How to go with the flow: flow matching in bioinformatics and computational biology

Alex Morehead[1*†], Lazar Atanackovic[2†], Akshata Hegde[3†], Yanli Wang[3†], Frimpong Boadu[3†], Joel Selvaraj[3†], Alexander Tong[4], Aditi Krishnapriyan[5,1], Jianlin Cheng[3*]

[1*]Lawrence Berkeley National Laboratory, Berkeley, California, USA.
[2]Electrical & Computer Engineering, Vector Institute, University of Toronto, Toronto, Canada.
[3*]Electrical Engineering & Computer Science, NextGen Precision Health, University of Missouri, Columbia, Missouri, USA.
[4]Mila - Quebec AI Institute, Université de Montréal, Montréal, Canada.
[5]Computer Science & Chemical Engineering, University of California-Berkeley, Berkeley, California, USA.

*Corresponding author(s). E-mail(s): acmwhb@lbl.gov; chengji@missouri.com;
Contributing authors: l.atanackovic@mail.utoronto.ca; aghktb@missouri.edu; yw7bh@missouri.edu; fbqc9@missouri.edu; jsbrq@missouri.edu; alexander.tong@mila.quebec; aditik1@berkeley.edu;
†These authors contributed equally to this work.

Abstract

Numerous problems in bioinformatics and computational biology can be framed as a task of learning a mapping from one state of a biological system to another relevant state or to explore novel data points across biologically-constrained spaces. However, manually deriving such mappings (e.g., to transform cells in a diseased state back into a healthy state) or extrapolating from existing datasets to create new data (e.g., for molecular design) is often nontrivial and can require extraordinary domain expertise and resources. Fortunately, the field of generative artificial intelligence (AI) has introduced a new training paradigm referred to as (conditional) flow matching, which has emerged as a promising solution to this problem, with broad applicability in computer vision, natural language processing, and the physical and life sciences. Flow matching is a powerful and

principled (data-driven) framework for efficiently learning a mapping between arbitrary pairs of high-dimensional data distributions, making it well suited for addressing problems in molecular and cell biology. In this Review, we characterize the theoretical foundations of flow matching and its applications in biomolecular modeling (e.g., for proteins, DNA/RNA, small molecules, and their interactions) and single/multi-cellular modeling (e.g., for cell phenotyping and imaging), each contributing towards the development of an AI-based virtual cell. Lastly, this review highlights open-source flow matching methods and discusses future directions in flow-based generative modeling for bioinformatics and computational biology.

Keywords: Flow matching, generative AI, bioinformatics, computational biology

1 Introduction

The field of generative AI has developed numerous important applications for biological and biomedical data analysis to date. For instance, Frazer et al. [1] introduced Bayesian variational autoencoders (VAEs) [2] for evolutionary protein sequence modeling, while Cui et al. [3] created a generative pre-trained transformer (GPT)-style [4] foundation model [5] for single-cellular data. As first of their kind results, Anand and Huang [6] generated protein structures *in silico* using a deep generative adversarial network (GAN) and further experimentally validated structures generated using an autoregressive model [7]. Interestingly, applications of generative AI in bioinformatics have also adapted ideas from non-equilibrium statistical physics to develop denoising diffusion models [8] for high-quality molecule [9] and protein [10] generation and even robust translation between biomedical imaging techniques [11].

More recently, generative AI has been experiencing a surge of research interest in developing new deep learning (DL) algorithms [12] for sampling from arbitrary (e.g., biological) data distributions based on principles from optimal transport (OT) theory (Box 1). The most prominent of such new methods are based primarily on a new paradigm of generative modeling called **flow matching (FM)** [13]. At a high level, FM refers to a class of methods designed to learn a transformation that maps samples from a simple, tractable source distribution—such as a standard Gaussian—into samples from a more complex target distribution. The target distribution is typically defined by an empirical dataset and may represent intricate structures or patterns present in real-world data. The overarching goal of FM is to construct a continuous, learnable transport path that progressively aligns the source distribution with the target, facilitating efficient sample generation and enabling applications in generative modeling, density estimation, and probabilistic inference (see Box 2 for more details).

Developing powerful FM generative models could unlock several promising new application areas within bioinformatics and computational biology, including obtaining end-to-end learnable mappings between functional states of 3D (structural) biomolecules and single or multi-cellular systems. With data-driven mappings like these, the development of an AI-based virtual cell (AIVC) [14] may be within the

scientific community's grasp in the coming decade, ushering in a new era of rapid biological hypothesis generation and testing via techniques from computational molecular and cell biology. In Section 2 of this Review, we first introduce the theoretical foundations of FM and how its core techniques may be improved in the coming years. We then discuss in Section 3 how FM has already been applied within diverse biological modeling contexts to accelerate scientific discovery. Lastly, in Section 4, we provide readers with an outlook on the future of FM's role in bioinformatics.

2 Fundamentals of flow matching

In this section, we will adopt the standardized FM notation of Lipman et al. [15]. At its core, FM theory is centered on learning to transport samples between arbitrary pairs of distributions, one often referenced as a (easy-to-sample) source distribution $p_0 = p$ and the other denoted as a (empirical) target distribution $p_1 = q$, where p and q are distributions defined over $\mathbb{R}^d$. Notably, what sets FM apart from previous popular generative modeling frameworks such as diffusion models [16, 17] is that, with FM, the source distribution p_0 is not restricted to be Gaussian in nature and can instead be any distribution of one's choosing [18]. Flow matching can thus be viewed as a generalization of diffusion models. Further, one can subsequently employ optimal transport paths to considerably reduce the time required to sample new data points from the generative model. Excitingly, this opens the possibility to use FM to learn arbitrary mappings between different states of (biological) systems to accelerate data modeling and analysis in numerous scientific domains.

2.1 Designing a probability path

Concretely, to learn such generative paths for data sampling, FM builds a **probability path** $(p_t)_{0\leq t\leq 1}$, where each p_t also denotes a distribution over $\mathbb{R}^d$. With this path defined, FM theory turns to training a deep neural network to predict the **vector field** needed to convert samples from the source distribution p_0 to the target distribution p_1 along the probability path p_t. Namely, using a trained vector field network, FM enables one to generate a new data point from the target distribution $X_1 \sim q$ by randomly sampling a point from the source distribution $X_0 \sim p$ and subsequently solving the ordinary differential equation (ODE) the vector field characterizes. Specifically, with neural network parameters θ, this time-dependent vector field $u^\theta : [0,1] \times \mathbb{R}^d \to \mathbb{R}^d$ determines the time-dependent **flow** $\psi : [0,1] \times \mathbb{R}^d \to \mathbb{R}^d$ defined as:

$$\frac{\mathrm{d}}{\mathrm{d}t}\psi_t(x) = u_t^\theta(\psi_t(x)), \text{ with } \psi_t := \psi(t,x) \text{ and } \psi_0(x) = x. \tag{1}$$

With FM, we say that the vector field u_t^θ *generates* the probability path p_t if, for its corresponding flow ψ_t, the following statement is satisfied:

$$X_t := \psi_t(X_0) \sim p_t \text{ for } X_0 \sim p_0. \tag{2}$$

Now, we proceed to design our probability path p_t. For the sake of notational simplicity, let our source distribution $p := p_0 = \mathcal{N}(x|0, I)$ denote a standard Gaussian. The probability path p_t is then the aggregation of conditional probability paths

$p_{t|1}(x|X_1)$ which are *conditioned* on the training dataset's (ground-truth) data points X_1. With $p_{t|1}(x|X_1) = \mathcal{N}(x|tX_1, (1-t)^2 I)$, this means we can express p_t as the *conditional optimal transport path*:

$$p_t(x) = \int p_{t|1}(x|X_1) q(X_1) \mathrm{d}X_1 \tag{3}$$

which allows one to define the random variable $X_t \sim p_t$ simply via the linear combination:

$$X_t = tX_1 + (1-t)X_0. \tag{4}$$

2.2 Defining a training objective

After designing the probability path p_t, we now turn to defining the training objective of our vector field network u_t^θ. The **standard** FM **loss** is defined as:

$$\mathcal{L}_{\mathrm{FM}}(\theta) = \mathbb{E}_{t,X_t} \|u_t^\theta(X_t) - u_t(X_t)\|^2, \text{ with } t \sim \mathcal{U}[0,1] \text{ and } X_t \sim p_t. \tag{5}$$

Unfortunately, calculating the ground-truth vector field $u_t(X_t)$ is computationally intractable, as doing so requires integrating over all possible transformations between two high-dimensional distributions. Instead, in each training iteration, we can condition the loss on a single random example X_1 from the training dataset. To this end, we first reference Equation 4 to define the conditional random variables $X_{t|1}$ as:

$$X_{t|1} = tX_1 + (1-t)X_0, \text{ where } X_0 \sim p_{t|1}(\cdot|X_1) = \mathcal{N}(\cdot|tX_1, (1-t)^2 I). \tag{6}$$

Then, solve for X_0 in Equation 6 as $X_0 = \frac{X_{t|1} - tX_1}{1-t}$. We then calculate the (tractable) derivative $u_t(X_{t|1}|X_1) = \frac{\mathrm{d}}{\mathrm{d}t} X_{t|1} = \frac{\mathrm{d}}{\mathrm{d}t}(tX_1 + (1-t)X_0) = X_1 - X_0$ which, after substituting X_0 to obtain $u_t(X_{t|1}|X_1) = X_1 - \frac{X_{t|1} - tX_1}{1-t}$, simplifies to the following **conditional vector field**:

$$u_t(X_{t|1}|X_1) = \frac{X_1 - X_{t|1}}{1-t}. \tag{7}$$

Notably, this conditional vector field *generates* the conditional probability path $p_{t|1}(\cdot|X_1)$, which subsequently yields the following **conditional** FM **loss**:

$$\mathcal{L}_{\mathrm{CFM}}(\theta) = \mathbb{E}_{t,X_1,X_{t|1}} \|u_t^\theta(X_{t|1}) - u_t(X_{t|1}|X_1)\|^2, \tag{8}$$

where $X_{t|1} = tX_1 + (1-t)X_0$. From a practical perspective, previous works [13, 18] importantly have demonstrated the equality of the following gradients of Equations 5 and 8:

$$\nabla_\theta \mathcal{L}_{\mathrm{FM}}(\theta) = \nabla_\theta \mathcal{L}_{\mathrm{CFM}}(\theta), \tag{9}$$

which allows one to optimize the vector field network u_t^θ for both training objectives simultaneously (n.b., the first intractable and the second tractable). Lastly, if we consider the Gaussian flow $\psi_t(x) = \mu_t(X_1) + \sigma_t(X_1)x = tX_1 + (1-t)x$, let

$X_{t|1} = X_0$, and solve $u_t(X_{t|1}|X_1) = \frac{\mathrm{d}}{\mathrm{d}t}\psi_t(X_{t|1}) = \frac{\mathrm{d}}{\mathrm{d}t}\psi_t(X_0) = X_1 - X_0$, one can plug $u_t(X_{t|1}|X_1)$ of Equation 7 into Equation 8 to derive the following **Gaussian**-based **optimal transport** FM **loss** [15]:

$$\mathcal{L}_{\text{CFM}}^{\text{OT, Gauss}}(\theta) = \mathbb{E}_{t,X_0,X_1}\|u_t^\theta(X_t) - (X_1 - X_0)\|^2, \tag{10}$$

where $X_0 \sim \mathcal{N}(0, I)$.

2.3 Sampling with a learned vector field

To sample a new data point X_1 from a (random) easy-to-obtain starting point X_0, one can use the trained vector field network u_t^θ to solve the corresponding *push-forward* ODE. Namely, using a simple update rule of the *Euler method*:

$$X_{t+h} = X_t + hu_t^\theta(X_t), \text{ where } h = n^{-1} > 0 \text{ determines the step size,} \tag{11}$$

one can arrive at a novel sample X_1 in a fixed number of steps $n \in \mathbb{N}$ iteratively. Note that many other ODE solvers can be substituted for the Euler method [19]. In particular, many prior works on FM adopt the Dormand–Prince (DOPRI5) method as a common ODE solver for generative modeling due to its empirically-improved performance compared to an Euler solver.

2.4 Improving flow matching models

While FM models offer an elegant and theoretically grounded framework for generative modeling, several challenges must be addressed to improve their empirical performance, particularly in bioinformatics applications involving high-dimensional, structured, and multimodal data. To this end, recent work has introduced multiple avenues for enhancement. For example, Rectified Flows [20] solve a nonlinear least squares optimization problem to improve the sampling speed of FM, in some cases enabling high-quality sample generation in a single ODE solver step. Moreover, Generator Matching [21] provides a theoretical unification of various diffusion and FM methods, which may yield new multi-modal applications of FM models in future work. See Table 1 for a comprehensive listing of open-source methods for implementing and improving FM models.

3 Applications of flow matching in bioinformatics

In this section, we outline several applications of FM in bioinformatics, in particular those that contribute to the development of an AIVC for multi-scale biological simulation and modeling (see Figure 1), which we posit will spark important new research directions in flow-based generative AI for molecular and cellular biology in the coming years. Towards this end, Section 3.1 characterizes FM applications for molecular modeling, while Sections 3.2, 3.3, and 3.4 describe recent applications of FM in single/multi-cellular modeling and bioimaging, respectively. To complement the following sections, Table 2 lists the latest FM methods for bioinformatics. For a

historical timeline of the development of FM methods and their applications in bioinformatics, please see Figure 2. Further, for practical advice on how to build and apply FM models in bioinformatics applications, please see Box 3.

3.1 Flow matching for molecular modeling

In this section, we discuss selected FM generative models for 3D biomolecules, as listed in Table 2, encompassing small molecules, peptides, proteins, RNAs and DNAs. Towards this end, we begin by introducing readers to the notion of representing geometric data types with generative models.

3.1.1 Introduction to geometric generative modeling

In three-dimensional space, a *rigid body* is an object whose shape and internal distances remain unchanged during motion. The **configuration** of a rigid body is fully specified by its **position** and **orientation**. The position describes the location of a reference point (e.g., the center of mass) in space, while the orientation defines how the body is rotated about that point relative to a fixed reference frame. In some contexts, rigid bodies also exhibit intrinsic **handedness** or **chirality**, a structural property preserved under translation and rotation but inverted by reflection.

A rigid body's configuration can be modified using three fundamental operations: **translation**, which moves the body through space without changing its orientation or shape; **rotation**, which changes the body's orientation by rotating it about an axis through a fixed point; and **reflection**, which flips the body across a plane, reversing its handedness. However, reflections are not physically realizable for many biological systems due to fixed chirality.

These geometric transformations correspond to structured mathematical spaces known as **manifolds**. A manifold is a topological space that locally resembles Euclidean space and allows for continuity and differentiability [22–24]. This smooth structure enables the definition of derivatives, such as velocity and acceleration, making the motion of rigid bodies mathematically tractable. The configuration space of a rigid body also has an algebraic structure known as a **group**, where the group operation corresponds to composing two motions into a single equivalent motion. When a group is also a smooth manifold with smooth group operations, it is called a **Lie group**. Examples of Lie groups relevant to rigid bodies include $SO(3)$, which represents rotations, and $SE(3)$, which represents full rigid motions (both translation and rotation) [15, 23, 24].

For rigid body configurations in 3D, the configuration space combines translations and rotations: translations span $\mathbb{R}^3$, a 3-dimensional Euclidean manifold, while rotations lie on $SO(3)$, a compact 3-dimensional manifold topologically equivalent to a 3-sphere with antipodal points identified. Together, the full configuration space is the product manifold $SE(3) = SO(3) \ltimes \mathbb{R}^3$, a 6-dimensional Lie group representing all possible positions and orientations of a rigid body [15, 23, 24]. As a smooth manifold, $SE(3)$ can also be equipped with a Riemannian metric, turning it into a **Riemannian manifold** that supports definitions of angles, distances, and geodesics — essential for modeling flows and probability distributions on $SE(3)$.

This geometric framework is particularly relevant for modeling biological macromolecules such as proteins. Proteins are chains of amino acids that fold into specific three-dimensional structures, which determine their biological functions. Each amino acid residue can be treated as a rigid body with a well-defined position and orientation in space [22–25]. In structural data formats (e.g., PDB files), the position of a residue is specified by the (x, y, z) coordinates of a reference atom (commonly the C_α) in a global Cartesian coordinate system. The orientation of each residue is described as a rotation relative to the same reference frame, using a local rigid frame derived from backbone atoms. The origin $(0, 0, 0)$ of the global frame is arbitrary and fixed by convention; the protein does not need to be centered there unless explicitly adjusted during preprocessing. Proteins also exhibit chirality due to the stereochemistry of their L-amino acids and right-handed α-helices. This handedness is intrinsic and preserved in modeling, since applying reflections would violate stereochemical constraints and produce biologically invalid structures. The full rigid motion of each residue — combining both position and orientation — is mathematically represented as an element of $SE(3)$, and the entire protein backbone can be described as a sequence of frames $\{(\mathbf{t}^{(n)}, R^{(n)})\}_{n=1}^{N} \in SE(3)^N$.

Among all possible manifolds, $SE(3)$ is the natural space for modeling protein backbones because it encodes all possible rigid motions in 3D space, is a 6-dimensional Lie group with well-defined geometry and symmetries, supports equivariant modeling (i.e., global rigid motions of the protein preserve structural validity), and as a Riemannian manifold, allows the use of gradients, geodesics, and optimal transport methods [22–25].

Generative modeling of protein backbones involves learning probability distributions over sequences of rigid body configurations, which naturally live in $SE(3)^N$ [10, 16, 26]. To generate valid and diverse protein structures, the generative model must respect the manifold structure, symmetries, and smoothness of $SE(3)$. Traditional diffusion-based generative models have shown success in Euclidean spaces but exhibit limitations when applied to manifolds like $SE(3)$. Diffusion models introduce stochastic noise and then learn to reverse the noising process iteratively, which can require thousands of inference steps and may struggle with manifold constraints unless carefully adapted.

Flow matching addresses these limitations by directly learning a time-dependent vector field that transports points from a simple prior distribution toward the data distribution along continuous paths on the manifold [15, 27]. The learned vector field respects the geometry of $SE(3)$, since it operates in the tangent space at each point, and the training objective can explicitly incorporate the Riemannian metric. This results in smooth, deterministic flows that remain on the manifold throughout the generation process. Compared to diffusion-based models, FM offers four key advantages: (1) it requires fewer inference steps while maintaining sample quality; (2) its implementations are often simpler than their diffusion-based counterparts; (3) it offers explicit control over distribution couplings; and (4) it does so while preserving the (e.g., equivariant) properties of the underlying data manifold.

3.1.2 Generative modeling of proteins

Designing proteins that jointly satisfy geometric fidelity, physicochemical realism, and functional constraints is a substantially complex task, owing to proteins' large conformational space and intertwined sequence–structure dependencies. Traditional diffusion-based models, such as RFdiffusion, have made headway but remain computationally expensive and often decouple sequence from structure. Recent advances in FM provide a unifying and faster alternative: by learning continuous velocity fields on the product manifold $\mathrm{SE}(3)^N$ (for backbones) or on mixed discrete–continuous spaces (for sequences and torsions), these models transform isotropic noise into valid protein configurations in a handful of integration steps. Because $SE(3)^N$ naturally represents the space of protein backbones, where each residue is modeled as a rigid body in $SE(3)$, FM is a principled and efficient method for learning generative models that respect both the geometry of individual residues and the collective symmetries of protein structure.

In this subsection, we survey the state of the art—FrameFlow [22], FoldFlow [23, 24], FlowPacker [25], Multiflow [28], CoFlow [29], ProtFlow [30], OriginFlow [31], D-Flow [32], and FlowDesign [33]—highlighting how each leverages equivariant architectures, optimal-transport objectives, and discrete-state flows to accelerate backbone generation, side-chain packing, and holistic co-design while improving diversity, accuracy, and experimental success rates.

FrameFlow: Fast Protein Backbone Generation via SE(3) Flow Matching

FrameFlow is a generative method for fast protein backbone generation using SE(3) FM. It builds upon FrameDiff [22], a diffusion-based model that generates three-dimensional protein backbones in the frame representation, where each residue is encoded as a local SE(3) frame specifying its position and orientation. This representation respects the geometric structure of backbones and achieves state-of-the-art performance in *de novo* design.

Protein structures naturally reside on the manifold $\mathrm{SE}(3)^N$, representing N independent rigid-body transformations. In FrameDiff, generation is formulated as a stochastic diffusion process on $\mathrm{SE}(3)^N$, with Gaussian noise added to translations and Brownian motion to rotations. The model learns to reverse this process by iteratively denoising with a learned score function.

FrameFlow improves efficiency by replacing stochastic diffusion with *flow matching* [13]. Instead of stepwise denoising, it learns a continuous, time-dependent vector field that deterministically transports noisy samples to clean data along geodesics of SE(3). This vector field is trained using conditional FM, where closed-form geodesic interpolations are computed via exponential and logarithmic maps for rotations and linear interpolation for translations.

The metric on SE(3) combines inner products on SO(3) and $\mathbb{R}^3$, treating rotations and translations independently but coherently. The prior distribution over noisy samples is uniform over rotations and Gaussian over translations, ensuring SE(3)-invariance. Training minimizes a time-scaled loss between predicted and true clean frames at each intermediate time t, guiding samples along the correct geodesic path.

To preserve geometric consistency and equivariance, FrameFlow adopts an architecture based on FrameDiff [22], which combines Invariant Point Attention (IPA) layers and transformers. It imputes oxygen atoms from backbone geometry, following RFdiffusion [10], and retains other hyperparameters and self-conditioning from FrameDiff.

Several enhancements improve stability and efficiency: weighting the rotation loss to match translation scale, clipping the scaling factor at $t \approx 1$ to avoid divergence, using an alternative rotational prior during training to avoid degenerate geodesics, pre-aligning noisy and clean samples to remove global rotations, and applying an exponential schedule for rotations during inference.

Overall, FrameFlow learns an equivariant vector field on SE(3) that moves noisy samples to clean configurations along geodesics. It achieves significant speedup and improved designability over FrameDiff, while preserving diversity and novelty, making it a strong candidate for scalable *de novo* protein design.

FoldFlow: SE(3)-Stochastic Flow Matching for Protein Backbone Generation

Bose et al. [23] [24] introduce FoldFlow, a family of generative models for protein backbone design that operate over the space of 3D rigid body motions, formalized as the Lie group SE(3). Leveraging the FM paradigm [34], FoldFlow learns to generate protein structures by modeling continuous-time dynamics on this geometric space. The goal is to learn a probability density ρ_t over the space of protein backbone configurations, which are invariant to global translation and rotation, and to sample novel, physically plausible backbones from this distribution.

Configuration space. Each protein backbone consists of N residues. Each residue is represented as a rigid body in 3D, specified by a translation and a rotation (mathematically an element of SE(3), the group of rigid motions). The full configuration of the backbone lives in $\mathrm{SE}(3)^N$, the Cartesian product of N copies of SE(3). Since only the internal shape of the backbone (relative positions and orientations of residues) matters, and not its absolute position in 3D space, the model centers each backbone by subtracting its center of mass. This projects the configurations into the translation-invariant subspace $\mathrm{SE}(3)^N_0$, where the center of mass is fixed at the origin.

SE(3) can be written as a semidirect product of rotations (SO(3)) and translations ($\mathbb{R}^3$), with a natural Riemannian metric. This metric enables the use of the logarithmic map, which projects points on the manifold to the tangent space at a reference point, locally flattening the manifold, and the exponential map, which maps tangent vectors back onto the manifold. The $\mathrm{SE}(3)^N_0$ product space retains a product Riemannian structure, enabling separate per-residue flows and independent treatment of rotations and translations. The main paper focuses heavily on modeling **rotations** (SO(3)), since flows in $\mathbb{R}^3$ are relatively straightforward and have already been explored in prior work [13, 18, 27]. The treatment of translations is detailed in the supplementary material.

Model variants. The paper introduces three variants of FoldFlow, in increasing order of expressiveness and robustness:

- **FoldFlow-Base.** Models deterministic flows on SO(3) by constructing conditional paths between source (ρ_0) and target (ρ_1) rotations. For each pair (r_0, r_1), the path is constructed along the geodesic on SO(3). The learned velocity field matches the geodesic velocity, computed using the logarithmic map and parallel transport to the current tangent space. While simple and principled, these paths are not globally optimal over the distribution space and can lead to instability in training.
- **FoldFlow-OT.** Improves on FoldFlow-Base by constructing conditional paths using optimal transport (OT). It solves the Monge–Kantorovich OT problem to find the globally shortest (Wasserstein) path between the distributions ρ_0 and ρ_1 under the Riemannian metric. This results in conditional paths that are shorter, straighter, and more stable, with lower variance during training while retaining deterministic dynamics.
- **FoldFlow-SFM.** Further improves the model by introducing stochastic flows using stochastic differential equations (SDEs). Deterministic flows (Base and OT) tend to stay too close to the support of the training data and may struggle in high-dimensional spaces. SDE-based flows, such as Brownian bridges, add controlled noise, improving exploration of the configuration space and enabling generation of novel yet plausible backbones. In $\mathbb{R}^3$, Brownian bridges have closed-form conditional distributions, while in SO(3) they are approximated by backward simulation or isotropic Gaussians. This simulation-free approximation, using isotropic Gaussians, enhances training efficiency while maintaining robustness. The stochastic flows improve diversity and robustness at the cost of slightly higher computational complexity.

Translations in $\mathbb{R}^3$ (detailed in the supplementary material of the paper). FoldFlow models translations in $\mathbb{R}^3$ using conditional Gaussian probability paths, where each path interpolates linearly between a standard Gaussian prior and the target sample, with variance scaled over time. To improve path optimality, the authors incorporate optimal transport-inspired flows using McCann interpolants. These interpolants correspond to straight-line interpolations between pairs of samples under the optimal transport plan, minimizing the global transportation cost and producing shorter, more stable paths. This improves both training and inference efficiency. Rotational and translational flows are then combined to generate complete residue configurations in $\mathrm{SE}(3)_0^N$.

Experimental results. FoldFlow was evaluated on synthetic SO(3) distributions and real protein backbone design tasks. On synthetic data, all variants accurately captured multimodal distributions, with FoldFlow-OT achieving the lowest Wasserstein distances. This confirms that optimal transport improves path quality. Additionally, FoldFlow-SFM matched simulated SDE performance while being more efficient. On PDB data, FoldFlow outperformed FrameDiff across all metrics, achieving higher designability, diversity, and novelty. FoldFlow-OT produced the most designable structures, while FoldFlow-SFM generated the most novel backbones, demonstrating the benefits of optimal transport and stochasticity. FoldFlow also achieved substantially faster and more efficient inference than baselines, closing much of the gap to RFDiffusion [10] despite using smaller models and fewer resources.

FlowPacker: Protein Side-Chain Packing with Torsional Flow Matching

Prior to this method, we have discussed approaches to generate the full 3D structures of proteins. Another important aspect of protein structure is the configuration of side-chains, which are crucial for determining how the protein folds, interacts with other molecules, and performs its biological function. Predicting the side-chain conformations, also known as side-chain packing, is a key subproblem in structural bioinformatics. This paper, FlowPacker [25], addresses the side-chain packing problem using an FM approach.

Historically, side-chain packing has been tackled using physics-based models that rely on empirical scoring functions, rotamer libraries (collections of statistically frequent side-chain conformations observed in experimentally determined structures), and stochastic search methods. However, these approaches tend to be slow, prone to getting stuck in local minima, and often produce suboptimal results.

DiffPack [35] introduced a diffusion-based generative model that autoregressively predicts side-chain torsion angles through a denoising process. While effective, this method relies on a stochastic diffusion process. FlowPacker improves on DiffPack by replacing the stochastic diffusion process with FM, which provides a more efficient and deterministic generative modeling paradigm based on torsional FM. This is further enhanced by combining it with an equivariant graph neural network (EquiformerV2 [36]).

The model learns a continuous vector field that maps noisy torsion angles to their clean ground-truth configurations along geodesic paths on the torus manifold, which is the natural space for angular variables. This enables simulation-free training and significantly faster inference while maintaining or improving generation accuracy. The vector field is trained using a conditional FM loss, which is computed using closed-form exponential and logarithmic maps tailored for angular data.

To incorporate geometric context, FlowPacker uses EquiformerV2, an equivariant graph attention network that operates on tensor-valued features. This architecture preserves the three-dimensional symmetries of protein structures and improves model expressivity without increasing parameter count.

FlowPacker outperforms prior side-chain packing methods in both accuracy and runtime. It supports partial inpainting, where only the missing side chains are predicted while the others are kept fixed. It also generalizes effectively to multimeric complexes, including antibody–antigen interfaces.

The training procedure includes a vector field regression objective and explicitly handles angular symmetries, such as the threefold symmetry of methyl groups, by reducing angles to equivalent intervals. Notably, the model achieves robust performance without requiring higher-order equivariant tensors, which makes it an efficient and scalable solution for full-atom protein structure modeling.

MultiFlow: Generative Flows on Discrete State-Spaces: Enabling Multimodal Flows with Applications to Protein Co-Design

Designing proteins involves both discrete data (amino acid sequences) and continuous data (3D structures), yet most generative models only handle one or the other. Campbell et al. [28] address this by introducing **MultiFlow Discrete Flow Models**

(DFMs), a framework that enables flow-based generation over discrete spaces, and pairs it with standard flows to jointly model sequence and structure.

The architecture combines two key components:

- A **Discrete Flow Module**, based on *Continuous-Time Markov Chains (CTMCs)*, models transitions between categorical states such as amino acids.
- A **Continuous Flow Module** generates 3D atomic coordinates using standard FM techniques.

These components are trained together to learn a shared latent space that supports flexible generation: sequences from structures, structures from sequences, or both from noise. The joint training encourages alignment between the modalities and enables efficient, multimodal sampling.

In protein co-design benchmarks, DFMs achieve **state-of-the-art performance**, producing chemically valid sequences and stable 3D structures with high diversity and fast inference. By uniting discrete and continuous flows in one architecture, this work presents a scalable and practical solution for complex biological generation tasks.

CoFlow: Co-design protein sequence and structure in discrete space via generative flow

In protein design, co-design—the simultaneous generation of both protein sequence and structure—is becoming increasingly vital in protein engineering. Traditional two-step approaches, which generate either the structure or sequence first, fall short by neglecting the deep interdependence between the two. While recent models like ESM-3 [37] attempt to address this by jointly modeling sequence and structure, they often struggle with *unconditional generation*, producing outputs that are not always physically plausible.

CoFlow [29] presents a novel solution by introducing a *discrete flow model* grounded in continuous-time Markov chains (CTMCs). This approach begins with a fully masked protein and incrementally reveals both the sequence and structure in a coordinated, stepwise manner. In contrast to conventional diffusion-based models, CoFlow's framework is notably more stable, computationally efficient, and amenable to user control—qualities that are particularly advantageous for directed protein design.

At the core of CoFlow is a bidirectional Transformer architecture augmented with Fourier-based temporal embeddings. These features endow the model with temporal awareness during the generation process, enabling it to maintain coherence between sequence and structure predictions across time steps.

Importantly, CoFlow is not limited to a single application. Its versatility allows it to support a range of protein design tasks, including:

- **Unconditional generation:** Designing novel proteins from scratch
- **Folding:** Predicting structure from a known sequence
- **Inverse folding:** Predicting sequence from a known structure
- **Motif scaffolding:** Completing partial structures around functional motifs

In summary, CoFlow offers a significant paradigm shift in protein design by treating sequence and structure as a unified, co-evolving system. Through its temporally controlled, bidirectional generation process, it produces designs that are not only physically realistic but also highly applicable to real-world protein engineering challenges.

ProtFlow: Fast Protein Sequence Design via Flow Matching on Compressed Protein Language Model Embeddings

ProtFlow [30] offers a fresh direction in protein design by combining speed, accuracy, and structural awareness in one unified framework. It integrates three key components: rich semantic embeddings from the protein language model **ESM-2** [38], a lightweight compressor that reduces embedding size by 16× for efficiency, and a FM-based generator that learns the optimal path from Gaussian noise to realistic protein representations. Using a technique called *Reflow* [20], ProtFlow generates complete protein sequences in a single step.

This approach marks a significant shift from traditional diffusion-based models like **EvoDiff** [39], which require hundreds of sampling steps. Instead, ProtFlow uses a direct ODE path in latent space, drastically reducing inference time while preserving structure quality and alignment with natural protein distributions.

ProtFlow performs strongly across diverse design tasks. It outperforms models like **DiMA** [40], **EvoDiff** [39], and **ProteinGAN** [41] in general protein and peptide generation, scoring higher on pLDDT, perplexity, and distribution metrics. For antimicrobial peptides, it ranks first in Pamp and Pmic scores and excels at generating peptides effective against *E. coli*. It is also the first model to support joint antibody design, successfully modeling both heavy and light chains. Trained on 1.4 million antibody pairs, it surpasses **GPT-3.5** [42], **ESM-2**, and **dWJS** [43] on 15 structure-property metrics.

What makes ProtFlow especially promising is how it combines the speed of autoregressive models with the distributional power of diffusion models. It delivers fast, one-step generation with globally consistent outputs—enabled by the semantic depth of pretrained pLMs.

With its unique balance of efficiency, reliability, and biological insight, ProtFlow signals the next generation of protein design tools. It offers a scalable and practical solution for antibody engineering, peptide therapeutics, and AI-powered bioengineering.

OriginFlow: Robust and Reliable de novo Protein Design A Flow-Matching-Based Protein Generative Model Achieves Remarkably High Success Rates

OriginFlow [31] is a new *de novo* protein generation model that uses **Flow Matching** (optimal transport path learning) to directly generate target protein structures from Gaussian noise. Compared to traditional diffusion models like **RFdiffusion** [10] or **Chroma** [44], OriginFlow achieves faster convergence, lower training cost, and stronger structural controllability.

The model supports a wide range of design tasks, including unconditional structure generation, functional motif embedding, symmetric assembly, multi-chain complex design, and binder generation. Its architecture integrates Invariant Point Attention **IPA**, **graph neural networks (GNNs)**, and structure-aware attention, and it handles **SE(3)** transformations (rotation and translation) to accommodate proteins of various lengths and topologies.

One of OriginFlow's most impressive results is its high binder design success rate—a longstanding challenge in AI-based protein design. While models like **AlphaProteo** [45] show variable success (5–80%) and **RFdiffusion** relies mostly on simulations, **OriginFlow achieves up to 90% experimental success**, making AI-designed binders practically usable for the first time.

This marks more than a technical milestone—it signals a shift in how proteins can be designed and validated. By reducing the failure rate between structure generation and functional testing, OriginFlow enables:

- **Low-throughput experiments** to replace expensive high-throughput screening;
- A streamlined **"generation → validation"** workflow for binder development;
- **Real integration of AI** into biopharma pipelines.

From **AlphaFold** [46] to **OriginFlow**, we're seeing the evolution from structure prediction to functional design. OriginFlow delivers a breakthrough in controllability, diversity, and expressibility—bringing AI protein design one step closer to real-world application. If AlphaFold unlocked nature's protein code, OriginFlow may be the toolbox that helps AI build the next generation of biological functions.

D-Flow: Multi-modality Flow Matching for D-peptide Design

D-Flow [32] is a new framework for *de novo* D-peptide design using D-amino acids. Since most natural proteins and peptides are made from L-amino acids, generating D-peptides remains a challenge for deep learning due to limited data.

D-Flow addresses this by supporting both unconditional protein design pretrained on the PDB [47] dataset with FM and receptor-conditioned D-peptide generation. Benchmarks show that D-Flow performs especially well in designing peptides composed entirely of D-residues.

What sets D-Flow apart are three key innovations:

- **Multi-flow architecture:** Four flow models handle sequence and structural aspects, including residue positions, rotations, and torsion angles.
- **Two-stage training:** It is first pretrained on the PDB dataset without conditioning, then fine-tuned using receptor-specific data. A mirror-image algorithm helps overcome the lack of D-protein data.
- **Structural adapter:** A lightweight attention-based module integrates insights from protein language models (PLMs) and Invariant Point Attention (IPA), effectively capturing both sequence and structural features.

Collectively, D-Flow offers a powerful solution to D-peptide design by combining flow-based modeling, structural awareness, and mirror-image augmentation—paving the way for discovering therapeutic peptides that target previously undruggable sites.

FlowDesign: Improved Design of Antibody CDRs Through Flow Matching and Better Prior Distributions

Designing antibody CDRs is challenging due to their variability and the need to jointly model discrete sequences and continuous structures. While diffusion models have advanced this field, they are often slow, struggle with discrete amino acids, and rely on simplistic Gaussian priors. **FlowDesign** [33] overcomes these issues by using a FM framework that enables fast sampling, flexible priors, and direct discrete sequence modeling.

The architecture combines an **SE(3)-equivariant structural encoder** with a **discrete flow-based sequence decoder**. Unlike diffusion's iterative denoising, FlowDesign learns a velocity field that maps noise to data in a single step. Discrete FM applied to sequences avoids approximation, and learned priors on CDR structures make generation more data-driven and biologically realistic.

FlowDesign outperforms existing models in benchmarks, achieving higher sequence recovery, lower RMSD, and better energy scores. In real-world tests, it designed HIV-targeting antibodies that bind more strongly and neutralize a broader range of variants than a clinical antibody, confirmed through BLI and pseudovirus assays.

Overall, FlowDesign demonstrates how flow-based generative modeling can deliver faster, more accurate, and biologically meaningful antibody design.

3.1.3 Generative modeling of small molecules

Designing 3D small molecules with high structural fidelity and chemical validity is a central challenge in computational chemistry and drug discovery. Traditional generative approaches—such as graph-based autoencoders and diffusion models—have demonstrated strong performance but often suffer from limited scalability, slow sampling, or difficulty in handling the discrete-continuous nature of molecular data. Recent advances in FM offer a promising alternative, enabling efficient and accurate one-shot generation by learning continuous velocity fields that map noise to molecular structures. In this subsection, we explore several state-of-the-art models—SemlaFlow [48], FlowMol [49], ADiT [26], and ET-Flow [50]—which integrate geometric deep learning, categorical feature modeling, and symmetry-aware architectures to push the boundaries of fast and faithful 3D molecular generation.

SemlaFlow: Efficient 3D Molecular Generation with Latent Attention and Equivariant Flow Matching

Generating 3D molecules is essential for tasks like drug discovery, but most current models are either slow—requiring hundreds of sampling steps—or struggle to produce chemically valid structures. **SemlaFlow** [48] tackles this by combining speed and accuracy, generating high-quality molecules in as few as **20 steps**, while maintaining strong performance on standard benchmarks.

At the heart of the method is the **Semla architecture**—a message-passing neural network that's *E(3)-equivariant* (meaning it respects 3D symmetries) and uses *latent attention*. Instead of processing atom interactions at full scale, Semla compresses the messages into a lower-dimensional latent space and uses multi-head attention to

focus on the most important relationships. This makes the model much more efficient without losing detail.

For the generative process, SemlaFlow uses **equivariant FM**, learning a smooth velocity field to transform noise into valid molecular structures. It also introduces a *scale-optimal transport* regularizer, which keeps atomic movements realistic and stable during generation.

The results are impressive: SemlaFlow achieves **state-of-the-art performance** on datasets like QM9 and GEOM-Drugs, while being up to **100× faster** than traditional models. On top of that, the authors introduce more meaningful evaluation metrics based on molecular energy and chemical stability—areas where SemlaFlow also excels.

In short, SemlaFlow is a smart combination of geometric learning, efficient architecture, and fast generative modeling. It shows that with the right design choices, 3D molecular generation can be both high-quality and lightning fast.

FlowMol: Mixed Continuous and Categorical Flow Matching for 3D De Novo Molecule Generation

Generating 3D molecular structures involves producing both accurate atomic coordinates and discrete chemical properties, such as atom and bond types. While diffusion models have recently become state-of-the-art for this task, their iterative sampling procedures are computationally expensive. In contrast, FM offers a more efficient alternative by learning a direct mapping from random noise to a complete molecule in a single step. However, FM is inherently designed for continuous data, whereas molecules exhibit a combination of continuous (e.g., 3D positions) and categorical (e.g., atom types) features. Bridging this modality gap is the central challenge addressed in the paper.

To handle categorical data, Dunn and Koes [49] explore two strategies. The first, called **SimplexFlow**, embeds discrete categories into smooth probability distributions over a simplex, allowing FM to operate in a continuous domain. Though mathematically elegant, this approach complicates training and reduces flexibility. The second method is simpler and more practical: during training, categorical variables are treated as continuous with added noise; at generation time, the outputs are projected back to the nearest valid category. Surprisingly, this approach outperforms the more complex alternative.

These methods are unified in a model named **FlowMol** [49], which represents molecules as graphs, with atoms as nodes and bonds as edges. The architecture includes *Molecule Update Blocks* that iteratively refine atom features, 3D coordinates, and bond types. To ensure rotational equivariance—the property that the model's outputs rotate consistently with its inputs—FlowMol employs *Geometric Vector Perceptrons* (GVPs). It also incorporates specific mechanisms to handle molecular chirality, a critical factor in chemical functionality.

FlowMol is trained using *optimal transport* to align atoms between noisy inputs and target molecules, thereby learning a smooth velocity field that transforms noise into valid molecular structures in a single pass. As a result, FlowMol achieves generation quality on par with diffusion models while being more than ten times faster.

In summary, FlowMol provides a fast, flexible, and geometrically principled framework for 3D molecular generation, effectively bridging the gap between continuous and categorical data.

ADiT: All-atom Diffusion Transformers: Unified generative modelling of molecules and materials

All-atom Diffusion Transformers (ADiT) [26] propose a unified solution to a longstanding challenge in generative modeling: simultaneously addressing molecular and material structure generation. While molecules and crystalline materials both consist of atoms in three-dimensional space, prior generative models have typically treated them separately, relying on distinct architectures and domain-specific assumptions. ADiT departs from this trend by introducing a single model architecture and training objective capable of generating both molecular and crystalline structures within a common framework.

The method employs a Transformer-based autoencoder to map both molecules and materials into a shared latent space. This encoder jointly processes 3D coordinates, atom types, and, when applicable, lattice vectors. Crucially, instead of conducting the diffusion process directly in atomic space—a strategy that often requires specialized, symmetry-aware networks—ADiT performs denoising in latent space using a Transformer-based decoder. This design significantly reduces computational complexity and enables faster sampling.

One of the key strengths of ADiT lies in its architectural simplicity. The model does not rely on graph-specific components or inductive biases, yet achieves competitive and, in many cases, state-of-the-art results across diverse datasets. It also exhibits strong scalability: increasing model size consistently improves performance. Additional features, such as classifier-free guidance, provide flexible output control, and joint training across molecular and crystalline domains has been shown to enhance generalization.

Overall, ADiT represents a notable shift toward general-purpose generative modeling at the atomistic level. Its ability to handle heterogeneous data types within a single Transformer-based framework—while maintaining efficiency and scalability—positions it as a promising direction for future work in unified atom-level generation.

ET-Flow: Equivariant Flow-Matching for Molecular Conformer Generation

Equivariant Transformer Flow (ET-Flow) [50] is a new method designed to predict low-energy molecular conformations directly from molecular graphs—a key challenge in computational drug discovery due to the complex interplay of chemical, physical, and biological factors.

ET-Flow is lightweight, simple, and scalable. It operates on full atomic coordinates and leverages an **equivariant transformer architecture** [51], which captures molecular geometry more effectively than conventional models such as EGNN [52]. To better guide the FM process, ET-Flow introduces a **harmonic prior** [53, 54] that aligns well with target conformations, helping the model learn to generate accurate low-energy structures.

To further improve sample quality at inference time, ET-Flow includes a **stochastic sampler**, and to address issues with molecular **chirality**, it applies a *post hoc correction strategy*. This correction is both computationally efficient and enhances the realism of the generated conformers.

Compared to state-of-the-art approaches, ET-Flow generates more precise and physically valid conformers, while being a lighter model and faster during inference.

3.1.4 Generative modeling of nucleotides

Generating biologically meaningful nucleotide sequences lies at the heart of synthetic biology, functional genomics, and RNA-based therapeutics. However, nucleotide sequences present a unique modeling challenge: they are discrete, drawn from a small fixed alphabet, yet encode complex, high-dimensional biological functions and regulatory constraints. Traditional generative approaches, such as autoregressive models, capture local dependencies but suffer from limited scalability due to their inherently sequential sampling. Discrete diffusion models offer parallelism and controllability but introduce mismatches between continuous stochastic dynamics and categorical sequence structure, often resulting in inefficiencies and degraded performance.

FM has recently emerged as a promising alternative, offering efficient, simulation-free learning of generative vector fields that deterministically map noise distributions to data. When adapted to operate directly on the probability simplex or statistical manifolds, FM enables biologically grounded, controllable generation of nucleotide sequences with strong performance across tasks. This section surveys a suite of recent flow-based generative models that advance the state-of-the-art in DNA and RNA sequence design, RNA structure generation, and sequence-structure co-design under diverse biological constraints. These models combine geometric insights, principled probabilistic interpolation, and domain-specific conditioning to unlock scalable, interpretable, and high-fidelity generation of nucleic acid molecules.

Generative flow matching for DNA sequence design.

Designing DNA sequences that satisfy diverse biochemical, structural, and functional constraints is a central challenge in synthetic biology, genomics, and biotechnology. Traditional approaches often rely on heuristic optimization or rigid generative models that lack flexibility, interpretability, or support for differentiable end-to-end learning. FM offers a promising alternative by enabling smooth, simulation-free transport between probability distributions. However, applying FM to discrete domains such as DNA sequences introduces unique challenges, chiefly, the incompatibility of standard Euclidean interpolants with categorical data. In response, recent advances have introduced principled geometric extensions of FM tailored to the simplex of categorical distributions. This section explores two such frameworks—Dirichlet Flow Matching and Fisher Flow Matching—which address these limitations through distributional and Riemannian formulations, respectively, and set new performance benchmarks in generative DNA design.

Dirichlet Flow Matching for DNA Sequence Design

However, naively adapting FM to the categorical setting-such as linearly interpolating on the probability simplex-has proven inadequate, introducing discontinuities and collapsing support that fundamentally limit its applicability to biological sequence generation. Hence, in this work, Stark et al. [55] introduce Dirichlet flow matching (Dirichlet FM)-a novel FM framework that operates natively on the probability simplex by using time-evolving Dirichlet distributions as intermediate probability paths. The method defines a smooth, continuous vector field for transporting a uniform Dirichlet prior to data distributions concentrated on the vertices of the simplex, thus addressing the pathological behaviors seen in linear FM. The core mathematical innovation lies in deriving a closed-form relationship between Dirichlet mixture scores and flow vector fields, enabling both classifier-based and classifier-free guidance. This allows for not only accurate unconditional generation, but also controllable, condition-guided generation of DNA sequences. Furthermore, the authors demonstrate the amenability of the method to distillation, achieving one-step sequence generation with minimal performance degradation, an O(L) speed-up over autoregressive baselines.

Dirichlet FM outperforms discrete diffusion and autoregressive baselines on multiple DNA design benchmarks. In promoter sequence generation, it achieves lower MSE in predicting regulatory profiles compared to state-of-the-art models. In enhancer datasets (melanoma and fly brain), Dirichlet FM achieves superior distributional fidelity (measured via Fréchet's biological distance), and its classifier-free guidance improves both sample quality and conditional specificity. Importantly, the distilled variant enables single-step generation, offering dramatic reductions in inference time. Dirichlet FM establishes a new direction for flow-based generative modeling of discrete biological sequences.

Fisher Flow Matching for DNA Sequence Design

While Dirichlet FM offered a principled start, it is constrained by its reliance on symmetric Dirichlet interpolants, which limit the flexibility of source distributions and lead to complex, numerically unstable parameterizations, especially near the simplex boundaries. These shortcomings impede practical applicability, particularly for practical tasks in biological sequence generation requiring non-uniform priors or precise control over distributional paths.

The recently introduced Fisher Flow Matching (Fisher-Flow) [56] builds on the FM paradigm but advances it with a principled geometric foundation. It redefines discrete data generation by treating categorical distributions as points on a statistical manifold equipped with the Fisher-Rao Riemannian metric. This framework exploits a key isometry: the probability simplex endowed with the Fisher-Rao metric is geometrically equivalent to the positive orthant of a hypersphere. As a result, Fisher-Flow performs FM on this hyperspherical space (denoted S_d^+), allowing for the construction of smooth, closed-form geodesic flows between any source and target distribution. This geometric approach not only stabilizes training and improves sample quality but also

accommodates arbitrary priors—something DFM could not efficiently handle. Moreover, by incorporating Riemannian optimal transport, Fisher-Flow further optimizes the flow trajectories, reducing variance and improving convergence.

Fisher-Flow significantly outperforms Dirichlet FM across DNA promoter and enhancer generation tasks. On promoter sequence generation, it achieves lower mean squared error (MSE) between predicted and target transcription profiles, indicating more accurate control over biological function. For enhancer sequences, perplexity scores are notably reduced compared to both DFM and autoregressive baselines, confirming its superior modeling of complex biological sequence distributions. The improvements are consistent across both unconditional and conditionally guided generation scenarios. Importantly, Fisher-Flow remains scalable, adaptable, and numerically robust, even in high-dimensional categorical settings, and its flows can be generalized to sequences via Cartesian product manifolds.

Fisher-Flow establishes a new state-of-the-art in discrete flow-based generative modelling for DNA sequence design. It not only addresses the key limitations of Dirichlet FM but also introduces a unifying geometric foundation grounded in information geometry.

Generative flow matching for RNA sequence design.

RNA design presents a multifaceted challenge at the intersection of molecular biology and machine learning, requiring the generation of sequences that satisfy diverse structural, functional, and contextual constraints. Unlike DNA, RNA assumes complex secondary and tertiary structures and exhibits dynamic regulatory roles, making its generative modeling particularly demanding. Existing approaches to RNA design, such as RfamGen, Ribodiffusion, and RDesign—though effective in specific contexts—are often narrowly scoped, with rigid architectures that limit extensibility and hinder integration of heterogeneous biological information. FM, with its ability to model smooth, distributional transformations in a simulation-free and differentiable manner, offers a compelling foundation for unifying these disparate tasks. This section introduces RNACG [57], a universal RNA design framework that leverages Dirichlet FM [55] and a multimodal transformer backbone to achieve high fidelity, conditional sequence generation across diverse RNA design objectives.

RNACG: A Universal Framework for RNA Sequence Design via Flow Matching

Addressing the limitations, Gao and Lu [57] present RNACG (RNA Conditional Generator), a modular and extensible RNA design framework that builds upon FM principles, incorporating Dirichlet-based probabilistic paths and a transformer-based architecture. RNACG generalizes RNA design by supporting a variety of conditional inputs-including RNA family identifiers, tertiary structure coordinates, and 5'UTR translation efficiency targets-within a single model. At its core, RNACG employs Dirichlet FM, allowing it to generate RNA sequences as smooth transformations of noise into structured nucleotide distributions. The architecture is built around a multimodal diffusion transformer (mm-DiT) that supports condition-specific encoders and a weighted loss function designed to focus learning on areas of high uncertainty.

Crucially, RNACG introduces a principled and parameter-efficient solution: with fewer than 5 million parameters, it achieves performance comparable to or exceeding state-of-the-art models like Ribodiffusion and RDesign across a variety of benchmarks.

RNACG demonstrates competitive performance across multiple RNA design tasks. In family-specific generation, it outperforms traditional covariance model-based methods like cmemit and rivals RfamGen, particularly in scenarios lacking secondary structure constraints. For RNA tertiary structure inverse folding, RNACG matches or surpasses the accuracy of Ribodiffusion and RDesign on long sequences, despite using a fraction of the model parameters. Furthermore, RNACG shows strong potential in predictive applications, achieving comparable translation efficiency prediction to the UTR-LM model. By unifying diverse RNA design paradigms under a single, flow-based framework, RNACG extends the capabilities of Dirichlet FM to RNA-specific challenges and significantly improves generality, interpretability, and computational efficiency.

Generative flow matching for RNA structure design.

Designing RNA molecules with specific structural and functional properties is a grand challenge in molecular engineering, with broad implications for synthetic biology, therapeutics, and molecular diagnostics. Unlike proteins, RNA structures exhibit substantial conformational flexibility, with their biological roles often dictated by dynamic structural ensembles rather than static folds. Traditional approaches to RNA structure design—whether based on autoregressive models, diffusion processes, or inverse folding—struggle to efficiently capture this complexity, especially under conditioning from molecular partners like proteins or ligands. Moreover, the computational cost and rigidity of fine-tuning large backbone predictors such as RoseTTAFold2NA further constrain their applicability in real-world design tasks. FM, with its simulation-free training, efficient conditioning, and geometric expressivity, has emerged as a powerful paradigm to bridge these gaps. In this section, we survey recent advances in flow-based RNA structure design—including models that co-generate sequences and structures, operate directly in SE(3) space, and support conditioning on proteins, ligands, and base-pair constraints—thereby laying the foundation for unified, scalable, and physically grounded RNA generative frameworks.

RNAFlow: Protein-Conditioned RNA Structure and Sequence Design via Inverse Folding-Based Flow Matching

RNAFlow [58] introduces a novel protein-conditioned FM framework that unifies RNA sequence and structure generation. A key feature of its architecture is the integration of an inverse folding network (Noise-to-Seq) with a fixed, pretrained RF2NA structure prediction module. Rather than generating static structures, RNAFlow leverages a FM objective to learn smooth transitions from random noise to structured RNA conformations, guided by protein context. Crucially, the denoising model is built on a geometric graph neural network (GNN) architecture that represents RNA and protein backbones as 3D point clouds and learns node- and edge-level interactions using GVP-GNN layers. This design enables the model to encode fine-grained spatial relationships while conditioning on both protein structure and sequence. Importantly, it

models RNA structural dynamics by introducing **Traj-to-Seq**, an extension that conditions sequence design on a trajectory of predicted structures-effectively capturing conformational ensembles. This dual-module setup allows RNAFlow to avoid fine-tuning RF2NA while still maintaining structural fidelity and enabling conformational generalization. Furthermore, an output rescoring module selects optimal samples post-generation, boosting the model's performance without sacrificing speed or sample diversity.

RNAFlow substantially outperforms existing baselines-including MMDiff and LSTM models-on both RNA structure and sequence generation tasks, achieving lower root-mean-square deviation (RMSD) and higher native sequence recovery across various benchmarks. The model also performs well on novel protein targets and generates sequences with structural and sequence novelty beyond the training set. On a practical motif-scaffolding task for the GRK2 protein, RNAFlow designed aptamers with superior structural accuracy and sequence fidelity compared to state-of-the-art methods. By combining FM with inverse folding and protein conditioning, RNAFlow significantly advances the state of RNA generative modeling, offering a scalable, accurate, and biologically meaningful framework for co-designing RNA sequences and structures in complex molecular contexts.

RiboFlow: Ligand-Conditioned RNA Sequence-Structure Co-Design via Synergistic Flow Matching

Though significant advancements in protein-ligand and protein-protein modeling have been driven by co-design frameworks, there still exist unique challenges when using these techniques to RNA-ligand systems. With dynamic conformations controlled by both global backbone positioning and local torsion angles that are tightly coupled to their sequence, RNA molecules have far greater structural flexibility than proteins. This complexity causes them to be particularly challenging to model computationally. This inherent complexity is not taken into consideration by the majority of current RNA design models, and they typically do not explicitly include ligand geometry in the generation process. This significantly reduces their ability to capture the functional binding interactions that are essential for therapeutic applications.

RiboFlow [59] addresses this gap through a synergistic FM framework that co-designs RNA sequences and structures explicitly conditioned on ligand geometry. The model integrates three complementary representations: RNA backbone frames to capture global orientation, torsion angles to model local flexibility, and sequence features for nucleotide identity. This multi-scale representation is trained under a conditional FM objective that operates on both SE(3) space and continuous torsion manifolds-allowing the model to learn physically coherent transitions from noise to RNA-ligand complexes. A key innovation is its ligand-guided conditioning mechanism, which embeds spatial constraints from ligand structures directly into the generation process. To support large-scale training and benchmarking, the authors introduce RiboBind, a curated dataset of over 1,500 RNA-ligand complexes—by far the largest and most comprehensive resource of its kind.

RiboFlow achieves notable gains over existing RNA design baselines. Across a range of benchmarks, it demonstrates over twofold improvement in AlphaFold3 binding

metrics and a 50% boost in structural validity. The model exhibits strong controllability, generalizing well to both unconditional and ligand-guided scenarios, and performs robustly in low-data settings-an important feature for therapeutic applications. By maintaining sequence-structure consistency while modeling RNA's conformational flexibility, RiboFlow enables high-fidelity design of ligand-binding RNAs, a task previously out of reach for flow-based or diffusion-based models.

RNA-EFM: Energy-Based Flow Matching for Protein-Conditioned RNA Co-Design

While RiboFlow introduced ligand-conditioned RNA design by modeling multi-scale RNA conformations in response to small-molecule geometry, RNA-EFM extends these principles to the more complex setting of protein-conditioned RNA co-design. Designing RNAs that bind proteins poses unique challenges due to the size, flexibility, and multi-contact nature of RNA-protein interfaces. Capturing both local backbone dynamics and global spatial alignment within a unified generative framework remains difficult, particularly under limited structural data. RNA-EFM [60] addresses this gap by embedding protein-RNA interaction constraints directly into the FM process.

RNA-EFM introduces an energy-based FM framework that co-generates RNA sequences, torsions, and backbone structures conditioned on protein geometry. The model incorporates explicit energy terms derived from protein-RNA contact maps, guiding generation toward physically favorable conformations. A graph neural network encodes both RNA and protein features, enabling flow prediction to respect both molecular context and energetic plausibility. Unlike RiboFlow, which focuses on ligand-guided design in torsion and frame space, RNA-EFM brings energetic awareness into protein-guided flow modeling.

RNA-EFM achieves strong performance across protein-RNA benchmarks, improving sequence recovery and structural fidelity over prior baselines. Its integration of energy-based priors into FM allows it to generalize well in few-shot settings and maintain biophysical coherence in generated complexes. Together with ligand-guided approaches like RiboFlow, RNA-EFM advances the scope of flow-based RNA design toward controllable, interaction-specific generation, opening new avenues for synthetic RNA engineering.

RNA-FrameFlow and RNAbpFlow: Flow-Based Models for RNA 3D Structure Generation

Accurate generation of RNA 3D structures remains a critical barrier in computational RNA design, particularly in tasks that demand structural precision as a foundation for downstream sequence optimization and functional inference. While earlier models like RNA-EFM and RiboFlow extended FM to RNA-protein and RNA-ligand co-design, respectively, the problem of generating full 3D RNA backbones—either unconditionally or conditioned on secondary structure—has been less thoroughly explored. Two recent models, RNA-FrameFlow [61] and RNAbpFlow [62], directly address this gap, offering complementary approaches to the generation of RNA backbone ensembles through SE(3)-equivariant FM.

RNA-FrameFlow tackles the problem of unconditional RNA 3D backbone generation, adapting techniques from protein modeling to accommodate RNA's increased structural complexity. By representing RNA backbones through both torsion angles and rigid-body frames, the model enables continuous and equivariant flow in SE(3) space. It introduces auxiliary losses on atomic distances and local frame alignment to enforce structural plausibility, while data augmentation strategies like clustering and fragment cropping improve generalization. RNA-FrameFlow demonstrates the capacity to generate diverse, geometrically consistent backbones across a range of sequence lengths, with over 40% of sampled structures surpassing a TM-score threshold of 0.45. This model serves as a critical building block for sequence-conditioned or context-aware RNA design.

Building upon this, RNAbpFlow introduces conditional RNA structure generation, integrating nucleotide sequence and base-pairing information—including noncanonical interactions—as explicit inputs. By conditioning the flow on both sequence and pairing constraints, RNAbpFlow generates all-atom 3D RNA structures that are consistent with target secondary structures. The model maintains SE(3) equivariance and expands the representational capacity of FrameFlow by incorporating base-pair-specific losses at both 2D (pair presence) and 3D (inter-nucleotide geometry) levels. Benchmarks demonstrate that RNAbpFlow significantly outperforms unconditioned models and traditional folding pipelines such as RNAJP, and even surpasses specialized structure predictors like DRfold and RhoFold+ on CASP15 targets. Conditioning on accurate base-pair maps improves both global folding accuracy and local atomic detail.

Together, RNA-FrameFlow and RNAbpFlow establish a robust FM foundation for RNA structure generation. While RNA-FrameFlow addresses the unconstrained generation of diverse RNA backbones, RNAbpFlow provides a pathway for secondary-structure-constrained modeling that is directly applicable to inverse folding and functional design. Their incorporation of SE(3)-equivariant architectures and multi-level geometric losses ensures both physical realism and computational scalability. These models complement earlier co-design efforts by offering upstream structural generation tools essential for integrated, end-to-end RNA design workflows in synthetic biology and therapeutic development.

3.1.5 Generative modeling of biomolecular interactions

Understanding and predicting the structural and energetic consequences of biomolecular interactions remains a central challenge in computational biology and drug discovery. Recent advances in generative modeling—particularly those leveraging geometric flows and continuous normalizing flows—have enabled the direct generation of bound complex structures and affinity predictions from unbound inputs, circumventing the need for traditional docking pipelines or multiple sequence alignments. In this section, we review three state-of-the-art generative frameworks that exemplify these advances: FlowDock [63], which applies conditional FM to protein–ligand docking; NeuralPLexer3 [64], which employs physics-inspired normalizing flows to predict complex structures across diverse biomolecular types; and FlexDock [65], which adapts FM for the task of switch-state protein-ligand (pocket-based) docking. Together,

these models highlight the growing promise of generative approaches in capturing interaction-induced conformational changes and enabling scalable, accurate structure and affinity prediction.

FlowDock: Geometric Flow Matching for Generative Protein–Ligand Docking and Affinity Prediction

FlowDock, introduced by Morehead and Cheng [63], presents a novel generative FM framework for protein–ligand docking and binding affinity prediction. Unlike previous models that are limited to single-ligand scenarios or rely on multiple sequence alignments, FlowDock employs conditional FM (CFM) to directly map unbound (apo) protein structures to their bound (holo) forms, accommodating arbitrary numbers of ligands simultaneously. It also delivers both structural confidence scores and predicted affinities for each complex, enabling efficient virtual screening. FlowDock achieved a 51% blind docking success rate on the PoseBusters Benchmark dataset using apo inputs—outperforming single-sequence AlphaFold 3 without alignment data—and matched or exceeded the performance of single-sequence AlphaFold 3 and Chai-1 on the DockGen-E benchmark.

Beyond docking accuracy, FlowDock demonstrated strong affinity prediction capabilities: in the CASP16 ligand category, it ranked among the top 5 methods across 140 protein–ligand complexes. This performance confirms its dual utility in structure generation and pharmacological scoring. The model's integration of geometric flow dynamics, interpretable ODE trajectories, and graph-based heavy-atom representations provides both accuracy and efficiency, marking it as a state-of-the-art tool with promising applications in drug discovery.

NeuralPLexer3: Accurate Biomolecular Complex Structure Prediction with Flow Models

NeuralPLexer3 (NP3) [64] introduces a physics-inspired, continuous normalizing flow model to predict high-resolution structures of biomolecular complexes—including proteins, small molecules, nucleic acids, and covalent ligands—directly from sequence and topology. Its architecture features an encoder–decoder flow framework, enhanced with informative priors (e.g., relaxed globular polymer distributions), optimal-transport-based atom permutation correction, and vector-field reparameterization. These innovations enable NP3 to generate physically plausible conformations in mere seconds on a single GPU, markedly faster than diffusion-based predecessors like AlphaFold 3.

Rigorous benchmarking demonstrates NP3's state-of-the-art accuracy: it achieves a 78.4% success rate (RMSD < 2Å and physically valid) on the PoseBusters Benchmark dataset, exceeding AlphaFold 3's 73.1% and outperforming docking tools like Vina and EquiBind. Across NP3's new open-source NPBench data suite, which spans monomers, protein–protein interfaces, nucleic acids, and covalent modifications, NP3 matches or surpasses AF2-Multimer. Crucially, on NP3's new ConfBench data suite—designed to assess ligand-induced conformational shifts—it exhibits superior ability to predict both apo and holo states. NP3 also provides reliable confidence scoring, enabling stratification of predictions by predicted accuracy.

Composing Unbalanced Flows for Flexible Docking and Relaxation

Corso et al. [65] recently introduced Unbalanced Flow Matching (UFM), a principled generalization of FM tailored to modeling transport between biological structure distributions, particularly in flexible molecular docking. Classical FM techniques impose strict marginal preservation, which limits their ability to capture the complexity of protein conformational changes. UFM relaxes this constraint by allowing controlled deviations from exact mass conservation via a penalized divergence term. This trade-off enables simpler, more learnable flows while maintaining sufficient fidelity to the target distribution. The authors theoretically derive UFM as an upper bound to a joint objective balancing sample efficiency and approximation accuracy, and show that compositions of short unbalanced flows correspond to local likelihood gradient steps toward the target distribution.

Empirically, the authors instantiate UFM in FlexDock, a two-stage generative framework for flexible (pocket-based) protein-ligand docking. The first stage (manifold docking) models large-scale, low-dimensional motions using flows over torsions and frames; the second stage (structure relaxation) refines atomic positions via Euclidean flows with energy-based constraints. FlexDock significantly improves docking accuracy and physical plausibility on the PDBBind benchmark, outperforming prior methods such as DiffDock-Pocket and ReDock. Notably, FlexDock increases the percentage of energetically plausible poses (as measured by PoseBusters checks [66]) from 30% to 73%. These results demonstrate the practical benefits of UFM in modeling biologically realistic transitions between complex structural distributions.

3.1.6 Generative modeling of biomolecular dynamics

Biomolecules are inherently dynamic entities, often transitioning between multiple conformations to carry out their biological functions. Capturing this structural heterogeneity is essential for understanding allostery, binding mechanisms, and functional regulation. Traditional structure prediction methods, including AlphaFold and ESMFold, focus on generating a single static structure and thus fall short of modeling the conformational ensembles observed in reality. Recent developments in FM generative modeling offer a promising avenue for learning full distributions over biomolecular structures, enabling the generation of physically consistent and diverse structural trajectories. In this section, we highlight such approaches, focusing on methods like AlphaFlow [67], FMRC [68], MDGen [69], and Onsager–Machlup-based action minimization [70]. These models extend static predictors into dynamic generators, offering a unified probabilistic framework for simulating biomolecular motions conditioned on sequence or initial structure.

AlphaFlow (& ESMFlow): AlphaFold Meets Flow Matching for Generating Protein Ensembles

In this work, Jing et al. [67] address the limitation of single-structure prediction in models like AlphaFold and ESMFold by developing a framework for generating protein conformational ensembles. While AlphaFold and similar predictors like ESMFold and OmegaFold are highly accurate at generating static protein structures, real proteins often exist as dynamic ensembles that shift between multiple conformations

depending on their functional state or environment. Previous attempts to capture this diversity have involved modifying the multiple sequence alignment (MSA) input to AlphaFold during inference, but this strategy does not apply to models like ESMFold or OmegaFold that do not rely on MSAs. To overcome this limitation, the authors repurpose AlphaFold and ESMFold as generative models using a FM framework. This allows the models to learn and sample full distributions over protein structures, enabling principled generation of conformational ensembles directly from sequence.

In their method, the authors reinterpret AlphaFold and ESMFold as denoising networks and integrate them into a FM framework for protein ensemble generation. The goal is to model the distribution of protein structures (i.e., structural ensembles) conditioned on sequence, $p(x \mid A)$, by training the model to denoise noisy structural inputs. Here, A denotes the input amino acid sequence of the protein, and $x \in \mathbb{R}^{3\times N}$ represents the 3D coordinates of the protein structure, where N is the number of residues. The architecture of AlphaFold is modified to accept an additional noisy structure input and time embedding, leveraging its existing template embedding stack. This allows AlphaFold and ESMFold to function as conditional denoisers without requiring major architectural changes.

Flow Matching for Optimal Reaction Coordinates of Biomolecular System

Zhang et al. [68] introduce Flow Matching for Reaction Coordinates (FMRC), a novel deep-learning framework that identifies optimal reaction coordinates (RCs) for biomolecular reversible dynamics. FMRC reformulates theoretical principles—lumpability and decomposability—into a conditional probability framework amenable to deep generative modeling. Unlike traditional methods that explicitly approximate the transfer operator or its eigenfunctions, FMRC instead trains an encoder–decoder neural network to match the flow of trajectory data, implicitly capturing the system's dominant slow dynamics. FMRC is demonstrated on three biomolecular systems, where it consistently outperforms existing techniques (TICA, SRV, SPIB) by enabling the construction of higher-quality Markov state models (MSMs) based on improved kinetic separation.

Beyond benchmarking, the authors apply FMRC in an enhanced sampling context. In a proof-of-concept case using an adaptive sampling dataset of the dipeptide Ala_2, the learned RC successfully guided bias deposition via OPES sampling, yielding converged free energy landscapes comparable to much longer traditional simulations. The framework offers a generalizable approach to extract interpretable low-dimensional RCs, which can aid in both MSM construction and enhanced sampling strategies. The authors' supporting information includes in-depth theoretical analysis demonstrating equivalence between FMRC's objectives and traditional definitions of lumpability/decomposability, implementation details for comparator methods (TICA, SRV, SPIB), architectures, training protocols, and extensive supplementary figures validating MSM performance and sampling convergence.

Generative Modeling of Molecular Dynamics Trajectories

MDGen [69] is a recent flow-based generative modeling approach that learns to produce full molecular dynamics (MD) trajectories rather than just individual frames or

transition densities. By tokenizing trajectories based on roto-translation and torsional angle offsets and using a Scalable Interpolant Transformer (SiT) backbone enhanced with long-context architectures like Hyena, MDGen models time-series of molecular states in an SE(3)-invariant manner. The framework supports conditioning on different trajectory segments to perform a variety of tasks—forward simulation (given initial structure), interpolation (transition path sampling), upsampling (increasing frame rate), and inpainting (filling in partial structures)—thereby covering both forward and inverse dynamics tasks.

In experiments with tetrapeptides (transferable to unseen sequences), MDGen accurately reproduced free-energy surfaces, autocorrelation and Markov-state kinetics, realistic transition paths, and fine-scale fast motions absent in coarse trajectories. Moreover, in the context of molecular design, MDGen's inpainting conditioned on partial structures outperformed static-frame inverse folding methods in sequence recovery. The authors also extend MDGen to protein monomer ensemble generation, demonstrating its capability to produce realistic structural distributions and outperform baseline subsampling approaches using AlphaFold.

Action-Minimization Meets Generative Modeling: Efficient Transition Path Sampling with the Onsager-Machlup Functional

Raja et al. [70] propose a novel, zero-shot transition path sampling (TPS) method by repurposing pre-trained generative models—namely denoising diffusion and FM—as stochastic dynamical systems. They show that candidate paths produced by these models can be endowed with an Onsager–Machlup (OM) action functional, which quantifies their likelihood under an overdamped Langevin-like dynamics induced by the model's learned score field. Optimizing the OM action using gradient-based methods effectively identifies high-probability transition trajectories that bridge stable molecular states, without requiring any task-specific retraining.

Experiments across increasingly complex systems validate the approach: on a 2D Müller–Brown potential, it efficiently recovers committor functions and kinetics; for alanine dipeptide, it yields accurate free-energy barriers, outperforming metadynamics; for fast-folding proteins like Chignolin and Trp-Cage, it produces atomistic transition paths matching reference MD, but at a fraction of computational cost; and for tetrapeptides, it demonstrates zero-shot generalization—extracting realistic transition ensembles even on new sequences. Overall, this framework represents a significant step toward scalable, model-agnostic TPS by leveraging action-based optimization over generative model trajectories.

3.2 Flow matching for single-cellular modeling

Understanding how cellular phenotypes evolve in response to perturbations is a foundational goal in systems biology and precision medicine. Yet, the high dimensionality of single-cell measurements and the combinatorial nature of experimental conditions—spanning drugs, genetic edits, and microenvironmental factors—render exhaustive profiling infeasible. Generative modeling offers a tractable solution by learning mappings from observed to unobserved cellular states, enabling virtual phenotyping and hypothesis generation.

Among recent approaches, FM has emerged as a scalable and interpretable framework for conditional generation. By learning continuous-time vector fields that transport unperturbed cell-state distributions to perturbed ones, FM enables simulation-free inference over complex, high-dimensional phenotypic spaces. These models leverage principles from optimal transport, score-based modeling, and neural ODEs to generate realistic single-cell states under diverse perturbation regimes.

In this section, we survey advances in FM-based generative modeling, beginning with CellFlow [71], a leading method that integrates multimodal biological embeddings with permutation-invariant architectures. We then highlight methods that model cellular dynamics, account for population heterogeneity, and incorporate biologically grounded priors. Together, these methods demonstrate how FM enables predictive, data-driven exploration of cellular phenotypes across biological systems and perturbation spaces. Table 2 displays relevant methods in this domain.

3.2.1 Generative modeling of cellular phenotypes

Modeling cellular phenotypes in response to diverse perturbations is a central challenge in systems biology and therapeutic discovery. The combinatorial explosion of possible conditions—ranging from drug cocktails to genetic edits—renders comprehensive experimental measurement infeasible, especially at single-cell resolution. Generative models offer a compelling alternative: by learning to simulate cellular responses under unobserved conditions, they enable *in silico* exploration of the vast phenotypic landscape. Recent work has introduced FM as a powerful framework for this task, allowing efficient and interpretable modeling of conditional distributions over high-dimensional cellular states. In this section, we highlight CellFlow [71] and CFGen [72], state-of-the-art generative frameworks that leverage optimal transport, biological embeddings, and permutation-invariant architectures to model realistic single-cell phenotypes under complex perturbation regimes. CellFlow integrates optimal transport, biological embeddings, and permutation-invariant architectures to model multi-modal cellular responses at scale, enabling virtual phenotyping and systematic perturbation design. CFGen extends this framework with discrete likelihoods and compositional conditioning, enabling precise, multi-modal simulation of single-cell data across biological and experimental contexts. Together, these models establish flow-based generative learning as a foundation for atlas augmentation and in silico perturbation design.

CellFlow Enables Generative Single-Cell Phenotype Modeling with Flow Matching

Combinatorial perturbation screens with single-cell resolution hold immense promise for decoding the complex relationship between the cellular state and external stimuli. However, the scale of potential perturbations, which spans drugs, cytokines, gene knockouts, and morphogen gradients, makes exhaustive experimental profiling impractical. This limitation has prompted a growing demand for computational models that can generalize beyond observed conditions to predict cellular phenotypes in unseen experimental settings.

CellFlow, introduced by Klein et al. [71], represents a significant advance in this space. Built on the principle of FM, CellFlow learns a conditional vector field

that maps unperturbed single-cell distributions to perturbed ones, guided by a set of context-specific biological embeddings. Unlike earlier methods that rely on explicit pairing of control and treated cells, CellFlow uses **optimal transport** to align distributions, enabling it to disentangle intrinsic cellular heterogeneity from perturbation-specific effects. Perturbations, including genetic, chemical, and environmental cues, are encoded using molecular fingerprints or protein language models and aggregated via **permutation-invariant attention mechanisms**, allowing flexible modeling of complex combinatorial conditions.

CellFlow demonstrates broad applicability and strong performance across a range of biological systems. In immune profiling, it accurately predicts donor-specific cytokine responses from a 10-million-cell PBMC atlas. In developmental biology, it captures gene knockout effects in zebrafish embryos at multiple stages, reconstructing tissue-specific phenotypes even at extrapolated time points. In drug response prediction, it recovers dose-dependent transcriptional shifts and synergistic effects of combinatorial treatments, outperforming established baselines in multiple benchmarks. Notably, in cell fate engineering and organoid development, CellFlow identifies previously unobserved cell states and generates realistic distributions under novel morphogen regimens, supporting in silico screening of differentiation protocols.

By integrating **biological prior knowledge, probabilistic flow-based modeling, and scalable inference**, CellFlow offers a unified and interpretable framework for generative phenotype modeling. Its ability to extrapolate across unseen perturbations, cell types, and developmental contexts marks a key step toward **virtual phenotyping**, enabling efficient hypothesis generation, experiment prioritization, and rational design of perturbation screens. As such, CellFlow exemplifies the potential of FM to bridge data scarcity and biological complexity at single-cell resolution.

Multi-Modal and Multi-Attribute Generation of Single Cells with CFGen

Existing deep generative models typically operate on normalized continuous gene expression data, neglecting the inherent discreteness, multi-modality (e.g., RNA and ATAC), and attribute-guidability of real-world single-cell assays.

To overcome these limitations, Palma et al. [72] propose CFGen (CellFlow for Generation), a flow-matching–based generative model that explicitly models discrete count distributions (negative binomial for RNA, Bernoulli for ATAC) and supports conditioning on multiple biological and technical attributes. CFGen decouples generation into sampling size factors and attribute labels, then transports Gaussian latent samples into discrete data using guided FM. This design preserves statistical accuracy for over-dispersed counts, enforces biologically realistic dispersion–mean trends, and supports compositional guidance in multi-label settings.

CFGen demonstrates superior performance across unimodal (RNA), multimodal (RNA+ATAC), and multi-attribute generation tasks. It outperforms state-of-the-art baselines—including scGAN, scDiffusion, scVI, and MultiVI—on distribution matching metrics, rare cell-type augmentation, and batch correction. Notably, CFGen delivers biologically plausible outputs such as correct zero-count sparsity, balanced mean–variance trends, and reliable conditional sampling across combinations of donor, cell type, and batch.

3.2.2 Flow matching in cellular dynamics and trajectories

The destructive nature of single-cell sequencing poses a fundamental challenge for understanding cellular dynamics: it precludes direct observation of individual cell trajectories over time. As a result, computational models must reconstruct dynamic processes—such as development, differentiation, and response to perturbation—from static population snapshots. Traditional approaches often rely on simplifying assumptions (e.g., mass conservation, Euclidean geometry, or deterministic flows) and require simulation-based training, which limits scalability and biological realism. Recently, FM has emerged as a powerful, simulation-free alternative for modeling such dynamics. By learning continuous-time vector fields from static distributions, FM enables scalable, flexible, and interpretable inference of cellular trajectories. This section surveys recent advances extending FM to address the complexities of single-cell dynamics, including stochasticity, growth, inter-individual variability, non-Euclidean geometry, and cyclic behaviors. These innovations—spanning methods such as CFM [18], $[SF]^2M$ [73], GENOT [74], Meta FM [75], and Curly-FM [76]—collectively redefine how we model biological processes from single-cell data, bridging theory and practice in dynamic systems inference.

Conditional Flow Matching: Efficient, Scalable Modeling via Minibatch Optimal Transport

Continuous normalizing flows (CNFs) offer a promising foundation for modeling cell-state transitions. Yet, standard CNF training requires costly simulations and assumes simple source distributions, limiting their practicality for complex biological data. Diffusion models avoid simulation during training but are computationally expensive at inference due to their stochastic nature. To address these trade-offs, in this work, Tong et al. [18] introduce a unified and highly general framework called Conditional Flow Matching (CFM), which combines the advantages of CNFs and diffusion models while overcoming their limitations. CFM defines a family of simulation-free training objectives that enable CNFs to be trained directly from static data pairs via regression, without requiring time integration. This includes a novel variant—Optimal Transport Conditional Flow Matching (OT-CFM)—which leverages optimal transport plans to generate straighter, more efficient flows that approximate dynamic optimal transport and Schrödinger bridges, even in high dimensions.

The highlight of the framework lies in its ability to handle arbitrary source and target distributions, enabling training with complex empirical data such as single-cell measurements. The authors demonstrate that by incorporating minibatch optimal transport, OT-CFM retains the structural benefits of full OT while remaining computationally tractable, even on large datasets. This allows for simulation-free learning of biologically meaningful trajectories and stochastic interpolants in both low-dimensional systems and high-dimensional settings like image generation or single-cell omics.

Extensive experiments confirm the method's utility. OT-CFM achieves state-of-the-art results on synthetic and real datasets for generative modeling, image translation, and single-cell trajectory inference. In particular, it significantly outperforms previous CNF and diffusion approaches in terms of both sample quality and

computational efficiency. In single-cell benchmarks, OT-CFM accurately interpolates held-out timepoints across developmental trajectories, surpassing prior methods such as TrajectoryNet and diffusion Schrödinger bridges.

Simulation-Free Schrödinger Bridges via Score and Flow Matching: A Scalable Approach to Learning Cell Dynamics

The destructive nature of single-cell assays forces us to reconstruct cellular trajectories indirectly, using independent population snapshots captured at different time points. Schrödinger bridges offer a principled way to model the most likely stochastic evolution between two observed states, making them a natural fit for inferring cellular dynamics from single-cell data. Yet in practice, most existing methods for learning such bridges depend on repeatedly simulating stochastic differential equations—a process that is not only computationally demanding but also poorly suited for the high-dimensional, noisy data typical of genomic applications.

Tong et al. [73] propose **[SF]^{2}M** (Simulation Free Score and Flow Matching), a novel simulation-free method for solving Schrödinger bridge problems without simulating the stochastic dynamics during optimization. The core idea is to frame stochastic generative modeling as a regression problem using static samples from source and target distributions. By combining score matching (used in diffusion models) and FM (used in continuous normalizing flows), [SF]^{2}M learns stochastic dynamics without requiring simulation during training. The method is grounded in entropy-regularized optimal transport, which allows it to scale efficiently even in high dimensions.

A key strength of [SF]^{2}M is its versatility. It accurately reconstructs known Schrödinger bridges in synthetic benchmarks and performs competitively in generative modeling and optimal transport tasks. Most notably, it is the first method capable of learning Schrödinger bridges in high-dimensional single-cell settings. The authors demonstrate this by recovering complex developmental trajectories, modeling Waddington-like landscapes, and even inferring gene regulatory networks from time-stamped transcriptomic data. The ability to incorporate manifold-aware costs and model stochasticity directly in gene expression space positions [SF]^{2}M as a biologically realistic and scalable alternative to deterministic or simulation-based methods.

Stochastic Flow Matching for Flexible Transport Modeling: The GENOT Framework

While optimal transport (OT) has become central in these efforts [77], classical discrete solvers are computationally intensive, non-parametric, and poorly suited for out-of-sample generalization. Neural OT solvers alleviate some of these issues but remain limited in flexibility, particularly in their reliance on deterministic maps, Euclidean cost functions, mass-conserving constraints, and intra-domain mappings.

To address these challenges, Klein et al. [74] introduce GENOT (Generative Entropic Neural Optimal Transport), a neural OT framework grounded in FM and designed to handle both linear and quadratic entropic OT, including their unbalanced and fused variants. GENOT learns stochastic transport plans rather than deterministic Monge maps, enabling the modeling of biological variability and uncertainty. The method supports arbitrary cost functions, essential for non-Euclidean data

geometries common in single-cell analyses, and relaxes mass conservation, accommodating cellular growth, death, and outliers. Furthermore, GENOT extends to the Gromov-Wasserstein (GW) and Fused GW (FGW) settings, enabling alignment across incomparable domains such as multi-modal or spatially misaligned datasets.

GENOT demonstrates robust performance across several key single-cell tasks. It accurately infers branching developmental trajectories in the mouse pancreas, predicts drug responses with calibrated uncertainty estimates, and facilitates ATAC-to-RNA cross-modal translation using GW and FGW formulations. These applications underscore the method's versatility in handling complex, heterogeneous biological data. In short, GENOT represents a significant step toward a general-purpose, expressive, and interpretable transport-based modeling framework for single-cell genomics.

Meta Flow Matching: Extending Flow Matching Paradigms for Heterogeneous Biological Systems

While Conditional FM (CFM) and GENOT have advanced the modeling of cellular dynamics through simulation-free deterministic flows and flexible stochastic optimal transport, respectively, both primarily focus on fixed or pairwise distributions. CFM excels in efficiently learning continuous flows from static snapshot pairs, offering scalable and simulation-free inference ideal for single-domain trajectory reconstruction. GENOT expands this framework by embracing stochastic transport plans and entropic regularization, enabling flexible handling of biological uncertainty and multi-modal or cross-domain alignments.

Meta Flow Matching (Meta FM), introduced by Atanackovic et al. [75], addresses a complementary but critical limitation: the assumption of a fixed initial population. Biological systems, particularly in personalized medicine and single-cell genomics, often feature diverse initial states across individuals or samples, where inter-population variability carries essential biological meaning. Meta FM innovatively integrates this heterogeneity by amortizing flow learning over entire populations. It employs graph neural networks to embed the structure of entire populations, conditioning the flow fields on these embeddings. This design enables Meta FM to generalize flow predictions across varying initial distributions, capturing personalized trajectories and enabling the modeling of interacting particles (e.g., cells) that traditional pairwise methods cannot. The approach is demonstrated on a large-scale multi-patient single-cell drug response dataset, where Meta FM accurately models individual treatment trajectories and predicts unseen cellular states. By integrating population-level embeddings with FM on the Wasserstein manifold, Meta FM provides a scalable, flexible, and biologically interpretable tool for modeling complex dynamic systems where sample heterogeneity and personalized inference are paramount.

Lifting Flow Matching to Distributions of Distributions: The WFM Framework

Many biological contexts demand a higher level of abstraction, where each sample itself is a distribution. This is particularly relevant in spatial transcriptomics, where cellular microenvironments—the distributions of neighboring gene expression profiles—carry critical functional meaning. Existing generative models cannot directly operate in

these distributional settings, particularly when samples vary in size, structure, or dimensionality.

To address this, Haviv et al. [78] introduce Wasserstein Flow Matching (WFM), a principled generalization of FM lifted to the space of probability distributions over distributions. WFM learns vector fields over the Wasserstein space, using optimal transport geometry to interpolate between entire distributions, whether represented analytically (as Gaussians) or empirically (as point clouds). For the Gaussian setting, the authors develop a closed-form solution via Bures–Wasserstein Flow Matching (BW-FM). For general distributions, they employ entropic optimal transport to compute approximate geodesics and parameterize vector fields using transformer architectures that respect permutation invariance and scale naturally to high dimensions.

WFM is evaluated across a diverse set of generative modeling tasks, including 2D and 3D shape synthesis, spatial transcriptomics, and scRNA-seq-based niche modeling. In spatial datasets, WFM demonstrates a unique capacity to generate cellular neighborhoods that reflect both molecular signatures and spatial context, outperforming prior methods that require voxelization or fixed-size samples. Notably, BW-FM can synthesize biologically meaningful niches—such as those arising during embryonic gut development or immune responses to infection—by accurately modeling the mean–covariance structure of local gene expression profiles.

Geometry-Aware Flow Matching: Learning Trajectories on the Data Manifold with Metric Flow Matching

The majority of FM methods interpolate between distributions, assuming Euclidean geometry, which often results in straight-line paths that stray off the data manifold. In biological contexts like single-cell transcriptomics, where data lie on curved, low-dimensional manifolds, such assumptions can lead to unrealistic reconstructions of cellular dynamics. Capturing nonlinear trajectories that respect the intrinsic geometry of data remains an open challenge.

To address this, Kapusniak et al. [79] introduce Metric Flow Matching (Metric FM), a simulation-free extension of Conditional FM (CFM) that learns interpolations guided by a data-induced Riemannian metric. Rather than assuming straight paths between samples, Metric FM constructs geodesic interpolants—paths that stay close to the data manifold by minimizing a kinetic energy functional tied to the learned metric. This ensures that the modeled dynamics remain biologically plausible and respect the structure of the underlying data.

A central contribution of Metric FM is its ability to operate entirely in ambient Euclidean space while implicitly enforcing manifold constraints. It avoids simulations by training a lightweight neural interpolator to approximate geodesics, which are then used to regress a flow field over time. The authors propose OT-Metric FM, a variant that incorporates optimal transport couplings to improve alignment between the source and target distributions. Empirical results across synthetic, vision, and single-cell benchmarks show that Metric FM consistently outperforms its Euclidean counterpart (OT-CFM), especially in reconstructing realistic intermediate states.

In single-cell trajectory inference, Metric FM achieves state-of-the-art performance, surpassing leading methods such as $[SF]^2M$, WLF, and CNF-based baselines on datasets from CITE-seq, Multiome, and embryoid body development. By aligning flows with the true geometry of gene expression data, Metric FM enables more accurate and interpretable modeling of dynamic cellular processes, without the need for temporal supervision or simulation.

Modeling Cell Dynamics and Interactions with Unbalanced Mean Field Schrödinger Bridge

Reconstructing cellular trajectories from single-cell snapshot data is also challenged by population unbalancedness (cells divide and die unevenly) and cell–cell interactions influencing state transitions. While prior flow and Schrödinger bridge methods have addressed mass variation, they assume independence of individual cells and ignore the critical role of intercellular communication.

To fill this gap, Zhang et al. [80] introduce an Unbalanced Mean-Field Schrödinger Bridge (UMFSB) framework and its practical instantiation, CytoBridge, which jointly models state dynamics, population growth, and cell–cell interactions. Drawing on mean-field theory, UMFSB generalizes unbalanced stochastic transport to include pairwise interaction terms, allowing cells to influence each other's trajectories through learned potentials. CytoBridge implements this by training neural networks to simultaneously regress velocity fields, growth functions, and interaction forces, all without relying on simulations.

Through evaluations on synthetic regulatory circuits and real single-cell RNA-seq datasets, CytoBridge accurately disentangles state transitions, growth, and intercellular interactions. It effectively suppresses biologically implausible trajectories and reconstructs coherent developmental landscapes, especially in systems shaped by signaling and local cellular context.

In summary, UMFSB and CytoBridge mark a substantial advance in flow-based single-cell modeling, extending the Schrödinger bridge paradigm to interaction-aware, unbalanced dynamics.

Curly Flow Matching for Learning Non-gradient Field Dynamics

Most FM frameworks assume that biological processes evolve along gradient fields-a simplifying assumption that underlies many trajectory inference models built on optimal transport. However, this assumption breaks down in important biological contexts, such as the cell cycle, where dynamics are inherently non-gradient and periodic. These rotational dynamics and similar priors (e.g., on cell trajectories) cannot be captured by conventional flow-based or Schrödinger bridge methods that rely on drift-free or potential-driven reference processes.

To address this limitation, Petrović et al. [76] introduce Curly Flow Matching (Curly-FM)-a novel framework for learning non-gradient field dynamics from single-cell data. Curly-FM frames the trajectory inference problem as a Schrödinger bridge with a non-zero reference drift, enabling the model to match both the marginal cell state distributions and the velocity field inferred from RNA-velocity. This stands

in contrast to prior approaches, which either ignore velocity information or assume dynamics follow energy-minimizing paths.

Curly-FM adopts a two-stage simulation-free approach. First, it learns a neural interpolant that matches a reference velocity field constructed from RNA-velocity estimates. Second, it trains a generative model to approximate stochastic transport paths aligned with both the reference field and population marginals. This design allows Curly-FM to model realistic cyclic behaviors that are inaccessible to conventional models.

In experiments on synthetic systems and the Deep Cycle single-cell RNA-seq dataset, Curly-FM outperforms existing flow-based models-including Conditional FM (CFM), OT-CFM, and TrajectoryNet-particularly in capturing circular trajectories and periodic dynamics. Quantitatively, it achieves superior alignment with RNA-velocity fields while maintaining competitive performance on generative accuracy metrics.

Extending Flow Matching to Unbalanced Populations: Joint Velocity-Growth Flow Matching for Single-Cell Dynamics Modeling

While Conditional FM (CFM) and $[SF]^2M$ have shown promise in learning simulation-free dynamics from static single-cell data, they operate under a key assumption: that cell populations are mass-conserving. In reality, biological systems such as developmental tissues, hematopoietic lineages, and tumor microenvironments often undergo non-conserved dynamics, with cells dividing, dying, or expanding at different rates. These processes induce mass variation over time, which cannot be captured by traditional balanced transport models.

To bridge this gap, Wang et al. [81] introduce Velocity–Growth Flow Matching (VGFM), a simulation-free method that extends the FM framework to jointly model state transitions and population growth. VGFM decomposes cellular dynamics into two coupled components: a velocity field, representing gene expression changes, and a growth field, capturing cell expansion or shrinkage. This formulation draws from a dynamic relaxation of optimal transport, where mass is allowed to evolve alongside state, rather than being strictly preserved.

Technically, VGFM leverages a regression-based training objective similar to that in CFM but augments it with a growth-aware loss. This allows it to learn both the transport paths and the local proliferation or depletion of cells across time. Crucially, VGFM avoids simulations entirely, ensuring scalability and training efficiency. The model is regularized using Wasserstein distance to match the evolving distributions of cells, and can incorporate auxiliary data such as RNA velocity if available.

In evaluations on synthetic benchmarks and real datasets—including embryoid bodies, hematopoiesis, and multi-modal CITE-seq data—VGFM accurately reconstructs both temporal trajectories and population-level mass changes. It outperforms simulation-based baselines and prior FM methods, particularly in settings where cell numbers shift non-uniformly or where time-resolved data are sparse.

By incorporating biologically realistic growth into flow-based modeling, VGFM extends the frontier of simulation-free trajectory inference. Similar to Wasserstein Lagrangian Flows by Neklyudov et al. [82], VGFM enables interpretable learning of

both cellular state and population dynamics from snapshot data, offering a powerful tool for understanding development, regeneration, and disease progression in high-resolution single-cell studies.

Modeling Complex System Dynamics with Flow Matching across Time and Conditions

Traditional optimal transport and FM frameworks typically model transitions between two distributions—limiting their ability to capture complex biological processes that unfold across multiple time points or conditions. This pairwise constraint is particularly restrictive in settings like drug response profiling or developmental biology, where datasets often span several perturbations and temporal stages.

To address this, Rohbeck et al. [83] propose Multi-Marginal Flow Matching (MMFM), a principled extension of FM that operates over multiple distributions simultaneously. MMFM models a shared vector field that transports samples smoothly across all observed marginals-whether defined by time, treatment, or other conditions-using spline-guided interpolation and classifier-free guidance. This formulation enables the learning of globally consistent trajectories without requiring simulation or time-resolved supervision.

The key innovation lies in integrating multi-marginal optimal transport into a FM framework, allowing the model to generalize across structured experimental designs. Applied to large-scale single-cell datasets of immune cells exposed to diverse kinase inhibitors over multiple time points, MMFM accurately reconstructs observed dynamics and predicts unseen combinations of time and condition, outperforming prior pairwise approaches.

In summary, MMFM extends FM to the multi-marginal setting, enabling scalable and biologically realistic modeling of systems governed by condition-specific and temporal variation.

3.3 Flow matching for multi-cellular modeling

While FM has shown promise in modeling cellular state transitions from high-dimensional transcriptomic or proteomic data, extending this framework to image-based phenotypic modeling opens new avenues for understanding cellular morphology. High-content imaging assays capture rich spatial and structural information that reflects the integrated outcome of cellular processes. However, generative modeling of these data—particularly under perturbations—remains a significant challenge due to batch effects, variability across experimental conditions, and the complexity of cellular morphology.

Recent advances have begun to address these challenges by applying FM to multi-cellular image data, enabling the generation of realistic, interpretable perturbation outcomes at the pixel level. In this subsection, we highlight CellFlux [84], a conditional FM framework that learns to simulate morphological changes in response to chemical and genetic perturbations across diverse experimental settings. By combining flow-based modeling with architectural innovations and batch-aware conditioning, CellFlux (listed in Table 2) exemplifies the potential of FM to bridge generative modeling and high-resolution cellular imaging.

CellFlux: Simulating Cellular Morphology Changes via Flow Matching

CellFlux [84] tackles the longstanding ambition of building a "virtual cell" — a generative model able to predict how a cell's morphology will change under chemical or genetic perturbations. The authors argue that existing image-based phenotypic profiling tools either overlook batch effects or model each perturbation in isolation, limiting biological fidelity. They instead recast the task as a transformation from a source distribution (control-well images) to a target distribution (perturbed-well images) within the same experimental batch, a formulation naturally suited to flow-based generative modeling.

CellFlux is a U-Net-based conditional FM model that learns a continuous velocity field transporting control images to their perturbed counterparts. The authors incorporate classifier-free guidance and Gaussian-based noise augmentation to improve the generated image fidelity. Empirically, CellFlux is evaluated on three public high-content screening datasets – BBBC021 (chemical) [85, 86], RxRx1 [87] (genetic), and JUMP [88] (combined). It achieves a significantly reduced Fréchet Inception Distance (FID) score and improved Mode-of-Action (MoA) prediction accuracy over prior generative baselines such as PhenDiff [89] and IMPA [90]. It also generalizes better on held-out perturbations, which were never seen during training, showing its broad applicability.

Beyond closing the performance gap, CellFlux effectively corrects the batch effects by keeping generated images within the correct batch distribution when seeded with a control image from that batch. FM enables smooth, bidirectional interpolation between cellular states, offering a novel lens on intermediate cell morphologies and perturbation dynamics. As such, it is a promising building block for "virtual cell" platforms in drug discovery and personalized medicine. In the future, the authors believe scaling up CellFlux to process diverse cell types and a wide range of perturbations will enable the full potential of virtual cell modeling.

3.4 Flow matching for bioimaging

Advances in FM have begun to transform image-based biological data analysis by providing scalable, probabilistic models capable of capturing complex spatial structures and offering principled uncertainty quantification. In bioimaging contexts—such as cryo-electron microscopy and medical image segmentation—data are often high-dimensional, noisy, and structurally intricate, posing challenges for both discriminative and generative models. FM frameworks offer a promising alternative by learning continuous-time vector fields that bridge simple priors and rich, structured outputs, enabling flexible inference without reliance on simulation or task-specific fine-tuning.

This section highlights two recent applications that exemplify the breadth of FM in bioimaging (Table 2). CryoFM [91] introduces a foundation model for cryo-EM density maps, capable of supporting diverse tasks such as denoising and 3D structure reconstruction through unconditional flow modeling. FlowSDF [92] tackles medical image segmentation using a novel conditional flow over signed-distance functions, producing accurate and smooth masks while offering uncertainty estimates. Together, these works illustrate how FM can unify representation learning and generative modeling in imaging-based biological systems.

CryoFM: A Flow-Based Foundation Model for Cryo-EM Densities

The EMDB database contains a vast number of 47k cryo-EM density maps of macromolecular structures. However, the wealth of information is not well utilized, and most data-driven methods are limited to task-specific implementation. CryoFM [91] overcomes this challenge by introducing a foundation model that can learn the distribution of high-quality density maps and generalize effectively to various downstream tasks such as denoising and 3D reconstruction of density maps without requiring further task-specific fine-tuning.

CryoFM is an unconditional FM model that learns a vector field $v_\Theta(t, \mathbf{x}_t)$, whose corresponding probability flow generates the data distribution $p_0(\mathbf{x}_0)$ of high-quality protein densities. During inference, given an observation y, a likelihood term $p_t(\mathbf{y}|\mathbf{x}_t)$ is introduced to change the unconditional vector field $v_\Theta(t, \mathbf{x}_t)$ to a conditional one $v_\Theta(t, \mathbf{x}_t|\mathbf{y})$, such that we can sample from the posterior distribution $p_0(\mathbf{x}_0|\mathbf{y})$. By adjusting the likelihood term in the flow posterior sampling methodology, the authors were able to successfully apply CryoFM for various downstream tasks such as anisotropic and spectral noise denoising, missing-wedge restoration, and ab initio 3D reconstruction of density maps from 2D class averages of particle images without requiring any further fine-tuning of the model. However, the denoising was limited to synthetic noise created by the authors and performed limitedly in denoising real-world noisy cryo-EM density maps. In spite of its shortcomings, the extensive application of CryoFM across different downstream tasks shows the potential of an unconditional FM-based foundational model for imaging techniques.

FlowSDF: Flow Matching for Medical Image Segmentation Using Distance Transforms

FlowSDF [92] addresses a central limitation of contemporary generative medical image segmentation models. Diffusion and flow–based methods have recently shown promise for modeling the distribution of segmentation masks, yet they work on binary masks whose class-wise discontinuous statistics make learning difficult and often yield unnaturally distorted boundaries. The authors instead cast segmentation as a conditional probability flow over signed-distance functions (SDFs) [93], a representation that smoothly encodes boundary proximity and naturally interpolates between classes.

FlowSDF is a conditional FM framework that learns a vector field guiding an ordinary differential equation from a simple Gaussian prior to the data distribution of SDFs conditioned on an input image. Compared with prior variance-exploding SDE approaches, this deterministic flow formulation offers stable training, faster inference, and exact likelihood evaluation. By sampling multiple trajectories, the model also delivers uncertainty maps (variance of generated masks) for free, providing clinically relevant confidence estimates.

Evaluation on public nuclei (MoNuSeg) [94] and gland (GlaS) [95] microscopic image datasets demonstrates that FlowSDF reaches or surpasses state-of-the-art segmentation accuracy while producing smoother, more realistic contours. One of the highlights of FlowSDF is the absence of jagged artifacts common in binary-mask generators. In conclusion, FlowSDF shows that pairing FM with distance-transform

representations bridges generative modeling and classical level-set methods, yielding a robust, probabilistic segmentation tool that is both accurate and uncertainty-aware.

4 Outlook

The rapid emergence of FM as a unifying paradigm for generative modeling marks a pivotal moment in the computational life sciences. Across molecular, cellular, and imaging domains, FM has already demonstrated significant advantages over previous simulation-based generative models, offering simulation-free training, efficient sampling, principled incorporation of geometry, and interpretable, controllable mappings between biological states. As surveyed in this Review, FM-based models now set the state-of-the-art in a wide array of bioinformatics tasks: from de novo protein and RNA design, to conditional single-cell phenotype prediction, to generative modeling of high-content imaging data. Yet, the field is only beginning to realize the full potential of flow-based generative modeling in biology.

From foundational advances to biological impact

The past two years have seen foundational advances in FM theory and practice. New formulations—such as optimal transport FM, discrete and manifold-valued flows, and simulation-free Schrödinger bridges—have expanded the scope of generative modeling to settings previously inaccessible to diffusion or normalizing flow methods. These innovations have enabled the modeling of complex, multimodal, and structured biological data, accommodating discrete sequences, geometric manifolds (e.g., SE(3) for proteins), and high-dimensional empirical distributions (e.g., single-cell omics or cellular images). Crucially, FM models have begun to bridge the gap between generative modeling and biological interpretability. By learning explicit vector fields and probability paths, FM provides a transparent lens on the transformations between biological states—be they conformational transitions in biomolecules, cell-state changes under perturbation, or morphological shifts in response to drugs. This interpretability, coupled with the ability to condition on rich biological context (e.g., protein partners, ligand structures, perturbation embeddings), positions FM as a cornerstone for the next generation of AI-driven virtual biology.

Unifying and extending the generative toolkit

A striking trend is the unification of previously disparate generative modeling approaches under the FM umbrella. Recent work has shown that FM subsumes diffusion models, continuous normalizing flows, and optimal transport as special cases, while enabling new hybrid and multimodal architectures. This theoretical convergence is mirrored by practical advances: models such as FrameFlow, FoldFlow, FlowDock, MultiFlow, and CellFlow demonstrate how FM can be flexibly adapted to discrete, continuous, or mixed data; to Euclidean, Riemannian, or statistical manifolds; and to both unconditional and richly conditional settings. Looking forward, we anticipate three key directions for extending the FM toolkit:

1. **Generalization to new data modalities and biological scales.** FM has already proven effective for sequences, structures, and images. Extensions to spatial omics, multi-modal single-cell data, multi-scale tissue modeling, and even population-level dynamics are natural next steps. Lifting FM to operate over distributions of distributions, as in WFM, or to model time-evolving population heterogeneity, as in Meta FM and VGFM, will be essential for capturing the complexity of biological systems.
2. **Principled incorporation of biological priors and constraints.** Future FM models will increasingly encode domain knowledge: physical symmetries, energetic constraints, evolutionary couplings, or experimentally derived priors. Advances in geometry-aware and energy-based flows, as well as integration with large pre-trained biological language models, will enable more accurate, controllable, and biologically meaningful generation.
3. **Scalable, interpretable, and uncertainty-aware modeling.** As FM models are deployed in high-stakes applications—such as therapeutic design, synthetic biology, or personalized medicine—scalability and interpretability become paramount. Efficient distillation and one-step generative flows (e.g., via Reflow or distilled FM), uncertainty quantification (as in FlowSDF), and explainable vector field visualization will be critical for building trust and enabling scientific discovery.

Toward the artificial intelligence-based virtual cell

A central vision emerging from recent FM advances is the realization of an AIVC: a unified, generative framework capable of simulating the molecular, structural, and phenotypic consequences of genetic, chemical, and environmental perturbations. FM models are uniquely well-suited to this challenge. By enabling efficient, simulation-free mappings across molecular, cellular, and imaging spaces—and by supporting conditional, context-aware generation—FM provides the scaffolding for an end-to-end, multi-scale virtual cell platform. Realizing this vision will require continued progress in several areas. Integrating FM models across biological scales and modalities—e.g., coupling protein–protein interaction flows with single-cell phenotype generators, or linking nucleotide design flows with 3D chromatin structure models—will demand advances in compositional and multi-modal FM architectures. Robust benchmarking, open-source tool development, and community standards for biological generative modeling (see Tables 1 and 2) will be essential to ensure reproducibility, interoperability, and broad adoption.

Open challenges and future research directions

Despite its promise, FM-based generative modeling faces open challenges. Theoretical questions remain regarding optimality, convergence, and expressivity of learned flows, especially in high-dimensional or data-sparse regimes. Efficient training and inference for very large or multi-modal biological datasets remains an area of active development. Furthermore, the integration of FM with experimental design—enabling closed-loop, AI-driven hypothesis generation and validation—remains largely unexplored. Key future directions include:

- **Learning on limited or noisy data:** Developing robust, data-efficient FM models that can generalize from scarce or noisy biological measurements.
- **Integration with experimental feedback:** Coupling FM-based generators with experimental platforms (e.g., high-throughput screening, CRISPR perturbation) for active learning and closed-loop discovery.
- **Interpretability and validation:** Building tools for visualizing, interpreting, and experimentally validating learned flows, ensuring that generative models yield biologically meaningful and actionable hypotheses.
- **Ethical and societal considerations:** As generative AI becomes increasingly influential in biological design, careful consideration of ethical, safety, and regulatory implications will be critical.

Conclusion

In summary, flow matching has rapidly matured from a theoretical curiosity to a practical, unifying framework for generative modeling in bioinformatics and computational biology. Its principled foundation, flexibility, and scalability have catalyzed breakthroughs across molecular, cellular, and imaging domains. As the field advances, flow matching is poised to underpin the next wave of AI-driven biological discovery, design, and simulation—bringing the vision of the virtual cell, and ultimately virtual biology, ever closer to reality.

References

[1] Frazer, J., Notin, P., Dias, M., Gomez, A., Min, J.K., Brock, K., Gal, Y., Marks, D.S.: Disease variant prediction with deep generative models of evolutionary data. Nature **599**(7883), 91–95 (2021)

[2] Kingma, D.P., Welling, M.: Auto-encoding variational bayes. In: International Conference on Learning Representations (2014)

[3] Cui, H., Wang, C., Maan, H., Pang, K., Luo, F., Duan, N., Wang, B.: scgpt: toward building a foundation model for single-cell multi-omics using generative ai. Nature Methods **21**(8), 1470–1480 (2024)

[4] Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F.L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al.: Gpt-4 technical report. arXiv preprint arXiv:2303.08774 (2023)

[5] Bommasani, R., Hudson, D.A., Adeli, E., Altman, R., Arora, S., Arx, S., Bernstein, M.S., Bohg, J., Bosselut, A., Brunskill, E., et al.: On the opportunities and risks of foundation models. arXiv preprint arXiv:2108.07258 (2021)

[6] Anand, N., Huang, P.: Generative modeling for protein structures. Advances in neural information processing systems **31** (2018)

[7] Anand, N., Eguchi, R., Mathews, I.I., Perez, C.P., Derry, A., Altman, R.B., Huang, P.-S.: Protein sequence design with a learned potential. Nature communications **13**(1), 746 (2022)

[8] Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. Advances in neural information processing systems **33**, 6840–6851 (2020)

[9] Hoogeboom, E., Satorras, V.G., Vignac, C., Welling, M.: Equivariant diffusion for molecule generation in 3d. In: International Conference on Machine Learning, pp. 8867–8887 (2022). PMLR

[10] Watson, J.L., Juergens, D., Bennett, N.R., Trippe, B.L., Yim, J., Eisenach, H.E., Ahern, W., Borst, A.J., Ragotte, R.J., Milles, L.F., *et al.*: De novo design of protein structure and function with rfdiffusion. Nature **620**(7976), 1089–1100 (2023)

[11] Shome, D., Sarkar, P., Etemad, A.: Region-disentangled diffusion model for high-fidelity ppg-to-ecg translation. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 38, pp. 15009–15019 (2024)

[12] LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. nature **521**(7553), 436–444 (2015). **This article characterizes foundational algorithms and applications of deep learning in artificial intelligence and beyond.**

[13] Lipman, Y., Chen, R.T., Ben-Hamu, H., Nickel, M., Le, M.: Flow matching for generative modeling. In: The Eleventh International Conference on Learning Representations (2023). **This article reports the first modern formulation of flow matching for generative AI.**

[14] Bunne, C., Roohani, Y., Rosen, Y., Gupta, A., Zhang, X., Roed, M., Alexandrov, T., AlQuraishi, M., Brennan, P., Burkhardt, D.B., *et al.*: How to build the virtual cell with artificial intelligence: Priorities and opportunities. Cell **187**(25), 7045–7063 (2024)

[15] Lipman, Y., Havasi, M., Holderrieth, P., Shaul, N., Le, M., Karrer, B., Chen, R.T., Lopez-Paz, D., Ben-Hamu, H., Gat, I.: Flow matching guide and code. arXiv preprint arXiv:2412.06264 (2024)

[16] Yang, L., Zhang, Z., Song, Y., Hong, S., Xu, R., Zhao, Y., Zhang, W., Cui, B., Yang, M.-H.: Diffusion models: A comprehensive survey of methods and applications. ACM Computing Surveys **56**(4), 1–39 (2023)

[17] Guo, Z., Liu, J., Wang, Y., Chen, M., Wang, D., Xu, D., Cheng, J.: Diffusion models in bioinformatics and computational biology. Nature reviews bioengineering **2**(2), 136–154 (2024)

[18] Tong, A., Fatras, K., Malkin, N., Huguet, G., Zhang, Y., Rector-Brooks, J.,

Wolf, G., Bengio, Y.: Improving and generalizing flow-based generative models with minibatch optimal transport. Transactions on Machine Learning Research (2024). **This article introduces the first generally tractable training objective for flow matching in generative AI.**

[19] Iserles, A.: A First Course in the Numerical Analysis of Differential Equations. Cambridge University Press, Cambridge (2009)

[20] Liu, X., Gong, C., *et al.*: Flow straight and fast: Learning to generate and transfer data with rectified flow. In: The Eleventh International Conference on Learning Representations (2023). **This article presents the first few-step formulation of generative flow matching.**

[21] Holderrieth, P., Havasi, M., Yim, J., Shaul, N., Gat, I., Jaakkola, T., Karrer, B., Chen, R.T.Q., Lipman, Y.: Generator matching: Generative modeling with arbitrary markov processes. In: The Thirteenth International Conference on Learning Representations (2025). **This article comprehensively generalizes flow matching across modalities and data types.** https://openreview.net/forum?id=RuP17cJtZo

[22] Yim, J., Trippe, B.L., De Bortoli, V., Mathieu, E., Doucet, A., Barzilay, R., Jaakkola, T.: Se (3) diffusion model with application to protein backbone generation. arXiv preprint arXiv:2302.02277 (2023)

[23] Bose, J., Akhound-Sadegh, T., Huguet, G., Fatras, K., Rector-Brooks, J., Liu, C.-H., Nica, A.C., Korablyov, M., Bronstein, M.M., Tong, A.: Se(3)-stochastic flow matching for protein backbone generation. In: The Twelfth International Conference on Learning Representations (2024). **This article reports one of the first applications of geometric generative flow matching in bioinformatics.**

[24] Huguet, G., Vuckovic, J., Fatras, K., Thibodeau-Laufer, E., Lemos, P., Islam, R., Liu, C., Rector-Brooks, J., Akhound-Sadegh, T., Bronstein, M., *et al.*: Sequence-augmented se (3)-flow matching for conditional protein generation. Advances in neural information processing systems **37**, 33007–33036 (2024)

[25] Lee, J.S., Kim, P.M.: Flowpacker: protein side-chain packing with torsional flow matching. Bioinformatics **41**(3), 010 (2025)

[26] Joshi, C.K., Fu, X., Liao, Y.-L., Gharakhanyan, V., Miller, B.K., Sriram, A., Ulissi, Z.W.: All-atom diffusion transformers: Unified generative modelling of molecules and materials. In: International Conference on Machine Learning (2025). PMLR

[27] Albergo, M.S., Vanden-Eijnden, E.: Building normalizing flows with stochastic interpolants. arXiv preprint arXiv:2209.15571 (2022)

[28] Campbell, A., Yim, J., Barzilay, R., Rainforth, T., Jaakkola, T.: Generative flows on discrete state-spaces: Enabling multimodal flows with applications to protein co-design. In: International Conference on Machine Learning, pp. 5453–5512 (2024). PMLR. **This article was the first to generalize flow matching to discrete data types.**

[29] Yang, S., Ju, L., Cheng, P., Zhou, J., Cai, Y., Feng, D.: Co-design protein sequence and structure in discrete space via generative flow. Bioinformatics **41**(5), 248 (2025)

[30] Kong, Z., Zhu, Y., Xu, Y., Zhou, H., Yin, M., Wu, J., Xu, H., Hsieh, C.-Y., Hou, T., Wu, J.: ProtFlow: Fast Protein Sequence Design via Flow Matching on Compressed Protein Language Model Embeddings. arXiv preprint arXiv:2504.10983 (2025)

[31] Yan, J., Cui, Z., Yan, W., Chen, Y., Pu, M., Li, S., Ye, S.: Robust and Reliable de novo Protein Design: A Flow-Matching-Based Protein Generative Model Achieves Remarkably High Success Rates. bioRxiv. preprint, 2025-04 (2025)

[32] Wu, F., Xu, T., Jin, S., Tang, X., Xu, Z., Zou, J., Hie, B.: D-Flow: Multi-modality Flow Matching for D-peptide Design. arXiv preprint arXiv:2411.10618 (2024)

[33] Wu, J., Kong, X., Sun, N., Wei, J., Shan, S., Feng, F., Wu, F., et al.: Flowdesign: Improved design of antibody cdrs through flow matching and better prior distributions. Cell Systems **16**(6) (2025)

[34] Chen, R.T., Lipman, Y.: Flow matching on general geometries. arXiv preprint arXiv:2302.03660 (2023)

[35] Zhang, Y., Zhang, Z., Zhong, B., Misra, S., Tang, J.: Diffpack: A torsional diffusion model for autoregressive protein side-chain packing. Advances in Neural Information Processing Systems **36**, 48150–48172 (2023)

[36] Liao, Y.-L., Wood, B., Das, A., Smidt, T.: Equiformerv2: Improved equivariant transformer for scaling to higher-degree representations. arXiv preprint arXiv:2306.12059 (2023)

[37] Hayes, T., Rao, R., Akin, H., Sofroniew, N.J., Oktay, D., Lin, Z., Verkuil, R., et al.: Simulating 500 million years of evolution with a language model. Science, 0018 (2025)

[38] Lin, Z., Akin, H., Rao, R., Hie, B., Zhu, Z., Lu, W., Smetanin, N., Verkuil, R., Kabeli, O., Shmueli, Y., et al.: Language models of protein sequences at the scale of evolution enable accurate structure prediction. https://github.com/facebookresearch/esm. Meta AI Technical Report (2022)

[39] Alamdari, S., Thakkar, N., Berg, R., Tenenholtz, N., Strome, R., Moses, A.M.,

Lu, A., Fusi, N., Amini, A.P., Yang, K.K., et al.: Protein generation with evolutionary diffusion: sequence is all you need. bioRxiv (2023). Preprint, doi:10.1101/2023.09.11.556673

[40] Meshchaninov, V., Strashnov, P., Shevtsov, A., Nikolaev, F., Ivanisenko, N., Kardymon, O., Vetrov, D.: Diffusion on language model encodings for protein sequence generation. arXiv preprint (2024). arXiv:2403.03726

[41] Repecka, D., Jauniskis, V., Karpus, L., Rembeza, M., Zrimec, J., Poviloniene, S., Rokaitis, I., Laurynenas, A., Abuajwa, W., Savolainen, O., *et al.*: Expanding functional protein sequence spaces using generative adversarial networks. Nature Machine Intelligence **3**(4), 324–333 (2021)

[42] OpenAI: GPT-3.5 Technical Overview. https://platform.openai.com/docs/models/gpt-3-5. Accessed July 2025 (2023)

[43] Frey, N.C., Berenberg, D., Zadorozhny, K., Kleinhenz, J., Lafrance-Vanasse, J., Hotzel, I., Wu, Y., Ra, S., Bonneau, R., Cho, K., et al.: Protein discovery with discrete walk-jump sampling. arXiv preprint arXiv:2306.12360 (2023)

[44] Ingraham, J.B., Baranov, M., Costello, Z., Barber, K.W., Wang, W., Ismail, A., Frappier, V., Lord, D.M., Ng-Thow-Hing, C., Van Vlack, E.R., *et al.*: Illuminating protein space with a programmable generative model. Nature **623**(7989), 1070–1078 (2023)

[45] Zambaldi, V., La, D., Chu, A.E., Patani, H., Danson, A.E., Kwan, T.O.C., Frerix, T., Schneider, R.G., Saxton, D., Thillaisundaram, A., Wu, Z., Moraes, I., Lange, O., Papa, E., Stanton, G., Martin, V., Singh, S., Wong, L.H., Bates, R., Kohl, S.A., Abramson, J., Senior, A.W., Alguel, Y., Wu, M.Y., Aspalter, I.M., Bentley, K., Bauer, D.L.V., Cherepanov, P., Hassabis, D., Kohli, P., Fergus, R., Wang, J.: De novo design of high-affinity protein binders with alphaproteo. Technical report, Google DeepMind (2024). Technical Report. https://storage.googleapis.com/deepmind-media/DeepMind.com/Blog/alphaproteo-generates-novel-proteins-for-biology-and-health-research/AlphaProteo2024.pdf

[46] Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., Tunyasuvunakool, K., Bates, R., Žídek, A., Potapenko, A., *et al.*: Highly accurate protein structure prediction with alphafold. nature **596**(7873), 583–589 (2021)

[47] Berman, H.M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T.N., Weissig, H., Shindyalov, I.N., Bourne, P.E.: The protein data bank. Nucleic Acids Research **28**(1), 235–242 (2000) https://doi.org/10.1093/nar/28.1.235

[48] Irwin, R., Tibo, A., Janet, J.P., Olsson, S.: Semlaflow–efficient 3d molecular generation with latent attention and equivariant flow matching. arXiv preprint arXiv:2406.07266 (2024)

[49] Dunn, I., Koes, D.R.: Mixed continuous and categorical flow matching for 3d de novo molecule generation. ArXiv, 2404 (2024)

[50] Hassan, M., Shenoy, N., Lee, J., Stärk, H., Thaler, S., Beaini, D.: Et-flow: Equivariant flow-matching for molecular conformer generation. In: Advances in Neural Information Processing Systems, vol. 37, pp. 128798–128824 (2024)

[51] Thölke, P., De Fabritiis, G.: Torchmd-net: Equivariant transformers for neural network based molecular potentials. arXiv preprint arXiv:2202.02541 (2022)

[52] Garcia Satorras, V., Hoogeboom, E., Welling, M.: E(n) equivariant graph neural networks. In: International Conference on Machine Learning (ICML), pp. 9323–9332 (2021). PMLR

[53] Jing, B., Erives, E., Pao-Huang, P., Corso, G., Berger, B., Jaakkola, T.: Eigenfold: Generative protein structure prediction with diffusion models. arXiv preprint arXiv:2304.02198 (2023)

[54] Stark, H., Jing, B., Barzilay, R., Jaakkola, T.: Harmonic prior self-conditioned flow matching for multi-ligand docking and binding site design. In: NeurIPS 2023 AI for Science Workshop (2023)

[55] Stark, H., Jing, B., Wang, C., Corso, G., Berger, B., Barzilay, R., Jaakkola, T.: Dirichlet flow matching with applications to dna sequence design. In: International Conference on Machine Learning, pp. 46495–46513 (2024). PMLR

[56] Davis, O., Kessler, S., Petrache, M., Ceylan, I., Bronstein, M., Bose, J.: Fisher flow matching for generative modeling over discrete data. Advances in Neural Information Processing Systems **37**, 139054–139084 (2024)

[57] Gao, L., Lu, Z.J.: Rnacg: A universal rna sequence conditional generation model based on flow-matching. arXiv preprint arXiv:2407.19838 (2024)

[58] Nori, D., Jin, W.: RNAFlow: RNA Structure & Sequence Design via Inverse Folding-based Flow Matching. arXiv preprint arXiv:2405.18768 (2024)

[59] Ma, R., Zhang, Z., Wang, Z., Hua, C., Zhou, Z., Cao, F., Rao, J., Zheng, S.: Riboflow: Conditional de novo rna sequence-structure co-design via synergistic flow matching. arXiv preprint arXiv:2503.17007 (2025)

[60] Abir, A.R., Zhang, L.: Rna-efm: Energy based flow matching for protein-conditioned rna sequence-structure co-design. bioRxiv, 2025–02 (2025)

[61] Anand, R., Joshi, C.K., Morehead, A., Jamasb, A.R., Harris, C., Mathis, S.V., Didi, K., Hooi, B., Lio, P.: RNA-frameflow: Flow matching for de novo 3d RNA backbone design. Transactions on Machine Learning Research (2025)

[62] Tarafder, S., Bhattacharya, D.: Rnabpflow: Base pair-augmented se (3)-flow

matching for conditional rna 3d structure generation. bioRxiv (2025)

[63] Morehead, A., Cheng, J.: Flowdock: Geometric flow matching for generative protein-ligand docking and affinity prediction. In: Intelligent Systems for Molecular Biology (ISMB) (2025)

[64] Qiao, Z., Ding, F., Dresselhaus, T., Rosenfeld, M.A., Han, X., Howell, O., Iyengar, A., Opalenski, S., Christensen, A.S., Krishna Sirumalla, S., et al.: Neuralplexer3: Physio-realistic biomolecular complex structure prediction with flow models. arXiv e-prints, 2412 (2024)

[65] Corso, G., Somnath, V.R., Getz, N., Barzilay, R., Jaakkola, T., Krause, A.: Composing unbalanced flows for flexible docking and relaxation. In: The Thirteenth International Conference on Learning Representations (2025). https://openreview.net/forum?id=gHLWTzKiZV

[66] Buttenschoen, M., Morris, G.M., Deane, C.M.: Posebusters: Ai-based docking methods fail to generate physically valid poses or generalise to novel sequences. Chemical Science **15**(9), 3130–3139 (2024)

[67] Jing, B., Berger, B., Jaakkola, T.: Alphafold meets flow matching for generating protein ensembles. In: Forty-first International Conference on Machine Learning (2024)

[68] Zhang, M., Zhang, Z., Wu, H., Wang, Y.: Flow matching for optimal reaction coordinates of biomolecular systems. Journal of Chemical Theory and Computation **21**(1), 399–412 (2024)

[69] Jing, B., Stärk, H., Jaakkola, T., Berger, B.: Generative modeling of molecular dynamics trajectories. Advances in Neural Information Processing Systems **37**, 40534–40564 (2024)

[70] Raja, S., Sipka, M., Psenka, M., Kreiman, T., Pavelka, M., Krishnapriyan, A.S.: Action-minimization meets generative modeling: Efficient transition path sampling with the onsager-machlup functional. In: Forty-second International Conference on Machine Learning (2025)

[71] Klein, D., Fleck, J.S., Bobrovskiy, D., Zimmermann, L., Becker, S., Palma, A., Dony, L., Tejada-Lapuerta, A., Huguet, G., Lin, H.-C., et al.: Cellflow enables generative single-cell phenotype modeling with flow matching. bioRxiv, 2025–04 (2025)

[72] Palma, A., Richter, T., Zhang, H., Lubetzki, M., Tong, A., Dittadi, A., Theis, F.J.: Multi-modal and multi-attribute generation of single cells with CFGen. In: The Thirteenth International Conference on Learning Representations (2025). https://openreview.net/forum?id=3MnMGLctKb

[73] Tong, A., Malkin, N., Fatras, K., Atanackovic, L., Zhang, Y., Huguet, G., Wolf, G., Bengio, Y.: Simulation-free schrödinger bridges via score and flow matching. In: AISTATS, pp. 1279–1287 (2024). https://proceedings.mlr.press/v238/y-tong24a.html

[74] Klein, D., Uscidda, T., Theis, F., Cuturi, M.: Genot: Entropic (gromov) wasserstein flow matching with applications to single-cell genomics. Advances in Neural Information Processing Systems **37**, 103897–103944 (2024)

[75] Atanackovic, L., Zhang, X., Amos, B., Blanchette, M., Lee, L.J., Bengio, Y., Tong, A., Neklyudov, K.: Meta flow matching: Integrating vector fields on the wasserstein manifold. arXiv preprint arXiv:2408.14608 (2024)

[76] Petrović, K., Atanackovic, L., Kapusniak, K., Bronstein, M.M., Bose, J., Tong, A.: Curly flow matching for learning non-gradient field dynamics. In: ICLR 2025 Workshop on Machine Learning for Genomics Explorations (2025)

[77] Bunne, C., Stark, S.G., Gut, G., Del Castillo, J.S., Levesque, M., Lehmann, K.-V., Pelkmans, L., Krause, A., Rätsch, G.: Learning single-cell perturbation responses using neural optimal transport. Nature methods **20**(11), 1759–1768 (2023)

[78] Haviv, D., Pooladian, A.-A., Pe'er, D., Amos, B.: Wasserstein flow matching: Generative modeling over families of distributions. arXiv preprint arXiv:2411.00698 (2024)

[79] Kapusniak, K., Potaptchik, P., Reu, T., Zhang, L., Tong, A., Bronstein, M., Bose, J., Di Giovanni, F.: Metric flow matching for smooth interpolations on the data manifold. Advances in Neural Information Processing Systems **37**, 135011–135042 (2024)

[80] Zhang, Z., Wang, Z., Sun, Y., Li, T., Zhou, P.: Modeling cell dynamics and interactions with unbalanced mean field schr\" odinger bridge. arXiv preprint arXiv:2505.11197 (2025)

[81] Wang, D., Jiang, Y., Zhang, Z., Gu, X., Zhou, P., Sun, J.: Joint velocity-growth flow matching for single-cell dynamics modeling. arXiv preprint arXiv:2505.13413 (2025)

[82] Neklyudov, K., Brekelmans, R., Tong, A., Atanackovic, L., Liu, Q., Makhzani, A.: A computational framework for solving Wasserstein lagrangian flows. In: Salakhutdinov, R., Kolter, Z., Heller, K., Weller, A., Oliver, N., Scarlett, J., Berkenkamp, F. (eds.) Proceedings of the 41st International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 235, pp. 37461–37485. PMLR, ??? (2024). https://proceedings.mlr.press/v235/neklyudov24a.html

[83] Rohbeck, M., De Brouwer, E., Bunne, C., Huetter, J.-C., Biton, A., Chen, K.Y., Regev, A., Lopez, R.: Modeling complex system dynamics with flow matching across time and conditions. In: The Thirteenth International Conference on Learning Representations (2025)

[84] Zhang, Y., Su, Y., Wang, C., Li, T., Wefers, Z., Nirschl, J., Burgess, J., Ding, D., Lozano, A., Lundberg, E., *et al.*: Cellflux: Simulating cellular morphology changes via flow matching. In: Proceedings of the 42nd International Conference on Machine Learning (2025)

[85] Caie, P.D., Walls, R.E., Ingleston-Orme, A., Daya, S., Houslay, T., Eagle, R., Roberts, M.E., Carragher, N.O.: High-content phenotypic profiling of drug response signatures across distinct cancer cells. Molecular Cancer Therapeutics **9**(6), 1913–1926 (2010) https://doi.org/10.1158/1535-7163.MCT-09-1148 https://aacrjournals.org/mct/article-pdf/9/6/1913/1886797/1913.pdf

[86] Ljosa, V., Sokolnicki, K.L., Carpenter, A.E.: Annotated high-throughput microscopy image sets for validation. Nat. Methods **9**(7), 637 (2012)

[87] Sypetkowski, M., Rezanejad, M., Saberian, S., Kraus, O., Urbanik, J., Taylor, J., Mabey, B., Victors, M., Yosinski, J., Sereshkeh, A.R., Haque, I., Earnshaw, B.: Rxrx1: A dataset for evaluating experimental batch correction methods. In: 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 4285–4294 (2023). https://doi.org/10.1109/CVPRW59228.2023.00451

[88] Chandrasekaran, S.N., Ackerman, J., Alix, E., Ando, D.M., Arevalo, J., Bennion, M., Boisseau, N., Borowa, A., Boyd, J.D., Brino, L., Byrne, P.J., Ceulemans, H., Ch'ng, C., Cimini, B.A., Clevert, D.-A., Deflaux, N., Doench, J.G., Dorval, T., Doyonnas, R., Dragone, V., Engkvist, O., Faloon, P.W., Fritchman, B., Fuchs, F., Garg, S., Gilbert, T.J., Glazer, D., Gnutt, D., Goodale, A., Grignard, J., Guenther, J., Han, Y., Hanifehlou, Z., Hariharan, S., Hernandez, D., Horman, S.R., Hormel, G., Huntley, M., Icke, I., Iida, M., Jacob, C.B., Jaensch, S., Khetan, J., Kost-Alimova, M., Krawiec, T., Kuhn, D., Lardeau, C.-H., Lembke, A., Lin, F., Little, K.D., Lofstrom, K.R., Lotfi, S., Logan, D.J., Luo, Y., Madoux, F., Marin Zapata, P.A., Marion, B.A., Martin, G., McCarthy, N.J., Mervin, L., Miller, L., Mohamed, H., Monteverde, T., Mouchet, E., Nicke, B., Ogier, A., Ong, A.-L., Osterland, M., Otrocka, M., Peeters, P.J., Pilling, J., Prechtl, S., Qian, C., Rataj, K., Root, D.E., Sakata, S.K., Scrace, S., Shimizu, H., Simon, D., Sommer, P., Spruiell, C., Sumia, I., Swalley, S.E., Terauchi, H., Thibaudeau, A., Unruh, A., Waeter, J., Van Dyck, M., Staden, C., Warchoł, M., Weisbart, E., Weiss, A., Wiest-Daessle, N., Williams, G., Yu, S., Zapiec, B., Żyła, M., Singh, S., Carpenter, A.E.: Jump cell painting dataset: morphological impact of 136,000 chemical and genetic perturbations. bioRxiv (2023) https://doi.org/10.1101/2023.03.23.534023 https://www.biorxiv.org/content/early/2023/03/24/2023.03.23.534023.full.pdf

[89] Bourou, A., Boyer, T., Gheisari, M., Daupin, K., Dubreuil, V., De Thonel, A., Mezger, V., Genovesio, A.: Phendiff: Revealing subtle phenotypes with diffusion models in real images, pp. 358–367. Springer, Berlin, Heidelberg (2024). https://doi.org/10.1007/978-3-031-72384-1_34

[90] Palma, A., Theis, F.J., Lotfollahi, M.: Predicting cell morphological responses to perturbations using generative modeling. Nat. Commun. **16**(1), 505 (2025)

[91] Zhou, Y., Li, Y., Yuan, J., Gu, Q.: CryoFM: A flow-based foundation model for cryo-EM densities. In: The Thirteenth International Conference on Learning Representations (2025). https://openreview.net/forum?id=T4sMzjy7fO

[92] Bogensperger, L., Narnhofer, D., Falk, A., Schindler, K., Pock, T.: Flowsdf: Flow matching for medical image segmentation using distance transforms. International Journal of Computer Vision **133**(7), 4864–4876 (2025) https://doi.org/10.1007/s11263-025-02373-y

[93] Bogensperger, L., Narnhofer, D., Ilic, F., Pock, T.: Score-based generative models for medical image segmentation using signed distance functions. In: Köthe, U., Rother, C. (eds.) Pattern Recognition, pp. 3–17. Springer, Cham (2024)

[94] Kumar, N., Verma, R., Sharma, S., Bhargava, S., Vahadane, A., Sethi, A.: A dataset and a technique for generalized nuclear segmentation for computational pathology. IEEE Transactions on Medical Imaging **36**(7), 1550–1560 (2017) https://doi.org/10.1109/TMI.2017.2677499

[95] Sirinukunwattana, K., Pluim, J.P.W., Chen, H., Qi, X., Heng, P.-A., Guo, Y.B., Wang, L.Y., Matuszewski, B.J., Bruni, E., Sanchez, U., Böhm, A., Ronneberger, O., Cheikh, B.B., Racoceanu, D., Kainz, P., Pfeiffer, M., Urschler, M., Snead, D.R.J., Rajpoot, N.M.: Gland segmentation in colon histology images: The glas challenge contest. Medical Image Analysis **35**, 489–502 (2017) https://doi.org/10.1016/j.media.2016.08.008

[96] Paszke, A.: Pytorch: An imperative style, high-performance deep learning library. arXiv preprint arXiv:1912.01703 (2019)

[97] Huang, C.-W., Krueger, D., Lacoste, A., Courville, A.: Neural autoregressive flows. In: International Conference on Machine Learning, pp. 2078–2087 (2018). PMLR

[98] Chen, R.T., Rubanova, Y., Bettencourt, J., Duvenaud, D.K.: Neural ordinary differential equations. Advances in neural information processing systems **31** (2018). **This article introduces the theoretical foundations necessary for generative flow matching of differential systems.**

[99] Kingma, D.P., Dhariwal, P.: Glow: Generative flow with invertible 1x1 convolutions. Advances in neural information processing systems **31** (2018)

[100] Grathwohl, W., Chen, R.T., Bettencourt, J., Sutskever, I., Duvenaud, D.: Ffjord: Free-form continuous dynamics for scalable reversible generative models. In: International Conference on Learning Representations (2019)

[101] Dupont, E., Doucet, A., Teh, Y.W.: Augmented neural odes. Advances in neural information processing systems **32** (2019)

[102] Chen, R.T., Behrmann, J., Duvenaud, D.K., Jacobsen, J.-H.: Residual flows for invertible generative modeling. Advances in Neural Information Processing Systems **32** (2019)

[103] Liu, J., Kumar, A., Ba, J., Kiros, J., Swersky, K.: Graph normalizing flows. Advances in Neural Information Processing Systems **32** (2019)

[104] Wu, H., Köhler, J., Noé, F.: Stochastic normalizing flows. Advances in neural information processing systems **33**, 5933–5944 (2020)

[105] Hodgkinson, L., Heide, C., Roosta, F., Mahoney, M.W.: Stochastic continuous normalizing flows: training sdes as odes. In: Uncertainty in Artificial Intelligence, pp. 1130–1140 (2021). PMLR

[106] Garcia Satorras, V., Hoogeboom, E., Fuchs, F., Posner, I., Welling, M.: E (n) equivariant normalizing flows. Advances in Neural Information Processing Systems **34**, 4181–4192 (2021)

[107] Zhang, Q., Chen, Y.: Diffusion normalizing flow. Advances in neural information processing systems **34**, 16280–16291 (2021)

[108] Gabrié, M., Rotskoff, G.M., Vanden-Eijnden, E.: Adaptive monte carlo augmented with normalizing flows. Proceedings of the National Academy of Sciences **119**(10), 2109420119 (2022)

[109] Richter-Powell, J., Lipman, Y., Chen, R.T.: Neural conservation laws: A divergence-free perspective. Advances in Neural Information Processing Systems **35**, 38075–38088 (2022)

[110] Albergo, M.S., Boffi, N.M., Vanden-Eijnden, E.: Stochastic interpolants: A unifying framework for flows and diffusions. arXiv preprint arXiv:2303.08797 (2023)

[111] Albergo, M.S., Goldstein, M., Boffi, N.M., Ranganath, R., Vanden-Eijnden, E.: Stochastic interpolants with data-dependent couplings. In: International Conference on Machine Learning, pp. 921–937 (2024). PMLR

[112] Davtyan, A., Sameni, S., Favaro, P.: Efficient video prediction via sparsely conditioned flow matching. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 23263–23274 (2023)

[113] Wu, L., Wang, D., Gong, C., Liu, X., Xiong, Y., Ranjan, R., Krishnamoorthi, R., Chandra, V., Liu, Q.: Fast point cloud generation with straight flows. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 9445–9454 (2023)

[114] Neklyudov, K., Brekelmans, R., Severo, D., Makhzani, A.: Action matching: Learning stochastic dynamics from samples. In: International Conference on Machine Learning, pp. 25858–25889 (2023). PMLR

[115] Lee, S., Kim, B., Ye, J.C.: Minimizing trajectory curvature of ode-based generative models. In: International Conference on Machine Learning, pp. 18957–18973 (2023). PMLR

[116] Dao, Q., Phung, H., Nguyen, B., Tran, A.: Flow matching in latent space. arXiv preprint arXiv:2307.08698 (2023)

[117] Chen, R.T.Q., Lipman, Y.: Flow matching on general geometries. In: The Twelfth International Conference on Learning Representations (2024). **This article reports a theoretical framework crucial to developing flow matching models for geometric data.** https://openreview.net/forum?id=g7ohDlTITL

[118] Klein, L., Krämer, A., Noé, F.: Equivariant flow matching. Advances in Neural Information Processing Systems **36**, 59886–59910 (2023)

[119] Neklyudov, K., Brekelmans, R., Tong, A., Atanackovic, L., Liu, Q., Makhzani, A.: A computational framework for solving wasserstein lagrangian flows. arXiv preprint arXiv:2310.10649 (2023)

[120] Fischer, J.S., Gui, M., Ma, P., Stracke, N., Baumann, S.A., Ommer, B.: Boosting latent diffusion with flow matching. arXiv preprint arXiv:2312.07360 (2023)

[121] Hu, V.T., Yin, W., Ma, P., Chen, Y., Fernando, B., Asano, Y.M., Gavves, E., Mettes, P., Ommer, B., Snoek, C.G.: Motion flow matching for human motion synthesis and editing. arXiv preprint arXiv:2312.08895 (2023)

[122] Liu, X., Zhang, X., Ma, J., Peng, J., *et al.*: Instaflow: One step is enough for high-quality diffusion-based text-to-image generation. In: The Twelfth International Conference on Learning Representations (2023)

[123] Kerrigan, G., Migliorini, G., Smyth, P.: Functional flow matching. In: International Conference on Artificial Intelligence and Statistics, pp. 3934–3942 (2024). PMLR

[124] Gat, I., Remez, T., Shaul, N., Kreuk, F., Chen, R.T., Synnaeve, G., Adi, Y., Lipman, Y.: Discrete flow matching. Advances in Neural Information Processing Systems **37**, 133345–133385 (2024)

[125] Nguyen, B., Nguyen, B., Nguyen, V.A.: Bellman optimal stepsize straightening of flow-matching models. arXiv preprint arXiv:2312.16414 (2023)

[126] Yang, L., Zhang, Z., Zhang, Z., Liu, X., Xu, M., Zhang, W., Meng, C., Ermon, S., Cui, B.: Consistency flow matching: Defining straight flows with velocity consistency. arXiv preprint arXiv:2407.02398 (2024)

[127] Gui, M., Schusterbauer, J., Prestel, U., Ma, P., Kotovenko, D., Grebenkova, O., Baumann, S.A., Hu, V.T., Ommer, B.: Depthfm: Fast generative monocular depth estimation with flow matching. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 39, pp. 3203–3211 (2025)

[128] Chen, Y., Goldstein, M., Hua, M., Albergo, M.S., Boffi, N.M., Vanden-Eijnden, E.: Probabilistic forecasting with stochastic interpolants and föllmer processes. In: Proceedings of the 41st International Conference on Machine Learning, pp. 6728–6756 (2024)

[129] Wang, C., Li, X., Qi, L., Ding, H., Tong, Y., Yang, M.-H.: Semflow: Binding semantic segmentation and image synthesis via rectified flow. Advances in Neural Information Processing Systems **37**, 138981–139001 (2024)

[130] Kim, M., Lee, Y., Kang, S., Oh, J., Chong, S., Yun, S.-Y.: Preference alignment with flow matching. Advances in Neural Information Processing Systems **37**, 35140–35164 (2024)

[131] Kerrigan, G., Migliorini, G., Smyth, P.: Dynamic conditional optimal transport through simulation-free flows. Advances in Neural Information Processing Systems **37**, 93602–93642 (2024)

[132] Hu, V., Wu, D., Asano, Y., Mettes, P., Fernando, B., Ommer, B., Snoek, C.: Flow matching for conditional text generation in a few sampling steps. In: Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 2: Short Papers), pp. 380–392 (2024)

[133] Cheng, C., Li, J., Peng, J., Liu, G.: Categorical flow matching on statistical manifolds. Advances in Neural Information Processing Systems **37**, 54787–54819 (2024)

[134] Kornilov, N., Mokrov, P., Gasnikov, A., Korotin, A.: Optimal flow matching: Learning straight trajectories in just one step. Advances in Neural Information Processing Systems **37**, 104180–104204 (2024)

[135] Nisonoff, H., Xiong, J., Allenspach, S., Listgarten, J.: Unlocking guidance for discrete state-space diffusion and flow models. In: The Thirteenth International Conference on Learning Representations

[136] Zhang, X.N., Pu, Y., Kawamura, Y., Loza, A., Bengio, Y., Shung, D., Tong,

A.: Trajectory flow matching with applications to clinical time series modelling. Advances in Neural Information Processing Systems **37**, 107198–107224 (2024)

[137] Stoica, G., Ramanujan, V., Fan, X., Farhadi, A., Krishna, R., Hoffman, J.: Contrastive flow matching. arXiv preprint arXiv:2506.05350 (2025) arXiv:2506.05350 [cs.CV]

[138] Lin, H., Li, S., Ye, H., Yang, Y., Ermon, S., Liang, Y., Ma, J.: Tfg-flow: Training-free guidance in multimodal generative flow. arXiv preprint arXiv:2501.14216 (2025)

[139] Bartosh, G., Vetrov, D., Naesseth, C.A.: Sde matching: Scalable and simulation-free training of latent stochastic differential equations. arXiv preprint arXiv:2502.02472 (2025)

[140] Yim, J., Campbell, A., Foong, A.Y., Gastegger, M., Jiménez-Luna, J., Lewis, S., Satorras, V.G., Veeling, B.S., Barzilay, R., Jaakkola, T., et al.: Fast protein backbone generation with se (3) flow matching. arXiv preprint arXiv:2310.05297 (2023)

[141] Song, Y., Gong, J., Xu, M., Cao, Z., Lan, Y., Ermon, S., Zhou, H., Ma, W.-Y.: Equivariant flow matching with hybrid probability transport for 3d molecule generation. Advances in Neural Information Processing Systems **36**, 549–568 (2023)

[142] Hurst, A., Lerer, A., Goucher, A.P., Perelman, A., Ramesh, A., Clark, A., Ostrow, A., Welihinda, A., Hayes, A., Radford, A., et al.: Gpt-4o system card. arXiv preprint arXiv:2410.21276 (2024)

[143] Esser, P., Kulal, S., Blattmann, A., Entezari, R., Müller, J., Saini, H., Levi, Y., Lorenz, D., Sauer, A., Boesel, F., *et al.*: Scaling rectified flow transformers for high-resolution image synthesis. In: Forty-first International Conference on Machine Learning (2024)

[144] Abramson, J., Adler, J., Dunger, J., Evans, R., Green, T., Pritzel, A., Ronneberger, O., Willmore, L., Ballard, A.J., Bambrick, J., *et al.*: Accurate structure prediction of biomolecular interactions with alphafold 3. Nature **630**(8016), 493–500 (2024)

[145] Salakhutdinov, R.: Learning deep generative models. Annual Review of Statistics and Its Application **2**(1), 361–385 (2015)

[146] Peyré, G., Cuturi, M., *et al.*: Computational optimal transport: With applications to data science. Foundations and Trends® in Machine Learning **11**(5-6), 355–607 (2019)

[147] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N.,

Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017)

[148] Corso, G., Stark, H., Jegelka, S., Jaakkola, T., Barzilay, R.: Graph neural networks. Nature Reviews Methods Primers **4**(1), 17 (2024)

[149] Geiger, M., Smidt, T.: e3nn: Euclidean neural networks. arXiv preprint arXiv:2207.09453 (2022)

[150] Krishnapriyan, A., Gholami, A., Zhe, S., Kirby, R., Mahoney, M.W.: Characterizing possible failure modes in physics-informed neural networks. Advances in neural information processing systems **34**, 26548–26560 (2021)

Acknowledgements. This research used resources of the National Energy Research Scientific Computing Center (NERSC), a U.S. Department of Energy User Facility, using NERSC award DDR-ERCAP 0034574 awarded to AM. This work was also financially supported by the U.S. National Science Foundation [grant number DBI2308699, awarded to JC] and the U.S. National Institutes of Health [grant numbers: R01GM146340 and R01GM093123, awarded to JC].

Author Contributions.

- **Alex Morehead:** Investigation; writing - original draft; writing - reviewing and editing.
- **Lazar Atanackovic:** Investigation; writing - original draft; writing - reviewing and editing.
- **Akshata Hegde:** Investigation; writing - original draft; writing - reviewing and editing.
- **Yanli Wang:** Investigation; writing - original draft; writing - reviewing and editing.
- **Frimpong Boadu:** Investigation; writing - original draft; writing - reviewing and editing.
- **Joel Selvaraj:** Investigation; writing - original draft; writing - reviewing and editing.
- **Alexander Tong:** Investigation; writing - original draft; writing - reviewing and editing.
- **Aditi Krishnapriyan:** Investigation; writing - original draft; writing - reviewing and editing.
- **Jianlin Cheng:** Conceptualization; investigation; writing - challenges and future perspectives, writing - reviewing and editing.

Competing interests. The authors declare no competing interests.

Related links.

- Accompanying paper list as a GitHub repository: https://github.com/amorehead/awesome-generative-flows.
- Meta AI's flow matching guide and code [15]: https://ai.meta.com/research/publications/flow-matching-guide-and-code.

- A PyTorch-friendly library [96] for training conditional flow matching models [18]: https://github.com/atong01/conditional-flow-matching.

Key points.

- Flow matching is a generative artificial intelligence technology that can be applied to numerous problems in computer vision, natural language processing, and bioinformatics.
- Flow matching models can learn to transport samples between any pair of data distributions.
- Flow matching has contributed greatly to computational biomolecule prediction and design, biomolecular dynamics modeling, cellular data modeling, and bioimaging.
- Although flow matching is promising compared to well-known generative modeling methods such as diffusion models and generative adversarial networks, applying it effectively to individual problems often requires considerable domain expertise.

Tables.

Method name	Year	Brief description	Source Code
Neural Autoregressive Flows [97]	2018	Foundational model	GitHub
Neural (latent) ODE [98]	2018	Foundational model	GitHub
GLOW [99]	2018	Foundational generative flows	GitHub
FFJORD [100]	2019	Improves computational efficiency of [98]	GitHub
Augmented Neural ODEs [101]	2019	Foundational neural ODE	GitHub
Residual Flows [102]	2019	Invertible generative flows	GitHub
Graph Normalizing Flows [103]	2019	Generative geometric flows	Github
Stochastic Normalizing Flows [104, 105]	2020	Generative stochastic flows	GitHub
Equivariant Normalizing Flows [106]	2021	Generative equivariant flows	GitHub
Diffusion Normalizing Flow [107]	2021	Generative stochastic flows	GitHub
Monte Carlo Flows [108]	2022	Flows for MCMC	GitHub
Divergence-free Neural Conservation Laws [109]	2022	Flows with divergence-free neural networks	GitHub
Rectified Flows [20]	2022	Straight and fast flow trajectories	GitHub
Flow Matching for Generative Modeling [13, 15]	2022	Generative flows with Gaussian probability paths	GitHub
Stochastic Interpolants (InterFlow) [27, 110, 111]	2022	Building flows with stochastic interpolants & diffusion	GitHub
RIVER [112]	2022	Flow matching for video prediction	GitHub
Point Straight Flow [113]	2022	Generating point clouds with Straight and fast flows	GitHub
Action Matching [114]	2023	Generative optimal transport with arbitrary paths	GitHub
OT Conditional Flow Matching [18]	2023	Generative (conditional) continuous flows	GitHub
Fast-ODE [115]	2023	Minimizing trajectory curvature of ODE-based generative models	GitHub
Latent Flow Matching [116]	2023	Flow matching in the latent space	GitHub
$[SF]^2M$ [73]	2023	Simulation-free stochastic dynamics	GitHub
Riemannian Flow Matching [117]	2023	Generative geometric flows	GitHub
Equivariant Flow Matching [118]	2023	Generative equivariant flows	GitHub
Wasserstein Lagrangian Flows [119]	2023	Unified flows and optimal transport	GitHub
FM Boosting [120]	2023	Boosting latent diffusion models with flow matching	GitHub
Motion Flow Matching [121]	2023	Flow matching for human motion synthesis and editing	GitHub
InstaFlow [122]	2023	Flows for accelerating text-to-image generation	GitHub
Multimodal Flow Matching [28]	2024	Continuous-discrete data generation	GitHub
Functional Flow Matching [123]	2024	Flow matching for infinite dimensional spaces	GitHub
Dirichlet Flow Matching [55]	2024	Discrete data generation	GitHub
Discrete Flow Matching [124]	2024	Discrete data generation	GitHub
Fisher Flow Matching [56]	2024	Discrete data generation	GitHub
Bellman Optimal Stepsize Straightening [125]	2024	Distilling and fine-tuning flow matching generative models	GitHub
Consistency Flow Matching [126]	2024	Learning strait flows with self-consistency in velocity fields	GitHub
Metric Flow Matching [79]	2024	Flows on the data manifold	GitHub
Depth FM [127]	2024	Flow matching for depth estimation	GitHub
Probabilistic Forecasting with Interpolants [128]	2024	Stochastic interpolants for probabilistic forecasting of dynamical systems	GitHub
SemFlow [129]	2024	Flows for semantic segmentation & semantic image synthesis	GitHub
Preference Flow Matching [130]	2024	Flow matching for preference-based reinforcement learning	GitHub
COT-FM [131]	2024	Conditional optimal transport with simulation-free flows	GitHub
FlowSeq [132]	2024	Flow matching for conditional text generation	GitHub
Statistical Flow Matching [133]	2024	Flows on the manifold of parameterized probability measures	GitHub
Optimal Flow Matching [134]	2024	Shorter flow trajectories	GitHub
Discrete Guidance [135]	2024	Guidance of discrete state space flow and diffusion models	GitHub
Meta Flow Matching [75]	2024	Flows across measures and modeling sample interactions	GitHub
Trajectory Flow Matching [136]	2024	Simulation-free modeling of stochastic time-series	GitHub
Generator Matching [21]	2024	Generalized multi-modal flow matching	GitHub
Wasserstein Flow Matching [78]	2024	Flows for families of distributions	GitHub
Multi-Marginal Flow Matching [83]	2025	Flows with smooth spline-based interpolation	GitHub
Contrastive Flow Matching [137]	2025	Unique conditional flows	GitHub
TFG-Flow [138]	2025	Continous-discrete flow guidance	GitHub
SDE Matching [139]	2025	Simulation-free latent SDEs	GitHub

Table 1 Open-source methods for implementing and improving flow matching models. O(S)DE, ordinary (stochastic) differential equation; MCMC, Markov chain Monte Carlo.

Research area	Applications	Method name	Flow conditioning	Network architecture	Source Code
Molecular modeling	Protein sequence generation	ProtFlow [30]	Conditioned	Transformer	GitHub
	Protein structure generation	FrameFlow [140]	Unconditioned	SE(3)-equivariant transformer	GitHub
	Protein structure generation	FoldFlow [23, 24]	Unconditioned	SE(3)-equivariant transformer	GitHub
	Protein side-chain packing	FlowPacker [25]	Conditioned	SE(3)-equivariant transformer	GitHub
	Protein sequence & structure generation	Multiflow [28]	Unconditioned	SE(3)-equivariant transformer	GitHub
	Protein sequence & structure generation	CoFlow [29]	Conditioned	Transformer	GitHub
	Protein sequence & structure generation	OriginFlow [31]	Unconditioned/Conditioned	Transformer	GitHub
	Antibody protein sequence & structure generation	FlowDesign [33]	Conditioned	SE(3)-equivariant transformer	-
	Peptide protein sequence & structure generation	D-Flow [32]	Conditioned	Transformer	GitHub
	Small molecule generation	MolFM [141]	Unconditioned	Geometric GNN	GitHub
	Small molecule generation	SemlaFlow [48]	Unconditioned	Geometric GNN	GitHub
	Small molecule generation	FlowMol [49]	Unconditioned	Geometric GNN	GitHub
	Small molecule conformer prediction	ET-Flow [50]	Conditioned	Transformer	GitHub
	Small molecule & materials generation	ADiT [26]	Unconditioned	All-atom transformer	GitHub
	DNA sequence generation	Dirichlet FM [55]	Unconditioned	Transformer with FM on simplex	GitHub
	DNA sequence generation	Fisher-Flow [56]	Unconditioned	Hyperspherical (Fisher-Rao) FM	GitHub
	RNA sequence generation	RNACG [57]	Conditioned	Multimodal Diffusion Transformer with Dirichlet FM	-
	RNA sequence & structure generation	RNAFlow [58]	Conditioned	Geometric GNN	GitHub
	Ligand-conditioned RNA sequence & structure generation	RiboFlow [59]	Conditioned	SE(3) flow with torsion angle modeling	-
	Protein-conditioned RNA sequence & structure generation	RNA-EFM [60]	Conditioned	Energy-based SE(3) flow with protein contacts	-
	RNA structure generation	RNA-FrameFlow [61]	Unconditioned	Torsion-based SE(3)-equivariant transformer	GitHub
	RNA structure prediction	RNAbpFlow [64]	Conditioned	Base-pair-augmented SE(3)-equivariant transformer	GitHub
	Biomolecular interactions	FlowDock [63]	Conditioned	SE(3)-equivariant transformer	GitHub
	Biomolecular interactions	NeuralPLexer3 [64]	Conditioned	All-atom transformer	GitHub
	Biomolecular interactions	FlexDock [65]	Conditioned	SE(3)-equivariant GNN	GitHub
	Biomolecular dynamics	AlphaFlow [67]	Conditioned	SE(3)-equivariant transformer	GitHub
	Biomolecular dynamics	FMRC [68]	Conditioned	Fully-connected neural network	GitHub
	Biomolecular dynamics	MDGen [69]	Conditioned	Transformer	GitHub
	Biomolecular dynamics	OM-TPS [70]	Conditioned	Diffusion or flow-based neural network	GitHub
Single-cellular modeling	Cell phenotype modeling	CellFlow [71]	Conditioned	Residual neural network	GitHub
	Phenotype generation	CFGen [72]	Compositional (multi-attribute)	Flow matching with discrete likelihoods	GitHub
	Simulation-free flows	CFM [18]	Conditioned	Conditional FM (regression)	GitHub
	Stochastic bridges	[SF]^{2}M [73]	Unconditioned	Score-FM	GitHub
	Stochastic flow alignment	GENOT [74]	Unconditioned	Entropic Gromov–Wasserstein flows	GitHub
	Population-conditioned flow modeling	Meta FM [75]	Population-conditioned	FM-based GNN	GitHub
	Distribution-level modeling	Wasserstein FM [78]	Distributional	Bures–Wasserstein FM	GitHub
	Manifold-constrained flows	Metric FM [79]	Metric-conditioned	Geodesic interpolant + flow field	GitHub
	Interaction-aware bridges	CytoBridge [80]	Conditioned (mean-field)	Unbalanced mean-field Schrödinger bridge	-
	Rotational dynamics	Curly-FM [76]	Velocity-conditioned	Schrödinger bridge with drift	-
	Growth-aware flows	VGFM [81]	Conditioned (growth + velocity)	Flow and growth field regression	-
	Multi-context flow learning	MMFM [83]	Conditioned (Time + condition)	Global vector field + spline guidance	GitHub
Multi-cellular modeling	Cell imaging	CellFlux [84]	Conditioned	U-Net	GitHub
Bioimaging	Cryo-EM denoising	CryoFM [91]	Conditioned	Transformer	-
	Microscopic Image Segmentation	FlowSDF [92]	Conditioned	U-Net	GitHub

Table 2 Flow matching methods for bioinformatics. SE(3), the special Euclidean group in $\mathbb{R}^3$; GNN, graph neural network; FM, flow matching; U-Net, U-shaped neural network.

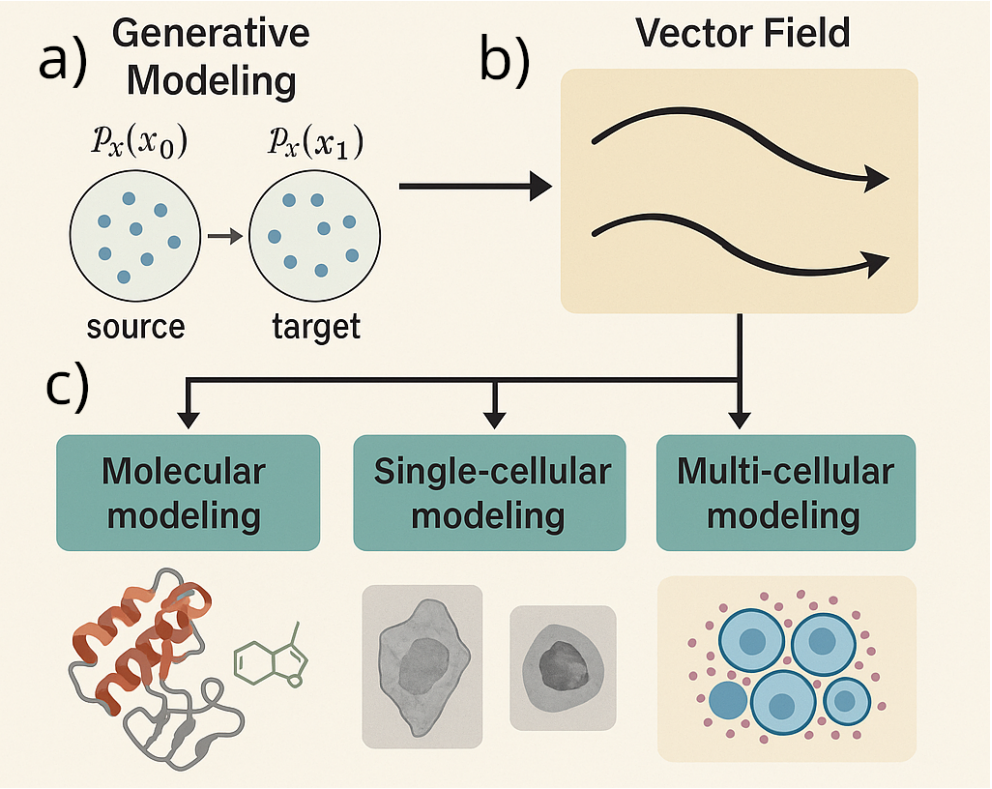

Fig. 1 An overview of flow matching and its applications in bioinformatics. a, Generative modeling for optimal transport of samples derived from a source distribution to a target distribution. **b**, An illustration of flow matching's standard vector field parametrization. **c**, Application areas of flow matching in bioinformatics relevant to developing an AI-based virtual cell.

Figures.

Figure legends.

- Figure 1: An overview of flow matching and its applications in bioinformatics towards building an AI-based virtual cell.
- Figure 2: A timeline illustrating the history of flow matching model development and the application of flow matching methods in bioinformatics.

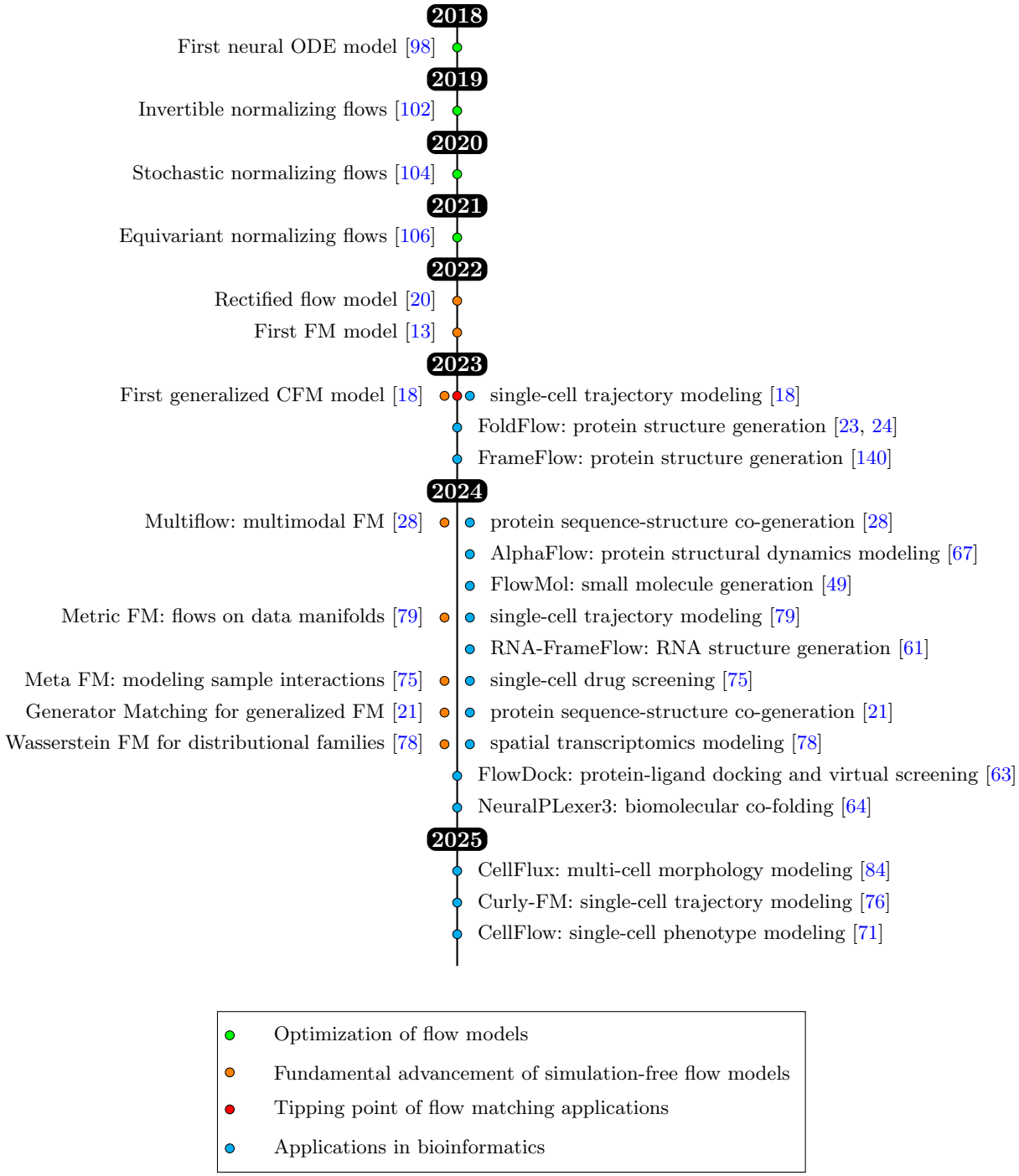

Fig. 2 Timeline of flow matching model development and applications in bioinformatics.

Boxes.

Box 1

Deep learning and generative modeling

Deep learning (DL) [12] represents a subset of artificial intelligence and machine learning research focused on developing many-layer (i.e., deep) artificial neural networks. Each neuron in an artificial neural network receives input data features and provides output data representations to a subsequent layer of interconnected neurons. Through their hierarchical structure, deep neural networks can learn increasingly abstract yet informative representations of the network's input data, which can be used for predicting properties of unseen data or generating new data entirely.

In the realm of generative modeling, DL underpins advanced techniques such as diffusion models [16] and flow matching [13]. Diffusion models generate data by simulating a process that gradually transforms simple noise into complex structures, achieving impressive results in image, text, and graph synthesis. Flow matching, a related approach, trains models to learn optimal transport paths between data distributions, offering efficient and scalable solutions for generating high-quality samples. These methods exemplify the versatility of DL in capturing and reproducing the underlying distributions of complex datasets. Some of the most well-known examples of successful generative DL applications include GPT-4o for multi-modal natural language understanding and reasoning [142], Stable Diffusion for image generation [143], and AlphaFold 3 for biomolecule 3D structure prediction [144].

Box 2

Key concepts related to flow matching models

Generative models [145]: Models that learn to generate new data samples from a learned distribution, often by transforming noise into structured outputs such as images, sequences, or graphs. These form the basis of most generative AI applications to date.

Optimal transport [146]: The study of transforming probability distributions by minimizing a cost function over mappings, often used to compare or interpolate between distributions.

Continuous normalizing flows [98] (CNFs): Generative models that transform a simple base distribution into a target distribution using invertible flows parameterized by neural ordinary differential equations, allowing exact likelihood computation.

Conditional flow matching [15] (CFM): A simulation-free method for training continuous normalizing flows by regressing a vector field to known conditional transitions, enabling efficient and scalable generative modeling.

Rectified flow [20]: A simplified flow model first popularized in the field of computer vision, where data points follow straight-line paths in latent space, which improves sample efficiency and aligns with optimal transport objectives.

Vector field parametrization [15]: A core modeling component in flow matching methods, where a neural network is trained to predict the time-dependent velocity vector that transports data points along learned trajectories from a source to the target distribution.

Diffusion models [8]: A class of generative models that learn to reverse a diffusion (noise injection) process, closely related to flow matching (FM) methods but typically trained with score-based objectives over noisy data.

Transformers [147]: Neural network architectures based on self-attention mechanisms, widely used in sequence and graph modeling, including in flow-based generative tasks due to their scalability and expressiveness.

Graph neural networks [148] (GNNs): Architectures that operate on graph-structured data by iteratively updating node representations via message passing, commonly used when modeling social, physical, or biomolecular structures.

SE(3)-equivariant networks [149]: Neural network models that preserve geometric equivariance under the special Euclidean group in $\mathbb{R}^3$ (i.e., the group SE(3)) consisting of 3D rotations and translations, important for learning over molecular or physical systems where symmetry is important.

Box 3

A pragmatic guide for developing and applying flow matching models in bioinformatics

1. Define the mapping task. Clearly identify the biological states or data distributions between which you aim to learn a mapping (e.g., perturbed vs. unperturbed cellular states, random vs. ground-truth 3D structures, or latent phenotype trajectories). Flow matching is best suited for problems that can be expressed as transporting samples between two distributions in $\mathbb{R}^d$.

2. Choose appropriate data representations. Select feature encodings that capture relevant biological information while being compatible with neural network inputs. For molecules, this may include graph-based representations or 3D atomic coordinates; for cellular data, it may involve gene expression vectors, imaging embeddings, or latent phenotypes.

3. Select a model class and vector field parameterization. Use architectures tailored to the domain: SE(3)-equivariant or (rotation-augmented) transformer-based networks for 3D molecular modeling, convolutional or transformer-based encoders for high-dimensional cellular data, and GNNs for biomolecular interaction tasks. Consider ensuring the vector field respects certain (non-trivial) symmetries when necessary.

4. Define the source distribution p_0 and sampling strategy. While a standard Gaussian is often used for p_0, FM permits more biologically meaningful priors (e.g., empirical distributions or structured noise priors) when justified by domain knowledge.

5. Design the training objective and probability path. Choose a probability path p_t (e.g., linear interpolation, optimal transport geodesic) and loss formulation (e.g., squared vector field deviation or dual potential-based objectives) appropriate for the application and model capacity.

6. Validate with domain-relevant metrics. Evaluate generated samples using biologically meaningful metrics, such as structural accuracy [64] (e.g., RMSD, TM-score), functional annotations, or predictive consistency with downstream tasks.

7. Integrate constraints when applicable. For improved biological fidelity, incorporate inductive biases such as geometric constraints, known interaction priors, or energy functions, either directly in the model or during post-processing. However, note that doing so may make model training more complex or challenging [150].

8. Use and contribute to open-source FM libraries. Several frameworks support FM training, including Meta AI's beginner-friendly `flow_matching` GitHub repository. Starting with open-source implementations can greatly accelerate model development and reproducibility.

Videos. Illustrations of generative flow matching sampling trajectories driven by the FlowDock model for protein-ligand docking:

- https://github.com/BioinfoMachineLearning/FlowDock/blob/7713d103ab40f32299d4c2b91c3cc5d94cfc3aa3/img/6I67.gif
- https://github.com/BioinfoMachineLearning/FlowDock/blob/7713d103ab40f32299d4c2b91c3cc5d94cfc3aa3/img/L1003.gif
- https://github.com/BioinfoMachineLearning/FlowDock/blob/7713d103ab40f32299d4c2b91c3cc5d94cfc3aa3/img/L3068.gif
- https://github.com/BioinfoMachineLearning/FlowDock/blob/7713d103ab40f32299d4c2b91c3cc5d94cfc3aa3/img/L3132.gif
- https://github.com/BioinfoMachineLearning/FlowDock/blob/7713d103ab40f32299d4c2b91c3cc5d94cfc3aa3/img/L3158.gif
- https://github.com/BioinfoMachineLearning/FlowDock/blob/7713d103ab40f32299d4c2b91c3cc5d94cfc3aa3/img/T1152.gif