Efficient 3D Molecular Generation with Flow Matching and Scale Optimal Transport

Anonymous Authors[1]

Abstract

Generative models for 3D drug design have gained prominence recently for their potential to design ligands directly within protein pockets. Current approaches, however, often suffer from very slow sampling times or generate molecules with poor chemical validity. Addressing these limitations, we propose *Semla*, a scalable E(3)-equivariant message passing architecture. We further introduce a molecular generation model, *MolFlow*, which is trained using flow matching along with scale optimal transport, a novel extension of equivariant optimal transport. Our model produces state-of-the-art results on benchmark datasets with just 100 sampling steps. Crucially, *MolFlow* samples high quality molecules with as few as 20 steps, corresponding to a two order-of-magnitude speed-up compared to state-of-the-art, without sacrificing performance. Furthermore, we highlight limitations of current evaluation methods for 3D generation and propose new benchmark metrics for unconditional molecular generators. Finally, using these new metrics, we compare our model's ability to generate high quality samples against current approaches and further demonstrate *MolFlow*'s strong performance.

1. Introduction

Generative models for 3D drug design have recently seen a surge of interest due to their potential to design binders directly within protein pockets. Some recently proposed models have attempted to directly generate ligands within protein pockets (Peng et al., 2022; Guan et al., 2023; Schneuing et al., 2022). More thorough analysis, however, revealed that many of these models generate ligands with unrealistic binding poses (Harris et al., 2023). Others have attempted to train unconditional 3D molecular generators as a starting point (Hoogeboom et al., 2022; Song et al., 2024; Vignac et al., 2023; Hua et al., 2024; Morehead & Cheng, 2023; Xu et al., 2024; Le et al., 2023). Specifically, models which apply diffusion (Ho et al., 2020; Song et al., 2020) to molecular coordinates have been particular popular. However, these models also suffer significant practical limitations; namely, they almost all require hundreds or even thousands of forward passes during generation, making them impractical for most downstream applications. Many also generate chemically unrealistic or poor quality samples when applied to datasets of drug-like molecules.

For molecular generators which represent molecules as strings or 2D graphs, fine-tuning for specific protein pockets has proven very fruitful (Blaschke et al., 2020; Loeffler et al., 2024; Atance et al., 2022) and is currently standard practice in the field. Frequently, these models are guided into optimised chemical spaces using reinforcement learning (RL). This approach, while very effective, requires that high quality molecules can be sampled very quickly. Existing 3D molecular generators, which use fully-connected message passing, exhibit very poor scaling to larger molecules and larger model sizes. State-of-the-art unconditional generators (Vignac et al., 2023; Le et al., 2023) take minutes to sample a single batch, making them impractical for RL-based fine-tuning.

In this work we tackle this problem from two directions. Firstly, we introduce a novel equivariant architecture for 3D molecular generation which exhibits significantly better efficiency and scalability than existing approaches. Even with 3 times as many parameters as the current state-of-the-art, our model processes a batch of molecules more than twice as quickly. Secondly, we introduce a novel type of optimal transport, *scale optimal transport*, which extends equivariant optimal transport (Klein et al., 2024; Song et al., 2024) to account for different sizes of molecules. Using our model architecture, we then train an unconditional molecular generator, *MolFlow*, using flow matching with scale optimal transport. With only 100 sampling steps our model outperforms all existing approaches on two benchmark datasets. We further show that with as few as 20 sampling steps *MolFlow*'s performance is comparable to the existing state-

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

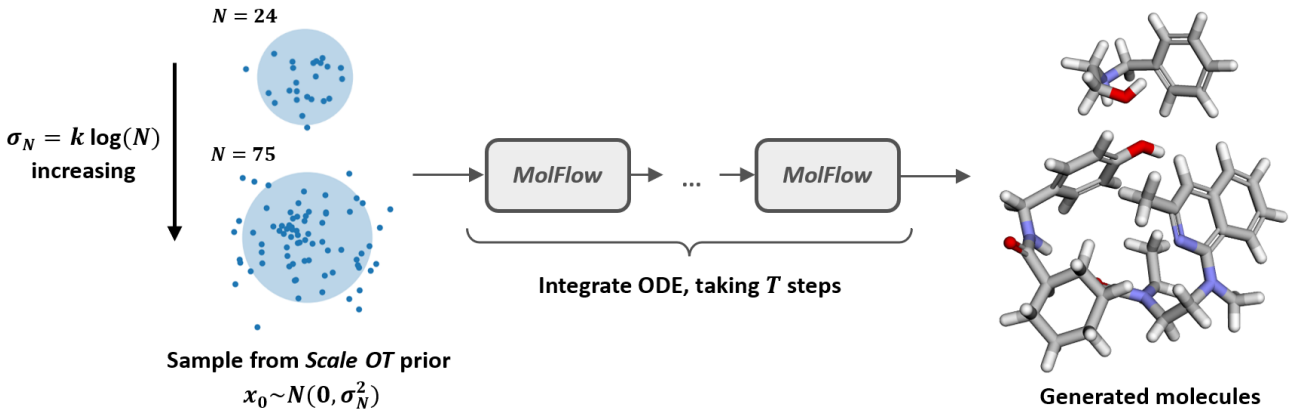

Figure 1. An overview of our approach. Firstly, prior coordinate noise is sampled according to the respective scale of the molecules (atom and bond type noise is not shown for brevity). Our trained model, *MolFlow*, transports this noise to the data distribution to generate a molecular sample.

of-the-art while producing a 2-orders-of-magnitude speedup in sampling times. Finally, we highlight issues with current evaluation metrics and introduce two new benchmark metrics for unconditional molecular generation to combat these. We provide an overview of our molecular generation approach using scale optimal transport in Figure 1.

2. Background

Flow Matching Flow matching seeks to learn a generative process which transports samples from a noise distribution p_{noise} to samples from a data distribution p_{data}. Conditional flow matching (CFM) has emerged in different flavors as an effective way to train flow matching models (Albergo & Vanden-Eijnden, 2023; Liu et al., 2022b; Lipman et al., 2022). CFM works in a simulation-free manner by interpolating between noise and a data sample $x_1 \sim p_{\text{data}}(\mathbf{x}_1)$. To do this a time-dependent conditional flow $p_{t|1}(\cdot|\mathbf{x}_1)$ is defined, from which a conditional vector field $u_t(\mathbf{x}_t|\mathbf{x}_1)$ can be derived. Typically, a model $v_t^\theta(\mathbf{x}_t)$ is trained to regress the vector field, but other formulations (Campbell et al., 2024; Stark et al., 2023) have trained a model to estimate the distribution $p_{1|t}^\theta(\cdot|\mathbf{x}_t)$, which reconstructs clean data from noisy data. The vector field can then be constructed using the expectation:

$$v_t^\theta(\mathbf{x}_t) = \mathbb{E}_{\tilde{\mathbf{x}}_1 \sim p_{1|t}^\theta(\mathbf{x}_1|\mathbf{x}_t)}([u_t(\mathbf{x}_t|\tilde{\mathbf{x}}_1)]) \tag{1}$$

Samples can then be generated by integrating the vector field with an arbitrary ODE solver.

The static optimal transport (OT) problem seeks to identify a coupling π between samples from two distributions that minimises a cost function (Monge, 1781). Identifying the π which is the *2-Wasserstein OT map* defined by

$$W_2^2 = \inf_{\pi \in \Pi} \int c(x_0, x_1)\, \pi(\mathrm{d}x_0, \mathrm{d}x_1) \tag{2}$$

where $c(x_0, x_1) = \|x_0 - x_1\|_2$ is the cost function and Π is the set of all couplings, has proven to be a successful strategy to improve the efficiency of CFM. Previous work has proposed mini-batched (Pooladian et al., 2023; Tong et al., 2023) and group equivariant (Klein et al., 2024; Song et al., 2024) variations of this. In equivariant OT, the cost function is replaced by $\min_{g \in \mathcal{G}} \hat{c}(x_0, x_1) = \|x_0 - T_g x_1\|_2$ where T_g is a linear representation of $g \in \mathcal{G}$ (Serre et al., 1977).

Invariance and Equivariance Group invariance and equivariance are crucial properties to consider when designing models for 3D molecular generation. For a group $\mathcal{G}$, if T_g and P_g are linear representations of a group element $g \in \mathcal{G}$, then a probability density $p(\mathbf{x})$ is considered *invariant* with respect to $\mathcal{G}$ iff $p(T_g\mathbf{x}) = p(\mathbf{x})$ for all $g \in \mathcal{G}$, and a function f is considered *equivariant* to $\mathcal{G}$ iff $T_g(f(\mathbf{x})) = f(P_g(\mathbf{x}))$ for all $g \in \mathcal{G}$.

Köhler et al. (2020) showed that, if a base density $p_0(\mathbf{x}_0)$ is $\mathcal{G}$-invariant and a target density $p_1(\mathbf{x}_1)$ is generated by following a $\mathcal{G}$-equivariant vector field, then $p_1(\mathbf{x}_1)$ is also $\mathcal{G}$-invariant. We use this finding to ensure that the density of molecular coordinates learned by our model is $\mathcal{G}$-invariant by only applying equivariant updates and sampling coordinate noise from an isotropic Gaussian. For molecular generation we are concerned with the group $\mathcal{G} = \mathrm{E}(3) \times S_N$ where $\mathrm{E}(3)$ is the Euclidean group in 3 dimensions, encompassing translations, rotations and reflections, and S_N is the symmetric group for a set with N elements – the group of all possible permutations.

3. The *Semla* Architecture

Existing state-of-the-art models for 3D molecular generation (Hoogeboom et al., 2022; Vignac et al., 2023; Le et al., 2023) use fully-connected, multi-layer perceptron (MLP)-based message passing layers. However, the computational cost of such layers scales quadratically in both the feature dimension and the number of atoms. Consequently, these layers become a significant computational bottleneck when scaling to larger, drug-like molecules.

To alleviate this problem we propose *Semla* - a scalable equivariant model which uses multi-head latent graph attention, where message passing is performed on compressed latent representations. This extension allows us to scale the dimensionality of the node features and the number of learnable model parameters without leading to prohibitory increases in computational cost. We illustrate the *Semla* architecture in Fig. 2 and further expand on each component below.

Similarly to previous approaches, *Semla* makes use of $\mathrm{E}(3)$ invariant and equivariant features. Enforcing group symmetry provides strong inductive biases and improves sample complexity (Bietti et al., 2021; Tahmasebi & Jegelka, 2023; Hoogeboom et al., 2022). However, unlike previous molecular generation models, *Semla* does not distinguish between molecular coordinates and equivariant feature vectors, but rather treats them as a single learnable representation, which we refer to as *coordinate sets*. Coordinate sets are analogous to the sets of 3D geometric equivariant vectors found in other equivariant architectures (Jing et al., 2020; Schütt et al., 2021; Le et al., 2022). We argue that this representation has two key benefits over previous molecular generation approaches. Firstly, learnable coordinate sets provide much more expressivity than models which store only one set of coordinates per molecule (Hoogeboom et al., 2022; Vignac et al., 2023). Additionally, a joint representation of equivariant features allows for a simpler update mechanism – we can simply apply linear projections (without bias) to create and update the coordinate sets while maintaining equivariance.

To ensure stable training we use normalisation layers throughout the model. LayerNorm (Ba et al., 2016) is applied to invariant features, and, for equivariant features, we adapt the normalisation scheme from MiDi (Vignac et al., 2023) to allow for multiple coordinate sets. We hypothesise that this allows coordinate sets to learn equivariant features of different length scales, which helps to circumvent the problem of molecules of different sizes being normalised to have the same average vector norm. We further extend the normalisation to ensure that coordinates are zero-centred. An additional zero-centering, which we apply at the end of the model, ensures that the learned density is translation invariant (Garcia Satorras et al., 2021; Xu et al., 2022). For the remainder of this paper we will use $\phi_{inv}(\cdot)$ and $\phi_{equi}(\cdot)$ to refer to the normalisation functions for invariant and equivariant features, respectively.

Throughout this paper we denote invariant and equivariant features for an atom i as $\mathbf{h}_i \in \mathbb{R}^D$ and $\mathbf{x}_i \in \mathbb{R}^{S\times 3}$, respectively, where D is the model dimension and S is the number of coordinate sets. We also use N to refer to the number of atoms in the molecule. To simplify the notation we assume that operations applied to $\mathbf{x}_i$ implicitly correspond to the concatenation of the results of the operation applied to individual vectors, unless we make the coordinate set explicit using a superscript. For example, the norm of $\mathbf{x}_i$ is implicitly applied as $\|\mathbf{x}_i\| = \left[\|\mathbf{x}_i^1\|, \|\mathbf{x}_i^2\|, \ldots, \|\mathbf{x}_i^S\|\right]$.

3.1. Feature Feed-forward

The feed-forward component provides a simple feature update mechanism while also allowing the exchange of information between invariant and equivariant features. The feed-forward update is given as follows:

$$\tilde{\mathbf{h}}_k = \phi_{inv}(\mathbf{h}_k) \qquad \tilde{\mathbf{x}}_k = \mathbf{W}_\theta^1 \phi_{equi}(\mathbf{x}_k) \tag{3}$$

$$\mathbf{h}_i^{\text{ff}} = \mathbf{h}_i + \Phi_\theta(\tilde{\mathbf{h}}_i, \|\phi_{equi}(\mathbf{x}_i)\|) \tag{4}$$

$$\mathbf{x}_i^{\text{ff}} = \mathbf{x}_i + \mathbf{W}_\theta^2 \left(\sum_{j=1}^{S} \tilde{\mathbf{x}}_i^j \otimes \Psi_\theta(\tilde{\mathbf{h}}_i) \right) \tag{5}$$

where Φ_θ and Ψ_θ are learnable multi-layer perceptrons, $\mathbf{W}_\theta^1 \in \mathbb{R}^{P\times S}$ and $\mathbf{W}_\theta^2 \in \mathbb{R}^{S\times P}$ are learnable weight matrices with hyperparameter P, and $\otimes$ is the outer product. For simplicity we set $P = M$, the message size, which is defined below. Similarly to the transformer architecture (Vaswani et al., 2017), Φ_θ linearly maps features to $4D$, applies a non-linearity (we use SiLU (Elfwing et al., 2018) throughout) and then maps back to the model dimension D.

3.2. Equivariant Graph Attention

In this section, we introduce a novel attention mechanism for 3D graph structures. Like previously proposed attention mechanisms for equivariant architectures (Satorras et al., 2021; Le et al., 2022; Liao & Smidt, 2023), our model computes pairwise messages using a multi-layer perceptron (MLP). These pairwise messages are then split in two and passed to attention mechanisms for invariant and equivariant features, respectively. We describe each of these components in more detail below.

Latent Message Passing Pairwise messages are computed using a 2-layer MLP which combines invariant node features with pairwise dot products from the coordinate sets. Similarly to previous approaches we compute messages between all pairs of nodes in the graph. Unlike models such as EGNN (Satorras et al., 2021), MiDi (Vignac et al., 2023) and EQGAT (Le et al., 2022), however, we first compress

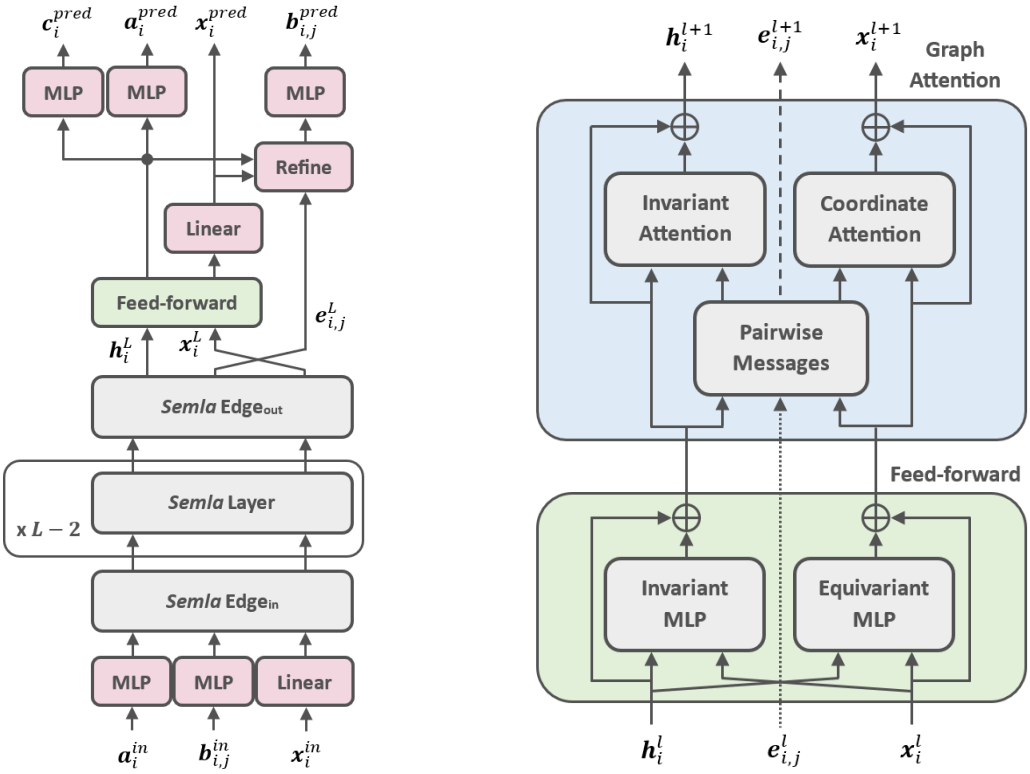

Figure 2. Architectural overview of a full *Semla* model (left) and one *Semla* layer (right).

the invariant node features into a smaller latent space, with dimensionality M, using a learnable linear map. This reduces the computational complexity of the pairwise MLP from $\mathcal{O}(N^2D^2)$ to $\mathcal{O}(N^2M^2)$ where $M \ll D$, leading to a significant reduction in the compute and memory overhead of the MLP, especially on larger molecules. It also allows us to scale the size of the invariant node features, D, independently of M. Formally, messages between nodes i and j, which are split into invariant and equivariant attention components, are computed as follows:

$$(\mathbf{m}_{i,j}^{(inv)}, \mathbf{m}_{i,j}^{(equi)}) = \Omega_\theta(\tilde{\mathbf{h}}_i, \tilde{\mathbf{h}}_j, \tilde{\mathbf{x}}_i \cdot \tilde{\mathbf{x}}_j) \tag{6}$$

$$\tilde{\mathbf{h}}_k = \mathbf{W}_\theta^3 \phi_{inv}(\mathbf{h}_k^{\text{ff}}) \qquad \tilde{\mathbf{x}}_k = \phi_{equi}(\mathbf{x}_k^{\text{ff}}) \tag{7}$$

where Ω_θ is the pairwise message MLP and $\mathbf{W}_\theta^3 \in \mathbb{R}^{M \times D}$ is a learnable weight matrix.

Invariant Feature Attention Once messages have been computed a softmax operation is applied to produce attention weights between pairs of nodes. These weights are then used to aggregate node features by taking a weighted average. Since the message vectors can, in general, be smaller than the node features, each scalar in the message vector attends to a fixed number of scalars within the node feature vectors. We note that this attention implementation generalises the attention mechanism found in EQGAT and related models such as the Point Transformer (Zhao et al., 2021), where each scalar in the message attends to exactly one scalar in the node features, and is very closely related to the multi-head attention mechanism adopted in GAT (Veličković et al., 2018; Brody et al., 2021) and the Transformer (Vaswani et al., 2017). We also make use of the recently proposed variance preserving aggregation mechanism (Schneckenreiter et al., 2024), which corresponds to multiplying the attended vectors by weights w_i^k. Overall, our invariant feature attention is computed as:

$$\alpha_{i,j}^k = \frac{\exp(m_{i,j}^{k,(inv)})}{\sum_{j'=1}^N \exp(m_{i,j'}^{k,(inv)})} \qquad w_i^k = \sqrt{\sum_{j=1}^N \alpha_{i,j}^k} \tag{8}$$

$$\tilde{\mathbf{h}}_i = \mathbf{W}_\theta^4 \phi_{inv}(\mathbf{h}_i^{\text{ff}}) \qquad \mathbf{a}_i^k = \sum_{j=1}^N \alpha_{i,j}^k \tilde{\mathbf{h}}_j^k \tag{9}$$

$$\mathbf{h}_i^{\text{out}} = \mathbf{h}_i^{\text{ff}} + \mathbf{W}_\theta^5 \Big(\Big\|_{k=1}^K w_i^k \mathbf{a}_i^k \Big) \tag{10}$$

where $\mathbf{W}_\theta^4 \in \mathbb{R}^{D \times D}$ and $\mathbf{W}_\theta^5 \in \mathbb{R}^{D \times D}$ are learnable weight matrices and $\|$ is the concatenation operation. Here, node features are split into K equally sized segments and each scalar attention score $\alpha_{i,j}^k$ attends to one segment $\tilde{\mathbf{h}}_j^k$. For simplicity we use $K = M$, the message size, which is 128 in all of our experiments.

Equivariant Feature Attention Similarly to the invariant features, messages for the equivariant features are used to apply an attention-based update. The attention function applied here is line with previous work (Satorras et al., 2021; Vignac et al., 2023; Le et al., 2022), however we extend it to allow for multiple coordinate sets. Notably, we also find the use of softmax normalisation on raw messages to be beneficial for overall model performance. Analogously to the invariant feature attention, we also apply variance preserving updates to the equivariant features. *Semla* attention for equivariant features is therefore defined as follows:

$$\hat{\mathbf{x}}_{i,j} = \frac{\tilde{\mathbf{x}}_j - \tilde{\mathbf{x}}_i}{\|\tilde{\mathbf{x}}_j - \tilde{\mathbf{x}}_i\|} \qquad \tilde{\mathbf{x}}_k = \mathbf{W}_\theta^6 \phi_{equi}(\mathbf{x}_k^{\text{ff}}) \tag{11}$$

$$\alpha_{i,j}^k = \frac{\exp(m_{i,j}^{k,(equi)})}{\sum_{j'=1}^N \exp(m_{i,j'}^{k,(equi)})} \tag{12}$$

$$a_i^k = \sum_{j=1}^N \alpha_{i,j}^k \hat{x}_{i,j}^k \qquad w_i^k = \sqrt{\sum_{j=1}^N \alpha_{i,j}^k} \tag{13}$$

$$\mathbf{x}_k^{\text{out}} = \mathbf{x}_k^{\text{ff}} + \mathbf{W}_\theta^7 \Big(\left[w_i^1 a_i^1, \ldots, w_i^K a_i^K \right]^T \Big) \tag{14}$$

3.3. Overall Architecture

As shown in Fig. 2 a full *Semla* model consists of a stack of L *Semla* layers along with embedding layers and MLPs for encoding the atom and bond types, and MLP prediction heads for producing unnormalised distributions for atoms, bonds and formal charges. Unlike EQGAT, our model does not carry edge features throughout the network. Instead, the first layer embeds bond information into the node features by passing the encoded bond features into the pairwise message module. Analogously, the final layer produces pairwise edge features which are then further updated through a bond refinement layer at the end of the network. This layer acts in a similar way to the pairwise message block described above but only updates the edge features. Absorbing bond information into the node features like this leads to a further increase in the efficiency of our model and, in our experiments, had little impact on generative performance.

All models we present in this paper are constructed from 12 *Semla* layers, with $D = 512$, $M = 128$ and $S = 64$. This corresponds to approximately 40M learnable parameters, which is more than 3 times as many than the current state-of-the-art, EQGAT-diff (Le et al., 2023). Despite this additional expressivity, our model processes batches more than twice as fast due to the efficiency improvements outlined above. We discuss this in more detail in the experiments section, below.

4. Flow Matching for Molecular Generation

To assess the ability of *Semla* to model distributions with $E(3) \times S_N$ symmetry we apply conditional flow matching (Lipman et al., 2022; Albergo & Vanden-Eijnden, 2023; Albergo et al., 2023; Liu et al., 2022b) with optimal transport to create a generative model for molecules, which we refer to as *MolFlow*. In this section we outline the training and sampling procedure for *MolFlow* with molecular structures. In the following we represent each molecule by a tuple $z = (\mathbf{x}, \mathbf{a}, \mathbf{b}, \mathbf{c})$ of coordinates, atom types, bond types and formal charges, respectively.

Training *MolFlow* As shown in existing conditional flow matching frameworks (Lipman et al., 2022; Albergo & Vanden-Eijnden, 2023; Campbell et al., 2024), training proceeds by firstly sampling: noise $z_0 \sim p_{\text{noise}}(z_0)$; data $z_1 \sim p_{\text{data}}(z_1)$; and a time $t \in [0, 1]$, and using these to sample from the time-dependent conditional flow $z_t \sim p_{t|1}(z|z_0, z_1)$. In practice, we apply *scale optimal transport* (OT) and sample noise from the scaled distribution $p_{\text{noise}}^{\pi}(z_0|n)$, where n is the number of atoms. We discuss this further, along with the OT transformation applied to the noise during training below. Previous work on molecular structure generation has found it beneficial to train models to predict data directly rather than noise (Le et al., 2023) or a vector field (Stark et al., 2023). *MolFlow* is therefore trained to learn a distribution $p_{1|t}^{\theta}(z_1|z_t)$ which predicts clean data from interpolated data using a *Semla* model with parameters θ. Different conditional flows are used for continuous and discrete data. For coordinates, we apply a linear interpolation with the addition of a small amount of Gaussian noise. Atom and bond types are sampled according to the recently proposed discrete flow models (DFM) framework (Campbell et al., 2024). The joint molecular interpolation is therefore given as follows:

$$\mathbf{x}_t \sim \mathcal{N}(t\mathbf{x}_1 - (1-t)\mathbf{x_0}, \sigma^2) \qquad t \sim \text{Beta}(\alpha, \beta) \tag{15}$$

$$a_t \sim \text{Cat}(t\delta(a_1) + (1-t)\frac{1}{|\mathcal{A}|}) \tag{16}$$

$$b_t \sim \text{Cat}(t\delta(a_1) + (1-t)\frac{1}{|\mathcal{B}|}) \tag{17}$$

Where $\mathcal{A}$ and $\mathcal{B}$ are the sets of atom and bond types, respectively, and $\delta(\cdot)$ is the one-hot encoding operation. We use (α, β) = (2.0, 1.0) and $\sigma = 0.2$ for all models presented in this paper. Notably, although our model predicts the formal charge for each atom, the charges do not participate in the generative flow-matching process and so there is no need to interpolate them.

After sampling a predicted molecule $\tilde{z}_1 \sim p_{1|t}^{\theta}(z_1|z_t)$ where $\tilde{z}_1 = (\tilde{\mathbf{x}}_1, \tilde{\mathbf{a}}_1, \tilde{\mathbf{b}}_1, \tilde{\mathbf{c}}_1)$, the model is trained with a mean-squared error loss function ($\mathcal{L}_{\text{MSE}}$) for coordinates, and cross-entropy losses ($\mathcal{L}_{\text{CE}}$) for atom types, bond types and

charges. The final loss for the model is then given by the weighted sum:

$$\mathcal{L}_{\text{MolFlow}} = \lambda_x \mathcal{L}_{\text{MSE}}(\tilde{\mathbf{x}}_1, \mathbf{x}_1) + \lambda_a \mathcal{L}_{\text{CE}}(\tilde{\mathbf{a}}_1, \mathbf{a}_1) \quad (18)$$

$$+\lambda_b \mathcal{L}_{\text{CE}}(\tilde{\mathbf{b}}_1, \mathbf{b}_1) + \lambda_c \mathcal{L}_{\text{CE}}(\tilde{\mathbf{c}}_1, \mathbf{c}_1) \quad (19)$$

We also make use of self-conditioning, which was originally proposed for diffusion models as way of reusing the model's previous prediction when sampling (Chen et al., 2023). To create a self-conditioned *MolFlow* model, we adopt the same training procedure as HarmonicFlow (Stark et al., 2023). We provide further details on this, along with the hyperparameters used for *MolFlow*, in Appendix C.

Sampling Molecules Once we have trained a *MolFlow* model, molecules can be generated by, firstly, sampling noise $z_0 \sim p^{\pi}_{\text{noise}}(z_0|n)$, and then integrating the ODE corresponding to the conditional flow $p_{t|1}$ from $t = 0$ to $t = 1$. For coordinates, the vector field corresponding to our choice of conditional flow is given by $\tilde{\mathbf{x}}_1 - \mathbf{x}_0$ where $\tilde{\mathbf{x}}_1$ is sampled from $p^{\theta}_{1|t}$ as shown above. We then apply an Euler solver to integrate the ODE with step sizes Δt as follows: $\mathbf{x}_{t+\Delta t} = \mathbf{x}_t + \Delta t(\tilde{\mathbf{x}}_1 - \mathbf{x}_0)$. We refer readers to DFM (Campbell et al., 2024) for the sampling procedure for atom and bond types. In practice, we found that taking logarithmically spaced steps resulted in better performance than using constant step sizes.

Scale Optimal Transport For molecular distributions the average length between atom coordinates and the centre of the molecules increases with the number of atoms N. In polymer theory the standard deviation of atom coordinates within self-avoiding polymer chains has been shown to scale with $\sqrt{N}$, known as the Flory radius (Rubinstein & Colby, 2003). Typically, however, for molecular generation models, coordinate noise is sampled from a unit Gaussian which doesn't reflect differences in molecular sizes. Instead we sample from a Gaussian distribution with a variance that depends on N, the number of atoms in $\mathbf{x}_1$. We produce a sample $\mathbf{x}^{\pi}_0$ from the noise distribution p^{π}_{noise} during training and inference as follows:

$$\mathbf{x}^{\pi}_0 = \begin{cases} f_\pi(\mathbf{x}_0, \mathbf{x}_1) & \text{training} \\ \mathbf{x}_0 & \text{inference} \end{cases} \qquad \mathbf{x}_0 \sim \mathcal{N}(0, \sigma^2_N) \quad (20)$$

Where $f_\pi(\mathbf{x}_0, \mathbf{x}_1)$ is the equivariant optimal transport (Klein et al., 2024; Song et al., 2024) transformation which corresponds to applying a permutation and rotation which minimises the transport cost (in this case the mean-squared error) between $\mathbf{x}_0$ and $\mathbf{x}_1$. Since existing 3D generative models already require N to be known at inference time, no additional restrictions are placed on the sampling when using scale OT.

Since it applies to long polymer chains, we consider the Flory radius as an upper-bound for small molecule drugs, and empirically find $\sigma_N = k \log(N)$ with $k = 0.2$ to be a good fit for a dataset of drug-like molecules. In Appendix A we provide further details on this approach and show that it leads to a significant reduction in the transport cost compared to equivariant optimal transport, especially for smaller molecules.

5. Experiments

In this section we provide results on benchmark 3D molecular generation tasks and compare the performance of our model to existing state-of-the-art approaches. In Appendix B, we provide results on ablation experiments which show how different components of the training, such as scale OT and self conditioning, affect performance on molecular generation tasks. We also provide samples from a *MolFlow* model trained on GEOM Drugs in Appendix D.

Evaluation Setup Two benchmark datasets, QM9 (Ramakrishnan et al., 2014) and GEOM Drugs (Axelrod & Gomez-Bombarelli, 2022), are used to assess *MolFlow*'s abilities as an unconditional molecular generator. Since QM9 contains only very small molecules GEOM Drugs serves a more useful benchmark for distinguishing model performance. We therefore provide the results on QM9 in Appendix B and discuss the GEOM Drugs results here. For both datasets we use the same data splits as MiDi and EQGAT-diff. To improve training times, however, we discard molecules with more than 72 atoms from the GEOM Drugs training set. This corresponds to about 1% of the training data. Validation and test sets are left unchanged. All metrics for *MolFlow* presented below are calculated by, firstly, sampling molecule sizes from the test set, and then generating molecules with the sampled number of atoms using the model, as outlined above. We use standard benchmark evaluation metrics: *atom stability*; *molecule stability*; *validity*; *uniqueness*; and *novelty*, which have been thoroughly described in previous works. We also provide a full description of these metrics in Appendix B. Additionally, for each baseline model we provide the number of function evaluations (NFE) required to sample one batch of molecules. These were taken from the respective publications, if available, or using the default values in the official codebases.

Molecular Generation Results Table 1 compares our model to MiDi and EQGAT-diff, which we regard as the current state-of-the-art, on this dataset. We do not include models which infer bonds since these models typically perform very poorly on larger molecules and often do not provide results for all evaluation metrics. We see that *MolFlow* outperforms existing state-of-the-art models on all key evalu-

Table 1. Molecular generation results on GEOM Drugs. Values for models we evaluated are given as an average over 5 runs, sampling 5000 molecules on each run, with standard deviations in subscripts.

Model	Atom Stab ↑	Mol Stab ↑	Valid ↑	Unique ↑	Novel ↑	NFE ↓
MiDi	**99.8**	91.6	77.8	**100.0**	**100.0**	500
EQGAT-diff	**99.8**$_{\pm 0.01}$	93.4$_{\pm 0.21}$	94.6$_{\pm 0.24}$	**100.0**$_{\pm 0.0}$	99.9$_{\pm 0.07}$	500
MolFlow (Ours)	**99.8**$_{\pm 0.01}$	**97.7**$_{\pm 0.07}$	**95.2**$_{\pm 0.22}$	**100.0**$_{\pm 0.0}$	99.6$_{\pm 0.09}$	100

ation metrics, while generating only marginally fewer novel molecules. In addition to requiring significantly fewer evaluation steps our model also requires much less compute for training. *MolFlow* trains for 200 epochs on a single Nvidia A100 GPU, compared to 800 epochs with 4 GPUs for EQGAT-diff.

Sampling Efficiency To further improve the efficiency of our model we tried varying the number of ODE integration steps and we provide a one-to-one comparison of these results with EQGAT-diff in Table 2. We also measure the time required by each model to sample 5000 molecules. For EQGAT-diff we measure the sample time using the given evaluation code with a batch size of 75, which was the largest multiple of 25 we could consistently fit on an Nvidia A100 40GB GPU. In the table we can see a stark difference in runtime between the two models. *MolFlow* provides comparable performance to EQGAT-diff with as few as 20 sampling steps, corresponding to a two order of magnitude speed-up in sampling time. Much of this improvement is due to the very small number of sampling steps, which we hypothesise is due to the use of optimal transport, since this aims to minimise the cost of transporting probability mass between distributions. However, we further see a two-fold improvement in the time per forward pass, which demonstrates the importance of designing efficient equivariant architectures for molecular generation.

Conformer Evaluation The validity and stability metrics presented here, however, only measure the 1D structure of the molecule (i.e. the SMILES string); they provide no information on the quality of the conformations. We can also see that recent models have saturated these metrics on both QM9 and GEOM Drugs, warranting other evaluation methods. To further compare model performance and to allow evaluation of 3D generation, we introduce energy and strain energy as new benchmark metrics for this task. The energy measures the quality of a conformer, considering typical bonded and non-bonded interactions. The energy $U(\mathbf{x})$ of a conformation is inversely related to its probability according to the Boltzmann distribution $p(\mathbf{x}) = Z^{-1} \exp\left(-U(\mathbf{x})/kT\right)$ where T is the temperature and k is the Boltzmann constant. The strain is given by the difference $U(\mathbf{x}) - U(\tilde{\mathbf{x}})$ where $\tilde{\mathbf{x}}$ is the *relaxed* (i.e. minimised) conformation for $\mathbf{x}$. We argue that these metrics provide a useful overview of the quality of the generated conformations and directly include measurements such as bond lengths and bond angles which have been proposed previously (Vignac et al., 2023). We use RDKit (Gred Landrum *et al.*, 2023) with an MMFF94 (Halgren, 1996) forcefield to calculate the energies and perform the minimisation.

Table 2 shows that *MolFlow* produces molecules with better energies and strain energies than EQGAT-diff, despite the significantly faster sampling time. Notably, however, molecules generated by EQGAT-diff have lower minimised energies than *MolFlow*, suggesting that their model is better at finding molecular conformations which have lower energy minima, while our model is better at producing lower strain energies. In ablation experiments in Appendix B we also show that the use of scale optimal transport is a crucial component for generating low energy conformations with our model.

6. Related Work

3D Molecular Generation In addition to the unconditional molecular generators we outlined above, a number of works have attempted to directly generate ligands within protein pockets (Peng et al., 2022; Guan et al., 2023; Schneuing et al., 2022). However, these models also suffer from the issues we outlined previously, including long-sampling times (100s - 1000s of seconds for 100 molecules (Schneuing et al., 2022)) and generating invalid chemical structures or molecules with very high strain energies (Harris et al., 2023). GraphBP (Liu et al., 2022a), an autoregressive model for protein-conditioned generation, is able to generate ligands faster, but suffers from worse docking scores than more recent diffusion models.

Flow Matching for 3D Structures Outside of small molecule design flow-matching has recently gained traction with generative models for biomolecules. FoldFlow (Bose et al., 2024) and FrameFlow (Yim et al., 2023) are both recently introduced flow-matching models for protein structure generation. Multiflow (Campbell et al., 2024) attempts to jointly generate protein sequence and structure and introduces the discrete flow models (DFM) framework for flow-matching generation of discrete data. Stark et al. (2024) also introduce a framework for flow-matching on discrete

Table 2. Comparison between EQGAT-diff and MolFlow with different numbers of sampling steps. Molecule stability and validity are given as percentages, energy and strain energy are measured in $\text{kcal} \cdot \text{mol}^{-1}$, NFE refers to the number of function evaluations required to sample one batch of molecules and sample time is measured by the number of seconds required to generate 5000 molecules, averaged over 5 runs.

Model	Mol Stab ↑	Valid ↑	Energy ↓	Strain ↓	NFE ↓	Sample Time ↓
EQGAT-diff	$93.4_{\pm 0.21}$	$94.6_{\pm 0.24}$	$148.8_{\pm 0.88}$	$140.2_{\pm 0.72}$	500	11521
MolFlow_{20}	$94.8_{\pm 0.28}$	$93.7_{\pm 0.36}$	$168.7_{\pm 3.1}$	$130.5_{\pm 3.1}$	20	167
MolFlow_{50}	$97.2_{\pm 0.25}$	$95.2_{\pm 0.40}$	$141.1_{\pm 2.1}$	$102.0_{\pm 2.1}$	50	412
MolFlow_{100}	$97.7_{\pm 0.07}$	$95.2_{\pm 0.22}$	$127.5_{\pm 2.1}$	$88.9_{\pm 1.3}$	100	821
Data	100.0	100.0	50.3	15.9	–	–

data, *DirichletFM*, and apply this to DNA sequence design. Finally, Verma et al. (2023) use a conjoined system of ODEs to train a model to jointly generate antibody sequences and structures.

7. Conclusion

In this work we have presented *Semla*, a novel equivariant message passing architecture which exhibits much better efficiency and scalability than existing approaches for unconditional molecular generation. Additionally, we introduced a novel type of optimal transport (OT), scale OT, which extends equivariant OT but noticeably reduces the transport cost. We then trained a *Semla* model for 3D molecular generation using flow matching with scale OT. Our model, *MolFlow*, achieves state-of-the-art results on molecular generation benchmarks and, crucially, is able to generate high quality molecules with only 20 sampling steps. We also highlighted issues with current molecular evaluation metrics and proposed the use of energy and strain for evaluating the quality of generated molecular conformations.

While we believe our model has made significant progress in solving key challenges for 3D molecular generators, many challenges remain. Firstly, the energies of the molecules generated by *MolFlow* are still significantly higher than that of the dataset; generating molecular coordinates with very high fidelity remains a problem for these models. Including further inductive biases or fine-tuning against an energy model could be an avenue to improve this in future work (Noé et al., 2019; Schreiner et al., 2023; Viguera Diez et al., 2024). Additionally, while *MolFlow* has shown significant efficiency improvements over existing methods, it still uses a full-connected message passing component, limiting it's scalability to larger molecular systems. We leave the further enhancement of the scalability of *Semla* to future work. We believe our model makes crucial step towards the practical application of 3D molecular generators, although we leave the integration of *MolFlow* into drug discovery workflows, either through RL-based fine-tuning or protein pocket conditioned generation, to future work.

Impact Statement

We acknowledge that, in its current form, our models have the potential to contribute towards malicious activities, such as helping to design toxic small molecules. However, we believe our model would form only a very small part of the resources required for such an application, and we strongly believe the potential positive contributions of this model, helping to design new therapeutics, for example, strongly outweigh the negatives.

References

Albergo, M. S. and Vanden-Eijnden, E. Building normalizing flows with stochastic interpolants. In *The Eleventh International Conference on Learning Representations*, 2023.

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

Atance, S. R., Diez, J. V., Engkvist, O., Olsson, S., and Mercado, R. De novo drug design using reinforcement learning with graph-based deep generative models. *Journal of Chemical Information and Modeling*, 62(20):4863–4872, October 2022.

Axelrod, S. and Gomez-Bombarelli, R. Geom, energy-annotated molecular conformations for property prediction and molecular generation. *Scientific Data*, 9(1):185, 2022.

Ba, J. L., Kiros, J. R., and Hinton, G. E. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

Bietti, A., Venturi, L., and Bruna, J. On the sample complexity of learning under geometric stability. *Advances in neural information processing systems*, 34:18673–18684, 2021.

Blaschke, T., Arús-Pous, J., Chen, H., Margreitter, C., Tyrchan, C., Engkvist, O., Papadopoulos, K., and Patronov,

A. Reinvent 2.0: an ai tool for de novo drug design. *Journal of chemical information and modeling*, 60(12): 5918–5922, 2020.

Bose, J., Akhound-Sadegh, T., Huguet, G., FATRAS, K., Rector-Brooks, J., Liu, C.-H., Nica, A. C., Korablyov, M., Bronstein, M. M., and Tong, A. SE(3)-stochastic flow matching for protein backbone generation. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=kJFIH23hXb.

Brody, S., Alon, U., and Yahav, E. How attentive are graph attention networks? In *International Conference on Learning Representations*, 2021.

Campbell, A., Yim, J., Barzilay, R., Rainforth, T., and Jaakkola, T. Generative flows on discrete state-spaces: Enabling multimodal flows with applications to protein co-design. *arXiv preprint arXiv:2402.04997*, 2024.

Chen, T., ZHANG, R., and Hinton, G. Analog bits: Generating discrete data using diffusion models with self-conditioning. In *The Eleventh International Conference on Learning Representations*, 2023.

Elfwing, S., Uchibe, E., and Doya, K. Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. *Neural networks*, 107:3–11, 2018.

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

Gred Landrum *et al.* Rdkit: Open-source cheminformatics. https://www.rdkit.org, 2023.

Guan, J., Qian, W. W., Peng, X., Su, Y., Peng, J., and Ma, J. 3d equivariant diffusion for target-aware molecule generation and affinity prediction. *arXiv preprint arXiv:2303.03543*, 2023.

Halgren, T. A. Merck molecular force field. i. basis, form, scope, parameterization, and performance of mmff94. *Journal of computational chemistry*, 17(5-6):490–519, 1996.

Harris, C., Didi, K., Jamasb, A. R., Joshi, C. K., Mathis, S. V., Lio, P., and Blundell, T. Benchmarking generated poses: How rational is structure-based drug design with generative models? *arXiv preprint arXiv:2308.07413*, 2023.

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

Hoogeboom, E., Satorras, V. G., Vignac, C., and Welling, M. Equivariant diffusion for molecule generation in 3d. In *International conference on machine learning*, pp. 8867–8887. PMLR, 2022.

Hua, C., Luan, S., Xu, M., Ying, Z., Fu, J., Ermon, S., and Precup, D. Mudiff: Unified diffusion for complete molecule generation. In *Learning on Graphs Conference*, pp. 33–1. PMLR, 2024.

Jing, B., Eismann, S., Suriana, P., Townshend, R. J. L., and Dror, R. Learning from protein structure with geometric vector perceptrons. In *International Conference on Learning Representations*, 2020.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Klein, L., Krämer, A., and Noé, F. Equivariant flow matching. *Advances in Neural Information Processing Systems*, 36, 2024.

Köhler, J., Klein, L., and Noe, F. Equivariant flows: Exact likelihood generative learning for symmetric densities. In *International Conference on Machine Learning*, pp. 5361–5370. PMLR, 2020.

Le, T., Noe, F., and Clevert, D.-A. Representation learning on biomolecular structures using equivariant graph attention. In *Learning on Graphs Conference*, pp. 30–1. PMLR, 2022.

Le, T., Cremer, J., Noé, F., Clevert, D.-A., and Schütt, K. Navigating the design space of equivariant diffusion-based generative models for de novo 3d molecule generation. *arXiv preprint arXiv:2309.17296*, 2023.

Liao, Y.-L. and Smidt, T. Equiformer: Equivariant graph attention transformer for 3d atomistic graphs. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=KwmPfARgOTD.

Lipman, Y., Chen, R. T., Ben-Hamu, H., Nickel, M., and Le, M. Flow matching for generative modeling. In *The Eleventh International Conference on Learning Representations*, 2022.

Liu, M., Luo, Y., Uchino, K., Maruhashi, K., and Ji, S. Generating 3d molecules for target protein binding. In *International Conference on Machine Learning*, pp. 13912–13924. PMLR, 2022a.

Liu, X., Gong, C., and Liu, Q. Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv preprint arXiv:2209.03003*, 2022b.

Loeffler, H. H., He, J., Tibo, A., Janet, J. P., Voronov, A., Mervin, L. H., and Engkvist, O. Reinvent 4: Modern ai–driven generative molecule design. *Journal of Cheminformatics*, 16(1):20, 2024.

Monge, G. Mémoire sur la théorie des déblais et des remblais. *Mem. Math. Phys. Acad. Royale Sci.*, pp. 666–704, 1781.

Morehead, A. and Cheng, J. Geometry-complete diffusion for 3d molecule generation. In *ICLR 2023-Machine Learning for Drug Discovery workshop*, 2023.

Noé, F., Olsson, S., Köhler, J., and Wu, H. Boltzmann generators: Sampling equilibrium states of many-body systems with deep learning. *Science*, 365(6457):eaaw1147, 2019.

Peng, X., Luo, S., Guan, J., Xie, Q., Peng, J., and Ma, J. Pocket2mol: Efficient molecular sampling based on 3d protein pockets. In *International Conference on Machine Learning*, pp. 17644–17655. PMLR, 2022.

Pooladian, A.-A., Ben-Hamu, H., Domingo-Enrich, C., Amos, B., Lipman, Y., and Chen, R. T. Q. Multisample flow matching: Straightening flows with minibatch couplings. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 28100–28127. PMLR, 23–29 Jul 2023.

Ramakrishnan, R., Dral, P. O., Rupp, M., and Von Lilienfeld, O. A. Quantum chemistry structures and properties of 134 kilo molecules. *Scientific data*, 1(1):1–7, 2014.

Reddi, S. J., Kale, S., and Kumar, S. On the convergence of adam and beyond. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=ryQu7f-RZ.

Rubinstein, M. and Colby, R. *Polymer Physics*. Oxford University Press, 2003. ISBN 9781613449431. URL https://books.google.se/books?id=EJrYoAEACAAJ.

Satorras, V. G., Hoogeboom, E., and Welling, M. E (n) equivariant graph neural networks. In *International conference on machine learning*, pp. 9323–9332. PMLR, 2021.

Schneckenreiter, L., Freinschlag, R., Sestak, F., Brandstetter, J., Klambauer, G., and Mayr, A. Gnn-vpa: A variance-preserving aggregation strategy for graph neural networks. In *The Second Tiny Papers Track at ICLR 2024*, 2024.

Schneuing, A., Du, Y., Harris, C., Jamasb, A., Igashov, I., Du, W., Blundell, T., Lió, P., Gomes, C., Welling, M., et al. Structure-based drug design with equivariant diffusion models. *arXiv preprint arXiv:2210.13695*, 2022.

Schreiner, M., Winther, O., and Olsson, S. Implicit transfer operator learning: Multiple time-resolution models for molecular dynamics. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=1kZx7JiuA2.

Schütt, K., Unke, O., and Gastegger, M. Equivariant message passing for the prediction of tensorial properties and molecular spectra. In *International Conference on Machine Learning*, pp. 9377–9388. PMLR, 2021.

Serre, J.-P. et al. *Linear representations of finite groups*, volume 42. Springer, 1977.

Song, Y., Sohl-Dickstein, J., Kingma, D. P., Kumar, A., Ermon, S., and Poole, B. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

Song, Y., Gong, J., Xu, M., Cao, Z., Lan, Y., Ermon, S., Zhou, H., and Ma, W.-Y. Equivariant flow matching with hybrid probability transport for 3d molecule generation. *Advances in Neural Information Processing Systems*, 36, 2024.

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

Stark, H., Jing, B., Wang, C., Corso, G., Berger, B., Barzilay, R., and Jaakkola, T. Dirichlet flow matching with applications to dna sequence design. *arXiv preprint arXiv:2402.05841*, 2024.

Tahmasebi, B. and Jegelka, S. Sample complexity bounds for estimating probability divergences under invariances. *arXiv preprint arXiv:2311.02868*, 2023.

Tong, A., Malkin, N., Huguet, G., Zhang, Y., Rector-Brooks, J., Fatras, K., Wolf, G., and Bengio, Y. Improving and generalizing flow-based generative models with minibatch optimal transport. *arXiv preprint arXiv:2302.00482*, 2023.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., and Bengio, Y. Graph attention networks. In *International Conference on Learning Representations*, 2018.

Verma, Y., Heinonen, M., and Garg, V. Abode: Ab initio antibody design using conjoined odes. In *International Conference on Machine Learning*, pp. 35037–35050. PMLR, 2023.

Vignac, C. and Frossard, P. Top-n: Equivariant set and graph generation without exchangeability. *arXiv preprint arXiv:2110.02096*, 2021.

Vignac, C., Osman, N., Toni, L., and Frossard, P. Midi: Mixed graph and 3d denoising diffusion for molecule generation. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 560–576. Springer, 2023.

Viguera Diez, J., Romeo Atance, S., Engkvist, O., and Olsson, S. Generation of conformational ensembles of small molecules via surrogate model-assisted molecular dynamics. *Machine Learning: Science and Technology*, 5(2):025010, April 2024. ISSN 2632-2153. doi: 10.1088/2632-2153/ad3b64. URL http://dx.doi.org/10.1088/2632-2153/ad3b64.

Xu, C., Wang, H., Wang, W., Zheng, P., and Chen, H. Geometric-facilitated denoising diffusion model for 3d molecule generation. *arXiv preprint arXiv:2401.02683*, 2024.

Xu, M., Yu, L., Song, Y., Shi, C., Ermon, S., and Tang, J. Geodiff: A geometric diffusion model for molecular conformation generation. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=PzcvxEMzvQC.

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.

Zhao, H., Jiang, L., Jia, J., Torr, P. H., and Koltun, V. Point transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 16259–16268, October 2021.

A. Scale Optimal Transport

Figure 3 shows the transport cost for different sizes of molecules. We measure the transport cost as the mean-squared error between a batch of molecular coordinates and a batch of coordinate noise sampled using either equivariant OT or scale OT. For each batch of molecules we sample 10 different batches of noise and calculate the mean transport cost. We use a batch of 100 as a representative example. Although the distribution is much wider, we only show transport cost values for molecules with between 20 and 70 atoms since almost all of the probability mass lies within this region. A histogram of the frequency of different sizes of molecules is also shown in Figure 3. We can see that scale OT provides a significant reduction in the transport cost overall. The reduction is very pronounced for smaller molecules but becomes less so as the molecular size increases. Even for molecules with between 40 and 50 atoms – the densest part of the distribution – scale OT still provides a noticeable reduction in the transport cost.

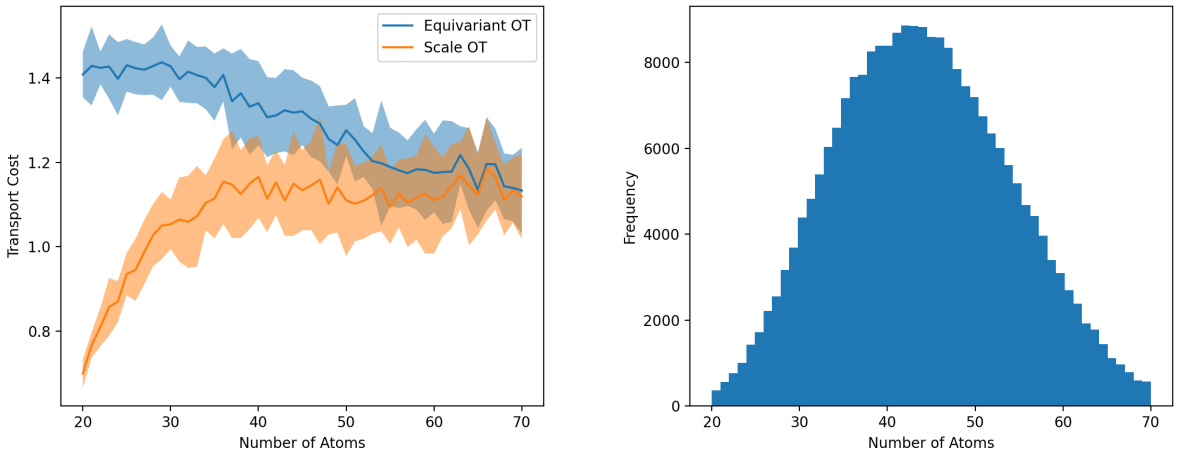

Figure 3. **Left**: average transport cost (mean squared error) for molecules of different sizes with both equivariant and scale optimal transport. Shaded areas show 2 standard deviations either side of the mean. **Right**: histogram of the number of atoms per molecule for the GEOM Drugs dataset.

B. Additional Experiments

In this section we provide results on the QM9 benchmark dataset, as well as results on ablation experiments showing the importance of various components of the model and training regime. Firstly, we provide a full definition for each of the existing benchmark metrics we have used:

- **Atom stability** measures the proportion of atoms which have the correct number of bonds, according to a pre-defined lookup table.
- **Molecule stability** then measures the proportion of generated molecules for which all atoms are stable.
- **Validity** is given by the proportion of molecules which can be successfully sanitized using RDKit.
- **Uniqueness** measures the proportion of generated molecules which are unique in their 1D (i.e. SMILES string) form.
- **Novelty** measures the proportion of molecules which are not in the training set.

Table 3 compares the performance of *MolFlow* with existing approaches. We compare *MolFlow* to a number of recently proposed models for 3D molecular generation, including both models which infer bonds based on coordinates and models which generate bonds directly. EDM (Hoogeboom et al., 2022), GCDM (Morehead & Cheng, 2023), MUDiff (Hua et al., 2024) and GFMDiff (Xu et al., 2024) are all diffusion-based models which infer bonds from atom positions. We also compare to EquiFM (Song et al., 2024) which uses flow-matching along with equivariant optimal transport to generate atom

Table 3. Molecular generation results on QM9. Values for models we evaluated are given as an average over 5 runs, sampling 5000 molecules on each run, with standard deviations given in subscripts. Models are grouped into those which infer bonds from coordinates (top) and those which generate bonds directly (bottom). Since some models only publish the proportion of molecules which are both unique and valid, results marked * are estimates for uniqueness.

Model	Atom Stab ↑	Mol Stab ↑	Valid ↑	Unique ↑	NFE ↓
EDM	98.7	82.0	91.9	98.9*	1000
GCDM	98.7	85.7	94.8	98.4*	1000
MUDiff	98.8	89.9	95.3	99.1	1000
GFMDiff	98.9	87.7	96.3	98.8*	500
EquiFM	98.9	88.3	94.7	98.7*	210
MiDi	99.8	97.5	97.9	97.6	500
EQGAT-diff	**99.9**$_{\pm 0.0}$	98.7$_{\pm 0.18}$	99.0$_{\pm 0.16}$	**100.0**$_{\pm 0.0}$	500
MolFlow (Ours)	**99.9**$_{\pm 0.0}$	**99.6**$_{\pm 0.06}$	**99.4**$_{\pm 0.12}$	97.5$_{\pm 0.20}$	100

types and coordinates; bonds are then inferred based on these. Finally, we also compare to MiDi (Vignac et al., 2023) and EQGAT-diff (Le et al., 2023), which we discussed in the main text. Following (Vignac & Frossard, 2021; Hoogeboom et al., 2022) we do not provide novelty scores on QM9. *MolFlow* is trained for 300 epochs on QM9 on a single Nvidia A100 GPU. From the table we can see that *MolFlow* matches or exceeds all models on all metrics other than uniqueness, despite using 5 times fewer sampling steps than MiDi and EQGAT-diff. Our model also significantly outperforms EquiFM, the only other flow-matching-based model in the table. We also note that since EquiFM uses an adaptive ODE solver the number of function evaluations in the table is an average and could be significantly higher for larger, more complex molecules.

In Table 4 we present the results of ablation experiments showing the importance of various model components. The results in the table work in the following way – *Baseline* refers to a MolFlow model with one coordinate set, using equivariant optimal transport and no self conditioning. The models below then build on each other, starting at the top and working down the table, adding the component given in the model column. For example, the model in the row marked by + Scale OT uses 64 coordinate sets and scale optimal transport. The final model is the same as presented in the main text but trained for only 100 epochs, as with the other models in Table 4.

The ablation results show that all three additions are important for model performance. Unlike using additional coordinate sets and self conditioning, scale optimal transport comes with no increase in training time. Scale OT leads to a significant reduction in the energies and strains of the generated molecules, although does come at the cost of slightly decreased validity. In practice we believe this is still a very beneficial trade-off – validity becomes much less important when it's close to 100% and the lower energies and strains show that the model is producing higher quality conformations.

Table 4. MolFlow ablation results on the Geom Drugs dataset. Each model was trained for 100 epochs. During evaluation an average over 5 evaluation runs was taken where each run consisted of sampling 5000 molecules with sizes sampled randomly from the test set.

Model	Mol Stab	Validity	Energy	Strain	Uniqueness	Novelty
Baseline	88.8$_{\pm 0.54}$	82.7$_{\pm 0.43}$	692.0$_{\pm 26.5}$	662.3$_{\pm 26.4}$	100.0$_{\pm 0.0}$	99.7$_{\pm 0.03}$
+ Coord Sets	96.2$_{\pm 0.11}$	95.0$_{\pm 0.12}$	318.7$_{\pm 4.08}$	269.3$_{\pm 3.48}$	99.9$_{\pm 0.01}$	99.8$_{\pm 0.06}$
+ Scale OT	95.9$_{\pm 0.23}$	91.9$_{\pm 0.14}$	199.5$_{\pm 2.12}$	161.4$_{\pm 2.12}$	100.0$_{\pm 0.0}$	99.7$_{\pm 0.08}$
+ Self Cond	97.4$_{\pm 0.18}$	94.6$_{\pm 0.24}$	135.9$_{\pm 2.0}$	98.4$_{\pm 2.30}$	100.0$_{\pm 0.0}$	99.7$_{\pm 0.06}$

C. Training Details

All models were trained with the AMSGrad (Reddi et al., 2018) variant of the Adam optimiser (Kingma & Ba, 2014) with a learning rate (LR) of 0.0003. We also apply linear LR warm-up, using 2000 warm-up steps for QM9 and 10000 warm-up steps for GEOM Drugs. During training we clip the norms of the gradients at 1.0 for all models. Loss weightings $(\lambda_x, \lambda_a, \lambda_b, \lambda_c) = (1.0, 0.2, 0.5, 1.0)$ were used to train QM9 models. The same weightings were used for GEOM Drugs, except we set $\lambda_b = 1.0$.

When training with self-conditioning half of the training batches are treated as normal and the other half are trained on as self-conditioning batches. In this case the batch is firstly processed by the model to generate conditioning inputs, these are then detached from the computation graph, and finally used as conditioning inputs for the model training step. In practice the conditioning inputs are concatenated with the interpolated data and embedded at the start of the model. For atom and bond types the conditioning inputs are Softmax-normalised probability distributions over the predicted categorical types.

In order to make the training as efficient as possible we place molecules in the dataset into buckets based on their size, and then form minibatches for training and evaluation within the buckets. This ensures that all batches contain similarly sized molecules so that the amount of padding within each batch is minimised. With this setup we can also apply a cost function to select the batch size for each bucket separately; since the amount of memory required to process a molecule increases quadratically with the number of atoms, this helps to balance the GPU memory consumption for each batch. In practice, though, we simply apply a linear cost function and use a batch size of 4096 atoms per batch for all our models. While we have found our bucketing scheme leads to a significant increase in training speed, it may also introduce additional bias into the training since molecules within each batch are no longer selected completely at random. Although we have not attempted to quantify this bias our results seem to show that bucketing is not significantly detrimental to performance.

D. Samples from MolFlow

In this section we present samples from *MolFlow* trained on GEOM Drugs. The samples were generated randomly but we have rotated them where necessary to aid visualisation.

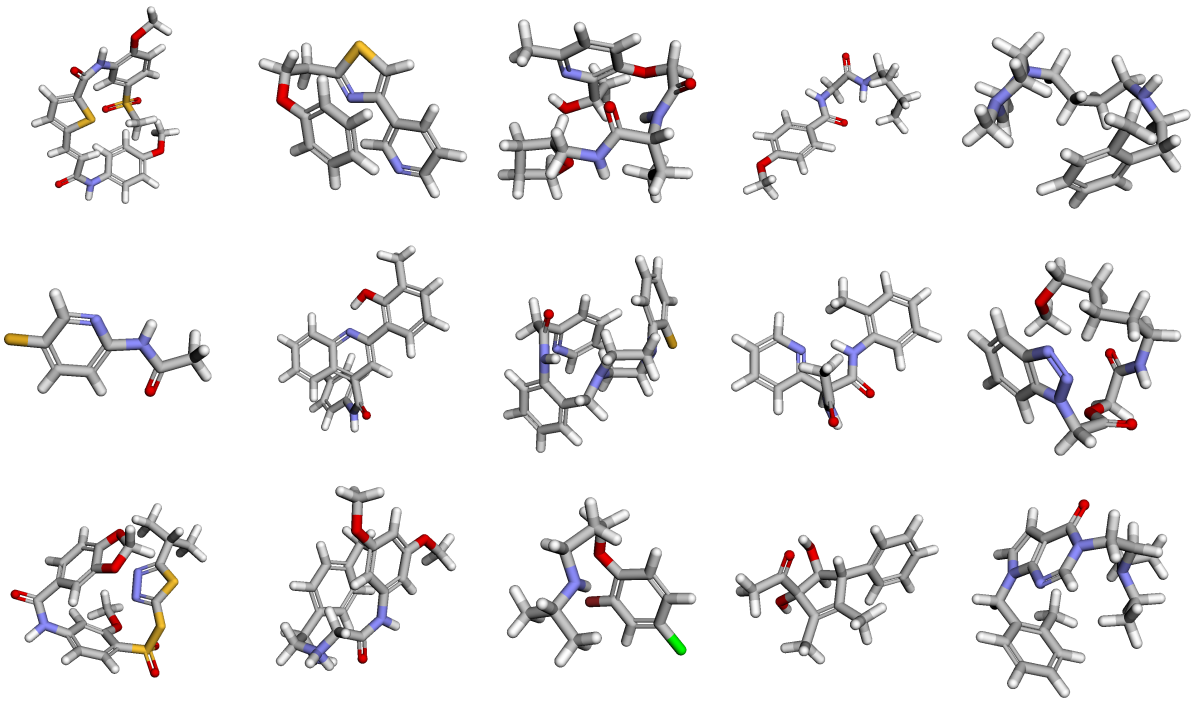

Figure 4. Random samples from a MolFlow model trained on GEOM Drugs, generated using 100 ODE integration steps.