance of $I$ and the chain rule,

$$\begin{aligned} \frac{\partial I}{\partial t}\left(t, g\left(x_0\right), g\left(x_1\right)\right) &= \frac{\partial}{\partial t} g\left(I\left(t, x_0, x_1\right)\right) \\ &= \mathrm{d}g_{I(t,x_0,x_1)}\left(\frac{\partial I}{\partial t}\left(t, x_0, x_1\right)\right) \ . \end{aligned}$$

☐

**Theorem 4.1.1.** *Let $G$ be an isometry group of $\mathsf{X}$, and let $x(t)$ be the stochastic interpolant defined by the interpolation path $I : [0,1] \times \mathsf{X} \times \mathsf{X} \to \mathsf{X}$ and the coupling $\nu$ between $Q_0$ and $Q_1$. If $\nu$ is $G$-invariant and $I$ is a $G$-equivariant, then the optimal velocity field $v^\star$ defined in Eq. (1.35) is $G$-equivariant for all $t \in [0,1]$.*

*Proof.* The optimal velocity field $v^\star$ can be expressed as

$$\begin{aligned} v^\star(t,x) &= \mathbb{E}_{(X_0,X_1)\sim\nu}\left[\frac{\partial I}{\partial t}(t,X_0,X_1) \mid X(t)=x\right] \\ &= \frac{1}{q_t(x)}\int_{\mathsf{X}^2}\frac{\partial I}{\partial t}(t,x_0,x_1)\;\delta_{I(t,x_0,x_1)}(x)\,\nu(\mathrm{d}x_0,\mathrm{d}x_1)\ . \end{aligned}$$

Since $q_t$ is $G$-invariant (Proposition C.1.2), we need to show that the integral

$$\mathcal{I}(x) = \int_{\mathsf{X}^2}\frac{\partial I}{\partial t}(t,x_0,x_1)\;\delta_{I(t,x_0,x_1)}(x)\,\nu(\mathrm{d}x_0,\mathrm{d}x_1)$$

is a $G$-equivariant vector field. Let $g\in G$,

$$\mathcal{I}(g(x)) = \int_{\mathsf{X}^2}\frac{\partial I}{\partial t}(t,x_0,x_1)\;\delta_{I(t,x_0,x_1)}(g(x))\;\nu(\mathrm{d}x_0,\mathrm{d}x_1)\ .$$

Make the change of variable $x_0 = g(u_0)$, $x_1 = g(u_1)$. Using Eq. (A.8), the $G$-invariance of $\nu$, the $G$-equivariance of $I$, Lemma C.1.1, Lemma C.1.3, the definition of the delta function and the linearity of the differential operator,

$$\begin{aligned} \mathcal{I}(g(x)) &= \int_{\mathsf{X}^2}\frac{\partial I}{\partial t}(t,g(u_0),g(u_1))\;\delta_{I(t,g(u_0),g(u_1))}(g(x))\;(g\times g)_{\#}\nu(\mathrm{d}u_0,\mathrm{d}u_1) \\ &= \int_{\mathsf{X}^2}\frac{\partial I}{\partial t}(t,g(u_0),g(u_1))\;\delta_{I(t,g(u_0),g(u_1))}(g(x))\;\nu(\mathrm{d}u_0,\mathrm{d}u_1) \\ &= \int_{\mathsf{X}^2}\frac{\partial I}{\partial t}(t,g(u_0),g(u_1))\;\delta_{g(I(t,u_0,u_1))}(g(x))\;\nu(\mathrm{d}u_0,\mathrm{d}u_1) \\ &= \int_{\mathsf{X}^2}\frac{\partial I}{\partial t}(t,g(u_0),g(u_1))\;\delta_{I(t,u_0,u_1)}(x)\;\nu(\mathrm{d}u_0,\mathrm{d}u_1) \\ &= \int_{\mathsf{X}^2}\mathrm{d}g_{I(t,u_0,u_1)}\left(\frac{\partial I}{\partial t}(t,u_0,u_1)\right)\delta_{I(t,u_0,u_1)}(x)\;\nu(\mathrm{d}u_0,\mathrm{d}u_1) \\ &= \int_{\mathsf{X}^2}\mathrm{d}g_x\left(\frac{\partial I}{\partial t}(t,u_0,u_1)\right)\delta_{I(t,u_0,u_1)}(x)\;\nu(\mathrm{d}u_0,\mathrm{d}u_1) \\ &= \mathrm{d}g_x\left(\int_{\mathsf{X}^2}\frac{\partial I}{\partial t}(t,u_0,u_1)\;\delta_{I(t,u_0,u_1)}(x)\;\nu(\mathrm{d}u_0,\mathrm{d}u_1)\right) \\ &= \mathrm{d}g_x(\mathcal{I}(x))\ . \end{aligned}$$

□

## C.2 Invariances of the target distribution

In this section we show that set of symmetries introduced in Section 2.2.4 form a *group*. We then show that both distributions in Eq. (2.3) and Eq. (2.5) are invariant under the action of such group.

**Definition C.2.1.** Let $\mathsf{X} = \mathcal{S}^N \times \mathcal{M}_L^N$ be the state space of a glass forming system with $N$ particles in a box of side $L$ with periodic boundary conditions. We define the *isometry group of* $\mathsf{X}$ as the set $G_{\mathsf{X}}$ of transformations $g_\sigma, g_u, g_M : \mathsf{X} \to \mathsf{X}$ defined by

$$g_\sigma(\boldsymbol{s}, \boldsymbol{r}) = \begin{pmatrix} \left(s_{\sigma(1)}, \ldots, s_{\sigma(N)}\right) \\ \left(\vec{r}_{\sigma(1)}, \ldots, \vec{r}_{\sigma(N)}\right) \end{pmatrix} \tag{C.3}$$

$$g_{\vec{u}}(\boldsymbol{s}, \boldsymbol{r}) = \begin{pmatrix} \boldsymbol{s} \\ (\boldsymbol{r} + \mathbf{1}_N \otimes \vec{u}) \,\%\, L \end{pmatrix} \tag{C.4}$$

$$g_M(\boldsymbol{s}, \boldsymbol{r}) = \begin{pmatrix} \boldsymbol{s} \\ \big((I_N \otimes M)\, \boldsymbol{r}\big) \,\%\, L \end{pmatrix} , \tag{C.5}$$

where $\sigma \in S_N$ is a permutation of $N$ elements, $\vec{u} \in \mathbb{R}^d$ is a translation vector, $M \in B(d)$ is a signed permutation hyperoctahedral group, and $\%\, L$ denotes the modulo operation with respect to $L$ applied component-wise.

**Lemma C.2.1.** *Let* $\sigma \in S_N$*, with corresponding permutation matrix* $A_\sigma \in \mathbb{Z}^{N \times N}$*,* $\vec{u} \in \mathbb{R}^d$*, and* $M \in B(d)$*. Then any transformation* $g \in G_{\mathsf{X}}$ *can be decomposed as* $g(x) = (g^s(x),\, g^r(x))$*, where* $g^s : \mathsf{X} \to \mathcal{S}^N$ *and* $g^r : \mathsf{X} \to \mathcal{M}_L^N$ *are defined by*

$$\begin{aligned} g^s(\boldsymbol{s}, \boldsymbol{r}) &\coloneqq A_\sigma \boldsymbol{s} \\ g^r(\boldsymbol{s}, \boldsymbol{r}) &\coloneqq (A_\sigma \otimes I_d)\, w_{M,\vec{u},L}(\boldsymbol{r}) \ , \end{aligned}$$

*and*

$$w_{M,\vec{u},L}(\boldsymbol{r}) \coloneqq \Big(\big((I_N \otimes M)\, \boldsymbol{r} + \mathbf{1}_N \otimes \vec{u}\big) \,\%\, L\Big) . \tag{C.6}$$

*Proof.* The result directly comes from combining Eqs. (C.3), (C.4) and (C.5). □

**Lemma C.2.2.** *Let* $\sigma \in S_N$*, with corresponding permutation matrix* $A_\sigma \in \mathbb{Z}^{N \times N}$*,* $\vec{u} \in \mathbb{R}^d$*, and* $M \in B(d)$*. Then any transformation* $g \in G_{\mathsf{X}}$ *can be decomposed as* $g = f_L \circ h_{A,b}$*, where* $f_L : \mathsf{X} \to \mathsf{X}$ *and* $h_{A,b} : \mathsf{X} \to \mathsf{X}$ *are defined as*

$$f_L(\boldsymbol{s}, \boldsymbol{r}) = \begin{pmatrix} \boldsymbol{s} \\ \boldsymbol{r} \,\%\, L \end{pmatrix} \tag{C.7}$$

$$h_{A,b}(\boldsymbol{s}, \boldsymbol{r}) = A \begin{pmatrix} \boldsymbol{s} \\ \boldsymbol{r} \end{pmatrix} + b , \tag{C.8}$$

*with*

$$A = \begin{pmatrix} A_\sigma & 0 \\ 0 & (A_\sigma \otimes I_d)\,(I_N \otimes M) \end{pmatrix} \in \mathbb{Z}^{N(d+1)\times N(d+1)} \tag{C.9}$$

$$b = \begin{pmatrix} 0 \\ \mathbf{1}_N \otimes \vec{u} \end{pmatrix} \in \mathbb{R}^{N(d+1)} \; . \tag{C.10}$$

*Proof.* From Lemma C.2.1,

$$\begin{aligned} g^s\left(\boldsymbol{s}, \boldsymbol{r}\right) &= A_\sigma \boldsymbol{s} \\ g^r\left(\boldsymbol{s}, \boldsymbol{r}\right) &= (A_\sigma \otimes I_d)\left(\big((I_N \otimes M)\,\boldsymbol{r} + \mathbf{1}_N \otimes \vec{u}\big)\,\%\,L\right) . \end{aligned}$$

Then, since particle permutations commute with the modulo operation,

$$g^r\left(\boldsymbol{s}, \boldsymbol{r}\right) = \big((A_\sigma \otimes I_d)\,(I_N \otimes M)\,\boldsymbol{r} + \mathbf{1}_N \otimes \vec{u}\big)\,\%\,L \; .$$

Combining these two expressions with Eqs. (C.9) and (C.10), gives the desired result. ☐

**Lemma C.2.3.** *Let $\boldsymbol{r}_1, \boldsymbol{r}_2 \in \mathbb{R}^{Nd}$, then*

$$\left((\boldsymbol{r}_1\ \%\ L) + \boldsymbol{r}_2\right)\ \%\ L = \left(\boldsymbol{r}_1 + \boldsymbol{r}_2\right)\ \%\ L \; . \tag{C.11}$$

*Proof.* Since $\boldsymbol{r}_1 = (\boldsymbol{r}_1\ \%\ L) + \boldsymbol{k}_1 L$, where $\boldsymbol{k}_1 \in \mathbb{Z}^{Nd}$, then

$$\begin{aligned} \left((\boldsymbol{r}_1\ \%\ L) + \boldsymbol{r}_2\right)\ \%\ L &= \left(\boldsymbol{r}_1 + \boldsymbol{r}_2 - \boldsymbol{k}_1\right)\ \%\ L \\ &= \left(\boldsymbol{r}_1 + \boldsymbol{r}_2\right)\ \%\ L \; . \end{aligned}$$ ☐

**Lemma C.2.4.** *Let $y \in \mathbb{R}$ and $\epsilon \in \{-1, +1\}$, then $(\epsilon y)\ \%\ L = (\epsilon\,(y\ \%\ L))\ \%\ L$.*

*Proof.* If $\epsilon = 1$ this is obviously true. If $\epsilon = -1$, we are trying to prove that

$$(-y)\ \%\ L = (-\,(y\ \%\ L))\ \%\ L \; .$$

By definition, we have

$$\begin{aligned} (-y)\ \%\ L &= -y - \left\lfloor \frac{-y}{L} \right\rfloor L \; , \\ (-\,(y\ \%\ L))\ \%\ L &= -y - \left(\left\lfloor \frac{-y}{L} + k \right\rfloor - k\right) L \; , \end{aligned}$$

where $k = \lfloor y/L \rfloor$. Using the fact that for any $x \in \mathbb{R}$ and $n \in \mathbb{N}$,

$$\lfloor x + n \rfloor - n = \lfloor x \rfloor \; ,$$

we get the intended result. ☐

**Lemma C.2.5.** *Let $M \in B(d)$ and $\vec{r} \in \mathbb{R}^d$, then $(M\vec{r})\ \%\ L = (M\,[\vec{r}\ \%\ L])\ \%\ L$.*

*Proof.* For any $M \in B(d)$, there exist $\epsilon \in \{-1{,}1\}^d$ and $\sigma \in \mathrm{S}_d$ such that for all $i, j \in [\![1, d]\!]$, $M_{i,j} = \epsilon(i)\delta_{\sigma(i),j}$. For all $i \in [\![1, d]\!]$, we have

$$((M\vec{r})\ \%\ L)_i = \left(\epsilon(i)\, \vec{r}_{\sigma(i)}\right)\ \%\ L, \quad (M\,[\vec{r}\ \%\ L])_i = \epsilon(i)\ [\vec{r}\ \%\ L]_{\sigma(i)}\ .$$

Use Lemma C.2.4 on each coordinate to conclude. ☐

**Lemma C.2.6.** *For all $a \in \mathbb{R}$, $-\lfloor \frac{a}{L} + \frac{1}{2} \rfloor = \operatorname{argmin}_{k\in\mathbb{Z}} |a + kL|$ holds.*

*Proof.* By definition,

$$\left\lfloor \frac{a}{L} \right\rfloor L \le a \le \left(\left\lfloor \frac{a}{L} \right\rfloor + 1\right) L\ ,$$

which means that

$$k^\star = \operatorname*{argmin}_{k\in\mathbb{Z}} |a + kL| \in \left\{-\left\lfloor \frac{a}{L} \right\rfloor, -\left(\left\lfloor \frac{a}{L} \right\rfloor + 1\right)\right\}\ .$$

<u>Case 1</u>: $k^\star = -\lfloor a/L \rfloor$ This means that $a/L$ is closer to $\lfloor a/L \rfloor$ than to $\lfloor a/L \rfloor + 1$ which implies that $a/L + 1/2$ is also closer to $\lfloor a/L \rfloor$ which means that

$$\left\lfloor \frac{a}{L} \right\rfloor = \left\lfloor \frac{a}{L} + \frac{1}{2} \right\rfloor\ ,$$

and that $k^\star = -\lfloor a/L + 1/2 \rfloor$.

<u>Case 2</u>: $k^\star = -(\lfloor a/L \rfloor + 1)$ This means that $a/L$ is closer to $\lfloor a/L \rfloor + 1$ than to $\lfloor a/L \rfloor$ which implies that $a/L + 1/2$ is also closer to $\lfloor a/L \rfloor + 1$ which means that

$$\left\lfloor \frac{a}{L} \right\rfloor + 1 = \left\lfloor \frac{a}{L} + \frac{1}{2} \right\rfloor\ ,$$

and that $k^\star = -\lfloor a/L + 1/2 \rfloor$. ☐

**Lemma C.2.7.** *Let $f : \mathbb{R}^k \to \mathbb{R}^+$ be defined for all $x \in \mathbb{R}^k$ as*

$$f(x) = \sum_{i=1}^{k} f_i(x_i), \quad f_i : \mathbb{R} \to \mathbb{R}^+\ .$$

*Then the minimum of $f$ decomposes as*

$$\operatorname*{argmin}_{x\in\mathbb{R}^k} f(x) = \operatorname*{argmin}_{x_1\in\mathbb{R}} f_1(x) \times \ldots \times \operatorname*{argmin}_{x_k\in\mathbb{R}} f_k(x)\ .$$

*Proof.* Suppose that there exist $\bar{x} \in \operatorname{argmin}_{x\in\mathbb{R}^k} f(x)$ and $\bar{x} \notin \operatorname{argmin}_{x_1\in\mathbb{R}} f_1(x) \times \ldots \times \operatorname{argmin}_{x_k\in\mathbb{R}} f_k(x)$.
Then there exists $i \in \{1,\ldots,k\}$ such that $\bar{x}_i \notin \operatorname{argmin}_{x_i\in\mathbb{R}} f_i(x)$. Let $\hat{x}_i \in \operatorname{argmin}_{x_i\in\mathbb{R}} f_i(x)$, then $\hat{x} \in \mathbb{R}^k$ defined as

$$\hat{x}_j = \begin{cases} \bar{x}_j & \text{if } j \neq i \\ \hat{x}_i & \text{otherwise} \end{cases} .$$

Then $f(\hat{x}) \leq f(\bar{x})$ which contradicts the assumptions. This gives the left-right inclusion, and the right-left one is trivial. ☐

**Lemma C.2.8.** *Let $\vec{r}_1, \vec{r}_2 \in \mathcal{M}_L$, the nearest image distance $d_{\mathcal{M}_L}$ can be written as*

$$d_{\mathcal{M}_L}(\vec{r}_1,\vec{r}_2) = \left\|\log_{\vec{r}_1}(\vec{r}_2)\right\|, \qquad \textit{for any } \vec{r}_1,\vec{r}_2 \in \mathcal{M}_L \ . \tag{C.12}$$

*Proof.* We can write the log map as $\log_{\vec{r}_1}(\vec{r}_2) = \vec{r}_2 - \vec{r}_1 + \vec{k}_{12}L$, where

$$\begin{aligned} \vec{k}_{12} &:= -\left\lfloor \frac{\vec{r}_2 - \vec{r}_1 + \frac{L}{2}}{L} \right\rfloor \\ &= -\left\lfloor \frac{\vec{r}_2 - \vec{r}_1}{L} + \frac{1}{2} \right\rfloor . \end{aligned}$$

Moreover,

$$\begin{aligned} d_{\mathcal{M}_L}(\vec{r}_1,\vec{r}_2) &= \min_{\vec{k}\in\mathbb{Z}^d} \left\|\vec{r}_1 - \vec{r}_2 + \vec{k}L\right\| \\ &= \min_{\vec{k}\in\mathbb{Z}^d} \left\|\vec{r}_2 - \vec{r}_1 + \vec{k}L\right\| \\ &= \min_{\vec{k}\in\mathbb{Z}^d} \left\|\vec{r}_2 - \vec{r}_1 + \vec{k}L\right\|^2 , \end{aligned}$$

which implies that

$$d_{\mathcal{M}_L}(\vec{r}_1,\vec{r}_2) = \min_{\vec{k}\in\mathbb{Z}^d} \sum_{i=1}^{d} \left(r_{2i} - r_{1i} + k_i L\right)^2 \ .$$

Using Lemma C.2.6 and the monotonicity of the square root on $\mathbb{R}^+$, we have that for all $i \in [\![1,d]\!]$,

$$\operatorname*{argmin}_{k\in\mathbb{Z}} \left(r_{2i} - r_{1i} + kL\right)^2 = k_{12i} \ .$$

By Lemma C.2.7, this implies that

$$\operatorname*{argmin}_{\vec{k}\in\mathbb{Z}^d} \sum_{i=1}^{d} \left(r_{2i} - r_{1i} + k_i L\right)^2 = \vec{k}_{12} \ ,$$

leading to

$$\begin{aligned} d_{\mathcal{M}_L}(\vec{r}_1, \vec{r}_2) &= \left\| \vec{r}_2 - \vec{r}_1 + \vec{k}_{12} L \right\| \\ &= \left\| \log_{\vec{r}_1}(\vec{r}_2) \right\| . \end{aligned} \qquad \square$$

**Lemma C.2.9.** *For any $\vec{r}_1, \vec{r}_2 \in \mathcal{M}_L$, the logarithmic map on the flat torus can be written*

$$\log_{\vec{r}_1}(\vec{r}_2) = \frac{L}{2\pi} \operatorname{atan2}\left( \sin\left( \frac{2\pi}{L} (\vec{r}_1 - \vec{r}_2) \right), \cos\left( \frac{2\pi}{L} (\vec{r}_1 - \vec{r}_2) \right) \right) . \tag{C.13}$$

*Proof.* For any $\vec{r}_1, \vec{r}_2 \in \mathcal{M}_L$, using

$$\arctan \tan \alpha = \alpha - \pi \left\lfloor \frac{\alpha}{\pi} + \frac{1}{2} \right\rfloor$$

and

$$\tan \alpha = \frac{\sin 2\alpha}{1 + \cos 2\alpha}$$

we get

$$\begin{aligned} \log_{\vec{r}_1}(\vec{r}_2) &= \left( \vec{r}_2 - \vec{r}_1 + \frac{L}{2} \right) \ \% \ L - \frac{L}{2} \\ &= \vec{r}_2 - \vec{r}_1 - L \left\lfloor \frac{\vec{r}_2 - \vec{r}_1 + \frac{L}{2}}{L} \right\rfloor \\ &= \frac{L}{\pi} \arctan\left( \tan\left( \frac{\pi}{L} (\vec{r}_2 - \vec{r}_1) \right) \right) \\ &= \frac{L}{\pi} \arctan\left( \frac{\sin\left( \frac{2\pi}{L} (\vec{r}_2 - \vec{r}_1) \right)}{1 + \cos\left( \frac{2\pi}{L} (\vec{r}_2 - \vec{r}_1) \right)} \right) \\ &= \frac{L}{2\pi} \operatorname{atan2}\left( \sin\left( \frac{2\pi}{L} (\vec{r}_1 - \vec{r}_2) \right), \cos\left( \frac{2\pi}{L} (\vec{r}_1 - \vec{r}_2) \right) \right) . \end{aligned} \qquad \square$$

**Lemma C.2.10.** *Let $\vec{r}_1, \vec{r}_2 \in \mathcal{M}_L$. For any $M \in B(d)$,*

$$\log_{(M\vec{r}_1) \ \% \ L}((M\vec{r}_2) \ \% \ L) = M \log_{\vec{r}_1}(\vec{r}_2) , \tag{C.14}$$

*holds.*

*Proof.* Let $x, y \in [0, L]$, using Lemma C.2.9

$$\log_{(-x) \ \% \ L}((-y) \ \% \ L) = \frac{L}{\pi} \arctan\left( \frac{\sin\left( \frac{2\pi}{L} ((-y) \ \% \ L - (-x) \ \% \ L) \right)}{1 + \cos\left( \frac{2\pi}{L} ((-y) \ \% \ L - (-x) \ \% \ L) \right)} \right) .$$

By definition, there exists $k, k' \in \mathbb{Z}$ such that

$$(-x)\ \%\ L = -x + kL, \quad \text{and } (-y)\ \%\ L = -y + k'L\ .$$

This leads to

$$\begin{aligned}
\frac{2\pi}{L}\left((-y)\ \%\ L - (-x)\ \%\ L\right) &= \frac{2\pi}{L}\left([-y + k'L] - [-x + kL]\right)\ , \\
&= -\frac{2\pi}{L}(y - x) + 2\pi \underbrace{(k - k')}_{\in \mathbb{Z}}\ .
\end{aligned}$$

Using the periodicity of the sine and cosine in the above formula, we get

$$\begin{aligned}
\log_{(-x)\ \%\ L}((-y)\ \%\ L) &= \frac{L}{\pi} \arctan\left(\frac{-\sin\left(\frac{2\pi}{L}(y - x)\right)}{1 + \cos\left(\frac{2\pi}{L}(y - x)\right)}\right) \\
&= -\log_x(y)\ .
\end{aligned}$$

Let $M \in B(d)$ and $\vec{r}_1, \vec{r}_2 \in \mathcal{M}_L$. By definition, for all $i \in [\![1, d]\!]$, $(M\vec{r}_1)_i = \epsilon_i (\vec{r}_1)_{\sigma(i)}$. Moreover, using the previous result,

$$\begin{aligned}
\left[\log_{(M\vec{r}_1)\ \%\ L}((M\vec{r}_2)\ \%\ L)\right]_i &= \log_{[(M\vec{r}_1)\ \%\ L]_i}([(M\vec{r}_2)\ \%\ L]_i) \\
&= \log_{((M\vec{r}_1)_i)\ \%\ L}(((M\vec{r}_2)_i)\ \%\ L) \\
&= \log_{\left(\epsilon_i(\vec{r}_1)_{\sigma(i)}\right)\ \%\ L}(\left(\epsilon_i(\vec{r}_2)_{\sigma(i)}\right)\ \%\ L) \\
&= \epsilon_i \log_{(\vec{r}_1)_{\sigma(i)}}((\vec{r}_2)_{\sigma(i)}) \\
&= [M \log_{\vec{r}_1}(\vec{r}_2)]_i\ . \qquad \square
\end{aligned}$$

**Lemma C.2.11.** *Let $\vec{r}_1, \vec{r}_2 \in \mathcal{M}_L$. For any $M \in B(d)$ and $\vec{u} \in \mathbb{R}^d$,*

$$\log_{(M\vec{r}_1 + \vec{u})\ \%\ L}\left((M\vec{r}_2 + \vec{u})\ \%\ L\right) = M \log_{\vec{r}_1}(\vec{r}_2)\ . \tag{C.15}$$

*Proof.* Applying Lemma C.2.3 twice, we get

$$\begin{aligned}
\log_{(\vec{r}_1 + \vec{u})\ \%\ L}\left((\vec{r}_2 + \vec{u})\ \%\ L\right) &= \left((\vec{r}_2 + \vec{u})\ \%\ L - (\vec{r}_2 + \vec{u})\ \%\ L + \frac{L}{2}\right)\ \%\ L - \frac{L}{2} \\
&= \left(\vec{r}_2 + \vec{u} - \vec{r}_2 - \vec{u} + \frac{L}{2}\right)\ \%\ L - \frac{L}{2} \\
&= \log_{\vec{r}_1\ \%\ L}\left(\vec{r}_2\ \%\ L\right)\ .
\end{aligned}$$

Using Lemma C.2.10 with $M\vec{r}_1 \in \mathcal{M}_L$ and $M\vec{r}_2 \in \mathcal{M}_L$ we can conclude. $\square$

**Lemma C.2.12.** *Let $\vec{r}, \vec{v} \in \mathbb{R}^d$, $M \in B(d)$ and $\vec{u} \in \mathbb{R}^d$ then*

$$\exp_{(M\vec{r}+\vec{u}) \ \% \ L}(M\vec{v}) = (M \exp_{\vec{r}}(\vec{v}) + \vec{u}) \ \% \ L \ . \tag{C.16}$$

*Proof.* From Lemma C.2.3,

$$\exp_{(M\vec{r}+\vec{u}) \ \% \ L}(M\vec{v}) = \left(\exp_{(M\vec{r}) \ \% \ L} M\vec{v} + \vec{u}\right) \ \% \ L \ .$$

Moreover, since the signed permutation matrix $M$ is characterisde by a permutation $\sigma$ and a vector $\vec{\epsilon} \in \{-1, 1\}^d$, for all $i \in [\![1, d]\!]$

$$\left[M\left(\vec{r} + \vec{v}\right)\right]_i = \epsilon_i \left(r_{\sigma(i)} + v_{\sigma(i)}\right) \ .$$

Using Lemma C.2.3 and Lemma C.2.4, we get

$$\begin{aligned} \left[\exp_{(M\vec{r}) \ \% \ L}(M\vec{v})\right]_i &= \left(\epsilon_i(r_{\sigma(i)} + v_{\sigma(i)})\right) \ \% \ L \\ &= \left(\epsilon_i\left[\left(r_{\sigma(i)} + v_{\sigma(i)}\right) \ \% \ L\right]\right) \ \% \ L \ . \end{aligned}$$

Additionally,

$$\left[(M \exp_{\vec{r}}(\vec{v})) \ \% \ L\right]_i = \left(\epsilon_i\left[\left(r_{\sigma(i)} + v_{\sigma(i)}\right) \ \% \ L\right]\right) \ \% \ L \ ,$$

which proves that $\exp_{(M\vec{r}) \ \% \ L}(M\vec{v}) = (M \exp_{\vec{r}}(\vec{v})) \ \% \ L$. Inserting this expression into the first equation and using Lemma C.2.3 we get

$$\begin{aligned} \exp_{(M\vec{r}+\vec{u}) \ \% \ L}(M\vec{v}) &= ((M \exp_{\vec{r}}(\vec{v})) \ \% \ L + \vec{u}) \ \% \ L \\ &= (M \exp_{\vec{r}}(\vec{v}) + \vec{u}) \ \% \ L \ . \end{aligned}$$

□

**Lemma C.2.13.** *Let $g \in G_{\mathsf{X}}$ be decomposed as in Lemma C.2.2, then $\mathrm{d}g_x = A$ for almost every $x \in \mathsf{X}$.*

*Proof.* Let $x \in \mathsf{X}$, by the chain rule

$$\mathrm{d}g_x = \mathrm{d}f_{L,h_{A,b}(x)} \, \mathrm{d}h_{A,b,x} \ .$$

The modulo has unit derivative (except from jump points, which have measure zero), thus $\mathrm{d}f_{L,h_{A,b}(x)} = I_{N(d+1)}$ almost everywhere. Since $h_{A,b}$ is affine, $\mathrm{d}h_{A,b,x} = A$ for all $x \in \mathsf{X}$, which concludes the proof. □

**Proposition C.2.14.** *The set $G_{\mathsf{X}}$ in Definition C.2.1 is a group under function composition.*

*Proof.* The identity is in $G_{\mathsf{X}}$ as it is a special case of the invariances. Let $g^1, g^2 \in G_{\mathsf{X}}$ we can decompose them as $g^1 = f_L \circ h_{A^1,b^1}$ and $g^2 = f_L \circ h_{A^2,b^2}$ where $A_1$ (respectively $A^2$) can be decomposed into $A^1_\sigma$, $M^1$ (respectively $A^2_\sigma$, $M^2$) and $b^1$ (respectively $b^2$) into $b^1 = \mathbf{1}_N \otimes c^1$ (respectively $b^2 = \mathbf{1}_N \otimes c^2$). Using both Lemmas C.2.5 and C.2.3, $g^1 \circ g^2$ can be written for any $x \in G_{\mathsf{X}}$ as

$$\begin{aligned}
&(g^1 \circ g^2)(x) = \\
&\quad = f_L \left[ \begin{pmatrix} A_{\sigma^1} A_{\sigma^2} & 0 \\ 0 & (A_{\sigma^1} A_{\sigma^2} \otimes I_d)\,(I_N \otimes M^1 M^2) \end{pmatrix} x + \big(\mathbf{1}_N \otimes (c^1 + M^1 c^2)\big) \right] ,
\end{aligned}$$

which shows, by C.2.2, that $g^1 \circ g^2 \in G_{\mathsf{X}}$. Finally, let $g \in G_{\mathsf{X}}$ be written under the same decomposition, using Lemmas C.2.5 and C.2.3 again, we have for any $x \in G_{\mathsf{X}}$

$$\begin{aligned}
g^{-1}(x) &= f_L \left[ \begin{pmatrix} A_{\sigma^{-1}} & 0 \\ 0 & (I_N \otimes M^{-1})(A_{\sigma^{-1}} \otimes I_d) \end{pmatrix} x - (\mathbf{1}_N \otimes c) \right] \\
&= f_L \left[ \begin{pmatrix} A_{\sigma^{-1}} & 0 \\ 0 & (A_{\sigma^{-1}} \otimes I_d)(I_N \otimes M^{-1}) \end{pmatrix} x + \big(\mathbf{1}_N \otimes (-c)\big) \right] .
\end{aligned}$$

which shows, by Lemma C.2.2, that $g^{-1} \in G_{\mathsf{X}}$. ☐

**Proposition C.2.15.** *The Boltzmann distribution in Eq.* (2.3) *and the Boltzmann distribution with fixed composition in Eq.* (2.5) *are invariant under the action of the group $G_{\mathsf{X}}$ defined in Definition C.2.1. Moreover, $G_{\mathsf{X}}$ is an isometry group of* $\mathsf{X}$.

*Proof.* The invariance with respect to permutations $g_\sigma$ trivially comes from the invariance of the sums. For translations $g_{\vec{u}}$ and box-symmetries $g_M$, we need to show that for all $x_1 = (\boldsymbol{s}_1, \boldsymbol{r}_1)\,, x_2 = (\boldsymbol{s}_2, \boldsymbol{r}_2) \in \mathsf{X}$, and for all $g = (g^{\boldsymbol{s}}, g^{\boldsymbol{r}}) \in G_{\mathsf{X}}$, the nearest image distance is invariant. Using Lemma C.2.8, the distance can be written as the norm of the logarithmic map, which is equivariant as per Lemma C.2.11. Using the fact that the scaling matrix is always orthogonal, we get the result. ☐

## C.3 Model's equivariance

**Proposition C.3.1.** *The interpolant in Eq. 4.1 is $G_{\mathsf{X}}$-equivariant.*

*Proof.* Let $g \in G_{\mathsf{X}}$, $t \in [0,1]$, $x_0, x_1 \in \mathsf{X}$ and write

$$I\,(t, x_0, x_1) = \begin{pmatrix} I^{\boldsymbol{s}}_L(t, x_0, x_1) \\ I^{\boldsymbol{r}}_L(t, x_0, x_1) \end{pmatrix} .$$

Then, using the decomposition of $g$ of Lemma C.2.1,

$$\begin{aligned} I^{\boldsymbol{s}}\left(t, g(x_0), g(x_1)\right) &= (1-t)A_\sigma \boldsymbol{s}_0 + tA_\sigma \boldsymbol{s}_1 \\ &= A_\sigma I^{\boldsymbol{s}}\left(t, x_0, x_1\right) \ . \end{aligned}$$

Moreover, for every $i \in [\![1, N]\!]$, using Lemma C.2.11 and Lemma C.2.12

$$\begin{aligned} \left[I^{\boldsymbol{r}}\left(t, g(x_0), g(x_1)\right)\right]_i &= \exp_{w_{M,\vec{u},L}\left(\vec{r}_{0\sigma(i)}\right)}\left(t \log_{w_{M,\vec{u},L}\left(\vec{r}_{0\sigma(i)}\right)}\left(w_{M,\vec{u},L}\left(\vec{r}_{1\sigma(i)}\right)\right)\right) \\ &= \exp_{w_{M,\vec{u},L}\left(\vec{r}_{0\sigma(i)}\right)}\left(tM \log_{\vec{r}_{0\sigma(i)}}\left(\vec{r}_{1\sigma_i}\right)\right) \\ &= \left(M \exp_{\vec{r}_{0\sigma_i}}\left(t \log_{\vec{r}_{0\sigma(i)}}\left(\vec{r}_{1\sigma(i)}\right)\right) + \vec{u}\right) \ \% \ L \\ &= \left[w_{M,\vec{u},L}\left(I^{\boldsymbol{r}}(t, x_0, x_1)\right)\right]_{\sigma(i)} \ . \end{aligned}$$

Using Lemma C.2.1 we can conclude. ☐

**Proposition C.3.2.** *Let $x(t)$ be the stochastic interpolant defined by the interpolation path in Eq.* (4.1) *and the coupling $\nu$ between $Q_0$ and $Q_1$. The optimal velocity field is*

$$v^*(t,x) = \mathbb{E}_{(x_0,x_1)\sim\nu}\left[\begin{array}{c} \boldsymbol{s}_0 - \boldsymbol{s}_1 \\ \left(\left(\boldsymbol{r}_1 - \boldsymbol{r}_0 + \frac{L}{2}\right) \% \, L - \frac{L}{2}\right) \% \, L \end{array} \middle| \, x(t) = I(t, x_0, x_1)\right] \ . \tag{C.17}$$

*Proof.* Let $t \in [0{,}1]$ and $x_0, x_1 \in G_{\mathsf{X}}$, write

$$I\left(t, x_0, x_1\right) = \begin{pmatrix} I^{\boldsymbol{s}}_L(t, x_0, x_1) \\ I^{\boldsymbol{r}}_L(t, x_0, x_1) \end{pmatrix} \ .$$

Then

$$\frac{\partial I^{\boldsymbol{s}}}{\partial t}\left(t, x_0, x_1\right) = \boldsymbol{s}_1 - \boldsymbol{s}_0 \ ,$$

and,

$$\frac{\partial I^{\boldsymbol{r}}}{\partial t}\left(t, x_0, x_1\right) = \left(\left(\boldsymbol{r}_1 - \boldsymbol{r}_0 + \frac{L}{2}\right) \% \, L - \frac{L}{2}\right) \% \, L \ .$$

Using the definition of $v^\star$ in Eq. (1.35) we can conclude. ☐

**Proposition C.3.3.** *The velocity field in Eq.* (4.7) *is $G_{\mathsf{X}}$-equivariant.*

*Proof.* Let $t \in [0{,}1]$ and $x \in \mathsf{X}$. Define $\psi_k(t,x) \coloneqq (\psi^{\boldsymbol{s}}_k(t,x), \psi^{\boldsymbol{r}}_k(t,x))$, $\Gamma_k(t,x) \coloneqq H^k$ and $\Lambda_k(t,x) \coloneqq M^k$. We start by showing by induction on $k \in \mathbb{N}$ that for all $g \in G_{\mathsf{X}}$

and $i, j \in [\![1, N]\!]$,

$$\begin{aligned} \psi_k(t, g(x)) &= g(\psi_k(t, x)) \ , \\ [\Gamma_k(t, g(x))]_i &= [\Gamma_k(t, x)]_{\sigma(i)} \ , \\ [\Lambda_k(t, g(x))]_{ij} &= [\Lambda_k(t, x)]_{\sigma(i)\sigma(j)} \ , \end{aligned}$$

Let $g(x) = (g^{\boldsymbol{s}}(x), g^{\boldsymbol{r}}(x))$ as in Lemma C.2.1. At $k = 0$, we have $\psi_0(t, g(x)) = g(x) = g(\psi_0(t, x))$ and for all $i \in [\![1, N]\!]$

$$\begin{aligned} [\Gamma_0(t, g(x))]_i &= (t, [g^{\boldsymbol{s}}(x)]_i) \\ &= \left(t, \boldsymbol{s}_{\sigma(i)}\right) \\ &= [\Gamma_0(t, x)]_{\sigma(i)} \ . \end{aligned}$$

Moreover, for all $i, j \in [\![1, N]\!]$

$$\begin{aligned} &[\Lambda_0(t, g(x))]_{ij} = \\ &\quad = \hat{\phi}_e\Big([\Gamma_0(t, g(x))]_i, [\Gamma_0(t, g(x))]_j, d_{\mathcal{M}_L}\Big([w_{M,\vec{u},L}(\boldsymbol{r})]_{\sigma(i)}, [w_{M,\vec{u},L}(\boldsymbol{r})]_{\sigma(j)}\Big)\Big) \ . \end{aligned}$$

Using the statement on $\Gamma_0$ and Proposition C.2.15 we get

$$\begin{aligned} [\Lambda_0(t, g(x))]_{ij} &= \hat{\phi}_e\Big([\Gamma_0(t, x)]_{\sigma(i)}, [\Gamma_0(t, x)]_{\sigma(j)}, d_{\mathcal{M}_L}\big(\vec{r}_{\sigma(i)}, \vec{r}_{\sigma(j)}\big)\Big) \\ &= [\Lambda_0(t, x)]_{\sigma(i)\sigma(j)} \ . \end{aligned}$$

Let $k \in \mathbb{N}$, assume the hypotesis holds at step $k$. Then, using the permutation invariance of the sum, for all $i \in [\![1, N]\!]$,

$$\begin{aligned} [\Gamma_{k+1}(t, g(x))]_i &= \hat{\phi}_h\left([\Gamma_k(t, g(x))]_i, \sum_{i \neq j} \hat{\phi}_m\left([\psi_k(t, g(x))]_{ij}\right) \ [\psi_k(t, g(x))]_{ij}\right) \\ &= \hat{\phi}_h\left([\Gamma_k(t, x)]_{\sigma(i)}, \sum_{i \neq j} \hat{\phi}_m\left([\psi_k(t, x)]_{\sigma(i)\sigma(j)}\right) \ [\psi_k(t, x)]_{\sigma(i)\sigma(j)}\right) \\ &= [\Gamma_{k+1}(t, x)]_{\sigma(i)} \ . \end{aligned}$$

Similarly,

$$[\Lambda_{k+1}(t, g(x))]_{ij} = [\Lambda_{k+1}(t, x)]_{\sigma(i)\sigma(j)} \ ,$$

and

$$\begin{aligned} \psi^{\boldsymbol{s}}_{k+1}(t, g(x)) &= g^{\boldsymbol{s}}(x) \\ &= g^{\boldsymbol{s}}\left(\psi^{\boldsymbol{s}}_k(t, x)\right) \ . \end{aligned}$$

For the position update:

$$\begin{aligned}
&\left[\psi^r_{k+1}(t,g(x))\right]_i = \\
&\qquad = \exp_{\left[\psi^r_k(t,g(x))\right]_i}\left(\sum_{i\neq j} \frac{\log_{\left[\psi^r_k(t,g(x))\right]_j}\left[\psi^r_k(t,g(x))\right]_i}{d_{\mathcal{M}_L}\left(\left[g^r(x)\right]_i,\left[g^r(x)\right]_j\right)+1}\phi_d\left(\left[\Lambda_k(t,x)\right]_{\sigma(i)\sigma(j)}\right)\right).
\end{aligned}$$

Using the induction hypothesis and Lemma C.2.11,

$$\begin{aligned}
\log_{\left[\psi^r_k(t,g(x))\right]_j}\left[\psi^r_k(t,g(x))\right]_i &= \log_{\left[g^r(\psi_k(t,x))\right]_j}\left[g^r(\psi_k(t,x))\right]_i \\
&= \log_{\left[w_{M,\vec{u},L}(\psi^r_k(t,x))\right]_{\sigma(j)}}\left[w_{M,\vec{u},L}(\psi^r_k(t,x))\right]_{\sigma(i)} \\
&= M \log_{\left[\psi^r_k(t,x)\right]_{\sigma(j)}}\left[\psi^r_k(t,x)\right]_{\sigma(i)}\ .
\end{aligned}$$

Using Lemma C.2.12,

$$\begin{aligned}
&\left[\psi^r_{k+1}(t,g(x))\right]_i = \\
&= \left(M \exp_{\left[\psi^r_k(t,x)\right]_{\sigma(i)}}\left(\sum_{i\neq j}\frac{\log_{\left[\psi^r_k(t,x)\right]_{\sigma(j)}}\left[\psi^r_k(t,x)\right]_{\sigma(i)}}{d_{\mathcal{M}_L}\left(\vec{r}_{\sigma(i)},\vec{r}_{\sigma(j)}\right)+1}\phi_d\left(\left[\Lambda_k(t,x)\right]_{\sigma(i)\sigma(j)}\right)\right)+\vec{u}\right) \% \, L
\end{aligned}$$

and using Proposition C.2.15 and the permutation invariance of the sum,

$$\begin{aligned}
\left[\psi^r_{k+1}(t,g(x))\right]_i &= \left[\left(M\left[\psi^r_{k+1}(t,g(x))\right]_i+\vec{u}\right)\ \%\ L\right]_{\sigma(i)} \\
&= g^r\left[\psi^r_{k+1}(t,x)\right]_i\ ,
\end{aligned}$$

which proves the recursion. Finally, using Lemma C.2.11 for the velocity field we get

$$\begin{aligned}
\hat{v}(t,g(x)) &= \begin{pmatrix} \mathbf{0}_N \\ \left\{\log_{\left[g^r(\boldsymbol{R})\right]_i}\left[\psi^r_K(t,g(x))\right]_i\right\}_{i=1}^N \end{pmatrix} \\
&= \begin{pmatrix} \mathbf{0}_N \\ \left\{M\log_{\left[\boldsymbol{R}\right]_{\sigma(i)}}\left[\psi^r_K(t,x)\right]_{\sigma(i)}\right\}_{i=1}^N \end{pmatrix} \\
&= A\,\hat{v}(t,x)\ .
\end{aligned}$$

Since $\mathrm{d}g_x = A$ by Lemma C.2.13, we can conclude. □

# Appendix D

# Supplementary material for Chapter 5

## D.1 Random organisation

Random organisation models were introduced as minimal dynamical descriptions of the experimentally observed transition between reversible and irreversible motion in periodically driven suspensions [122, 212, 221]. The dynamics are deliberately simple: at each discrete time step $t$, overlapping particles are randomly displaced according to a model-specific dynamical rule, and the dynamics are iterated until the system either reaches a stationary steady state or no overlapping pairs are left. Their simplicity has made random organisation models particularly attractive in recent years [121–124, 126, 212–214, 241, 242].

Two complementary aspects make random organisation models interesting. First, they are a paradigmatic example of nonequilibrium systems undergoing an *absorbing phase transition* between an active phase, where overlapping particles persist indefinitely, and an absorbing phase, where the system eventually reaches a configuration without overlaps [122, 243, 244] (see Section D.1.2). Second, they represent perhaps the simplest framework for generating hyperuniform states [121, 222, 223, 245] (see Section D.1.4). In this Appendix, we briefly review the main models and phenomenology of random organisation.

### D.1.1 Models

Depending on the specific dynamical rules, severals variants of random organisation models exist in the literature [122, 123, 126, 213]. Here, we focus on three representative cases.

In the model introduced by Tjhung and Berthier [122], overlapping particles at a given packing fraction $\phi$ are independently displaced by a random vector $\vec{\delta}_i$, whose direction is chosen uniformly on the unit circle and whose magnitude is drawn uniformly from the interval $[0, \epsilon]$.

Wilken *et al.* [213] proposed a variant of these dynamics known as *biased random organisation* (BRO), in which overlapping particles are displaced along the line connecting their centres, with a random magnitude in $[0, \epsilon]$ sampled independently for each particle. This modification does not change the nature of the absorbing phase transition, but shifts the critical packing fraction $\phi_c$ to substantially higher values [213].

Finally, in the *conserved biased random organisation* (CBRO) model of Hexner and Levine [123], overlapping pairs are also displaced along the line connecting their centres as in BRO, but the random magnitude in $[0, \epsilon]$ is the same for both particles in the pair. This rule enforces local conservation of the centre of mass, which makes the model hyperuniform throughout the active phase in both dilute [123] and crystalline [124] systems. It also suppresses long-wavelength vibrational modes, allowing for long-range order in two dimensions [124].

### D.1.2 Absorbing transition

Absorbing phase transitions are nonequilibrium phenomena in which the system can become trapped in a completely frozen state when a control parameter approaches a critical threshold [243, 244]. Similar to equilibrium phase transitions, they are characterised by universality classes, defined by critical exponents and scaling functions that depend only on general features of the dynamics, such as dimensionality and the symmetries of the order parameter. In random organisation models, this transition separates a phase in which particles overlap indefinitely (active phase) from a phase in which the system eventually reaches a configuration without overlaps (absorbing phase) [122]. The order parameter is the *activity* $f$, defined as the fraction of overlapping particles in the system. All the random organisation variants described above are believed to belong to the *conserved directed percolation* (CDP) universality class [122, 124, 213, 241], characterised by critical exponents that differ from standard directed percolation (DP) due to the presence of a conserved density field coupled to the order parameter [243]. Nonetheless, the precise nature of the transition remains under debate [222, 245–247]. In Fig. D.1, we present our numerical results for the CBRO model in two dimensions, showing the measured critical exponents for the activity $f$, the susceptibility associated to the fluctuations of the activity $\chi$, and the relaxation time $\tau_r$ [124].

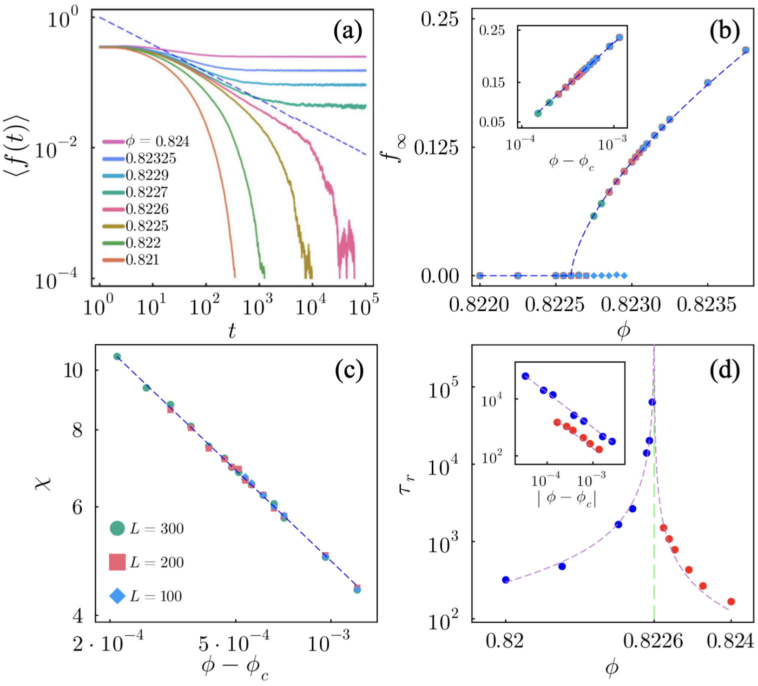

**Figure D.1:** Conserved directed percolation universality class. (a) The time dependence of the activity starting from a fully ordered structure reveals a critical packing fraction $\phi_c$. (b) Critical scaling of the average activity for three system sizes. (c) Diverging fluctuations of the activity at steady state. (d) Critical scaling of the relaxation time starting from an ordered configuration in the absorbing (blue) and active (red) phases. Reprinted from Ref. [124].

## D.1.3 Glass transition

Apart from a recent study [242], most works on random organisation have focused on monodisperse systems that crystallise at high density, while mixtures have typically been used only to avoid crystallisation, without examining the resulting glassy behaviour. In work currently in preparation, we study the interplay between random organisation and glassy dynamics in a binary mixture. We summarise the main results here, since the model we use is the same as the one employed in Chapter 5 to enhance hyperuniformity in selected configurations.

Specifically, we consider a two-dimensional model of $N$ particles in a square box of side $L$ with periodic boundary conditions. To suppress crystallisation, we use a bidisperse mixture in which 65% of the particles (type A) have diameter $\sigma$, and the remaining 35% (type B) have diameter $1.4\sigma$. We employ CBRO dynamics, with packing fraction $\phi = \left(\sum_i \pi\sigma_i^2\right)/(4L^2)$ and maximal displacement amplitude $\epsilon$ as control parameters.

In contrast to the monodisperse case [124], we find that this bidisperse model displays glassy behaviour at large $\phi$. To characterise its dynamics, we start from random configurations and run the dynamics at different values of $\phi$ and $\epsilon$ until the activity $f(t)$ reaches a steady state. We then measure the mean-squared displacement (Eq.(2.24)) and the self-intermediate scattering function (Eq.(2.25)) in the steady state. The results at fixed $\phi = 0.8$ for various $\epsilon$ in the active phase are shown in Fig. D.2(a-b). As $\epsilon$ decreases, the dynamics become significantly slower: the MSD develops a plateau at intermediate times, and the self-intermediate scattering function follows a two-step decay typical of glassy systems, with an increasingly pronounced plateau.

We extract the structural relaxation time $\tau_\alpha$ from the condition $F_s(q^\star, \tau_\alpha) = 1/\mathrm{e}$, and show its evolution with $\epsilon$ in Fig. D.2(c). The relaxation time crosses over from Arrhenius to super-Arrhenius behaviour as $\epsilon$ decreases, signalling a dynamical glass transition characterised by two-step relaxation. We emphasise that, for all values of $\epsilon$ shown, the steady-state activity lies in the range [0.88, 0.96], meaning that the system is well within the active phase and far from the absorbing transition. The observed slowing down is therefore a consequence of caging and collective particle rearrangements, and is not related to the critical slowing down associated with the absorbing transition.

Figure D.2(d) summarises the dynamical regimes in the $(\epsilon, \phi)$ plane. At sufficiently low $\phi$ and $\epsilon$, the system enters the absorbing phase. We locate the transition line separating the absorbing and active phases by monitoring the evolution of $f(t)$ across the parameter space, starting from random initial conditions: the transition is identified as the boundary between points where $f(t)$ relaxes to zero within the simulation time window and those where it remains finite. Above this critical line lies the active phase, which separates into an active liquid at large $\epsilon$ and an active glass with slow dynamics at small $\epsilon$. The crossover between these two regimes is marked by an iso-$\tau_\alpha$ line corresponding to $\tau_\alpha = 10^5$.

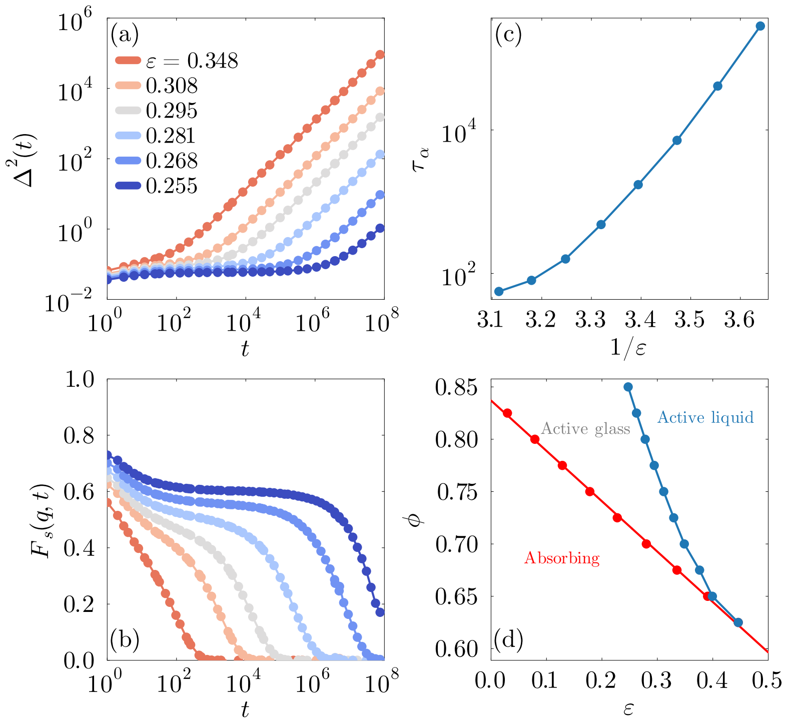

**Figure D.2:** Glass transition in CBRO. (a) Mean-squared displacement $\Delta^2(t)$ in the steady state for various $\epsilon$ at fixed $\phi = 0.8$. (b) Corresponding self-intermediate scattering function $F_s(q^\star, t)$. (c) Structural relaxation time $\tau_\alpha$ as a function of $\epsilon$ at fixed $\phi = 0.8$. (d) Phase diagram in the $(\epsilon, \phi)$ plane, showing the absorbing phase, active liquid, and active glass regions.

### D.1.4 Hyperuniformity

The emergence of hyperuniformity in random organisation models is subtle, as different variants suppress long-wavelength density fluctuations through distinct physical mechanisms. In practice, one can identify three separate sources of hyperuniformity. Exactly *at* the absorbing transition each model described above shows suppressed long-wavelength fluctuations as $S(q) \sim q^\alpha$, when $q \to 0$. However, the exact value of $\alpha$ is currently debated. It was long thought to fall within the CDP universality class with $\alpha \approx 0.45$ [124, 213, 222], consistent with a recent mapping to the quenched Edwards–Wilkinson model [246]. Even more recent renormalisation group calculations [247] suggest a distinction between dynamics that conserve the centre of mass and those that do not, which would imply two slightly different (and numerically extremely hard to disentangle) values of $\alpha$. Although well established, this form of hyperuniformity is challenging to observe numerically because it requires tuning exactly to the critical point. This is the main reason why we do not rely on this mechanism in Chapter 5 to enhance hyperuniformity in selected configurations.

A second source of hyperuniformity is related to jamming transition appearing at the $(\epsilon \to 0, \phi \to \phi_J)$ critical point in the BRO model [213, 241], which is even harder to tune numerically. It has been argued that this jamming transition is fundamentally different from the mean-field one, with critical behaviour instead controlled by random organisation itself. Regarding hyperuniformity, Refs. [213, 241] both report $\alpha \approx 0.45$ in two dimensions. That said, the exponent $\alpha$ at jamming is known to heavily depend on the preparation protocol [248–251], so these results should be interpreted with care.

Finally, CBRO introduces a third source of hyperuniformity. The centre-of-mass conservation imposed by the dynamics generates a locally anticorrelated noise that suppresses density fluctuations throughout the whole active phase [123, 124, 252, 253]. This mechanism has been shown to produce hyperuniform configurations in dilute [123] and crystalline systems [124], with a characteristic exponent $\alpha = 2$. However, even this type of hyperuniformity requires some care, particularly when dealing with kinetically arrested systems such as crystals and glasses [128]. Because the suppression of fluctuations originates from the dynamics, it affects only the *fluctuating* component $\delta\rho$ of the density field, and not the part associated with the average particle positions that forms the backbone of the solid. In crystals this backbone is trivially hyperuniform, since $S(q) \neq 0$ only at Bragg peaks, whereas in glasses it reflects the underlying disordered structure and is generically not hyperuniform. As a result, CBRO glasses are not fully hyperuniform: their backbone contribution remains finite at low $q$ and the resulting total structure factor does not vanish (see Fig. D.3).

We present a minimal coarse-grained model that captures this mechanism in the following Example.

**Example D.1.1** (Centre-of-mass conservation and hyperuniformity)**.** Consider the modifield Model B dynamics for the scalar field $\phi : \mathbb{R}^+ \times \mathbb{R}^d \to \mathbb{R}$

$$\frac{\partial \phi}{\partial t}(t, \vec{r}) = \Gamma \nabla^2 \frac{\delta \mathcal{F}}{\delta \phi} + \sqrt{2D}\, \nabla^2 \eta(t, \vec{r}) \ ,$$

where $\Gamma > 0$ is a damping coefficient, $D \geq 0$, $\mathcal{F}$ is a free-energy functional, and $\eta$ is white noise satisfying

$$\begin{aligned} \mathbb{E}\left[\eta\left(t, \vec{r}\right)\right] &= 0 \\ \mathbb{E}\left[\eta\left(t, \vec{r}\right) \eta\left(t', \vec{r}'\right)\right] &= \delta\left(t - t'\right) \delta\left(\vec{r} - \vec{r}'\right) \ , \end{aligned}$$

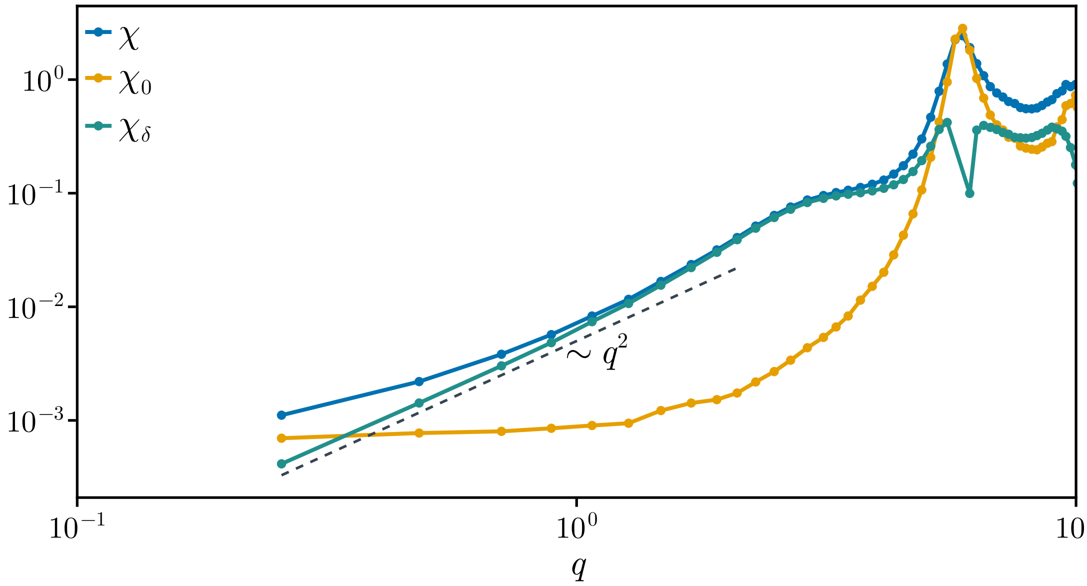

**Figure D.3:** Hyperuniformity in CBRO glasses. Total (blue), backbone (orange) and fluctuating (green) contributions to the $q$-dependent compressibility $\chi(q)$ for a CBRO glass at $\phi = 0.8$ and $\epsilon = 0.2$. The backbone contribution $\chi_0$ is approximated by computing the compressibility of the average particle positions, which shares the same low-$q$ behaviour [128]. $\chi_0$ remains finite as $q \to 0$ due to the disordered structure. The fluctuating part $\chi_\delta$ vanishes as $q^2$ due to the centre-of-mass conservation of CBRO dynamics. The resulting total compressibility $\chi$ remains finite at low $q$.

with the expectation taken over the noise distribution. Compared to standard Model B dynamics, where the stochastic term comes from a divergence [254, 255], here the (nonequilibrium) noise enters through a Laplacian. This modification enforces conservation of the centre of mass [123, 252, 253].

To model a solid vibrating around a reference configuration $\phi_0$, we choose the quadratic free-energy functional

$$\mathcal{F}[\phi] \;=\; \int_{\mathbb{R}^d} \left(K\,(\phi-\phi_0)^2 + \kappa\,\|\nabla\phi\|^2\right) \mathrm{d}\vec{r}\,,$$

where $K > 0$ is an elastic coefficient and $\kappa > 0$ is a stiffness parameter. This form of $\mathcal{F}$ leads to the linear stochastic differential equation

$$\frac{\partial\phi}{\partial t}(t,\vec{r}) = \Gamma\nabla^2\Big(K\big(\phi(t,\vec{r})-\phi_0(\vec{r})\big) - \kappa\nabla^2\phi(t,\vec{r})\Big) + \sqrt{2D}\,\nabla^2\eta(t,\vec{r})\,.$$

Introducing Fourier transforms

$$\begin{aligned}\hat{\phi}(t,\vec{q}) &= \int_{\mathbb{R}^d} \phi(t,\vec{r}) \exp\left(\mathrm{i}\vec{q}\cdot\vec{r}\right) \mathrm{d}\vec{r} \\ \hat{\eta}(t,\vec{q}) &= \int_{\mathbb{R}^d} \eta(t,\vec{r}) \exp\left(\mathrm{i}\vec{q}\cdot\vec{r}\right) \mathrm{d}\vec{r} \\ \hat{\phi}_0(\vec{q}) &= \int_{\mathbb{R}^d} \phi_0(\vec{r}) \exp\left(\mathrm{i}\vec{q}\cdot\vec{r}\right) \mathrm{d}\vec{r}\,,\end{aligned}$$

the equation becomes

$$\frac{\partial\hat{\phi}}{\partial t}(t,\vec{q}) + \Gamma\left(K+\kappa q^2\right) q^2\hat{\phi}(t,\vec{q}) = \Gamma K q^2\hat{\phi}_0(\vec{q}) - \sqrt{2D}q^2\hat{\eta}(t,\vec{q})\;.$$

Taking the expectation of the stationary state solution gives

$$\begin{aligned}\mathbb{E}\left[\hat{\phi}(\vec{q})\right] &= \frac{K}{K+\kappa q^2}\,\hat{\phi}_0(\vec{q}) \\ &\sim \hat{\phi}_0(\vec{q}), \quad \text{as } q\to 0\;,\end{aligned}$$

which shows that the average profile reproduces the reference configuration at long wavelengths. The backbone contribution to the structure factor (Eq. (2.18)) is then

$$\begin{aligned}S_0(\vec{q}) &= \frac{1}{N}\,\mathbb{E}\left[\hat{\phi}(\vec{q})\right]\mathbb{E}\left[\hat{\phi}(-\vec{q})\right] \\ &= \frac{K^2}{N\left(K+\kappa q^2\right)^2}\,\left|\hat{\phi}_0(\vec{q})\right|^2 \\ &\sim \left|\hat{\phi}_0(\vec{q})\right|^2, \quad \text{as } q\to 0\;.\end{aligned}$$

In crystals, $\hat{\phi}_0$ is nonzero only at Bragg peaks, so $S_0(q)=0$ at small $q$ and the backbone is trivially hyperuniform. In glasses, however, $\hat{\phi}_0$ does not vanish at small $q$, so $S_0$ approaches a constant and is not hyperuniform.

To extract fluctuations around the average progile, we set $\delta\hat{\phi}=\hat{\phi}-\mathbb{E}[\hat{\phi}]$, so that the fluctuating contribution to the structure factor (Eq. (2.19)) is

$$\begin{aligned}S_\delta(\vec{q}) &= \frac{1}{N}\,\mathbb{E}\left[\delta\hat{\phi}(\vec{q})\,\delta\hat{\phi}(-\vec{q})\right] \\ &= \frac{D\,q^2}{N\Gamma\left(K+\kappa q^2\right)} \\ &\sim q^2, \quad \text{as } q\to 0\;,\end{aligned}$$

so the fluctuating component is always hyperuniform with exponent $\alpha=2$, independent of the reference configuration.

The total structure factor $S = S_0 + S_\delta$ is therefore hyperuniform in crystals (consistent with Ref. [124]), but not in glasses, where $S_0$ dominates at low $q$. In liquids, the backbone term vanishes by translational invariance, and this dynamics indeed predicts $S(q) \sim q^2$ at small $q$, consistent with Ref [123].

## D.2 Reversibility of biased Monte Carlo

In this Appendix, we show that the MCMC scheme introduced in Ref. [215] and described in Section 5.1.2, which we use to sample from the biased distribution $\widetilde{P}$ defined in Eq. (5.1), indeed satisfies detailed balance with respect to $\widetilde{P}$. The algorithm resembles standard MH, except that the proposal configuration is generated by running a short MCMC trajectory targeting the *equilibrium* Boltzmann distribution $P$ of Eq. (2.3). At this stage it remains to clarify why the acceptance probability in Eq. (5.2) involves only the biasing potential $V$, with no explicit dependence on the underlying energy $U$ or on the detailed form of the proposal kernel $Q$. To clarify this point, we adopt a broader perspective. We show that, for an MH algorithm targeting a generic distribution $P$, whenever the proposal kernel $Q$ is reversible with respect to another measure $\mu$, the Hastings ratio acquires a particularly simple form: it involves only $P$ and $\mu$, and is entirely independent of the specific details of $Q$.

**Theorem D.2.1** (Hastings ratio for reversible proposals)**.** *Let* $(\mathsf{X}, \mathcal{E})$ *be a measurable space,* $P$ *and* $\mu$ *two probability measures on* $(\mathsf{X}, \mathcal{E})$*, and* $Q : \mathsf{X} \times \mathcal{E} \to \mathbb{R}^+$ *a Markov kernel. Consider the Metropolis–Hastings algorithm to sample from* $P$ *using* $Q$ *as proposal distribution. If* $Q$ *is reversible with respect to* $\mu$*, then the Hastings ratio defined in Eq.* (1.20) *simplifies to*

$$h(x, x') = \begin{cases} \dfrac{P\left(\mathrm{d}x'\right)\mu\left(\mathrm{d}x\right)}{P\left(\mathrm{d}x\right)\mu\left(\mathrm{d}x'\right)}, & \textit{if } P(\mathrm{d}x')Q(x', \mathrm{d}x) \sim P(\mathrm{d}x)Q(x, \mathrm{d}x') \\ 0, & \textit{otherwise} \end{cases} . \tag{D.1}$$

*Proof.* Let $R$ be the set where $P\left(\mathrm{d}x'\right) Q\left(x', \mathrm{d}x\right) \sim P\left(\mathrm{d}x\right) Q\left(x, \mathrm{d}x'\right)$. On $R^c$, the Hastings ratio is zero by definition. Conversely, using the reversibility of $Q$ with respect to $\mu$, for almost every $(x, x') \in R$,

$$\begin{aligned} h\left(x, x'\right) &= \frac{P\left(\mathrm{d}x'\right) Q\left(x', \mathrm{d}x\right)}{P\left(\mathrm{d}x\right) Q\left(x, \mathrm{d}x'\right)} \\ &= \frac{P\left(\mathrm{d}x'\right) Q\left(x', \mathrm{d}x\right)}{P\left(\mathrm{d}x\right) Q\left(x, \mathrm{d}x'\right)} \frac{Q\left(x, \mathrm{d}x'\right) \mu\left(\mathrm{d}x\right)}{Q\left(x', \mathrm{d}x\right) \mu\left(\mathrm{d}x'\right)} \\ &= \frac{P\left(\mathrm{d}x'\right) \mu\left(\mathrm{d}x\right)}{P\left(\mathrm{d}x\right) \mu\left(\mathrm{d}x'\right)} . \end{aligned} \qquad \square$$

**Corollary D.2.2** (Hastings ratio for biased Monte Carlo)**.** *In the settings of Theorem D.2.1, if $P$ is absolutely continuous with respect to $\mu$ with Radon–Nikodym derivative $f = \mathrm{d}P/\,\mathrm{d}\mu$, then the Hastings ratio in Eq.* (D.1) *simplifies to*

$$h(x, x') \; = \; \begin{cases} \dfrac{f\left(x'\right)}{f\left(x\right)}, & \textit{if } P(\mathrm{d}x')Q(x', \mathrm{d}x) \sim P(\mathrm{d}x)Q(x, \mathrm{d}x') \\ 0, & \textit{otherwise} \end{cases} . \tag{D.2}$$

*Proof.* Since $P$ is absolutely continuous with respec to $\mu$, then

$$P\left(\mathrm{d}x\right) \; = \; f\left(x\right) \mu\left(\mathrm{d}x\right) \; .$$

Let $R$ be the set where $P\left(\mathrm{d}x'\right) Q\left(x', \mathrm{d}x\right) \sim P\left(\mathrm{d}x\right) Q\left(x, \mathrm{d}x'\right)$. For almost every $(x, x') \in R$, using Theorem D.2.1,

$$\begin{aligned} h\left(x, x'\right) &= \frac{P\left(\mathrm{d}x'\right) \mu\left(\mathrm{d}x\right)}{P\left(\mathrm{d}x\right) \mu\left(\mathrm{d}x'\right)} \\ &= \frac{f\left(x'\right) \mu\left(\mathrm{d}x'\right) \mu\left(\mathrm{d}x\right)}{f\left(x\right) \mu\left(\mathrm{d}x\right) \mu\left(\mathrm{d}x'\right)} \\ &= \frac{f\left(x'\right)}{f\left(x\right)} . \end{aligned} \qquad \square$$

To see that these general results accounts for the simplification observed in Eq. (5.2), note that running a short MCMC trajectory targeting the Boltzmann distribution $P$ defines a proposal kernel $Q$ that is reversible with respect to $P$ itself. Therefore, in the BMC scheme, we can identify $P = \widetilde{P}$ and $\mu = P$ in Corollary D.2.2, so that $f(x) \propto \exp\big( - \beta\, V_\lambda\left(\Theta(x)\right) \big)$. Equation (5.2) then follows directly from Eq. (D.2).

# Appendix E

# Numerical implementations

The development of the Policy-guided Monte Carlo (PGMC) framework (see Chapter 3) led to the creation of two open-source Julia packages: `Arianna.jl` and `ParticlesMC.jl`. These packages were written in collaboration with Romain Simon and are available on GitHub at `https://github.com/TheDisorderedOrganization`. Our motivation was to provide a flexible framework for CPU-based MCMC simulations of physical systems: many research groups end up “reinventing the wheel” when it comes implementing MCMC algorithms for each new model. Both packages are written in Julia [256], which we think offers the best balance between performance, ease of use, and flexibility.

Although designed with PGMC in mind, these packages are not restricted to it. They were used, for instance, to generate the training data in Chapter 4 and to implement the biased sampling scheme in Chapter 5. They remain under active development, and further features are planned. Below is an overview of the two packages.

## E.1 Arianna.jl

`Arianna.jl` [146] is a Julia package for running (adaptive) MCMC simulations of physical systems. Existing MCMC software roughly falls into two categories. Packages aimed at computational statistics usually offer a broad set of algorithms, but they tend to be difficult to customise for physical systems, where one often needs a lot of control to implement domain-specific kernels. Conversely, packages developed within physics communities are frequently tailored to specific models or applications, which limits their extensibility. As a result, researchers often reimplement the same update rules and data storage each time a new model is studied, a procedure that is both time-consuming and error-prone. `Arianna.jl` is intended to bridge this gap by providing a lightweight and extensible framework for system-agnostic MCMC algorithms, while handling the technical parts of simulation efficiency.

**Figure E.1:** `Arianna.jl` logo.

### E.1.1 Main features

Main features of `Arianna.jl` include:

- **General MCMC engine:** A minimal framework that provides the essential components for MCMC sampling. Users define their system and transition kernel in a physics-oriented manner.
- **Extensible algorithms:** Built-in support for Metropolis–Hastings and Policy-guided Monte Carlo, with a structure that makes it easy to implement additional methods.
- **Parallel execution:** Support for synchronised chains on multiple CPU cores.
- **Simple user interface:** Simulations can be set up with a small amount of code while exposing the key parameters of each algorithm.
- **Flexible data management:** Simulation trajectories can be stored in custom formats, and users can attach callbacks to compute or record quantities during runtime.
- **Documentation:** Comprehensive documentation and examples are available at `https://thedisorderedorganization.github.io/Arianna.jl/stable/`.
- **Consistency checking:** A built-in linter tool inspects user-defined systems and transition kernels, helping catch missing definitions or inconsistencies before running simulations.

### E.1.2 Design strategy

Under the hood, `Arianna.jl` runs, at each step, a set of *algorithms* that operate on (possibly parallel) *systems* by performing actions that depend only on the *current* state of the system. Each algorithm has its own *scheduler*, which specifies at which steps of the simulation's "internal clock" it should be executed. This borad definition of algorithm allows to use the same framework to implement, for instance, different transition kernels, adaptive schemes such as PGMC, user-designed callbacks at specific times, data-management routines, and even molecular dynamics integrators. To keep this level of abstraction, `Arianna.jl` relies heavily on Julia's multiple dispatch system [256], which replaces class-based methods with functions having specialised methods for different combinations of argument types.

Despite its generality, the package is designed designed primarily for MCMC simulations of physical systems. To this end, as a core feature, it includes a general implementation of the Metropolis-Hastings algorithm that can be customised by defining a few essential methods for the system at hand. Following a physics-oriented approach, a transition kernel is defined by a *move* which combines the *action* of updating the current state of the system with the *policy* (*i.e.* proposal distribution) that specifies how to sample the action and how to evaluate its probability density. Multiple moves (*e.g.* displacements and swaps) with specific *weights* can be combined to build a composite transition kernel. `Arianna.jl` then handles proposing, accepting or rejecting moves, and updating the system efficiently.

Performance considerations guided several design choices. The system is *mutable*, and only the parts affected by a move are updated. This keeps large-system simulations fast on CPU, although it limits direct GPU support. As a consequence, the package is well suited for PGMC with a relatively small number of parameters, but not for policies based on large generative models.

Another important design choice is the support for multiple synchronised chains running in parallel on different CPU cores. While traditional MCMC simulations rarely require such synchronisation (and the overhead is often not justified), it becomes essential for certain techniques such as parallel tempering. PGMC also benefits from parallel chains, which can be used to obtain more stable gradient estimates before each policy update. Naturally, users can also run completely independent simulations, which is often preferred in high-performance computing (HPC) environments. While in conventional MCMC simulations this is often avoided as there is no need to syncronise multiple chains and single core simulations are easier to deal with in HPC cluster (also adds overhead), this feature is crucial in some applications such as parallel tempering. PGMC also benefits from parallel chains, as they can be used to get a better estimate of gradients before each update. Of course users can run independent simulations.

Since the resulting interface is quite abstract, the package includes a few simple example systems. The intention, however, is that `Arianna.jl` serves as a foundation upon which more specialised code can be built. Our own `ParticlesMC.jl` package is an example of this.

## E.2 ParticlesMC.jl

`ParticlesMC.jl` is a Julia package built on top of `Arianna.jl` for MCMC simulations of atomic and molecular particle systems. It is designed to be both efficient and straightforward to use, with a particular focus on classical liquids and glass-forming systems. Although it is not yet a registered package, it is available on GitHub at `https://github.com/TheDisorderedOrganization/ParticlesMC`.

Despite the large community working on computer simulations of liquids, most general-purpose software targets molecular dynamics rather than MCMC. However, for glass-forming systems, where conventional dynamics slows down dramatically at low temperature, MCMC can equilibrate systems efficiently through non-physical moves [140] (see Section 2.4.2). To the best of our knowledge, no general-purpose MCMC package currently provides an efficient and user-friendly framework for particle systems with custom moves. `ParticlesMC.jl` was developed to fill this gap by providing the essential components (*e.g.* interaction potentials, neighbour lists, data structures) needed to set up and run MCMC simulations of particle systems with minimal code.

ParticlesMC

**Figure E.2:** `ParticlesMC.jl` logo.

### E.2.1 Main features

Main features of `ParticlesMC.jl` include:

- **Support for atomic and molecular systems:** High-performance data structures, neighbour lists, and interaction potentials.

- **State-of-the-art moves:** Implementation of transition kernels widely used in glass-forming mixtures, including single-particle displacements, particle swaps, molecular flips, and intramolecular swaps for polymeric systems.

- **General ensembles (experimental):** Support for both NVT and NPT ensembles by internally adapting the target distribution.

- **Arianna features:** Inherits all features of `Arianna.jl`, including support for PGMC, parallel chains, and flexible data management.

- **Flexible execution:** Simulations can be run from scripts or via a command-line interface, which integrates naturally with HPC workflows.

- **Thorough testing:** New features are tested both at the code level and at the physical level, for instance by verifying detailed balance and checking that the correct distribution is sampled.

### E.2.2 Design strategy

The idea behind `ParticlesMC.jl` was to provide a package that avoids "reinventing the wheel" every time the interaction model changes, the spatial dimensionality is modified, or a new move is introduced. At the same time, the implementation had to remain *fast*. In particle systems, MCMC moves typically act on a small subset of the degrees of freedom (*e.g.*, a single particle). To implement these efficiently, repeated computations must be avoided and only the relevant parts of the system should be updated. `ParticlesMC.jl` relies on Julia's in-place operations to handle these updates efficiently under the hood while keeping the interface simple.

The package aims to handle the range of systems commonly used in the study of liquids and glasses. Soft vs. hard spheres, discrete vs. continuous polydispersity, and atomic vs. molecular systems often require different implementations and are usually handled by separate codes. `ParticlesMC.jl` is flexible enough to treat these cases within a single framework without rewriting everything from scratch. As with `Arianna.jl`, Julia's multiple dispatch system [256] plays an essential role, keeping types abstract types while while still allowing specialised implmentations to maximise performance.

Through `Arianna.jl`, the package inherits the ability to specify custom target distributions. This makes it possible, for example, to perform simulations in the NPT ensemble, or to target biased distribution for umbrella sampling. Annealing procedures are also being developed as an experimental feature.

For large systems, evaluating interaction potentials can be the main source of computational cost. Neighbour lists are therefore crucial [110, 111], but they can be difficult to implement correctly and are a common source of errors. `ParticlesMC.jl` provides several neighbour-list implementations and can, in some cases, choose an appropriate one automatically (currently experimental).

Interaction potentials are a core component to any particle simulation. `ParticlesMC.jl` includes common pair potentials (inverse-power-law, Lennard–Jones, Weeks–Chandler–Andersen) and predefined *models* corresponding to mixtures frequently used in studies of glass-forming liquids. These built-in models are meant as temporary conveniences, and may be moved to an external package in future versions to simplify maintenance and increase flexibility.

Finally, the package is designed to be easy to use in both local and HPC environments. Simulations can be launched through simple command-line TOML files specifying the system and algorithm parameters.

# Bibliography

[1] L. Berthier and G. Biroli, «Theoretical perspective on the glass transition and amorphous materials», Reviews of Modern Physics **83**, 587–645 (2011) (cit. on pp. 1, 29, 31, 47).

[2] P. G. Debenedetti and F. H. Stillinger, «Supercooled liquids and the glass transition», Nature **410**, 259–267 (2001) (cit. on pp. 1, 29).

[3] L. Berthier and D. R. Reichman, «Modern computational studies of the glass transition», Nature Reviews Physics **5**, 102–116 (2023) (cit. on pp. 1, 29, 30, 47, 79).

[4] C. A. Angell, «Formation of Glasses from Liquids and Biopolymers», Science **267**, 1924–1935 (1995) (cit. on pp. 1, 30).

[5] C. Scalliet, B. Guiselin, and L. Berthier, «Thirty Milliseconds in the Life of a Supercooled Liquid», Physical Review X **12**, 041028 (2022) (cit. on pp. 1, 50, 117).

[6] C. Alba-Simionesco and G. Tarjus, «A perspective on the fragility of glass-forming liquids», Journal of Non-Crystalline Solids: X **14**, 100100 (2022) (cit. on p. 1).

[7] P. N. Pusey and W. Van Megen, «Phase behaviour of concentrated suspensions of nearly hard colloidal spheres», Nature **320**, 340–342 (1986) (cit. on pp. 1, 29).

[8] G. L. Hunter and E. R. Weeks, «The Physics of the Colloidal Glass Transition», Reports on Progress in Physics **75**, 066501 (2012) (cit. on pp. 1, 29).

[9] T. G. Mason, J. Bibette, and D. A. Weitz, «Yielding and Flow of Monodisperse Emulsions», Journal of Colloid and Interface Science **179**, 439–448 (1996) (cit. on p. 1).

[10]B. Herzhaft, S. Kakadjian, and M. Moan, «Measurement and modeling of the flow behavior of aqueous foams using a recirculating pipe rheometer», Colloids and Surfaces A: Physicochemical and Engineering Aspects, A Collection of Papers Presented at the 5th European Conference on Foams, Emulsions, and Applications, EUFOAM 2004, University of Marne-la-Vallee, Champs Sur Marne (France), 5-8 July, 2004 **263**, 153–164 (2005) (cit. on pp. 1, 29).

[11]R. Candelier, O. Dauchot, and G. Biroli, «Building Blocks of Dynamical Heterogeneities in Dense Granular Media», Physical Review Letters **102**, 088001 (2009) (cit. on pp. 1, 29).

[12]L. Berthier, *Dynamical Heterogeneities in Glasses, Colloids, and Granular Media*, International Series of Monographs on Physics Ser v.150 (Oxford University Press, Incorporated, Oxford, 2011) (cit. on pp. 1, 29).

[13]S. Sastry, P. G. Debenedetti, and F. H. Stillinger, «Signatures of distinct dynamical regimes in the energy landscape of a glass-forming liquid», Nature **393**, 554–557 (1998) (cit. on pp. 1, 32, 63).

[14]N. Metropolis and S. Ulam, «The Monte Carlo Method», Journal of the American Statistical Association **44**, 335–341 (1949) (cit. on pp. 2, 5).

[15]H. Robbins and S. Monro, «A Stochastic Approximation Method», The Annals of Mathematical Statistics **22**, 400–407 (1951) (cit. on pp. 2, 17, 18, 59).

[16]S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, «Optimization by Simulated Annealing», Science **220**, 671–680 (1983) (cit. on pp. 2, 117).

[17]C. P. Robert and G. Casella, *Monte Carlo statistical methods*, 2. ed., softcover reprint of the hardcover 2. ed. 2004, Springer Texts in Statistics (Springer New York, New York, NY, 2010) (cit. on pp. 2, 5, 7, 16).

[18]J. Nocedal and S. J. Wright, *Numerical optimization*, Second edition, Springer Series in Operations Research and Financial Engineering (Springer, New York, NY, 2006) (cit. on pp. 2, 59–61, 115).

[19]S. P. Meyn and R. L. Tweedie, *Markov Chains and Stochastic Stability*, Communications and Control Engineering Series (Springer, London, 1993) (cit. on pp. 2, 9–12).

[20]N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, «Equation of State Calculations by Fast Computing Machines», The Journal of Chemical Physics **21**, 1087–1092 (1953) (cit. on pp. 2, 5, 14, 15).

[21]W. K. Hastings, «Monte Carlo sampling methods using Markov chains and their applications», Biometrika **57**, 97–109 (1970) (cit. on pp. 2, 14).

[22] H. Haario, E. Saksman, and J. Tamminen, «An Adaptive Metropolis Algorithm», Bernoulli **7**, 223 (2001) (cit. on pp. 2, 16).

[23] G. O. Roberts and J. S. Rosenthal, «Coupling and Ergodicity of Adaptive Markov Chain Monte Carlo Algorithms», Journal of Applied Probability **44**, 458–475 (2007) (cit. on pp. 2, 17).

[24] C. Andrieu and J. Thoms, «A tutorial on adaptive MCMC», Statistics and Computing **18**, 343–373 (2008) (cit. on pp. 2, 16, 18, 79).

[25] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*, Adaptive Computation and Machine Learning (The MIT press, Cambridge, Mass, 2016) (cit. on p. 2).

[26] D. J. Rezende and S. Mohamed, *Variational Inference with Normalizing Flows*, June 2016 (cit. on pp. 3, 19).

[27] L. Dinh, J. Sohl-Dickstein, and S. Bengio, *Density estimation using Real NVP*, Feb. 2017 (cit. on pp. 3, 19).

[28] G. Papamakarios, E. Nalisnick, D. J. Rezende, S. Mohamed, and B. Lakshminarayanan, «Normalizing Flows for Probabilistic Modeling and Inference», (cit. on pp. 3, 19).

[29] F. Noé, S. Olsson, J. Köhler, and H. Wu, «Boltzmann generators: Sampling equilibrium states of many-body systems with deep learning», Science **365**, eaaw1147 (2019) (cit. on pp. 3, 18, 19, 23, 55, 62, 79–81).

[30] Y. Lipman, R. T. Q. Chen, H. Ben-Hamu, M. Nickel, and M. Le, *Flow Matching for Generative Modeling*, Feb. 2023 (cit. on pp. 3, 23).

[31] M. S. Albergo, N. M. Boffi, and E. Vanden-Eijnden, *Stochastic Interpolants: A Unifying Framework for Flows and Diffusions*, Nov. 2023 (cit. on pp. 3, 5, 23, 26, 83, 86).

[32] S. Zhai et al., *Normalizing Flows are Capable Generative Models*, June 2025 (cit. on p. 3).

[33] M. Kumar, M. Babaeizadeh, D. Erhan, C. Finn, S. Levine, L. Dinh, and D. Kingma, *VideoFlow: A Conditional Flow-Based Model for Stochastic Video Generation*, Feb. 2020 (cit. on p. 3).

[34] I. Price et al., «Probabilistic weather forecasting with machine learning», Nature **637**, 84–90 (2025) (cit. on p. 3).

[35] L. Klein, A. Krämer, and F. Noé, *Equivariant flow matching*, Nov. 2023 (cit. on pp. 3, 27, 82, 84–86, 90, 98).

[36] I. Katsman, A. Lou, D. Lim, Q. Jiang, S.-N. Lim, and C. D. Sa, *Equivariant Manifold Flows*, Jan. 2022 (cit. on pp. 3, 26–28, 82–84, 98).

[37] B. K. Miller, R. T. Q. Chen, A. Sriram, and B. M. Wood, *FlowMM: Generating Materials with Riemannian Flow Matching*, June 2024 (cit. on pp. 3, 81, 82, 86, 89, 98).

[38] S. Ciarella, J. Trinquier, M. Weigt, and F. Zamponi, «Machine-learning-assisted Monte Carlo fails at sampling computationally hard problems», Machine Learning: Science and Technology **4**, 010501 (2023) (cit. on pp. 3, 62).

[39] J.-L. Barrat and L. Berthier, «Computer simulations of the glass transition and glassy materials», Comptes Rendus. Physique **24**, 57–72 (2024) (cit. on p. 3).

[40] G. Jung et al., «Roadmap on machine learning glassy dynamics», Nature Reviews Physics **7**, 91–104 (2025) (cit. on p. 3).

[41] T. A. Bojesen, «Policy-guided Monte Carlo: Reinforcement-learning Markov chain dynamics», Physical Review E **98**, 063303 (2018) (cit. on pp. 3, 18, 51, 57, 78, 119).

[42] L. Galliano, R. Rende, and D. Coslovich, «Policy-guided Monte Carlo on general state spaces: Application to glass-forming mixtures», The Journal of Chemical Physics **161**, 064503 (2024) (cit. on pp. 4, 51–53, 56, 66–69, 71, 73–76, 78, 119, 140).

[43] J. Wu, B. Chen, Y. Zhou, Q. Meng, R. Zhu, and Z.-M. Ma, *Riemannian Neural Geodesic Interpolant*, Apr. 2025 (cit. on pp. 4, 23–26, 82, 83, 85).

[44] J. Köhler, L. Klein, and F. Noé, *Equivariant Flows: Exact Likelihood Generative Learning for Symmetric Densities*, Oct. 2020 (cit. on pp. 4, 26, 27, 82–84, 98).

[45] L. Grenioux, L. Galliano, L. Berthier, G. Biroli, and M. Gabrié, *Riemannian Stochastic Interpolants for Amorphous Particle Systems*, Dec. 2025 (cit. on pp. 4, 40, 82, 88, 93, 94, 96–100).

[46] S. Torquato and F. H. Stillinger, «Local density fluctuations, hyperuniformity, and order metrics», Physical Review E **68**, 041113 (2003) (cit. on pp. 4, 42, 120).

[47] H. Tong and H. Tanaka, «Structural order as a genuine control parameter of dynamics in simple glass formers», Nature Communications **10**, 5596 (2019) (cit. on p. 4).

[48] T. Yanagishima, J. Russo, R. P. A. Dullens, and H. Tanaka, «Towards Glasses with Permanent Stability», Physical Review Letters **127**, 215501 (2021) (cit. on pp. 4, 103–105, 116, 120).

[49] X. Fan, D. Xu, J. Zhang, H. Hu, P. Tan, N. Xu, H. Tanaka, and H. Tong, *Ideal noncrystals: A possible new class of ordered matter without apparent broken symmetry*, Aug. 2024 (cit. on pp. 4, 103, 104, 113, 116, 120).

[50] F. Leoni, J. Russo, F. Sciortino, and T. Yanagishima, *Generating ultrastable glasses by homogenizing the local virial stress*, May 2024 (cit. on pp. 4, 103, 104, 116, 120).

[51] Y. Wang, Z. Qian, H. Tong, and H. Tanaka, «Hyperuniform disordered solids with crystal-like stability», Nature Communications **16**, 1398 (2025) (cit. on pp. 4, 104, 116, 120).

[52] H. Kahn and A. W. Marshall, «Methods of Reducing Sample Size in Monte Carlo Computations», Journal of the Operations Research Society of America **1**, 263–278 (1953) (cit. on p. 5).

[53] S. Brooks, ed., *Handbook of Markov chain Monte Carlo*, Chapman & Hall/CRC Handbooks of Modern Statistical Methods (CRC Press, Boca Raton, Fla, 2011) (cit. on pp. 5, 14).

[54] R. T. Q. Chen, Y. Rubanova, J. Bettencourt, and D. Duvenaud, *Neural Ordinary Differential Equations*, Dec. 2019 (cit. on pp. 5, 19, 23).

[55] P. J. Green, «Reversible jump Markov chain Monte Carlo computation and Bayesian model determination», Biometrika **82**, 711–732 (1995) (cit. on pp. 14, 51).

[56] L. Tierney, «A note on Metropolis-Hastings kernels for general state spaces», The Annals of Applied Probability **8**, `10.1214/aoap/1027961031` (1998) (cit. on p. 15).

[57] V. S. Borkar, *Topics in controlled Markov chains*, 1. publ, Pitman Research Notes in Mathematics Series 240 (Longman Scientific & Technical u.a, Harlow, 1991) (cit. on p. 16).

[58] A. Gelman, W. R. Gilks, and G. O. Roberts, «Weak convergence and optimal scaling of random walk Metropolis algorithms», The Annals of Applied Probability **7**, 110–120 (1997) (cit. on pp. 16, 17).

[59] W. R. Gilks, G. O. Roberts, and S. K. Sahu, «Adaptive Markov Chain Monte Carlo through Regeneration», Journal of the American Statistical Association **93**, 1045–1054 (1998) (cit. on p. 16).

[60] M. A. Miller, L. M. Amon, and W. P. Reinhardt, «Should one adjust the maximum step size in a Metropolis Monte Carlo simulation?», Chemical Physics Letters **331**, 278–284 (2000) (cit. on p. 17).

[61] A. Gelman and C. Pasarica, «Adaptively Scaling the Metropolis Algorithm Using Expected Squared Jumped Distance», SSRN Electronic Journal, `10.2139/ssrn.1010403` (2007) (cit. on pp. 18, 65).

[62] L. B. Rall et al., eds., *Automatic Differentiation: Techniques and Applications*, Vol. 120, Lecture Notes in Computer Science (Springer Berlin Heidelberg, Berlin, Heidelberg, 1981) (cit. on p. 18).

[63] H. Christiansen, F. Errica, and F. Alesiani, «Self-tuning Hamiltonian Monte Carlo for accelerated sampling», The Journal of Chemical Physics **159**, 234109 (2023) (cit. on pp. 18, 65).

[64] R. S. Sutton and A. Barto, *Reinforcement learning: an introduction*, Nachdruck, Adaptive Computation and Machine Learning (The MIT Press, Cambridge, Massachusetts, 2014) (cit. on pp. 18, 52, 54, 59, 78, 117).

[65] K.-W. Zhao, W.-H. Kao, K.-H. Wu, and Y.-J. Kao, «Generation of ice states through deep reinforcement learning», Physical Review E **99**, 062106 (2019) (cit. on pp. 18, 51).

[66] C. Wang, W. Chen, H. Kanagawa, and C. J. Oates, *Reinforcement Learning for Adaptive MCMC*, May 2024 (cit. on p. 18).

[67] L. Grenioux, A. Durmus, É. Moulines, and M. Gabrié, *On Sampling with Approximate Transport Maps*, Feb. 2024 (cit. on p. 18).

[68] M. S. Albergo, G. Kanwar, and P. E. Shanahan, «Flow-based generative models for Markov chain Monte Carlo in lattice field theory», Physical Review D **100**, 034515 (2019) (cit. on pp. 18, 19, 22, 80, 81).

[69] M. Gabrié, G. M. Rotskoff, and E. Vanden-Eijnden, «Adaptive Monte Carlo augmented with normalizing flows», Proceedings of the National Academy of Sciences **119**, e2109420119 (2022) (cit. on pp. 18, 80, 81, 100).

[70] C. Schönle and M. Gabrié, «Optimizing Markov Chain Monte Carlo Convergence with Normalizing Flows and Gibbs Sampling», in NeurIPS 2023 AI for Science Workshop (Dec. 2023) (cit. on pp. 18, 101).

[71] A. Cabezas, L. Sharrock, and C. Nemeth, *Markovian Flow Matching: Accelerating MCMC with Continuous Normalizing Flows*, May 2024 (cit. on p. 18).

[72] E. Mathieu and M. Nickel, *Riemannian Continuous Normalizing Flows*, Dec. 2020 (cit. on pp. 19–21).

[73] E. Hairer, «Solving Differential Equations on Manifolds», (cit. on p. 19).

[74] E. A. Coddington and N. Levinson, *Theory of Ordinary Differential Equations* (McGraw-Hill, 1955) (cit. on p. 20).

[75] W. Grathwohl, R. T. Q. Chen, J. Bettencourt, I. Sutskever, and D. Duvenaud, *FFJORD: Free-form Continuous Dynamics for Scalable Reversible Generative Models*, Oct. 2018 (cit. on p. 21).

[76] M. Hutchinson, «A stochastic estimator of the trace of the influence matrix for laplacian smoothing splines», Communications in Statistics - Simulation and Computation **19**, 433–450 (1990) (cit. on p. 21).

[77] D. P. Kingma and M. Welling, *Auto-Encoding Variational Bayes*, Dec. 2022 (cit. on pp. 22, 55).

[78] J. H. Huggins, M. Kasprzak, T. Campbell, and T. Broderick, *Validated Variational Inference via Practical Posterior Error Bounds*, Feb. 2020 (cit. on p. 22).

[79] L. S. Pontrjagin, *The mathematical theory of optimal processes*, 3. pr (Interscience Publ, New York, NY, 1965) (cit. on p. 23).

[80] M. S. Albergo and E. Vanden-Eijnden, *Building Normalizing Flows with Stochastic Interpolants*, Mar. 2023 (cit. on pp. 23, 83).

[81] R. T. Q. Chen and Y. Lipman, *Flow Matching on General Geometries*, Feb. 2024 (cit. on pp. 23, 82, 83, 85, 91, 119).

[82] A. Tong, K. Fatras, N. Malkin, G. Huguet, Y. Zhang, J. Rector-Brooks, G. Wolf, and Y. Bengio, *Improving and generalizing flow-based generative models with minibatch optimal transport*, Mar. 2024 (cit. on pp. 24, 84–86).

[83] J. Köhler, L. Klein, and F. Noé, *Equivariant Flows: sampling configurations for multi-body systems with symmetric energies*, Oct. 2019 (cit. on pp. 26, 83, 119).

[84] V. G. Satorras, E. Hoogeboom, F. B. Fuchs, I. Posner, and M. Welling, *E(n) Equivariant Normalizing Flows*, Jan. 2022 (cit. on pp. 27, 82, 83, 86, 87, 90, 91).

[85] J.-P. Hansen and I. R. McDonald, *Theory of simple liquids: with applications to soft matter*, 4th edition (Elsevier/AP, Amstersdam, 2013) (cit. on pp. 29, 35, 38, 40).

[86] A. P. Young and World Scientific (Firm), eds., *Spin glasses and random fields*, Series on Directions in Condensed Matter Physics v. 12 (World Scientific Pub. Co, Singapore River Edge, N.J, 1998) (cit. on p. 29).

[87] C. Scalliet, «Solides amorphes de la transition vitreuse à 1 Kelvin», Theses (Université de Montpellier, Sept. 2019) (cit. on p. 29).

[88] P. W. Anderson, «Through the Glass Lightly», Science **267**, 1615–1616 (1995) (cit. on p. 29).

[89] L. Berthier and M. D. Ediger, «Facets of glass physics», Physics Today **69**, 40–46 (2016) (cit. on pp. 31, 103).

[90] C. A. Angell, «Relaxation in liquids, polymers and plastic crystals — strong/fragile patterns and problems», Journal of Non-Crystalline Solids, Proceedings of the International Discussion Meeting on Relaxations in Complex Systems **131–133**, 13–31 (1991) (cit. on p. 30).

[91] F. C. Frank, «Supercooling of liquids», Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences **215**, 43–46 (1952) (cit. on p. 31).

[92] G. Adam and J. H. Gibbs, «On the Temperature Dependence of Cooperative Relaxation Properties in Glass-Forming Liquids», The Journal of Chemical Physics **43**, 139–146 (1965) (cit. on p. 31).

[93] M. Goldstein, «Viscous Liquids and the Glass Transition: A Potential Energy Barrier Picture», The Journal of Chemical Physics **51**, 3728–3739 (1969) (cit. on p. 31).

[94] U. Bengtzelius, W. Gotze, and A. Sjolander, «Dynamics of supercooled liquids and the glass transition», Journal of Physics C: Solid State Physics **17**, 5915 (1984) (cit. on p. 31).

[95] T. R. Kirkpatrick, D. Thirumalai, and P. G. Wolynes, «Scaling concepts for the dynamics of viscous liquids near an ideal glassy state», Physical Review A **40**, 1045–1054 (1989) (cit. on p. 31).

[96] D. Chandler and J. P. Garrahan, «Dynamics on the Way to Forming Glass: Bubbles in Space-Time», Annual Review of Physical Chemistry **61**, 191–217 (2010) (cit. on p. 31).

[97] C. P. Royall, F. Turci, S. Tatsumi, J. Russo, and J. Robinson, «The race to the bottom: approaching the ideal glass?», Journal of Physics: Condensed Matter **30**, 363001 (2018) (cit. on p. 32).

[98] R. J. Silbey, R. A. Alberty, and M. Bawendi, *Physical chemistry*, 4. ed (Wiley, Hoboken, NJ, 2005) (cit. on p. 33).

[99] H. Tanaka, «Two-order-parameter description of liquids. I. A general model of glass transition covering its strong to fragile limit», The Journal of Chemical Physics **111**, 3163–3174 (1999) (cit. on p. 33).

[100] G. Tarjus, S. A. Kivelson, Z. Nussinov, and P. Viot, «The frustration-based approach of supercooled liquids and the glass transition: a review and critical assessment», Journal of Physics: Condensed Matter **17**, R1143 (2005) (cit. on p. 33).

[101] J. Horbach and W. Kob, «Static and dynamic properties of a viscous silica melt», Physical Review B **60**, 3169–3181 (1999) (cit. on p. 33).

[102] H. Tanaka, «Two-order-parameter description of liquids. II. Criteria for vitrification and predictions of our model», The Journal of Chemical Physics **111**, 3175–3182 (1999) (cit. on p. 33).

[103] C. Hansen, F. Stickel, T. Berger, R. Richert, and E. W. Fischer, «Dynamics of glass-forming liquids. III. Comparing the dielectric $\alpha$- and $\beta$-relaxation of 1-propanol and o-terphenyl», The Journal of Chemical Physics **107**, 1086–1093 (1997) (cit. on p. 33).

[104] D. Coslovich, L. Galliano, and L. Costigliola, «Freezing, melting, and the onset of glassiness in binary mixtures», The Journal of Chemical Physics **162**, 061102 (2025) (cit. on pp. 33, 34, 46).

[105] W. Kob and H. C. Andersen, «Testing mode-coupling theory for a supercooled binary Lennard-Jones mixture. II. Intermediate scattering function and dynamic susceptibility», Physical Review E **52**, 4134–4153 (1995) (cit. on pp. 33, 34, 39, 62, 63).

[106] U. R. Pedersen, T. B. Schrøder, and J. C. Dyre, «Phase Diagram of Kob-Andersen-Type Binary Lennard-Jones Mixtures», Physical Review Letters **120**, 165501 (2018) (cit. on p. 34).

[107] C. J. Fullerton and L. Berthier, «Density controls the kinetic stability of ultrastable glasses», EPL (Europhysics Letters) **119**, 36003 (2017) (cit. on pp. 33, 103, 109–111).

[108] S. F. Swallen, K. L. Kearns, M. K. Mapes, Y. S. Kim, R. J. McMahon, M. D. Ediger, T. Wu, L. Yu, and S. Satija, «Organic Glasses with Exceptional Thermodynamic and Kinetic Stability», Science **315**, 353–356 (2007) (cit. on pp. 34, 103).

[109] C. Scalliet, L. Berthier, and F. Zamponi, «Absence of Marginal Stability in a Structural Glass», Physical Review Letters **119**, 205501 (2017) (cit. on p. 35).

[110] D. Frenkel and B. Smit, *Understanding molecular simulation: from algorithms to applications*, Third edition (ELSEVIER ACADEMIC PRESS, S.l., 2023) (cit. on pp. 35, 37, 40, 47, 65, 105, 106, 176).

[111] M. P. Allen and D. J. Tildesley, *Computer simulation of liquids*, 2nd ed (Oxford university press, Oxford, 2017) (cit. on pp. 36, 40, 47, 176).

[112] J. Paret, «Ordre caché dans les matériaux désordonnés», Theses (Université Montpellier, Nov. 2021) (cit. on p. 36).

[113] L. Berthier and T. A. Witten, «Glass transition of dense fluids of hard and compressible spheres», Physical Review E **80**, 021502 (2009) (cit. on p. 38).

[114] P. Charbonneau, E. I. Corwin, G. Parisi, and F. Zamponi, «Universal Microstructure and Mechanical Stability of Jammed Packings», Physical Review Letters **109**, 205501 (2012) (cit. on p. 38).

[115] B. Bernu, J. P. Hansen, Y. Hiwatari, and G. Pastore, «Soft-sphere model for the glass transition in binary alloys: Pair structure and self-diffusion», Physical Review A **36**, 4891–4903 (1987) (cit. on pp. 38, 62).

[116] U. R. Pedersen, T. B. Schrøder, and J. C. Dyre, «Repulsive Reference Potential Reproducing the Dynamics of a Liquid with Attractions», Physical Review Letters **105**, 157801 (2010) (cit. on p. 38).

[117] S. Torquato, «Hyperuniform states of matter», Physics Reports **745**, 1–95 (2018) (cit. on pp. 42, 43).

[118] A. Donev, F. H. Stillinger, and S. Torquato, «Unexpected Density Fluctuations in Jammed Disordered Sphere Packings», Physical Review Letters **95**, 090604 (2005) (cit. on p. 42).

[119] L. Berthier, P. Chaudhuri, C. Coulais, O. Dauchot, and P. Sollich, «Suppressed Compressibility at Large Scale in Jammed Packings of Size-Disperse Spheres», Physical Review Letters **106**, 120601 (2011) (cit. on pp. 42–44).

[120] R. L. Jack, I. R. Thompson, and P. Sollich, «Hyperuniformity and Phase Separation in Biased Ensembles of Trajectories for Diffusive Systems», Physical Review Letters **114**, 060601 (2015) (cit. on p. 42).

[121] E. Tjhung and L. Berthier, «Hyperuniform Density Fluctuations and Diverging Dynamic Correlations in Periodically Driven Colloidal Suspensions», Physical Review Letters **114**, 148301 (2015) (cit. on pp. 42, 105, 161).

[122] E. Tjhung and L. Berthier, «Criticality and correlated dynamics at the irreversibility transition in periodically driven colloidal suspensions», Journal of Statistical Mechanics: Theory and Experiment **2016**, 033501 (2016) (cit. on pp. 42, 105, 161, 162).

[123] D. Hexner and D. Levine, «Noise, Diffusion, and Hyperuniformity», Physical Review Letters **118**, 020601 (2017) (cit. on pp. 42, 105, 161, 162, 166, 167, 169).

[124] L. Galliano, M. E. Cates, and L. Berthier, «Two-Dimensional Crystals far from Equilibrium», Physical Review Letters **131**, 047101 (2023) (cit. on pp. 42, 105, 108, 161–166, 169).

[125] R. Maire and A. Plati, *Enhancing (quasi-)long-range order in a two-dimensional driven crystal*, May 2024 (cit. on p. 42).

[126] R. Maire, L. Galliano, A. Plati, and L. Berthier, «Hyperuniform Interfaces in Nonequilibrium Phase Coexistence», Physical Review Letters **135**, 227102 (2025) (cit. on pp. 42, 161, 162).

[127] Y. Jiao, T. Lau, H. Hatzikirou, M. Meyer-Hermann, Joseph C. Corbo, and S. Torquato, «Avian photoreceptor patterns represent a disordered hyperuniform solution to a multiscale packing problem», Physical Review E **89**, 022721 (2014) (cit. on p. 42).

[128] A. Ikeda and L. Berthier, «Thermal fluctuations, mechanical response, and hyperuniformity in jammed solids», Physical Review E **92**, 012309 (2015) (cit. on pp. 43, 106, 166, 167).

[129] N. W. Ashcroft and N. D. Mermin, *Solid state physics* (Saunders college publ, Fort Worth Philadelphia San Diego [etc.], 1976) (cit. on p. 43).

[130] F. H. Stillinger and T. A. Weber, «Hidden structure in liquids», Physical Review A **25**, 978–989 (1982) (cit. on p. 43).

[131] A. B. Bhatia and D. E. Thornton, «Structural Aspects of the Electrical Resistivity of Binary Alloys», Physical Review B **2**, 3004–3012 (1970) (cit. on p. 43).

[132] H. Tong and H. Tanaka, «Revealing Hidden Structural Order Controlling Both Fast and Slow Glassy Dynamics in Supercooled Liquids», Physical Review X **8**, 011041 (2018) (cit. on pp. 44, 45, 79, 107, 120).

[133] G. Parisi, *Theory of simple glasses: exact solutions in infinite dimensions* (Cambridge University Press, Cambridge, 2020) (cit. on p. 47).

[134] R. L. Jack, L. O. Hedges, J. P. Garrahan, and D. Chandler, «Preparation and Relaxation of Very Stable Glassy States of a Simulated Liquid», Physical Review Letters **107**, 275702 (2011) (cit. on pp. 47, 105).

[135] A. C. Maggs and W. Krauth, «Large-scale dynamics of event-chain Monte Carlo», Physical Review E **105**, 015309 (2022) (cit. on p. 47).

[136] F. Ghimenti, L. Berthier, and F. van Wijland, «Irreversible Monte Carlo Algorithms for Hard Disk Glasses: From Event-Chain to Collective Swaps», Physical Review Letters **133**, 028202 (2024) (cit. on pp. 47, 80).

[137] S. Singh, M. D. Ediger, and J. J. de Pablo, «Ultrastable glasses from in silico vapour deposition», Nature Materials **12**, 139–144 (2013) (cit. on pp. 47, 103).

[138] L. Berthier, «Revisiting the slow dynamics of a silica melt using Monte Carlo simulations», Physical Review E **76**, 011507 (2007) (cit. on p. 49).

[139] T. S. Grigera and G. Parisi, «Fast Monte Carlo algorithm for supercooled soft spheres», Physical Review E **63**, 045102 (2001) (cit. on pp. 49, 63, 66).

[140] A. Ninarello, L. Berthier, and D. Coslovich, «Models and Algorithms for the Next Generation of Glass Transition Studies», Physical Review X **7**, 021039 (2017) (cit. on pp. 49, 63, 66, 72, 103, 117, 174).

[141] L. Wang, A. Ninarello, P. Guan, L. Berthier, G. Szamel, and E. Flenner, «Low-frequency vibrational modes of stable glasses», Nature Communications **10**, 26 (2019) (cit. on p. 50).

[142] Q. Liao and L. Berthier, «Hierarchical Landscape of Hard Disk Glasses», Physical Review X **9**, 011049 (2019) (cit. on p. 50).

[143] C. Scalliet, L. Berthier, and F. Zamponi, «Nature of excitations and defects in structural glasses», Nature Communications **10**, 5102 (2019) (cit. on p. 50).

[144] G. Jung, M. Ozawa, G. Biroli, and L. Berthier, *Numerical investigation of the equilibrium Kauzmann transition in a two-dimensional atomistic glass*, July 2025 (cit. on pp. 50, 89, 90).

[145] R. Simon, J.-L. Barrat, and L. Berthier, *Molecular motion at the experimental glass transition*, June 2025 (cit. on pp. 50, 79).

[146] L. Galliano and R. Simon, *Arianna.jl*, TheDisorderedOrganization, Oct. 2025 (cit. on pp. 52, 57, 171).

[147] R. J. Williams, «Simple statistical gradient-following algorithms for connectionist reinforcement learning», Machine Learning **8**, 229–256 (1992) (cit. on pp. 55, 56).

[148] R. Rende, L. L. Viteritti, L. Bardone, F. Becca, and S. Goldt, «A simple linear algebra identity to optimize large-scale neural network quantum states», Communications Physics **7**, 260 (2024) (cit. on p. 55).

[149] P. L'Ecuyer, «A Unified View of the IPA, SF, and LR Gradient Estimation Techniques», Management Science **36**, 1364–1383 (1990) (cit. on p. 56).

[150] E. Greensmith, P. L. Bartlett, J. Baxter, and P. Com, «Variance Reduction Techniques for Gradient Estimates in Reinforcement Learning», (cit. on p. 59).

[151] S.-i. Amari, «Natural Gradient Works Efficiently in Learning», Neural Computation **10**, 251–276 (1998) (cit. on p. 60).

[152] S. M. Kakade, «A Natural Policy Gradient», in Advances in Neural Information Processing Systems, Vol. 14 (2001) (cit. on p. 60).

[153] J. Schulman, S. Levine, P. Moritz, M. I. Jordan, and P. Abbeel, *Trust Region Policy Optimization*, Apr. 2017 (cit. on pp. 60, 61).

[154] S. Amari, H. Nagaoka, S. Amari, and S. Amari, *Methods of information geometry*, trans. by D. Harada, Nachdruck, Translations of Mathematical Monographs 191 (American Mathematical Society, Providence, Rhode Island, 2007) (cit. on p. 61).

[155] S. Ruder, *An overview of gradient descent optimization algorithms*, June 2017 (cit. on p. 61).

[156] D. Wu, L. Wang, and P. Zhang, «Solving Statistical Mechanics Using Variational Autoregressive Networks», Physical Review Letters **122**, 080602 (2019) (cit. on pp. 62, 80, 81, 100).

[157] T. Marchand, M. Ozawa, G. Biroli, and S. Mallat, «Multiscale Data-Driven Energy Estimation and Generation», Physical Review X **13**, 041038 (2023) (cit. on p. 62).

[158] G. Jung, G. Biroli, and L. Berthier, «Normalizing flows as an enhanced sampling method for atomistic supercooled liquids», Machine Learning: Science and Technology **5**, 035053 (2024) (cit. on pp. 62, 82, 89, 91, 97, 99).

[159] A. D. S. Parmar, M. Ozawa, and L. Berthier, «Ultrastable metallic glasses in silico», Physical Review Letters **125**, 085505 (2020) (cit. on pp. 62, 63, 103, 117).

[160] P. J. Rossky, J. D. Doll, and H. L. Friedman, «Brownian dynamics as smart Monte Carlo simulation», The Journal of Chemical Physics **69**, 4628–4633 (1978) (cit. on p. 66).

[161] L. Berthier and W. Kob, «The Monte Carlo dynamics of a binary Lennard-Jones glass-forming mixture», Journal of Physics: Condensed Matter **19**, 205130 (2007) (cit. on p. 70).

[162] Y. S. Elmatad, D. Chandler, and J. P. Garrahan, «Corresponding States of Structural Glass Formers», The Journal of Physical Chemistry B **113**, 5563–5567 (2009) (cit. on p. 72).

[163] L. Berthier, P. Charbonneau, A. Ninarello, M. Ozawa, and S. Yaida, «Zero-temperature glass transition in two dimensions», Nature Communications **10**, 1508 (2019) (cit. on p. 72).

[164] D. Coslovich, M. Ozawa, and W. Kob, «Dynamic and thermodynamic crossover scenarios in the Kob-Andersen mixture: Insights from multi-CPU and multi-GPU simulations», The European Physical Journal E **41**, 62 (2018) (cit. on p. 72).

[165] J. P. Nilmeier, G. E. Crooks, D. D. L. Minh, and J. D. Chodera, «Nonequilibrium candidate Monte Carlo is an efficient tool for equilibrium simulation», Proceedings of the National Academy of Sciences **108**, E1009–E1018 (2011) (cit. on p. 78).

[166] S. Tamagnone, A. Laio, and M. Gabrié, *Coarse Grained Molecular Dynamics with Normalizing Flows*, July 2024 (cit. on p. 78).

[167] R. Vink, «A finite-temperature Monte Carlo algorithm for network forming materials», The Journal of Chemical Physics **140**, 104509 (2014) (cit. on p. 79).

[168] W. Krauth, «Event-Chain Monte Carlo: Foundations, Applications, and Prospects», Frontiers in Physics **9**, `10.3389/fphy.2021.663457` (2021) (cit. on p. 80).

[169] Y. Nishikawa, F. Ghimenti, L. Berthier, and F. Van Wijland, «Irreversible swap algorithms for soft sphere glasses», Physical Review E **111**, 045416 (2025) (cit. on p. 80).

[170] G. Corso, H. Stärk, B. Jing, R. Barzilay, and T. Jaakkola, *DiffDock: Diffusion Steps, Twists, and Turns for Molecular Docking*, Feb. 2023 (cit. on p. 81).

[171] T. Geffner, K. Didi, Z. Cao, D. Reidenbach, Z. Zhang, C. Dallago, E. Kucukbenli, K. Kreis, and A. Vahdat, *La-Proteina: Atomistic Protein Generation via Partially Latent Flow Matching*, July 2025 (cit. on p. 81).

[172] S. Lewis et al., «Scalable emulation of protein equilibrium ensembles with generative deep learning», Science **389**, eadv9817 (2025) (cit. on p. 81).

[173] S. Yang, K. Cho, A. Merchant, P. Abbeel, D. Schuurmans, I. Mordatch, and E. D. Cubuk, *Scalable Diffusion for Materials Generation*, June 2024 (cit. on pp. 81, 82, 99).

[174] A. Sriram, B. K. Miller, R. T. Q. Chen, and B. M. Wood, «FlowLLM: Flow Matching for Material Generation with Large Language Models as Base Distributions», (cit. on pp. 81, 82, 89, 98).

[175]P. Hoellmer et al., *Open Materials Generation with Stochastic Interpolants*, July 2025 (cit. on p. 81).

[176]K. Yang and D. Schwalbe-Koda, *A Generative Diffusion Model for Amorphous Materials*, July 2025 (cit. on pp. 82, 99).

[177]R. Jiao, W. Huang, P. Lin, J. Han, P. Chen, Y. Lu, and Y. Liu, *Crystal Structure Prediction by Joint Equivariant Diffusion*, Mar. 2024 (cit. on pp. 82, 86, 99).

[178]K. Ferguson, Y.-h. Chen, and L. B. Kara, *MDDM: A Molecular Dynamics Diffusion Model to Predict Particle Self-Assembly*, Sept. 2025 (cit. on pp. 82, 99).

[179]Z. Liu, W. Zhang, C. Schütte, and T. Li, *Riemannian Denoising Diffusion Probabilistic Models*, May 2025 (cit. on pp. 82, 99).

[180]M. Biloš and S. Günnemann, «Scalable Normalizing Flows for Permutation Invariant Densities», in Proceedings of the 38th International Conference on Machine Learning (July 2021), pp. 957–967 (cit. on p. 83).

[181]P. Wirnsberger, G. Papamakarios, B. Ibarz, S. Racanière, A. J. Ballard, A. Pritzel, and C. Blundell, «Normalizing flows for atomic solids», Machine Learning: Science and Technology **3**, 025009 (2022) (cit. on p. 83).

[182]K. Fatras, T. Séjourné, N. Courty, and R. Flamary, *Unbalanced minibatch Optimal Transport; applications to Domain Adaptation*, Mar. 2021 (cit. on p. 86).

[183]Y. Song, J. Gong, M. Xu, Z. Cao, Y. Lan, S. Ermon, H. Zhou, and W.-Y. Ma, *Equivariant Flow Matching with Hybrid Probability Transport*, Dec. 2023 (cit. on p. 86).

[184]R. Irwin, A. Tibo, J. P. Janet, and S. Olsson, *SemlaFlow – Efficient 3D Molecular Generation with Latent Attention and Equivariant Flow Matching*, Feb. 2025 (cit. on p. 86).

[185]H. W. Kuhn, «The Hungarian method for the assignment problem», Naval Research Logistics Quarterly **2**, 83–97 (1955) (cit. on p. 86).

[186]D. N. Perera and P. Harrowell, «Stability and structure of a supercooled liquid mixture in two dimensions», Physical Review E **59**, 5721–5743 (1999) (cit. on p. 89).

[187]G. Jung, G. Biroli, and L. Berthier, «Predicting Dynamic Heterogeneity in Glass-Forming Liquids by Physics-Inspired Machine Learning», Physical Review Letters **130**, 238202 (2023) (cit. on p. 89).

[188]D. P. Kingma and J. Ba, *Adam: A Method for Stochastic Optimization*, Jan. 2017 (cit. on p. 91).

[189]R. Pascanu, T. Mikolov, and Y. Bengio, *On the difficulty of training Recurrent Neural Networks*, Feb. 2013 (cit. on p. 91).

[190]J. R. Dormand and P. J. Prince, «A family of embedded Runge-Kutta formulae», Journal of Computational and Applied Mathematics **6**, 19–26 (1980) (cit. on p. 91).

[191]R. T. Q. Chen, *Torchdiffeq*, June 2021 (cit. on p. 91).

[192]E. Flenner and G. Szamel, «Hybrid Monte Carlo simulation of a glass-forming binary mixture», Physical Review E **73**, 061505 (2006) (cit. on p. 91).

[193]T. Akhound-Sadegh et al., *Iterated Denoising Energy Matching for Sampling from Boltzmann Densities*, June 2024 (cit. on p. 100).

[194]L. Richter and J. Berner, *Improved sampling via learned diffusions*, May 2024 (cit. on p. 100).

[195]A. Havens et al., *Adjoint Sampling: Highly Scalable Diffusion Samplers via Adjoint Matching*, May 2025 (cit. on p. 100).

[196]B. McNaughton, M. V. Milošević, A. Perali, and S. Pilati, «Boosting Monte Carlo simulations of spin glasses using autoregressive neural networks», Physical Review E **101**, 053312 (2020) (cit. on p. 100).

[197]L. M. D. Bono, F. Ricci-Tersenghi, and F. Zamponi, «Performance of machine-learning-assisted Monte Carlo in sampling from simple statistical physics models», Physical Review E **112**, 045307 (2025) (cit. on p. 100).

[198]D. Rehman, T. Akhound-Sadegh, A. Gazizov, Y. Bengio, and A. Tong, *FALCON: Few-step Accurate Likelihoods for Continuous Flows*, Dec. 2025 (cit. on p. 100).

[199]M. D. Parno and Y. M. Marzouk, «Transport Map Accelerated Markov Chain Monte Carlo», SIAM/ASA Journal on Uncertainty Quantification **6**, 645–682 (2018) (cit. on p. 101).

[200]S.-H. Li and L. Wang, «Neural Network Renormalization Group», Physical Review Letters **121**, 260601 (2018) (cit. on p. 101).

[201]M. S. Albergo, G. Kanwar, S. Racanière, D. J. Rezende, J. M. Urban, D. Boyda, K. Cranmer, D. C. Hackett, and P. E. Shanahan, «Flow-based sampling for fermionic lattice field theories», Physical Review D **104**, 114507 (2021) (cit. on p. 101).

[202]M. D. Ediger, «Perspective: Highly stable vapor-deposited glasses», The Journal of Chemical Physics **147**, 210901 (2017) (cit. on p. 103).

[203] S. S. Dalal and M. D. Ediger, «Molecular Orientation in Stable Glasses of Indomethacin», The Journal of Physical Chemistry Letters **3**, 1229–1233 (2012) (cit. on p. 103).

[204] K. L. Kearns, S. F. Swallen, M. D. Ediger, T. Wu, Y. Sun, and L. Yu, «Hiking down the Energy Landscape: Progress Toward the Kauzmann Temperature via Vapor Deposition», The Journal of Physical Chemistry B **112**, 4934–4942 (2008) (cit. on p. 103).

[205] C. Rodriguez-Tinoco, M. Gonzalez-Silveira, M. A. Ramos, and J. Rodriguez-Viejo, «Ultrastable glasses: new perspectives for an old problem», La Rivista del Nuovo Cimento **45**, 325–406 (2022) (cit. on p. 103).

[206] E. Flenner, L. Berthier, P. Charbonneau, and C. J. Fullerton, «Front-Mediated Melting of Isotropic Ultrastable Glasses», Physical Review Letters **123**, 175501 (2019) (cit. on p. 103).

[207] C. Herrero, C. Scalliet, M. D. Ediger, and L. Berthier, «Two-step devitrification of ultrastable glasses», Proceedings of the National Academy of Sciences **120**, e2220824120 (2023) (cit. on p. 103).

[208] L. Berthier, P. Charbonneau, E. Flenner, and F. Zamponi, «Origin of Ultrastability in Vapor-Deposited Glasses», Physical Review Letters **119**, 188002 (2017) (cit. on p. 103).

[209] G. Kapteijns, W. Ji, C. Brito, M. Wyart, and E. Lerner, «Fast generation of ultrastable computer glasses by minimization of an augmented potential energy», Physical Review E **99**, 012106 (2019) (cit. on p. 103).

[210] V. F. Hagh, S. R. Nagel, A. J. Liu, M. L. Manning, and E. I. Corwin, «Transient learning degrees of freedom for introducing function in materials», Proceedings of the National Academy of Sciences **119**, e2117622119 (2022) (cit. on p. 103).

[211] J. R. Dale, J. D. Sartor, R. C. Dennis, and E. I. Corwin, «Hyperuniform jammed sphere packings have anomalous material properties», Physical Review E **106**, 024903 (2022) (cit. on p. 103).

[212] L. Corté, P. M. Chaikin, J. P. Gollub, and D. J. Pine, «Random organization in periodically driven systems», Nature Physics **4**, 420–424 (2008) (cit. on pp. 105, 161).

[213] S. Wilken, R. E. Guerra, D. Levine, and P. M. Chaikin, «Random Close Packing as a Dynamical Phase Transition», Physical Review Letters **127**, 038002 (2021) (cit. on pp. 105, 161, 162, 165, 166).

[214] S. Anand, G. Zhang, and S. Martiniani, *Emergent universal long-range structure in random-organizing systems*, May 2025 (cit. on pp. 105, 161).

[215] P.-R. ten Wolde, M. J. Ruiz-Montero, and D. Frenkel, «Simulation of homogeneous crystal nucleation close to coexistence», Faraday Discussions **104**, 93–110 (1996) (cit. on pp. 105, 107, 169).

[216] P. G. Bolhuis, D. Chandler, C. Dellago, and P. L. Geissler, «TRANSITION PATH SAMPLING: Throwing Ropes Over Rough Mountain Passes, in the Dark», Annual Review of Physical Chemistry **53**, 291–318 (2002) (cit. on p. 105).

[217] L. O. Hedges, R. L. Jack, J. P. Garrahan, and D. Chandler, «Dynamic Order-Disorder in Atomistic Models of Structural Glass Formers», Science **323**, 1309–1313 (2009) (cit. on pp. 105, 106).

[218] L. Yao and R. L. Jack, «Heterogeneous nucleation in the random field Ising model», The Journal of Chemical Physics **159**, 244110 (2023) (cit. on pp. 105, 107).

[219] G. M. Torrie and J. P. Valleau, «Nonphysical sampling distributions in Monte Carlo free-energy estimation: Umbrella sampling», Journal of Computational Physics **23**, 187–199 (1977) (cit. on pp. 105, 106).

[220] G. M. Torrie and J. P. Valleau, «Monte Carlo free energy estimates using non-Boltzmann sampling: Application to the sub-critical Lennard-Jones fluid», Chemical Physics Letters **28**, 578–581 (1974) (cit. on pp. 105, 106).

[221] D. J. Pine, J. P. Gollub, J. F. Brady, and A. M. Leshansky, «Chaos and threshold for irreversibility in sheared suspensions», Nature **438**, 997–1000 (2005) (cit. on pp. 105, 161).

[222] D. Hexner and D. Levine, «Hyperuniformity of Critical Absorbing States», Physical Review Letters **114**, 110602 (2015) (cit. on pp. 105, 161, 163, 165).

[223] Q.-L. Lei and R. Ni, «Hydrodynamics of random-organizing hyperuniform fluids», Proceedings of the National Academy of Sciences **116**, 22983–22989 (2019) (cit. on pp. 105, 161).

[224] G. Voronoi, «Nouvelles applications des paramètres continus à la théorie des formes quadratiques. Premier mémoire. Sur quelques propriétés des formes quadratiques positives parfaites.», Journal für die reine und angewandte Mathematik (Crelles Journal) **1908**, 97–102 (1908) (cit. on p. 107).

[225] L. Berthier, D. Coslovich, A. Ninarello, and M. Ozawa, «Equilibrium Sampling of Hard Spheres up to the Jamming Density and Beyond», Physical Review Letters **116**, 238002 (2016) (cit. on pp. 108, 109, 111).

[226] D. Coslovich, M. Ozawa, and L. Berthier, «Local order and crystallization of dense polydisperse hard spheres», Journal of Physics: Condensed Matter **30**, 144004 (2018) (cit. on p. 108).

[227] E. Bitzek, P. Koskinen, F. Gähler, M. Moseler, and P. Gumbsch, «Structural Relaxation Made Simple», Physical Review Letters **97**, 170201 (2006) (cit. on p. 115).

[228] F. Ghimenti, L. Berthier, J. Kurchan, and F. van Wijland, *What do clever algorithms for glasses do? Time reparametrization at work*, Sept. 2024 (cit. on p. 117).

[229] C. Brito, E. Lerner, and M. Wyart, «Theory for Swap Acceleration near the Glass and Jamming Transitions for Continuously Polydisperse Particles», Physical Review X **8**, 031050 (2018) (cit. on p. 117).

[230] M. Shimada, D. Coslovich, H. Mizuno, and A. Ikeda, «Spatial structure of unstable normal modes in a glass-forming liquid», SciPost Physics **10**, 001 (2021) (cit. on p. 117).

[231] B. Hartke, «Global geometry optimization of clusters using a growth strategy optimized by a genetic algorithm», Chemical Physics Letters **240**, 560–565 (1995) (cit. on p. 117).

[232] D. Wales and J. Doye, «Global Optimization by Basin-Hopping and the Lowest Energy Structures of Lennard-Jones Clusters Containing up to 110 Atoms», The Journal of Physical Chemistry A **101**, 5111–5116 (1997) (cit. on p. 117).

[233] R. Kumar, G. M. Coli, M. Dijkstra, and S. Sastry, «Inverse design of charged colloidal particle interactions for self assembly into specified crystal structures», The Journal of Chemical Physics **151**, 084109 (2019) (cit. on p. 117).

[234] Z. M. Sherman, M. P. Howard, B. A. Lindquist, R. B. Jadrich, and T. M. Truskett, «Inverse methods for design of soft materials», The Journal of Chemical Physics **152**, 140902 (2020) (cit. on p. 117).

[235] P. Billingsley, *Probability and measure*, 3. ed, Wiley Series in Probability and Mathematical Statistics (Wiley, New York, NY, 1995) (cit. on p. 125).

[236] P. J. C. Spreij, «Measure Theoretic Probability» (cit. on p. 125).

[237] W. Rudin, *Real and complex analysis*, 3rd ed (McGraw-Hill, New York, 1987) (cit. on pp. 126, 128, 132).

[238]G. B. Folland, *Real analysis: modern techniques and their applications*, 2. ed, A Wiley-Interscience Publication (Wiley, New York Weinheim, 1999) (cit. on p. 133).

[239]P. Petersen, ed., *Riemannian Geometry*, Second Edition, Graduate Texts in Mathematics 171 (Springer Science + Business Media, LLC, New York, NY, 2006) (cit. on pp. 133, 135).

[240]X. Pennec, «Intrinsic Statistics on Riemannian Manifolds: Basic Tools for Geometric Measurements», Journal of Mathematical Imaging and Vision **25**, 127–154 (2006) (cit. on p. 137).

[241]S. Wilken, A. Z. Guo, D. Levine, and P. M. Chaikin, «Dynamical Approach to the Jamming Problem», Physical Review Letters **131**, 238202 (2023) (cit. on pp. 161, 162, 166).

[242]H.-D. Wang, B. Wang, Q.-L. Lei, and Y.-Q. Ma, *Anomalous Criticality of Absorbing State Transition toward Jamming*, Nov. 2025 (cit. on pp. 161, 163).

[243]M. Henkel, *Non-Equilibrium Phase Transitions: Volume 1: Absorbing Phase Transitions*, Theoretical and Mathematical Physics Ser (Springer Netherlands, Dordrecht, 2009) (cit. on pp. 161–163).

[244]R. Livi and P. Politi, *Nonequilibrium statistical physics: a modern perspective* (Cambridge university press, Cambridge, 2017) (cit. on pp. 161, 162).

[245]X. Ma, J. Pausch, and M. E. Cates, *Theory of Hyperuniformity at the Absorbing State Transition*, Oct. 2023 (cit. on pp. 161, 163).

[246]K. J. Wiese, «Hyperuniformity in the Manna Model, Conserved Directed Percolation and Depinning», Physical Review Letters **133**, 067103 (2024) (cit. on pp. 163, 165).

[247]X. Ma, J. Pausch, G. Pruessner, and M. E. Cates, «Hyperuniformity at the Absorbing State Transition: Perturbative RG for Random Organization», (cit. on pp. 163, 165).

[248]C. E. Zachary, Y. Jiao, and S. Torquato, «Hyperuniform Long-Range Correlations are a Signature of Disordered Jammed Hard-Particle Packings», Physical Review Letters **106**, 178001 (2011) (cit. on p. 166).

[249]J. Kim and S. Torquato, «Methodology to construct large realizations of perfectly hyperuniform disordered packings», Physical Review E **99**, 052141 (2019) (cit. on p. 166).

[250]C. E. Maher and S. Torquato, «Hyperuniformity scaling of maximally random jammed packings of two-dimensional binary disks», Physical Review E **110**, 064605 (2024) (cit. on p. 166).

[251]D. T. Dam, T. Kawasaki, A. Ikeda, and K. Miyazaki, *Hyperuniformity near jamming transition over a wide range of bidispersity*, July 2025 (cit. on p. 166).

[252]H. Ikeda, «Correlated noise and critical dimensions», Physical Review E **108**, 064119 (2023) (cit. on pp. 166, 167).

[253]F. D. Luca, X. Ma, C. Nardini, and M. E. Cates, «Hyperuniformity in phase ordering: the roles of activity, noise, and non-constant mobility», Journal of Physics: Condensed Matter **36**, 405101 (2024) (cit. on pp. 166, 167).

[254]P. C. Hohenberg and B. I. Halperin, «Theory of dynamic critical phenomena», Reviews of Modern Physics **49**, 435–479 (1977) (cit. on p. 167).

[255]U. C. Tauber, «Renormalization Group: Applications in Statistical Physics», Nuclear Physics B - Proceedings Supplements **228**, 7–34 (2012) (cit. on p. 167).

[256]J. Bezanson, A. Edelman, S. Karpinski, and V. B. Shah, «Julia: A Fresh Approach to Numerical Computing», SIAM Review **59**, 65–98 (2017) (cit. on pp. 171, 173, 175).

UNIVERSITÀ
DEGLI STUDI
DI TRIESTE

La borsa di dottorato è cofinanziata con risorse dell'Unione europea, NextGeneration EU - Piano Nazionale di Ripresa e Resilienza, Missione 4 – Componente 1 – Investimento 3.4 CUP J92B2200095000

Finanziato dall'Unione europea NextGenerationEU
Ministero dell'Università e della Ricerca
Italiadomani PIANO NAZIONALE DI RIPRESA E RESILIENZA
UNIVERSITÀ DEGLI STUDI DI TRIESTE