MiDi: Mixed Graph and 3D Denoising Diffusion for Molecule Generation

Clément Vignac[1*], Nagham Osman[2*], Laura Toni[2], and Pascal Frossard[1]

[1] LTS4, EPFL, Lausanne, Switzerland
[2] EEE Dept, University College London, London, United Kingdom

Abstract. This work introduces MiDi, a novel diffusion model for jointly generating molecular graphs and their corresponding 3D arrangement of atoms. Unlike existing methods that rely on predefined rules to determine molecular bonds based on the 3D conformation, MiDi offers an end-to-end differentiable approach that streamlines the molecule generation process. Our experimental results demonstrate the effectiveness of this approach. On the challenging GEOM-DRUGS dataset, MiDi generates 92% of stable molecules, against 6% for the previous EDM model that uses interatomic distances for bond prediction, and 40% using EDM followed by an algorithm that directly optimize bond orders for validity. Our code is available at github.com/cvignac/MiDi.

Keywords: Diffusion Model · Drug Discovery · Graph Generation.

1 Introduction

Modern drug discovery requires the development of effective machine learning models that can correctly capture the vast chemical space and sample from it. These models need to understand properties of molecules that depend both on their molecular graph and their conformation in the 3D space. The molecular graph (or 2D structure) determines the existence and type of the chemical bonds, and allows the identification of functional groups in a compound. This provides information about its chemical properties and enables to predict synthetic pathways. On the other hand, the 3D conformation of a compound plays a key role in its interaction with other molecules, and governs in particular its biological
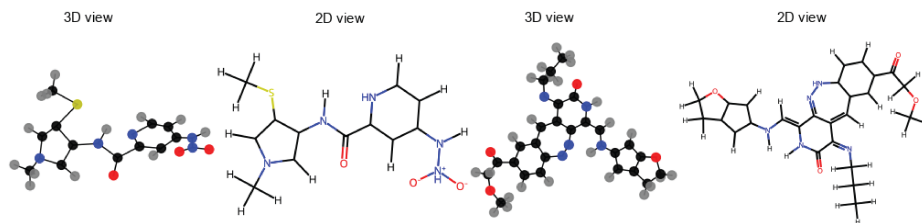

Fig. 1: Samples from our model. MiDi generates simultaneously a 2D graph structure and a 3D conformation that is consistent with this structure.

activity and binding affinity to proteins. To explore the chemical space adequately, it is therefore crucial to consider both aspects simultaneously.

Unfortunately, existing generative models for molecules are restricted to one of these two data modalities. While models that exclusively generate molecular graphs have been vastly researched [8], current 3D molecule generation are on the contrary only trained to generate conformers, thus ignoring bond information. These models rely on a subsequent step that predicts the 2D structure using either interatomic distances [16, 37] or chemical software such as OpenBabel [11]. As a result, these models are not end-to-end differentiable, which hampers their ability to be fully optimized for various downstream tasks. This severely limits the potential of 3D molecule generators, particularly for complex tasks like pocket-conditioned generation.

We propose here a new model, called Mixed Graph+3D Denoising Diffusion (MiDi), which overcomes this limitation by simultaneously generating a molecular graph and its corresponding 3D coordinates. MiDi represents molecules as graphs embedded in 3D that contain node features (atom types and formal charges) and edges features (bond types). Our model progressively corrupts data with noise, and trains a neural network to predict clean data from noisy inputs. New molecules can then be generated by sampling pure noise and iteratively denoising it with the neural network, similarly to other diffusion models [14, 42]. As the model is trained to denoise both the graph and 3D coordinates in tandem, it is able to produce stable molecular graphs that are consistent with the generated conformers.

While previous diffusion models for molecules relied on either Gaussian noise or discrete diffusion, MiDi uses both noise models simultaneously: the 3D coordinates are corrupted with Gaussian noise, while the other components use discrete diffusion which was found to be effective for graph generation [12, 47]. To further enhance the quality of the generated samples, we introduce a noise schedule whose parameters are adjusted to each component. Specifically, we add noise to the atom types and formal charges at a faster rate than to the coordinates and bond types. This encourages the denoising network to first focus on generating a realistic 3D conformation and corresponding bond types, before refining the atom types and formal charges.

Our second contribution considers the denoising network: the Transformer architecture we propose incorporates a novel *rEGNN* layer, which improves upon the popular EGNN layers [39] by leveraging features that are not translation-invariant. We show that, due to the use of Gaussian noise in the zero center-of-mass subspace of the molecules, the resulting model is nevertheless equivariant to translations and rotations, which is crucial for achieving high performance.

We showcase the effectiveness of our model on unconditional molecule generation. On the challenging GEOM-DRUGS dataset, the previous EDM model [16] can generate stable molecules at a rate of 5.5%, which can be improved to 40.3% by using the Open Babel bond prediction algorithm [35]. In contrast, the proposed method achieves 91.6% of stable molecules, demonstrating the superiority of our end-to-end differentiable approach. Furthermore, MiDi can be readily applied to

various drug-discovery tasks beyond unconditional generation, which confirms its versatility and potential for improving drug discovery pipelines.

2   Related Work

Concurrent work Concurrently to our work, [17] and [36] also proposed 2D+3D diffusion models for molecule generation. These models also leverage Gaussian diffusion for the 3D coordinates and discrete diffusion for the graph, using absorbing transitions in [36] and uniform transitions in [17]. Each model also features unique contributions: [17] proposes richer positional encodings for the transformer layer, while [36] introduces a guidance mechanism to help the network predict accurate bond lengths. MiDi is the only model that improves upon the standard EGNN layers, and that is capable of handling formal charges. It is also the only model that presents results on the more complex GEOM-DRUGS dataset with explicit hydrogens.

Molecule generation in 3D The idea of representing molecules as attributed 3D point cloud has been used both in one-shot settings [37] and in autoregressive methods such as GSchNet[11] . Recently, the Equivariant Diffusion Model (EDM) [16] was proposed for this task, improving significantly over previous results. This model was later extended by limiting the message-passing computations to neighboring nodes [19] and using a more expressive denoising network [34]. All these diffusion models can be conditioned on molecule-level properties using guidance mechanisms [4] or on another point cloud. Conditioning on a second point cloud has been employed to generate molecules that bind to a specific protein [7, 40] and to generate linkers between molecular fragments [20]. The main drawback of these models is that they do not learn the connectivity structure of the molecule. It needs to be obtained in a second stage using interatomic distances [16, 37] or specialized software such as Open Babel [35]. This results in limited performance for complex molecules, but also prevents end-to-end differentiability for downstream applications.

Graph Generation Another line of work has focused on generating graphs without associated 3D coordinates. Early denoising diffusion models for this task used Gaussian noise applied to the adjacency matrix [18, 25] or graph eigenvalues [30]. [47] and [12] however found that discrete diffusion is more effective, as it better respects the discrete nature of graphs. These diffusion models tend to outperform autoregressive methods except on validity metrics, as autoregressive models can perform validity checks at each sampling step [27, 29, 33]. In contrast to the proposed method, which operates at the node level, fragment-based methods [13, 24, 32] learn to combine chemically-relevant substructures from a fixed or learned dictionary [48], but are harder to adapt to 3D.

In order to generate molecules in 3D, graph-based models could be combined with conformer generation models [7, 52] which predict a 3D structure from

Table 1: Gaussian and categorical distributions enable the efficient computation of the key quantities involved in training diffusion models and sampling from them. Formulas for all parameters can be found in Section 3.

Noise model	Gaussian diffusion	Discrete diffusion
$q(\boldsymbol{z}_t\|\boldsymbol{z}_{t-1})$	$\mathcal{N}(\alpha_t\boldsymbol{z}_{t-1},\ \sigma_t^2\boldsymbol{I})$	$\boldsymbol{z}_{t-1}\,\boldsymbol{Q}_t$
$q(\boldsymbol{z}_t\|\boldsymbol{x})$	$\mathcal{N}(\bar{\alpha}_t\boldsymbol{x},\ \bar{\sigma}_t^2\boldsymbol{I})$	$\boldsymbol{x}\,\bar{\boldsymbol{Q}}_t$
$\int_x p_\theta(\boldsymbol{z}_{t-1}\|\boldsymbol{x},\boldsymbol{z}_t)dp_\theta(\boldsymbol{x}\|\boldsymbol{z}_t)$	$\mathcal{N}(\mu_t\hat{\boldsymbol{x}}+\nu_t\boldsymbol{z}_t,\tilde{\sigma}_t^2\boldsymbol{I})$	$\propto\sum_x p_\theta(x)(\boldsymbol{z}_t\boldsymbol{Q}_t'\odot\boldsymbol{x}\bar{\boldsymbol{Q}}_{t-1})$

an input graph. As these methods assume that the graph is known, they are able to exploit symmetries of the molecule (such as rotatable bonds), which is more difficult on unconditional generation tasks. Unfortunately, combining graph generation and conformer generation models would again break end-to-end differentiability and restrict performance.

Protein Generation While existing diffusion models for molecules operate on molecules of moderate size (up to 180 atoms), recent diffusion models for proteins have managed to scale to much larger structures [21, 41, 49, 50]. These methods leverage the chain structure of proteins, which implies that the adjacency matrix does not need to be predicted. Furthermore, instead of predicting 3D coordinates for each atom, they only predict the angles between successive C_α carbons, which significantly reduces the degrees of freedom and encodes roto-translation invariance in the representation. Those improvements are unfortunately specific to chain graphs, and cannot be used for arbitrary molecules.

3   Background

Denoising Diffusion Models Diffusion models consist of two essential elements: a noise model and a denoising neural network. The noise model q takes as input a data point x and generates a trajectory of increasingly corrupted data points $(z_1, ..., z_T)$. The corruption process is chosen to be Markovian, i.e.,

$$q(z_1, \ldots, z_T|x) = q(z_1|x)\prod_{t=2}^{T} q(z_t|z_{t-1}).$$

The denoising network ϕ_θ takes noisy data z_t as input, and learns to invert the diffusion trajectories. While it would be natural to naively train the network to predict z_{t-1}, this strategy would lead to noisy targets, as z_{t-1} depends on the sampled diffusion trajectory. Instead, modern diffusion models [14, 42, 43] predict the clean input x from z_t, or equivalently, the noise added to it. The diffusion sequences are then inverted by marginalizing over the network predictions $p_\theta(x|z_t)$:

$$p_\theta(z_{t-1}|z_t) = \int_x p_\theta(z_{t-1} \mid x, z_t)\ dp_\theta(x|z_t) \tag{1}$$

Although Eq. 1 leads to more efficient training, it requires the efficient computation of $p_\theta(z_{t-1}|x, z_t)$ and the integral, which is not always possible. Two main frameworks have been proposed under which Eq. 1 is tractable: Gaussian noise, which is suitable for continuous data, and discrete state-space diffusion for categorical data. Table 1 summarizes the main properties of the two related noise models.

Gaussian diffusion processes are defined by $q(z_t|z_{t-1}) \sim \mathcal{N}(\alpha_t \boldsymbol{z}_t, \sigma_t^2 \boldsymbol{I})$, where $(\alpha_t)_{t \leq T}$ controls how much signal is retained at each step and $(\sigma_t)_{t \leq T}$ how much noise is added. As normal distributions are stable under composition, we have $q(z_t|x) \sim \mathcal{N}(\bar{\alpha}_t \boldsymbol{z}_t, \bar{\sigma}_t^2 \boldsymbol{I})$, with $\bar{\alpha}_t = \prod_{s=1}^{t} \alpha_s$ and $\bar{\sigma}_t^2 = \sigma_t^2 - \alpha_t^2$. While any noise schedule is in principle possible, variance-preserving processes are most often used, which satisfy $\bar{\alpha}_t^2 + \bar{\sigma}_t^2 = 1$. The posterior of the transitions conditioned on x can also be computed in closed-form. It satisfies

$$q(z_{t-1}|z_t, x) \sim \mathcal{N}(\mu_t \; \boldsymbol{x} + \nu_t \; \boldsymbol{z}_t, \tilde{\sigma}_t^2 \mathbf{I}),$$

with $\mu_t = \bar{\alpha}_s(1 - \alpha_t^2 \bar{\sigma}_{t-1}^2/\bar{\sigma}_t^2)$, $\nu_t = \alpha_t \bar{\sigma}_{t-1}^2/\bar{\sigma}_t^2$ and $\tilde{\sigma}_t = \bar{\sigma}_{t-1}^2(1 - \alpha_t^2 \bar{\sigma}_{t-1}^2/\bar{\sigma}_t^2)$.

On the contrary, discrete diffusion considers that data points x belong to one of d classes [1]. The transition matrices $(\boldsymbol{Q}_1, ..., \boldsymbol{Q}_T)$ are square matrices of size $d \times d$ that represent the probability of jumping from one class to another at each time step. Given previous state z_{t-1}, the noise model for the next state z_t is a categorical distribution over the d possible classes which reads as $q(z_t|z_{t-1}) \sim \mathcal{C}(\boldsymbol{z}_{t-1}\boldsymbol{Q}_t)$, where $\boldsymbol{z}_{t-1}$ is a row vector encoding the class of z_{t-1}. Since the process is Markovian, we simply have $q(z_t = j|x) = [x\bar{\boldsymbol{Q}}^t]_j$ with $\bar{\boldsymbol{Q}}^t = \boldsymbol{Q}^1 \boldsymbol{Q}^2 ... \boldsymbol{Q}^t$. The posterior distribution $q(z_{t-1}|z_t, x)$ can also be computed in closed form using Bayes rule and the Markovian property. If $\odot$ denotes a pointwise product and $\boldsymbol{Q}'$ is the transpose of $\boldsymbol{Q}$, it can be written as

$$q(z^{t-1}|z^t, x) \propto \boldsymbol{z}^t \; (\boldsymbol{Q}^t)' \odot \boldsymbol{x} \; \bar{\boldsymbol{Q}}^{t-1}.$$

SE(3)-Equivariance with Diffusion Models Molecules are dynamic entities that can undergo translation and rotation, and the arrangement of their atoms does not have a predetermined order. To effectively model molecules using generative models and avoid augmenting the data with random transformations, it is essential to ensure that the models are equivariant to these inherent symmetries. In diffusion models, equivariance to a transformation group $\mathbb{G}$ can be achieved through several conditions. First, the noise model must be equivariant to the action of $\mathbb{G}$: $\forall g \in \mathbb{G}, q(g.z_t|g.x) = q(z_t|x)$. Second, the prior distribution q_∞ used at inference should be invariant to the group action, i.e., $q_\infty(g.z_T) = q_\infty(z_T)$, and this noise should be processed by an equivariant neural network in order to ensure that $p_\theta(g.z_{t-1}|g.z_t) = p_\theta(z_{t-1}|z_t)$. Finally, the network should be trained with a loss function that satisfies $l(p_\theta(g.x|g.z_t), g.x) = l(p_\theta(x|z_t), x)$. Together, these requirements create an architecture that is agnostic to the group elements used to represent the training data [16, 47, 52].

To ensure equivariance to the special Euclidean group SE(3), a number of architectures have been proposed as possible denoising networks for a diffusion

model [5, 10, 28, 44]. However, these networks can be computationally expensive due to their manipulation of spherical harmonics. As a result, many generative models for molecules [15, 19, 20, 40] use the more affordable EGNN layers [39]. At a high level, EGNN recursively updates the coordinates $(\boldsymbol{r}_i)$ of a graph with node features $(\boldsymbol{x}_i)$ and edge features $(\boldsymbol{y}_{ij})$ using:

$$\boldsymbol{r}_i \leftarrow \boldsymbol{r}_i + \sum_j c_{ij} \ m(||\boldsymbol{r}_i - \boldsymbol{r}_j||, \boldsymbol{x}_i, \boldsymbol{x}_j, \boldsymbol{y}_{ij})(\boldsymbol{r}_j - \boldsymbol{r}_i)$$

The crucial feature of this parameterization is that the message function m takes only rotation-invariant arguments. This, combined with the linear term in $\boldsymbol{r}_j - \boldsymbol{r}_i$, ensures that the network is rotation-equivariant. Finally, we note that the normalization term $c_{ij} = ||\boldsymbol{r}_i - \boldsymbol{r}_j|| + 1$ is necessary for numerical stability when concatenating many layers.

4   Proposed Model

We now present the Mixed Graph+3D denoising diffusion (MiDi) model. We represent each molecule as a graph $G = (\boldsymbol{x}, \boldsymbol{c}, \boldsymbol{R}, \boldsymbol{Y})$, where $\boldsymbol{x}$ and $\boldsymbol{c}$ are vectors of length n containing the type and formal charge associated to all atoms. The $n \times 3$ matrix $\boldsymbol{R} = [\boldsymbol{r}_i]_{1 \leq i \leq n}$ contains the coordinates of each atom, and $\boldsymbol{Y}$ is an $n \times n$ matrix containing the bond types. Similarly to previous diffusion models for graphs, we consider the absence of a bond as a particular bond type and generate dense adjacency tensors. We denote the one-hot encoding of $\boldsymbol{x}$, $\boldsymbol{c}$, and $\boldsymbol{Y}$ by $\boldsymbol{X}$, $\boldsymbol{C}$, and $\boldsymbol{Y}$, respectively. Time steps are denoted by superscripts, so, for example, $\boldsymbol{r}_i^t$ denotes the coordinates of node v_i at time t. The transpose of matrix $\boldsymbol{X}$ is denoted by $\boldsymbol{X}'$.

4.1   Noise Model

Our noise model corrupts the features of each node and edge independently, using a noise model that depends on the data type. For the positions, we use a Gaussian noise within the zero center-of-mass (CoM) subspace of the molecule $\boldsymbol{\epsilon} \sim \mathcal{N}^{\mathrm{CoM}}(\alpha^t \boldsymbol{R}^{t-1}, \ (\sigma^t)^2 \boldsymbol{I})$, which is required to obtain a roto-translation equivariant architecture [52]. This means that the noise follows a Gaussian distribution on the linear subspace of dimension $3(n-1)$ that satisfies $\sum_{i=1}^n \boldsymbol{\epsilon}_i = 0$.

For atom types, formal charges and bond types, we use discrete diffusion, where the noise model is a sequence of categorical distributions. We choose the marginal transition model proposed in [47]. For instance, when $\boldsymbol{m} \in \mathbb{R}^a$ represents the marginal distribution of atom types in the training set, we define $\boldsymbol{Q}_x^t = \alpha^t \boldsymbol{I} + \beta^t \ \mathbf{1}_a \boldsymbol{m}'$. We similarly define $\boldsymbol{Q}_c^t$ and $\boldsymbol{Q}_y^t$. The resulting noise model is given by:

$$q(G^t|G^{t-1}) \sim \mathcal{N}^{\mathrm{CoM}}(\alpha^t \boldsymbol{R}^{t-1}, \ (\sigma^t)^2 \boldsymbol{I}) \times \mathcal{C}(\boldsymbol{X}^{t-1} \boldsymbol{Q}_x^t) \times \mathcal{C}(\boldsymbol{C}^{t-1} \boldsymbol{Q}_c^t) \times \mathcal{C}(\boldsymbol{Y}^{t-1} \boldsymbol{Q}_y^t).$$
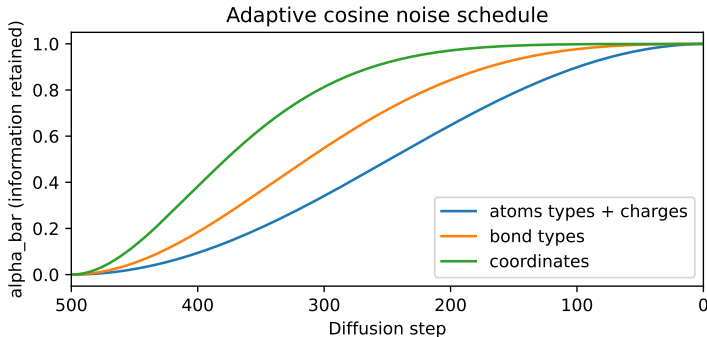
Fig. 2: The noise schedule is tuned separately for each component. Atom coordinates and bond types are denoised earlier during sampling, while atom types and formal charges are updated later in the process. Experimentally, the adaptive schedule allows to obtain better 3D conformers and more stable molecules.

When generating new samples, we define the posterior as a product as well:

$$p_\theta(G^{t-1}|G^t) = \prod_{1 \le i \le n} p_\theta(\boldsymbol{r}_i^{t-1}|G^t) p_\theta(\boldsymbol{x}_i^{t-1}|G^t) p_\theta(\boldsymbol{c}_i^{t-1}|G^t) \prod_{1 \le i,j \le n} p_\theta(\boldsymbol{Y}_{ij}^{t-1}|G^t),$$

We calculate each term by marginalizing over the network predictions. For instance,

$$p_\theta(x_i^{t-1}|G^t) = \int_{\boldsymbol{x}_i} p_\theta(x_i^{t-1} \mid x_i, G^t) \ dp_\theta(x_i|G^t)$$
$$= \sum_{x \in \mathcal{X}} q(x_i^{t-1}|x_i = x, G^t) \ p_\theta^X(x_i = x),$$

where $p_\theta^X(x_i = x)$ is the neural network estimate for the probability that node v_i in the clean graph G is of type x.

4.2 Adaptive Noise Schedule

Although the MiDi model corrupts the coordinates, atom types, bond types and formal charges simultaneously, these components do not play a symmetrical role. For instance, while the 2D connectivity structure can be predicted relatively well from the 3D conformation, the converse is not true as the conformation is not unique for a given structure. Similarly, the formal charges serve as an adjustable variable used to match the valency of each atom with its electronic structure, but they do not constitute a very fundamental property of the molecules.

Based on these observations, we propose an adaptation of the noise model in order to encourage the denoising network to first generate correctly the most important components, namely the atom coordinates and bond types, before moving on to predict the atom types and formal charges. To achieve this, we

modify the noise schedule to vary according to the component. We modify the popular cosine schedule by adding an exponent ν that controls the rate at which the noise is added to the model:

$$\bar{\alpha}^t = \cos\left(\frac{\pi}{2}\frac{(t/T + s)^{\nu}}{1 + s}\right)^2,$$

where the parameter ν can take the form of ν_r, ν_x, ν_y and ν_c for the atom coordinates, types, bond types, and charges, respectively. On the QM9 dataset, we use $\nu_r = 2.5$, $\nu_y = 1.5$, $\nu_x = \nu_c = 1$, while GEOM-DRUGS uses $\nu_r = 2$. The noise schedule used for QM9 is shown in Fig. 2. This choice means that rough estimates for the atom coordinates and the bond types are first generated at inference, before the other components start to play a significant role. This aligns with previous work on 2D molecular graph generation which found that predicting the bond types before the atom types is beneficial [31, 46].

4.3   Denoising Network

The denoising network takes a noisy graph as input and learns to predict the corresponding clean graph. It manipulates graph-level features $\boldsymbol{w}$, node coordinates $\boldsymbol{R}$, node features (atom types and formal charges, treated together in the matrix $\boldsymbol{X}$), and edge features $\boldsymbol{Y}$. Coordinates are treated separately from the other node features in order to guarantee SE(3) equivariance. The neural network architecture is summarized in Figure 3. It consists of a Transformer architecture [45], with a succession of self-attention module followed by normalization layers and feedforward networks. We give more details about the different blocks below.

Relaxed Equivariant Graph Neural Networks (rEGNNs) In our proposed method, we leverage the effective yet affordable EGNN layers [39] for processing the coordinates. However, we enhance these layers by exploiting the fact that, when the data and the noise reside in the zero Center-Of-Mass subspace, it is not necessary for the neural network to be translation invariant. This can be interpreted as defining a canonical pose for the translation group, which is a valid way to achieve equivariance [23, 26].

Rather than simply relying on pairwise distances $||\boldsymbol{r}_i - \boldsymbol{r}_j||_2$, we can therefore use other rotation invariant descriptors such as $||\boldsymbol{r}_i||_2$ or $\cos(\boldsymbol{r}_i, \boldsymbol{r}_j)$. We therefore propose the following *relaxedEGNN* (rEGNN) layer:

$$[\boldsymbol{\Delta}_r]_{ij} = \mathrm{cat}(||\boldsymbol{r}_i - \boldsymbol{r}_j||_2, ||\boldsymbol{r}_i||_2, ||\boldsymbol{r}_j||_2, \cos(\boldsymbol{r}_i, \boldsymbol{r}_j))$$
$$\boldsymbol{r}_i \leftarrow \boldsymbol{r}_i + \sum_j \phi_m(\boldsymbol{X}_i, \boldsymbol{X}_j, [\boldsymbol{\Delta}_r]_{ij}, \boldsymbol{Y}_{ij})\, (\boldsymbol{r}_j - \boldsymbol{r}_i)$$

Similar to EGNN layers, the rEGNN layer combines a rotation-invariant message function with a linear update in $\boldsymbol{r}_j - \boldsymbol{r}_i$, which guarantees rotation equivariance. Notably, the additional features $||\boldsymbol{r}_i||_2$, $||\boldsymbol{r}_j||_2$, and $\cos(\boldsymbol{r}_i, \boldsymbol{r}_j)$ are computed relative to the center-of-mass of the molecule, which is set to 0 by
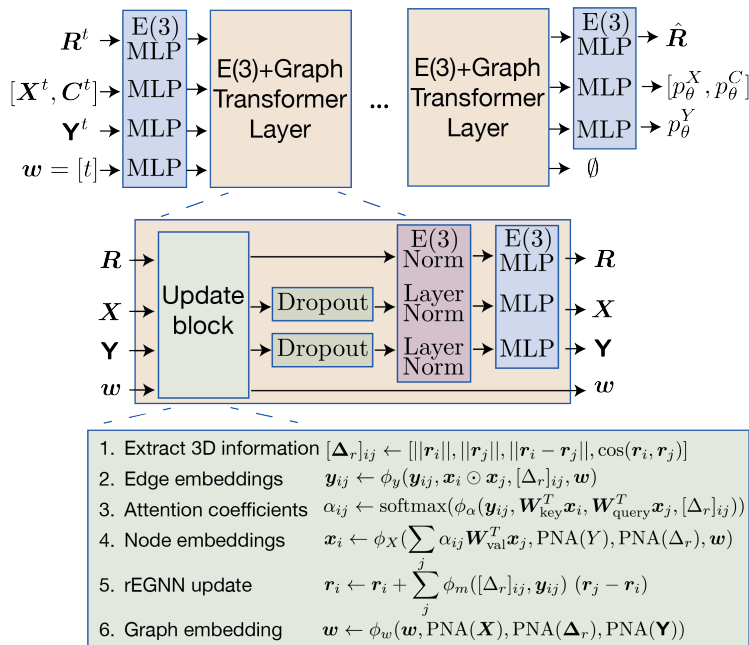
Fig. 3: The denoising neural network of MiDi jointly predicts the 2D graph and 3D coordinates of the clean graph from a noisy input. It follows a graph Transformer architecture with layers tailored to maintain SE(3) equivariance. In the update block, each component is updated using the other features. While the graph-level features $\boldsymbol{w}$ do not play a direct role in the final prediction, they serve as an effective means of storing and organizing pertinent information throughout the transformer layers.

definition. In our experiments, we have observed that these features facilitate the generation of a higher proportion of connected molecules, thereby mitigating an issue previously observed with both 3D [16] and 2D-based denoising diffusion models [47].

Update Block To improve the model's ability to process all features simultaneously, our new rEGNN layer is integrated into a larger update block that processes each component using all other ones. The edge features are first updated using $\boldsymbol{\Delta}_r$, the node features, and the global features. The node features are updated using a self-attention mechanism, where the attention coefficients also use the edge features and $\boldsymbol{\Delta}_r$. After the attention heads have been flattened, the obtained values are modulated by the pooled edge features, $\boldsymbol{\Delta}_r$ and the global features. For pooling pairwise features ($\boldsymbol{Y}$ and $\boldsymbol{\Delta}_r$) into node representations, we use PNA layers [6]: $\mathrm{PNA}(\boldsymbol{Y})_i = \boldsymbol{W}^T \mathrm{cat}[\mathrm{mean}, \min, \max, \mathrm{std}]_j(\boldsymbol{y}_{ij})$. The global features are updated by pooling all other features at the graph level. Finally, the

coordinates are updated using a rEGNN update, where the message function takes as input $\boldsymbol{\Delta}_r$ and the updated edge features. Note that we do not use the normalization term of EGNN: our layers are integrated in a Transformer architecture as discussed next, and we empirically found SE(3) normalization layers to be more effective than the EGNN normalisation term at controlling the magnitude of the activations.

Integration into a Transformer Architecture Transformers have proved to be a very efficient way to stabilize the self-attention mechanism over many layers. We describe below the changes to the feed-forward neural network and normalization layers that are required to ensure SE(3)-equivariance.

Our feed-forward neural network processes each component using MLPs applied in parallel on each node and each edge. As the coordinates cannot be treated separately (it would break SE(3)-equivariance), we define

$$\text{PosMLP}(\boldsymbol{R}) = \Pi^{\text{CoM}}(\text{MLP}(||\boldsymbol{R}||)\frac{\boldsymbol{R}}{||\boldsymbol{R}|| + \delta}) \in \mathbb{R}^{n \times 3},$$

where $||\boldsymbol{R}|| \in \mathbb{R}^{n \times 1}$ contains the norm $||\boldsymbol{r}_i||^2$ of each point, $\text{MLP}(||\boldsymbol{R}||) \in \mathbb{R}^{n \times 1}$ as well, δ is a small positive constant, and Π^{CoM} is the projection of the coordinates on the linear subspace with center-of-mass at 0:

$$\Pi^{\text{CoM}}(\boldsymbol{R})_i = \boldsymbol{r}_i - \frac{1}{n}\sum_{i=1}^{n}\boldsymbol{r}_i.$$

The choice of the normalization layer also depends on the problem symmetries: while batch normalization [22] is used in some graph transformer models [9], this layer is not equivariant in contrast to Set Normalization [53] or Layer Normalization [3]. For SE(3) equivariance, the normalization of [28] should be used. Applied to 3D coordinates, it writes

$$\text{E3Norm}(\boldsymbol{R}) = \gamma\frac{||\boldsymbol{R}||}{\bar{n} + \delta}\ \frac{\boldsymbol{R}}{||\boldsymbol{R}||} = \gamma\frac{\boldsymbol{R}}{\bar{n} + \delta}\quad \text{with}\quad \bar{n} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}||\boldsymbol{r}_i||^2},$$

with a learnable parameter $\gamma \in \mathbb{R}$ initialized at 1.

4.4   Training Objective

The denoising network of MiDi is trained to predict the clean molecule from a noisy input G^t, which is reflected in the choice of loss function used during model training. The estimation of the coordinates $\boldsymbol{R}$ is a regression problem that can simply be solved with mean-squared error, whereas the prediction p_θ^X for the atom types, p_θ^C for the formal charges and p_θ^Y for the bond types corresponds to a classification problem which can be addressed through a cross-entropy loss (CE in the equations). Note that the network's position predictions result in

pointwise estimates $\hat{\boldsymbol{R}}$, while for the other terms, the prediction is a distribution over classes. The final loss is a weighted sum of these components:

$$l(G, \hat{p}^G) = \lambda_r ||\hat{\boldsymbol{R}} - \boldsymbol{R}||^2 + \lambda_x \, \mathrm{CE}(\boldsymbol{X}, p_\theta^X) + \lambda_c \, \mathrm{CE}(\boldsymbol{C}, p_\theta^C) + \lambda_y \, \mathrm{CE}(\mathbf{Y}, p_\theta^Y)$$

The (λ_i) were initially chosen in order to balance the contribution of each term and cross-validated starting from this initial value. Our final experiments use $\lambda_r = 3$, $\lambda_x = 0.4$, $\lambda_c = 1$, $\lambda_y = 2$.

5 Experiments

5.1 Settings

We evaluate MiDi's performance on unconditional molecule generation tasks. To the best of our knowledge, MiDi is the first method to generate both the graph structure and the conformer simultaneously, leaving no end-to-end differentiable method to compare to. We therefore compare MiDi to 3D models on top of which a bond predictor is applied. We consider two such predictors: either a simple lookup table, as used in [16], or the optimization procedure of OpenBabel[3] [35] used in other works such as [20, 40]. The latter algorithm optimizes the bond orders of neighboring atoms in order to create a valid molecule, removing all control on the generated graphs. In terms of dataset comparison, EDM [16] was previously the only method that could scale up to the large GEOM-DRUGS dataset, so it is our only direct competitor in that case. For the QM9 dataset, we also compare MiDi's performance to that of the GSchNet method [11], which employed the OpenBabel algorithm and achieved good results.

To facilitate comparison with previous methods, such as [37] and [16], we benchmark our models on the full molecular graphs that include explicit hydrogens atoms. However, we acknowledge that, for most practical applications, hydrogen atoms can be inferred from the heavy atoms in the structure, and thus can be removed. In fact, methods trained solely on heavy atoms usually perform better since they consider smaller graphs.

We measure validity using the success rate of RDKit sanitization over 10,000 molecules. Uniqueness is the proportion of valid molecules with different canonical SMILES. Atom and molecule stability are metrics proposed in [38] – they are similar to validity but, in contrast to RDKit sanitization, they do not allow for adding implicit hydrogens to satisfy the valency constraints. Novelty is the proportion of unique canonical SMILES strings obtained that are not in the training set. Since all training molecules have a single connected component, we also measure the proportion of the generated molecules that are connected.

We also compare the histograms of several properties of the generated set with a test set. The atom and bond total variations (AtomTV and BondTV) measure the l1 distance between the marginal distribution of atom types and bond types, respectively, in the generated set and test set. The Wasserstein

---

[3] http://openbabel.org/wiki/Bond_Orders

distance between valencies is a weighted sum over the valency distributions for each atom types: $\text{ValencyW}_1 = \sum_{x \in \text{atom types}} p(x)\mathcal{W}_1(\hat{D}_{\text{val}}(x), D_{\text{val}}(x))$, where $p^X(x)$ is the marginal distribution of atom types in the training set, $\hat{D}_{\text{val}}(x)$ is the marginal distribution of valencies for atoms of type x in the generated set, $D_{\text{val}}(x)$ the same distribution in the test set. Here, the Wasserstein distance between histograms is used rather than total variation, as it allows to better respect the structure of ordinal data.

In previous methods, graph-based metrics were predominantly used. However, in our approach, we also introduce 3D metrics based on histograms of bond lengths and bond angles. This allows us to evaluate the efficacy of our approach not only in terms of the graph structure but also in generating accurate conformers. To this end, we report a weighted sum of the distance between bond lengths for each bond type:

$$\text{BondLenghtsW}_1 = \sum_{y \in \text{bond types}} p(y)\mathcal{W}_1(\hat{D}_{\text{dist}}(y), D_{\text{dist}}(y)),$$

where $p^Y(y)$ is the proportion of bonds of type y in the training set, $\hat{D}_{\text{dist}}(y)$ is the generated distribution of bond lengths for bond of type y, and $D_{\text{dist}}(y)$ is the same distribution computed over the test set. The output is value in Angstrom.

Finally, BondAnglesW_1 (in degrees) compares the distribution of bond angles (in degrees) for each atom type. We compute a weighted sum of these values using the proportion of each atom type in the dataset. This calculation is restricted to atoms with two or more neighbors to ensure that angles can be defined:

$$\text{BondAnglesW}_1(\text{generated}, \text{target}) = \sum_{x \in \text{atom types}} \tilde{p}(x)\mathcal{W}_1(\hat{D}_{\text{angles}}(x), D_{\text{angles}}(x)),$$

where $\tilde{p}^X(x)$ denotes the proportion of atoms of type x in the training set, restricted to atoms with two neighbors or more, and $D_{\text{angles}}(x)$ is the distribution of geometric angles of the form $\angle(\boldsymbol{r}_k - \boldsymbol{r}_i, \boldsymbol{r}_j - \boldsymbol{r}_i)$, where i is an atom of type x, and k and j are neighbors of i. The reported metrics are mean and 95% confidence intervals on 5 different samplings from the same checkpoint.

5.2   QM9

We first evaluate our model on the standard QM9 dataset [51] containing molecules with up to 9 heavy atoms. We split the dataset into 100k molecules for training, 20k for validation, and 13k for testing. Results are presented in Table 2. The *data* line represents the results of the training set compared with the test set, while the other entries compare the generated molecules to the test molecules. As we observe in Table 2, predicting the bonds only from the interatomic distances and atom types has limited performance. Therefore, MiDi outperforms EDM on 2D metrics, while obtaining similar 3D metrics for the generated conformers. It is worth noting that our list of allowed bonds is not identical to that used in [16, 38], which may explain why our results for EDM [16] do not match

Table 2: Unconditional generation on QM9 with explicit hydrogens with uniform and adaptive noise schedules. While MiDi outperforms the base EDM model on graph-based metrics, the Open Babel optimization procedure is very effective on this simple dataset, as the structures are simple enough for the bonds to be determined unambiguously from the conformation.

Metrics (↑)	Mol stable	At stable	Validity	Uniqueness	Novelty	Connected
Data	98.7	99.8	98.9	99.9	–	100.0
GSchNet	92.0	98.7	98.1	94.5	**80.5**	97.1
EDM	90.7	99.2	91.7	**98.5**	75.9	99.3
EDM + OBabel	**97.9**	**99.8**	**99.0**	**98.5**	77.8	99.7
MiDi (uniform)	$96.1_{\pm.2}$	$\mathbf{99.7}_{\pm.0}$	$96.6_{\pm.2}$	$97.6_{\pm.1}$	$64.9_{\pm.5}$	$99.8_{\pm.0}$
MiDi (adaptive)	$\mathbf{97.5}_{\pm.1}$	$\mathbf{99.8}_{\pm.0}$	$97.9_{\pm.1}$	$97.6_{\pm.1}$	$67.5_{\pm.3}$	$\mathbf{99.9}_{\pm.0}$

Metrics (↓)	Valency(e-2)	Atom(e-2)	Bond(e-2)	Angles (°)	Bond Lengths (e-2 Å)
Data	0.1	0.3	~ 0	0.12	~ 0
GSchNet	4.9	4.2	1.1	1.68	0.5
EDM	1.1	2.1	0.2	**0.44**	**0.1**
EDM + OBabel	1.1	2.1	**0.1**	**0.44**	**0.1**
MiDi (uniform)	$0.4_{\pm.0}$	$0.9_{\pm.0}$	$\mathbf{0.1}_{\pm0.0}$	$0.67_{\pm.02}$	$1.6_{\pm.7}$
MiDi (adaptive)	$\mathbf{0.3}_{\pm.0}$	$\mathbf{0.3}_{\pm.1}$	$\mathbf{0.0}_{\pm.0}$	$0.62_{\pm.02}$	$0.3_{\pm.1}$

those of the original paper perfectly. Nonetheless, the optimization algorithm of OpenBabel performs very well on this dataset of simple molecules. As QM9 contains molecules with only up to 9 atoms, the molecular conformations are easy to understand and the bonds can be determined easily.

5.3 GEOM-DRUGS

We then assess our model on the much larger GEOM-DRUGS dataset [2] which comprises 430,000 drug-sized molecules with an average of 44 atoms and up to 181 atoms. As this dataset features drug-like compounds, it is therefore better suited for downstream applications than QM9. We split the dataset into 80% for training, 10% for validation, and 10% for testing. For each molecule, we extract the 5 lowest energy conformations to build the dataset. Results are presented in Table 3. On this large dataset, we did not train the adaptive version of MiDi from scratch, but instead fine-tuned it using a checkpoint of MiDi with uniform noise schedule.

As this dataset contains molecules that are much more complex than those in QM9, the bonds in the molecules cannot be determined solely from pairwise distances. This explains why EDM, which performs relatively well on 3D-based metrics, produces very few valid and stable molecules. Furthermore, many structures in this dataset are too complex for the Open Babel algorithm. While the latter achieves good atom stability, there is at least one invalid atom in most molecules, leading to low molecular stability. The advantages of an end-to-end

Table 3: Unconditional generation on GEOM-Drugs with explicit hydrogens. EDM was previously the only method that scaled to this dataset. On this complex dataset, the benefits of an integrated models are very clear, as MiDi significantly outperforms Open Babel on most metrics. 95% confidence intervals are reported on five samplings of the same checkpoint.

Metrics (↑)	Mol stable	At stable	Validity	Uniqueness	Novelty	Connected
Data	99.9	99.9	99.8	100.0	–	100.0
EDM	5.5	92.9	**97.5**	99.9	**100.0**	35.6
EDM + OBabel	40.3	97.8	87.8	99.9	**100.0**	41.4
MiDi (uniform)	89.9$_{\pm.2}$	99.7$_{\pm.0}$	74.5$_{\pm.2}$	**100.0**$_{\pm.0}$	**100.0**$_{\pm.0}$	**90.5**$_{\pm.2}$
MiDi (adaptive)	**91.6**$_{\pm.2}$	**99.8**$_{\pm.0}$	77.8$_{\pm.2}$	**100.0**$_{\pm.0}$	**100.0**$_{\pm.0}$	90.0$_{\pm.3}$

Metrics (↓)	Valency(e-2)	Atom(e-2)	Bond(e-2)	Angles (°)	Bond Lengths (e-2 Å)
Data	0.1	0.1	2.5	0.05	~ 0
EDM	11.2	21.2	4.9	6.23	**0.2**
EDM + OBabel	28.5	21.2	4.8	6.42	**0.2**
MiDi (uniform)	2.9$_{\pm.0}$	3.9$_{\pm.1}$	**2.4**$_{\pm.0}$	1.43$_{\pm.002}$	1.1$_{\pm.2}$
MiDi (adaptive)	**0.8**$_{\pm.1}$	**3.8**$_{\pm.1}$	**2.4**$_{\pm.0}$	**1.07**$_{\pm.02}$	**0.2**$_{\pm.1}$

model that generates both a graph structure and its conformation are evident on this dataset: MiDi not only generates better molecular graphs, but also predicts 3D conformers with more realistic bond angles.

6   Conclusions

We propose MiDi, a novel denoising diffusion model that jointly generates a molecular graph and a corresponding conformation for this molecule. Our model combines Gaussian and discrete diffusion in order to define a noise model that is best suited to each component. The noise schedule is further adapted to the different components, with the network initially generating a rough estimate of the conformation and the graph structure, before tuning the atom types and charges. A graph transformer network is trained to denoise this model, that features novel rEGNN layers. While rEGNN layers manipulate features that are translation invariant, they still result in a SE(3) equivariant network when the input molecules are centered. On unconditional generation tasks with complex datasets, MiDi clearly outperforms prior 3D molecule generation methods that predict bonds from the conformation using predefined rules. While our model is currently evaluated on unconditional generation tasks, we believe that the end-to-end training of the graph structure and the conformation can offer even greater benefits for other downstream tasks such as pocket-conditioned generation.

7   Acknowledgements

Bibliography

[1] Austin, J., Johnson, D., Ho, J., Tarlow, D., van den Berg, R.: Structured denoising diffusion models in discrete state-spaces. Advances in Neural Information Processing Systems **34** (2021) 5

[2] Axelrod, S., Gomez-Bombarelli, R.: Geom: Energy-annotated molecular conformations for property prediction and molecular generation. arXiv preprint arXiv:2006.05531 (2020) 13

[3] Ba, J.L., Kiros, J.R., Hinton, G.E.: Layer normalization. arXiv preprint arXiv:1607.06450 (2016) 10

[4] Bao, F., Zhao, M., Hao, Z., Li, P., Li, C., Zhu, J.: Equivariant energy-guided sde for inverse molecular design. arXiv preprint arXiv:2209.15408 (2022) 3

[5] Brandstetter, J., Hesselink, R., van der Pol, E., Bekkers, E.J., Welling, M.: Geometric and physical quantities improve e (3) equivariant message passing. arXiv preprint arXiv:2110.02905 (2021) 6

[6] Corso, G., Cavalleri, L., Beaini, D., Liò, P., Veličković, P.: Principal neighbourhood aggregation for graph nets. Advances in Neural Information Processing Systems **33**, 13260–13271 (2020) 9

[7] Corso, G., Stärk, H., Jing, B., Barzilay, R., Jaakkola, T.: Diffdock: Diffusion steps, twists, and turns for molecular docking. arXiv preprint arXiv:2210.01776 (2022) 3

[8] Du, Y., Fu, T., Sun, J., Liu, S.: Molgensurvey: A systematic survey in machine learning models for molecule design. arXiv preprint arXiv:2203.14500 (2022) 2

[9] Dwivedi, V.P., Bresson, X.: A generalization of transformer networks to graphs. AAAI Workshop on Deep Learning on Graphs: Methods and Applications (2021) 10

[10] Gasteiger, J., Becker, F., Günnemann, S.: Gemnet: Universal directional graph neural networks for molecules. Advances in Neural Information Processing Systems **34**, 6790–6802 (2021) 6

[11] Gebauer, N.W., Gastegger, M., Schütt, K.T.: Symmetry-adapted generation of 3d point sets for the targeted discovery of molecules. arXiv preprint arXiv:1906.00957 (2019) 2, 3, 11

[12] Haefeli, K.K., Martinkus, K., Perraudin, N., Wattenhofer, R.: Diffusion models for graphs benefit from discrete state spaces. arXiv preprint arXiv:2210.01549 (2022) 2, 3

[13] Hajduk, P.J., Greer, J.: A decade of fragment-based drug design: strategic advances and lessons learned. Nature reviews Drug discovery **6**(3), 211–219 (2007) 3

[14] Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. In: Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., Lin, H. (eds.) Advances in Neural Information Processing Systems. vol. 33, pp. 6840–6851. Curran Associates, Inc. (2020), https://proceedings.neurips.cc/paper/2020/file/4c5bcfec8584af0d967f1ab10179ca4b-Paper.pdf 2, 4

[15] Hoogeboom, E., Nielsen, D., Jaini, P., Forré, P., Welling, M.: Argmax flows and multinomial diffusion: Learning categorical distributions. Advances in Neural Information Processing Systems **34** (2021) 6

[16] 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. PMLR (2022) 2, 3, 5, 9, 11, 12

[17] Hua, C., Luan, S., Xu, M., Ying, R., Fu, J., Ermon, S., Precup, D.: Mudiff: Unified diffusion for complete molecule generation. arXiv preprint arXiv:2304.14621 (2023) 3

[18] Huang, H., Sun, L., Du, B., Fu, Y., Lv, W.: Graphgdp: Generative diffusion processes for permutation invariant graph generation. arXiv preprint arXiv:2212.01842 (2022) 3

[19] Huang, L., Zhang, H., Xu, T., Wong, K.C.: Mdm: Molecular diffusion model for 3d molecule generation. arXiv preprint arXiv:2209.05710 (2022) 3, 6

[20] Igashov, I., Stärk, H., Vignac, C., Satorras, V.G., Frossard, P., Welling, M., Bronstein, M., Correia, B.: Equivariant 3d-conditional diffusion models for molecular linker design. arXiv preprint arXiv:2210.05274 (2022) 3, 6, 11

[21] Ingraham, J., Baranov, M., Costello, Z., Frappier, V., Ismail, A., Tie, S., Wang, W., Xue, V., Obermeyer, F., Beam, A., et al.: Illuminating protein space with a programmable generative model. bioRxiv (2022) 4

[22] Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: International conference on machine learning. pp. 448–456. PMLR (2015) 10

[23] Jaderberg, M., Simonyan, K., Zisserman, A., et al.: Spatial transformer networks. Advances in neural information processing systems **28** (2015) 8

[24] Jin, W., Barzilay, R., Jaakkola, T.: Hierarchical generation of molecular graphs using structural motifs. In: International Conference on Machine Learning. pp. 4839–4848. PMLR (2020) 3

[25] Jo, J., Lee, S., Hwang, S.J.: Score-based generative modeling of graphs via the system of stochastic differential equations. arXiv preprint arXiv:2202.02514 (2022) 3

[26] Kaba, S.O., Mondal, A.K., Zhang, Y., Bengio, Y., Ravanbakhsh, S.: Equivariance with learned canonicalization functions. In: NeurIPS 2022 Workshop on Symmetry and Geometry in Neural Representations (2022), https://openreview.net/forum?id=pVD1k8ge25a 8

[27] Liao, R., Li, Y., Song, Y., Wang, S., Nash, C., Hamilton, W.L., Duvenaud, D., Urtasun, R., Zemel, R.: Efficient graph generation with graph recurrent attention networks. In: NeurIPS (2019) 3

[28] Liao, Y.L., Smidt, T.: Equiformer: Equivariant graph attention transformer for 3d atomistic graphs. arXiv preprint arXiv:2206.11990 (2022) 6, 10

[29] Liu, Q., Allamanis, M., Brockschmidt, M., Gaunt, A.: Constrained graph variational autoencoders for molecule design. Advances in neural information processing systems **31** (2018) 3

[30] Luo, T., Mo, Z., Pan, S.J.: Fast graph generative model via spectral diffusion. arXiv preprint arXiv:2211.08892 (2022) 3

[31] Madhawa, K., Ishiguro, K., Nakago, K., Abe, M.: Graphnvp: An invertible flow model for generating molecular graphs. arXiv preprint arXiv:1905.11600 (2019) 8

[32] Maziarz, K., Jackson-Flux, H.R., Cameron, P., Sirockin, F., Schneider, N., Stiefl, N., Segler, M., Brockschmidt, M.: Learning to extend molecular scaffolds with structural motifs. In: International Conference on Learning Representations (ICLR) (2022) 3

[33] Mercado, R., Rastemo, T., Lindelöf, E., Klambauer, G., Engkvist, O., Chen, H., Bjerrum, E.J.: Graph networks for molecular design. Machine Learning: Science and Technology **2**(2), 025023 (2021) 3

[34] Morehead, A., Cheng, J.: Geometry-complete diffusion for 3d molecule generation. arXiv preprint arXiv:2302.04313 (2023) 3

[35] O'Boyle, N.M., Banck, M., James, C.A., Morley, C., Vandermeersch, T., Hutchison, G.R.: Open babel: An open chemical toolbox. Journal of cheminformatics **3**(1), 1–14 (2011) 2, 3, 11

[36] Peng, X., Guan, J., Liu, Q., Ma, J.: Moldiff: Addressing the atom-bond inconsistency problem in 3d molecule diffusion generation. arXiv preprint arXiv:2305.07508 (2023) 3

[37] Satorras, V.G., Hoogeboom, E., Fuchs, F., Posner, I., Welling, M.: E(n) equivariant normalizing flows. Advances in Neural Information Processing Systems **34** (2021) 2, 3, 11

[38] Satorras, V.G., Hoogeboom, E., Fuchs, F.B., Posner, I., Welling, M.: E (n) equivariant normalizing flows. arXiv preprint arXiv:2105.09016 (2021) 11, 12

[39] Satorras, V.G., Hoogeboom, E., Welling, M.: E (n) equivariant graph neural networks. arXiv preprint arXiv:2102.09844 (2021) 2, 6, 8

[40] 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) 3, 6, 11

[41] Shi, C., Wang, C., Lu, J., Zhong, B., Tang, J.: Protein sequence and structure co-design with equivariant translation. arXiv preprint arXiv:2210.08761 (2022) 4

[42] Sohl-Dickstein, J., Weiss, E.A., Maheswaranathan, N., Ganguli, S.: Deep unsupervised learning using nonequilibrium thermodynamics. In: Bach, F.R., Blei, D.M. (eds.) Proceedings of the 32nd International Conference on Machine Learning, ICML (2015) 2, 4

[43] Song, Y., Ermon, S.: Generative modeling by estimating gradients of the data distribution. Advances in Neural Information Processing Systems **32** (2019) 4

[44] Thomas, N., Smidt, T., Kearnes, S.M., Yang, L., Li, L., Kohlhoff, K., Riley, P.: Tensor field networks: Rotation- and translation-equivariant neural networks for 3d point clouds. CoRR **abs/1802.08219** (2018) 6

[45] 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) 8

[46] Vignac, C., Frossard, P.: Top-n: Equivariant set and graph generation without exchangeability. arXiv preprint arXiv:2110.02096 (2021) 8

[47] Vignac, C., Krawczuk, I., Siraudin, A., Wang, B., Cevher, V., Frossard, P.: Digress: Discrete denoising diffusion for graph generation. In: The Eleventh International Conference on Learning Representations (2023), `https://openreview.net/forum?id=UaAD-Nu86WX` 2, 3, 5, 6, 9

[48] Wang, Z., Nie, W., Qiao, Z., Xiao, C., Baraniuk, R., Anandkumar, A.: Retrieval-based controllable molecule generation. arXiv preprint arXiv:2208.11126 (2022) 3

[49] 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.: Broadly applicable and accurate protein design by integrating structure prediction networks and diffusion generative models. bioRxiv (2022) 4

[50] Wu, K.E., Yang, K.K., Berg, R.v.d., Zou, J.Y., Lu, A.X., Amini, A.P.: Protein structure generation via folding diffusion. arXiv preprint arXiv:2209.15611 (2022) 4

[51] Wu, Z., Ramsundar, B., Feinberg, E., Gomes, J., Geniesse, C., Pappu, A.S., Leswing, K., Pande, V.: Moleculenet: a benchmark for molecular machine learning. Chem. Sci. **9**, 513–530 (2018). https://doi.org/10.1039/C7SC02664A, `http://dx.doi.org/10.1039/C7SC02664A` 12

[52] Xu, M., Yu, L., Song, Y., Shi, C., Ermon, S., Tang, J.: Geodiff: A geometric diffusion model for molecular conformation generation. In: International Conference on Learning Representations (2022), `https://openreview.net/forum?id=PzcvxEMzvQC` 3, 5, 6

[53] Zhang, L.H., Tozzo, V., Higgins, J.M., Ranganath, R.: Set norm and equivariant skip connections: Putting the deep in deep sets (2022), `https://openreview.net/forum?id=MDT3OTEtaVY` 10

A   Additional Results

A.1   QM9 with Implicit Hydrogens

The results are presented in Table 4 for 5 samplings of the same checkpoint. While all methods overall achieve good results on this simple dataset, we observe that the lookup table of EDM sometimes fails to predict the bond type, resulting in much more invalid molecules than our model. Interestingly, the adaptive noise schedule that allowed for important improvements on the GEOM-DRUG dataset is not effective on this simpler dataset, and the uniform schedule seems to perform better. The reasons for this phenomenon. We finally observe that the bond length predictions are overall good for all methods, but that MiDi is not as precise as EDM, which can be explained by the fact that EDM uses both learning rate decay and an exponential moving average.

Table 4: Unconditional generation on QM9 with implicit hydrogens. On this simple dataset, all methods achieve very good results, although the lookup table of EDM sometimes fail to generate correct edge types, resulting in invalid molecules.

Metric ($\uparrow$)	Validity	Uniqueness	Novelty	Connected
Data	99.5	99.9	–	100
EDM	96.8	**96.6**	45.5	99.9
EDM + OBabel	**100.0**	96.1	45.4	99.9
MiDi (uniform)	$99.5_{\pm.1}$	$95.8_{\pm.2}$	$\mathbf{49.2}_{\pm0.4}$	$\mathbf{100}_{\pm.0}$
MiDi (adaptive)	$99.7_{\pm.0}$	$93.9_{\pm.2}$	$44.7_{\pm.5}$	$\mathbf{100}_{\pm.0}$

Metric ($\downarrow$)	Valency(e-2)	Atom(e-2)	Bond(e-2)	Bond Lengths(e-2 Å)
Data	0.6	0.1	0.1	0.3
EDM	4.3	**2.9**	0.9	**0.2**
EDM + OBabel	3.8	**2.9**	**0.3**	**0.2**
MiDi (uniform)	$\mathbf{2.3}_{\pm1.7}$	$3.1_{\pm3.7}$	$\mathbf{0.4}_{\pm.1}$	$0.8_{\pm.3}$
Midi (adaptive)	$5.9_{\pm.2}$	$11.7_{\pm0.2}$	$1.1_{\pm.0}$	$1.6_{\pm.7}$

Table 5: Unconditional generation on GEOM-DRUGS with implicit hydrogens.

Metric ($\uparrow$)	Validity	Uniqueness	Novelty	Connected
Data	100.0	100.0	–	99.8
MiDi (uniform)	$93.2_{\pm.1}$	$\mathbf{100.0}_{\pm.0}$	$\mathbf{100.0}_{\pm.0}$	$98.0_{\pm.1}$
MiDi (adaptive)	$\mathbf{96.7}_{\pm.1}$	$\mathbf{100.0}_{\pm.0}$	$\mathbf{100.0}_{\pm.0}$	$\mathbf{98.7}_{\pm.1}$

Metric ($\downarrow$)	Valency(e-2)	Atom(e-2)	Bond(e-2)	Bond Lengths(e-2 Å)
Data	~ 0	~ 0	7.9	~ 0
MiDi (uniform)	$13.6_{\pm.1}$	$9.5_{\pm.3}$	$8.8_{\pm.0}$	$1.6_{\pm.2}$
Midi (adaptive)	$\mathbf{5.3}_{\pm.1}$	$\mathbf{6.3}_{\pm0.1}$	$\mathbf{8.0}_{\pm.0}$	$\mathbf{0.6}_{\pm.3}$

A.2   GEOM-DRUGS with implicit hydrogens
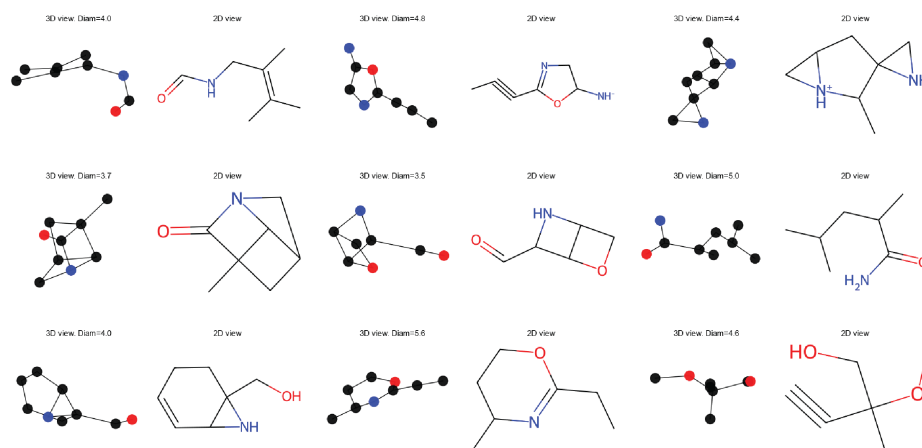
B   Samples from our model

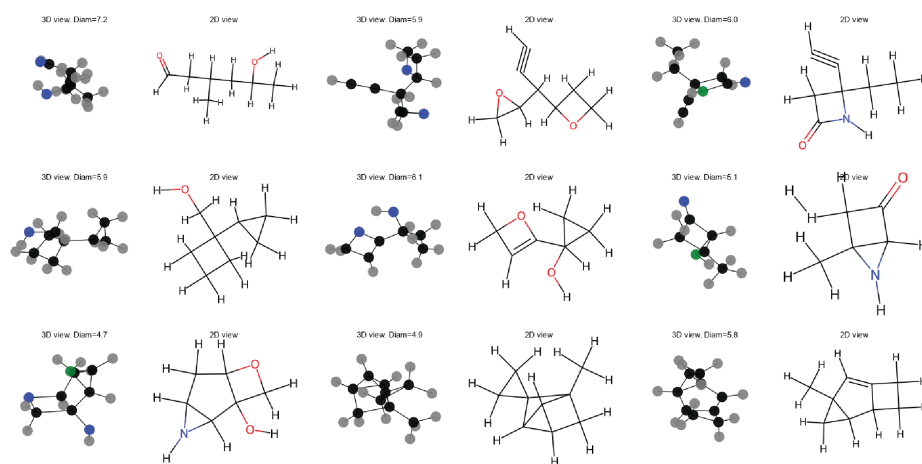Fig. 4: Non-curated samples on QM9 with implicit hydrogens.

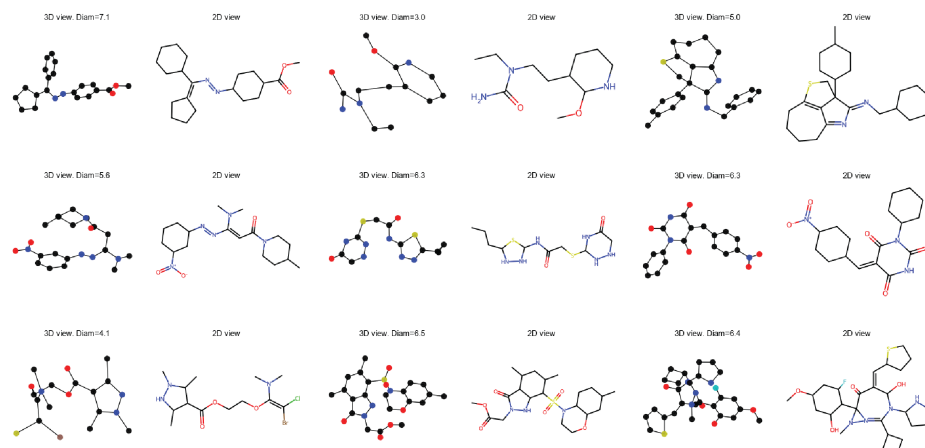Fig. 5: Non-curated samples on QM9 with explicit hydrogens.

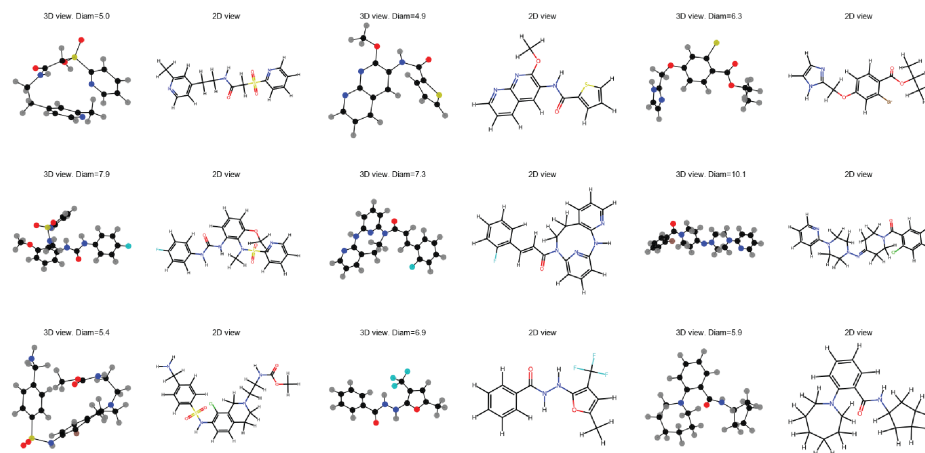Fig. 6: Non-curated samples on GEOM-drugs with implicit hydrogens.

Fig. 7: Non-curated samples on QEOM-drugs with explicit hydrogens.