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

Anonymous authors
Paper under double-blind review

Abstract

This work introduces MiDi, a diffusion model for jointly generating molecular graphs and corresponding 3D conformers. In contrast to existing models which derive molecular bonds from the conformation using predefined rules, MiDi streamlines the molecule generation process with an end-to-end differentiable model. Preliminary results demonstrate the benefits of this approach: on the complex GEOM-DRUGS dataset, our model generates significantly better molecular graphs than 3D-based models, and even surpasses specialized algorithms that directly optimize the bond orders for validity.

1 Introduction

The quest for novel, effective molecular compounds is a central challenge in modern drug discovery. To this end, it is crucial to consider both the 2D connectivity structure of the molecule, which determines its synthesis, and the 3D conformation, which governs its interaction with proteins. Yet, existing generative models for molecules are restricted to one of these two data modalities. To generate molecules in 3D, previous works typically train a network to generate conformers, and then predict the 2D structure using interatomic distances (Hoogeboom et al., 2022) or chemical software (Gebauer et al., 2019). These strategies show limited performance on complex tasks, and do not provide a differentiable signal for downstream applications.

Our Mixed Graph+3D Denoising Diffusion (MiDi) model addresses these limitations by jointly generating a graph and corresponding 3D coordinates. MiDi views molecules as attributed graphs whose nodes have 3D coordinates in addition to the atom types and charges, and whose edges have a bond type. The model is trained to generate all these components simultaneously, which allows for controlling both the quality of the generated graphs and the 3D conformation during training. It builds on the recent success of diffusion models (Sohl-Dickstein et al., 2015; Ho et al., 2020) on various data modalities including images (Dhariwal & Nichol, 2021), video (Ho et al., 2022) and proteins (Ingraham et al., 2022). MiDi utilises a Gaussian noise model for the coordinates, while formal charges, atom types and bond types are modeled using discrete diffusion, which was found to be beneficial for graph generation (Vignac et al., 2022; Haefeli et al., 2022).

A Transformer architecture is trained to denoise all these components simultaneously. It features *rEGNN* layers, a relaxation of the EGNN layers (Satorras et al., 2021b) that leverages more expressive features that are not translation invariant. We show that the SE(3) equivariance of the resulting model is still guaranteed, which is key to good performance.

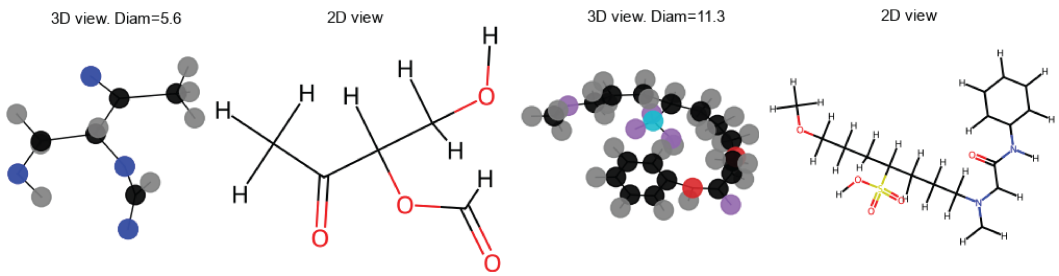


Figure 1: Selected samples from the MiDi model on QM9 (left) and GEOM-DRUGS (right).

Table 1: Gaussian and categorical distributions allow the efficient computation of the key quantities involved in training and sampling from diffusion models. Formulas for all parameters can be found in Appendix A. $\boldsymbol{Q}_t'$ denotes the transpose of $\boldsymbol{Q}_t$.

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

We demonstrate the effectiveness of our model on unconditional molecule generation. MiDi outperforms previous models that use interatomic distances for bond prediction. While the Open Babel bond prediction algorithm (O'Boyle et al., 2011) is very effective on the small QM9 dataset, our model outperforms existing models followed by Open Babel on the more complex Geom-Drugs dataset. Overall, MiDi constitutes an integrated model that can be used for various drug-discovery applications beyond unconditional generation.

2 Background

Diffusion models Diffusion models consist of two essential elements: a noise model, denoted by q, and a denoising neural network ϕ_θ. The noise model takes in a data point x and generates a trajectory $(z_1, ..., z_t)$ of increasingly corrupted data points. The corruption process is chosen to be Markovian, i.e., $q(z_1, \dots, z_T|x) = q(z_1|x) \prod_{t=2}^{T} q(z_t|z_{t-1})$. The denoising network ϕ_θ learns to predict the clean input x from z_t, or, equivalently, the noise that has been added to it. These predictions can then be used to invert the diffusion trajectories by marginalization:

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

This parametrization of diffusion models is more efficient than early diffusion models (Sohl-Dickstein et al., 2015) that were directly trained to predict z_{t-1} (which depends on the sampled diffusion trajectory, resulting in more noisy targets). Nevertheless, it requires to be able to compute efficiently both $q(z_t|x)$ and Eq. (1), which is not always possible. Two main frameworks have been proposed to make this computation efficient: Gaussian noise, which is well suited to continuous data, and discrete state-space diffusion for categorical data. In discrete diffusion, the noise is represented by transition matrices $(\boldsymbol{Q}_1, ..., \boldsymbol{Q}_T)$ such that $[\boldsymbol{Q}_t]_{ij}$ represents the probability of jumping from state i to state j. The noise model is therefore a categorical distribution which reads as $q(z_t|z_{t-1}) \sim \text{categ}(\boldsymbol{z}_{t-1}\boldsymbol{Q}_t)$, where $\boldsymbol{z}_{t-1}$ is seen as a row vector. The main properties of these two noise models are summarized in Table 1.

SE(3)-Equivariance with diffusion models Molecules can be translated, rotated, and their atoms do not have a natural order. To avoid having to augment data with random transformations, generative models should be equivariant to these symmetries. In diffusion models, equivariance to a group G is achieved through several requirements: first, the noise model should be invariant to the action of G: $\forall g \in G, q(g.z_t|g.x) = q(z_t|x)$. The prior distribution q_∞ used at inference should be invariant as well, 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, this 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 element used to represent the training data. We refer the reader to (Xu et al., 2022; Hoogeboom et al., 2022; Vignac et al., 2022) for detailed analyses.

For the special Euclidean group SE(3), many equivariant architectures have been proposed that could serve as the denoising network of a diffusion model (Thomas et al., 2018; Brandstetter et al., 2021; Gasteiger et al., 2021; Liao & Smidt, 2022). However, most of these networks require the costly manipulation of spherical harmonics. For this reason, many generative models for molecules (Hoogeboom et al., 2021; Igashov et al., 2022; Schneuing et al., 2022; Huang et al., 2022b) use the less expressive but cheap EGNN layers (Satorras et al., 2021b). At a high level, EGNN recursively

updates the coordinates $(\boldsymbol{p}_i)$ of a graph with node features $(\boldsymbol{x}_i)$ and edge features $(\boldsymbol{e}_{ij})$ using:

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

The key property of this parametrization is that the message function m takes only rotation-invariant arguments, which, combined with the linear term in $\boldsymbol{p}_j - \boldsymbol{p}_i$, ensures rotation equivariance. The normalization term $c_{ij} = ||\boldsymbol{p}_i - \boldsymbol{p}_j|| + 1$ is needed for numerical stability when stacking layers.

3 MODEL

In this section, we introduce the Mixed Graph+3D denoising diffusion model (MiDi). We model each molecule as a graph $G = (\boldsymbol{P}, \boldsymbol{x}, \boldsymbol{c}, \boldsymbol{E})$ where $\boldsymbol{P}$ is a $n \times 3$ matrix containing the position of each atom, $\boldsymbol{x}$ and $\boldsymbol{c}$ are vectors of length n containing the atom types and formal charge for each atom, and $\boldsymbol{E}$ is a $n \times n$ matrix containing the bond types. We denote $\boldsymbol{X}, \boldsymbol{C}, \mathbf{E}$ the one-hot encoding of $\boldsymbol{x}$, $\boldsymbol{c}$ and $\boldsymbol{E}$ respectively. As other diffusion models for graphs, we consider the absence of bond as a particular bond type. Superscripts denote time steps, so that, for example, $\boldsymbol{P}_i^t$ denotes the coordinates of atom i at time t.

3.1 NOISE MODEL

Our noise model corrupts the features of each node and edge independently, with a noise model depending on the data type. For the positions, we use a Gaussian noise $\boldsymbol{\epsilon}$ in the zero center-of-mass (CoM) subspace, which is required to obtain a roto-translation equivariant architecture (Xu et al., 2022). That is, 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 the atom types, the formal charges and the bond types, we use discrete diffusion, meaning that the noise model is a sequence of categorical distributions. We chose the marginal transition model defined in (Vignac et al., 2022). For example, if $\boldsymbol{m} \in \mathbb{R}^a$ is the marginal distribution of atom types in the training set, we define $\boldsymbol{Q}_t^X = \alpha_t \boldsymbol{I} + \beta_t\, \mathbf{1}_a \boldsymbol{m}'$. $\boldsymbol{Q}_t^C$ and $\boldsymbol{Q}_t^E$ are defined similarly.

The resulting model reads $q(G_t|G_{t-1}) \sim \mathcal{N}^{\text{CoM}}(\alpha_t \boldsymbol{P}^{t-1},\ \sigma_t^2 \boldsymbol{I}) \times \text{categ}(\boldsymbol{X}^{t-1}\boldsymbol{Q}_t^X) \times \text{categ}(\boldsymbol{C}^{t-1}\boldsymbol{Q}_t^C) \times \text{categ}(\mathbf{E}^{t-1}\boldsymbol{Q}_t^E)$. When generating new samples, we similarly define the posterior as a product:

$$p_\theta(G^{t-1}|G^t) = \prod_{1\le i\le n} p_\theta(\boldsymbol{p}_i^{t-1}|G^t) p_\theta(\boldsymbol{c}_i^{t-1}|G^t) p_\theta(\boldsymbol{x}_i^{t-1}|G^t) \prod_{1\le i,j\le n} p_\theta(\boldsymbol{E}_{ij}^{t-1}|G^t) \tag{2}$$

where each term is computed by marginalizing over the network predictions, for example,

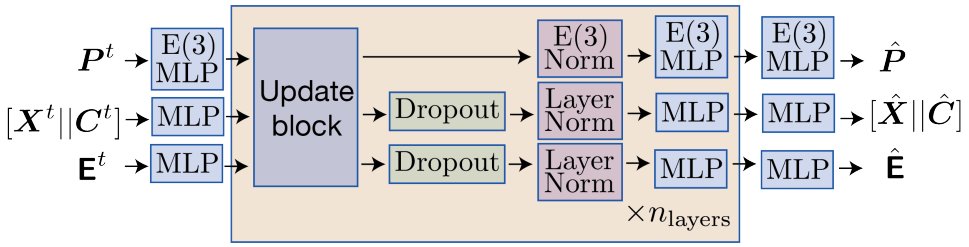
$$p_\theta(\boldsymbol{x}_i^{t-1}|G^t) = \int_{\boldsymbol{x}_i} p_\theta(\boldsymbol{x}_i^{t-1} \mid \boldsymbol{x}_i, G^t)\, dp_\theta(\boldsymbol{x}_i|G^t) = \sum_{x\in\mathcal{X}} q(\boldsymbol{x}_i^{t-1} \mid \boldsymbol{x}_i = x, G^t)\, \hat{p}_i^X(x).$$

For faster sampling, we directly predict G^{t-k} with $k > 1$ from G^t as proposed in Song et al. (2020). Empiricially, setting $k = 5$ does not seem to harm sampling quality.

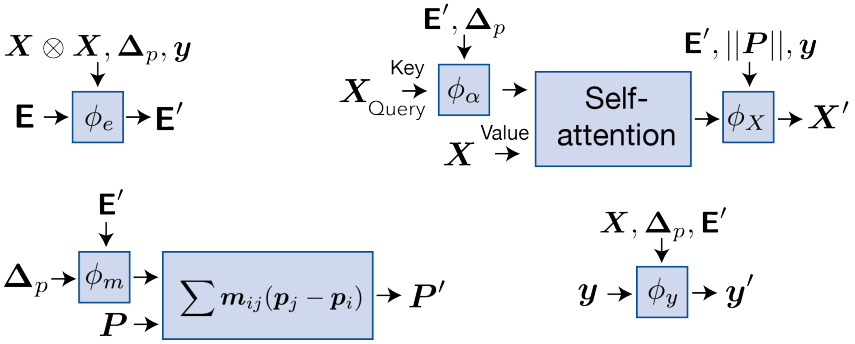
3.2 DENOISING NETWORK

The denoising network takes a noisy graph as input and tries to predict the corresponding clean graph. It manipulates node features (atom types and formal charges, treated together in a matrix $\boldsymbol{X}$), edge features $\mathbf{E}$, graph-level features $\boldsymbol{y}$ and node coordinates $\boldsymbol{P}$. Coordinates are treated separately from the other node features in order to guarantee SE(3) equivariance. The neural network architecture is summarized in Figure 2. It follows a Transformer architecture (Vaswani et al., 2017), with a succession of self-attention module followed by normalization layers and feedforward networks.

Relaxed Equivariant Graph Neural Networks (rEGNNs) As previous generative models for molecules in 3D, we start from the cheap but effective EGNN layers (Satorras et al., 2021b) to process the coordinates. To improve upon these layers, we observe that, when the data and the noise are both in the zero Center-Of-Mass subspace, there is no need for the neural network to be translation



(a) The denoising network of MiDi follows the standard Transformer architecture, except that the layers are adapted to preserve SE(3) equivariance.



(b) Architecture of an update block. Each component is updated using the other features. Graph-level features $\boldsymbol{y}$ are not used for the final prediction, but they are useful to store information efficiently in the transformer layers. $(\boldsymbol{\Delta}_p)_{ij} = \text{cat}(\boldsymbol{p}_i, \boldsymbol{p}_j, ||\boldsymbol{p}_i - \boldsymbol{p}_j||, \cos \boldsymbol{p}_i, \boldsymbol{p}_j)$.

Figure 2: The denoising neural network of MiDi simultaneously predicts the 2D graph and 3D coordinates of the clean graph from a noisy input.

invariant. This can be interpreted as defining a canonical pose for the translation group, which is a valid way to achieve equivariance (Jaderberg et al., 2015; Kaba et al., 2022). Instead of simply using pairwise distances $||\boldsymbol{p}_i - \boldsymbol{p}_j||_2$, we can therefore use other rotation invariant descriptors such as $||\boldsymbol{p}_i||$ or $\cos(\boldsymbol{p}_i, \boldsymbol{p}_j)$.

We therefore propose the following rEGNN layer:

$$[\boldsymbol{\Delta}_p]_{ij} = \text{cat}(||\boldsymbol{p}_i - \boldsymbol{p}_j||_2, ||\boldsymbol{p}_i||_2, ||\boldsymbol{p}_j||_2, \cos(\boldsymbol{p}_i, \boldsymbol{p}_j))$$
$$\boldsymbol{p}_i \leftarrow \boldsymbol{p}_i + \sum_j \phi_m(\boldsymbol{X}_i, \boldsymbol{X}_j, [\Delta_p]_{ij}, \mathbf{E}_{ij})\ (\boldsymbol{p}_j - \boldsymbol{p}_i)$$

Update block As our model simultaneously processes node features, edge features, graph level features and coordinates, this new layer is integrated in an update block. The edge features are first updated using $\boldsymbol{\Delta}_p$, 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 $\boldsymbol{\Delta}_p$. After the attention heads have been flattened, the obtained values are modulated by the pooled edge features, the norm of the coordinates and the global features. 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 $\boldsymbol{\Delta}_p$ and the updated edge features $\mathbf{E}'$. 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 that the normalisation term is unnecessary in this case. We also considered breaking the E(3) equivariance of EGNN to preserve only SE(3) properties, but found no practical benefit to it.

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

Our feedforward neural networks process 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

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

where $||\boldsymbol{P}|| \in \mathbb{R}^n$ contains the norm of each point, $\text{MLP}(||\boldsymbol{P}||) \in \mathbb{R}^n$ as well, ϵ is a small positive constant, and Π^{CoM} is the projection of the positions on the linear subspace with center of mass at 0.

The choice of the normalization layer also depends on the problem symmetries: while batch normalization (Ioffe & Szegedy, 2015) is used in the standard Transformer model, Transformers for sets and graphs instead use Set Normalization (Zhang et al., 2022) or Layer Normalization (Ba et al., 2016), which respect permutation equivariance. For SE(3) equivariance, the normalization of (Liao & Smidt, 2022) should be used. In the special case of coordinates, it writes:

$$\text{E3Norm}(\boldsymbol{P}) = \gamma\frac{||\boldsymbol{P}||}{\bar{n} + \epsilon}\frac{\boldsymbol{P}}{||\boldsymbol{P}||} = \gamma\frac{\boldsymbol{P}}{\bar{n} + \epsilon} \quad \text{with} \quad \bar{n} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}||p_i||^2},$$

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

3.3 Training objective

The network is trained to predict the clean molecule from a noisy input G^t. Correctly estimating the position $\boldsymbol{P}$ is a regression problem that can simply be solved with mean-squared error, while the prediction of the atom types, formal charges and bond types corresponds to a classification problem solved with a cross-entropy loss. Note that the network predictions for the positions is a pointwise estimate $\hat{\boldsymbol{P}}$, while for the other terms it is a distribution over the possible atom types, bond types or formal charge values. The final loss is a weighted sum of these components:

$$l(\hat{x}, x) = \lambda_0||\hat{\boldsymbol{P}} - \boldsymbol{P}||^2 + \lambda_1 \text{CE}(\boldsymbol{X}, \hat{p}^X) + \lambda_2 \text{CE}(\boldsymbol{C}, \hat{p}^C) + \lambda_3 \text{CE}(\mathbf{E}, \hat{p}^E)$$

4 Experiments

We evaluate MiDi on unconditional molecule generation task. Since it is, to the best of our knowledge, the first method to perform both the generation of the graph structure and the conformer simultaneously, we compare it 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 (Hoogeboom et al., 2022), or the optimization procedure[1] of OpenBabel (O'Boyle et al., 2011) used in other works such as (Igashov et al., 2022; Schneuing et al., 2022). This algorithm optimizes the bond orders of neighboring atoms in order to create a valid molecule, removing all control on the generated graphs. Our goal is to check whether MiDi can achieve similar results with a single end-to-end differentiable model.

As the molecules we consider have a single connected component, we define a molecule as valid if it passes RdKit sanitization filters and it is connected. We also use custom metrics that measure the ability to correctly capture several properties of the data distribution. These metrics are described in Appendix B.

4.1 QM9

We first evaluate our model on the standard QM9 dataset (Wu et al., 2018) containing molecules with up to 9 heavy atoms. We use 100k molecules for training, 20k for validation, and 13k for testing. We evaluate our model with both explicit and implicit Hydrogens (cf Appendix C). Results are presented in Tables 4 and 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 QM9 features simple molecules, predicting the bonds only from the interatomic distances and atom types is a relatively effective strategy. Yet, MiDi is still able to outperform EDM on 2D metrics, while retaining good 3D metrics performance.

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

Table 2: Results on QM9 with explicit hydrogens. MiDi clearly outperforms EDM on graph-based metrics. On this dataset of small graphs, Open Babel post-processing optimization is very effective.

2D metrics	Mol stable	At stable	Valid	Unique	AtomTV	BondTV	ValW1
Data	98.7	99.8	98.9	99.9	0.003	~ 0	0.001
GSchNet + OBabel	92.0	98.7	95.4	94.4	0.042	0.011	0.049
EDM	90.7	99.2	91.2	**98.5**	0.021	0.002	0.011
EDM + OBabel	**97.9**	**99.9**	**98.7**	**98.5**	0.021	**0.001**	0.011
MiDi (2D+3D)	95.7	**99.7**	96.5	98.0	**0.009**	**0.001**	**0.004**
3D metrics		Bond Lengths W1			Bond Angles W1		
Data		0.023			0.12		
GSchNet + OBabel		0.478			1.68		
EDM		**0.105**			0.44		
EDM + OBabel		0.131			0.44		
MiDi (2D+3D)		0.604			**0.42**		

Table 3: Results on GEOM-Drugs with explicit hydrogens. The lookup table of EDM fails to predict the bond types, resulting in few stable molecules. On this complex dataset, the benefits of an end-to-end model is clear, as MiDi surpasses Open Babel on most metrics.

2D metrics	Mol stable	At stable	Valid	Unique	AtomTV	BondTV	ValW1
Data	99.9	99.9	99.8	85.2	0.001	0.025	0.001
EDM	5.5	92.9	34.8	**100.0**	0.212	0.049	0.112
EDM + OBabel	40.3	97.8	35.3	**100.0**	0.212	0.048	0.285
MiDi (2D+3D)	**49.2**	**98.3**	**53.7**	**100.0**	**0.053**	**0.024**	**0.032**
3D metrics		Bond Lengths W1			Bond Angles W1		
Data		0.026			0.05		
EDM (3D)		**0.229**			6.23		
EDM + O. Babel		0.249			6.42		
MiDi (2D+3D)		0.741			**6.02**		

4.2 GEOM-DRUGS

We then evaluate our model on the much larger GEOM dataset (Axelrod & Gomez-Bombarelli, 2020) that features 430,000 drug-sized molecules with 44 atoms on average and up to 181 atoms. We keep 80% of these molecules for training, 10% for validation and 10% for testing. For each molecule, we extract the 5 lowest energy conformations to build the dataset.

As this dataset contains molecules that are much more complex than those in QM9, the bonds in the molecules cannot be only derived from pairwise distances. This explains why EDM, which scores relatively well on 3D based metrics, produces very few valid and stable molecules. On this dataset, the benefits of an end-to-end model that produces both a graph structure and its conformation is clear, as MiDi clearly outperforms EDM.

5 RELATED WORK

Molecule generation in 3D The idea of representing molecules as an attributed point cloud in 3d has been used within several framework, such as autoregressive models (Gebauer et al., 2019) or normalizing flows (Satorras et al., 2021a). Recently, diffusion models have proved to be very effective for this task: EDM (Hoogeboom et al., 2022) uses a fully-connected graph, while MDM (Huang et al., 2022b) only passes messages between neighboring atoms. These models can be conditioned on some molecule-level properties using guidance mechanisms (Bao et al., 2022) or on another point cloud. This strategy has been employed to generate molecules that bind to a specific protein (Corso et al., 2022; Schneuing et al., 2022) and to generate linkers between molecular fragments (Igashov et al., 2022). 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 a lookup table (Hoogeboom et al., 2022) or specialized software such as Open Babel (O'Boyle et al., 2011). This result in limited performance for complex molecules, but also prevents end-to-end differentiability for downstream applications.

On the contrary, conformer generation models (Xu et al., 2022; Corso et al., 2022) generate 3D coordinates given for a known graph structure. As the graph is known, these model can exploit internal symmetries of the molecule (such as the existence of rotatable bonds), which is more difficult on unconditional generation tasks.

Graph generation Another line of work has focused on generating graphs without associated 3D coordinates. This can be done with Gaussian diffusion models operating on the adjacency matrix entries (Jo et al., 2022; Huang et al., 2022a) or graph eigenvalues (Luo et al., 2022). Vignac et al. (2022) and Haefeli et al. (2022) however found that discrete diffusion is more effective, as it better respects the discrete nature of graphs. These diffusion models match the performance of autoregressive methods (Liu et al., 2018; Liao et al., 2019; Mercado et al., 2021) which typically perform validity checks at each sampling step.

Instead of operating at the node level, fragment-based methods (Hajduk & Greer, 2007; Jin et al., 2020; Maziarz et al., 2022) learn to combine chemically-relevant substructures from a fixed or learned dictionary (Wang et al., 2022). While these methods are normally used for 2D molecule generation, they can be used to produce conformers as well (Qiang et al., 2023). Fragment-based methods are often work very effective, and could constitute a promising extension of our model.

Protein generation While existing diffusion models for molecules operate on drug-size molecules of moderate size (up to ~ 200 atoms), recent diffusion models for proteins managed to scale to much larger structures (Watson et al., 2022; Ingraham et al., 2022; Wu et al., 2022; Shi et al., 2022). 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 the SE(3) symmetry in the representation. Those improvements are unfortunately specific to chain graphs, and cannot be used for arbitrary molecules.

6 Conclusion

We proposed MiDi, a denoising diffusion model that jointly generates a molecular graph and a corresponding conformation for this molecule. On complex datasets, MiDi clearly outperforms previous molecule generation methods in 3D, which would 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 will be all the more beneficial for downstream tasks such as pocket-conditioned generation.

References

Simon Axelrod and Rafael Gomez-Bombarelli. Geom: Energy-annotated molecular conformations for property prediction and molecular generation. *arXiv preprint arXiv:2006.05531*, 2020. 6

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016. 5

Fan Bao, Min Zhao, Zhongkai Hao, Peiyao Li, Chongxuan Li, and Jun Zhu. Equivariant energy-guided sde for inverse molecular design. *arXiv preprint arXiv:2209.15408*, 2022. 6

Johannes Brandstetter, Rob Hesselink, Elise van der Pol, Erik J Bekkers, and Max Welling. Geometric and physical quantities improve e (3) equivariant message passing. *arXiv preprint arXiv:2110.02905*, 2021. 2

Gabriele Corso, Hannes Stärk, Bowen Jing, Regina Barzilay, and Tommi Jaakkola. Diffdock: Diffusion steps, twists, and turns for molecular docking. *arXiv preprint arXiv:2210.01776*, 2022. 6, 7

Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021. 1

Johannes Gasteiger, Florian Becker, and Stephan Günnemann. Gemnet: Universal directional graph neural networks for molecules. *Advances in Neural Information Processing Systems*, 34: 6790–6802, 2021. 2

Niklas WA Gebauer, Michael Gastegger, and Kristof T Schütt. Symmetry-adapted generation of 3d point sets for the targeted discovery of molecules. *arXiv preprint arXiv:1906.00957*, 2019. 1, 6

Kilian Konstantin Haefeli, Karolis Martinkus, Nathanaël Perraudin, and Roger Wattenhofer. Diffusion models for graphs benefit from discrete state spaces. *arXiv preprint arXiv:2210.01549*, 2022. 1, 7

Philip J Hajduk and Jonathan Greer. A decade of fragment-based drug design: strategic advances and lessons learned. *Nature reviews Drug discovery*, 6(3):211–219, 2007. 7

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 6840–6851. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/4c5bcfec8584af0d967f1ab10179ca4b-Paper.pdf. 1

Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *arXiv preprint arXiv:2204.03458*, 2022. 1

Emiel Hoogeboom, Didrik Nielsen, Priyank Jaini, Patrick Forré, and Max Welling. Argmax flows and multinomial diffusion: Learning categorical distributions. *Advances in Neural Information Processing Systems*, 34, 2021. 2

Emiel Hoogeboom, Vicctor Garcia Satorras, Clément Vignac, and Max Welling. Equivariant diffusion for molecule generation in 3d. In *International Conference on Machine Learning*, pp. 8867–8887. PMLR, 2022. 1, 2, 5, 6, 7, 11

Han Huang, Leilei Sun, Bowen Du, Yanjie Fu, and Weifeng Lv. Graphgdp: Generative diffusion processes for permutation invariant graph generation. *arXiv preprint arXiv:2212.01842*, 2022a. 7

Lei Huang, Hengtong Zhang, Tingyang Xu, and Ka-Chun Wong. Mdm: Molecular diffusion model for 3d molecule generation. *arXiv preprint arXiv:2209.05710*, 2022b. 2, 6

Ilia Igashov, Hannes Stärk, Clément Vignac, Victor Garcia Satorras, Pascal Frossard, Max Welling, Michael Bronstein, and Bruno Correia. Equivariant 3d-conditional diffusion models for molecular linker design. *arXiv preprint arXiv:2210.05274*, 2022. 2, 5, 6

John Ingraham, Max Baranov, Zak Costello, Vincent Frappier, Ahmed Ismail, Shan Tie, Wujie Wang, Vincent Xue, Fritz Obermeyer, Andrew Beam, et al. Illuminating protein space with a programmable generative model. *bioRxiv*, 2022. 1, 7

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pp. 448–456. PMLR, 2015. 5

Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. Spatial transformer networks. *Advances in neural information processing systems*, 28, 2015. 4

Wengong Jin, Regina Barzilay, and Tommi Jaakkola. Hierarchical generation of molecular graphs using structural motifs. In *International Conference on Machine Learning*, pp. 4839–4848. PMLR, 2020. 7

Jaehyeong Jo, Seul Lee, and Sung Ju Hwang. Score-based generative modeling of graphs via the system of stochastic differential equations. *arXiv preprint arXiv:2202.02514*, 2022. 7

Sékou-Oumar Kaba, Arnab Kumar Mondal, Yan Zhang, Yoshua Bengio, and Siamak Ravanbakhsh. Equivariance with learned canonicalization functions. In *NeurIPS 2022 Workshop on Symmetry and Geometry in Neural Representations*, 2022. URL https://openreview.net/forum?id=pVD1k8ge25a. 4

Renjie Liao, Yujia Li, Yang Song, Shenlong Wang, Charlie Nash, William L. Hamilton, David Duvenaud, Raquel Urtasun, and Richard Zemel. Efficient graph generation with graph recurrent attention networks. In *NeurIPS*, 2019. 7

Yi-Lun Liao and Tess Smidt. Equiformer: Equivariant graph attention transformer for 3d atomistic graphs. *arXiv preprint arXiv:2206.11990*, 2022. 2, 5

Qi Liu, Miltiadis Allamanis, Marc Brockschmidt, and Alexander Gaunt. Constrained graph variational autoencoders for molecule design. *Advances in neural information processing systems*, 31, 2018. 7

Tianze Luo, Zhanfeng Mo, and Sinno Jialin Pan. Fast graph generative model via spectral diffusion. *arXiv preprint arXiv:2211.08892*, 2022. 7

Krzysztof Maziarz, Henry Richard Jackson-Flux, Pashmina Cameron, Finton Sirockin, Nadine Schneider, Nikolaus Stiefl, Marwin Segler, and Marc Brockschmidt. Learning to extend molecular scaffolds with structural motifs. In *International Conference on Learning Representations (ICLR)*, 2022. 7

Rocío Mercado, Tobias Rastemo, Edvard Lindelöf, Günter Klambauer, Ola Engkvist, Hongming Chen, and Esben Jannik Bjerrum. Graph networks for molecular design. *Machine Learning: Science and Technology*, 2(2):025023, 2021. 7

Noel M O'Boyle, Michael Banck, Craig A James, Chris Morley, Tim Vandermeersch, and Geoffrey R Hutchison. Open babel: An open chemical toolbox. *Journal of cheminformatics*, 3(1):1–14, 2011. 2, 5, 7

Bo Qiang, Yuxuan Song, Minkai Xu, Jingjing Gong, Hao Zhou, Wei-Ying Ma, and Yanyan Lan. A HIERARCHICAL FRAGMENT-BASED MODEL FOR 3d DRUG-LIKE MOLECULE GENERATION, 2023. URL https://openreview.net/forum?id=walno7E1F8w. 7

Victor Garcia Satorras, Emiel Hoogeboom, Fabian Fuchs, Ingmar Posner, and Max Welling. E(n) equivariant normalizing flows. *Advances in Neural Information Processing Systems*, 34, 2021a. 6

Victor Garcia Satorras, Emiel Hoogeboom, and Max Welling. E (n) equivariant graph neural networks. *arXiv preprint arXiv:2102.09844*, 2021b. 1, 2, 3

Arne Schneuing, Yuanqi Du, Charles Harris, Arian Jamasb, Ilia Igashov, Weitao Du, Tom Blundell, Pietro Lió, Carla Gomes, Max Welling, et al. Structure-based drug design with equivariant diffusion models. *arXiv preprint arXiv:2210.13695*, 2022. 2, 5, 6

Chence Shi, Chuanrui Wang, Jiarui Lu, Bozitao Zhong, and Jian Tang. Protein sequence and structure co-design with equivariant translation. *arXiv preprint arXiv:2210.08761*, 2022. 7

Jascha Sohl-Dickstein, Eric A. Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In Francis R. Bach and David M. Blei (eds.), *Proceedings of the 32nd International Conference on Machine Learning, ICML*, 2015. 1, 2

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 3, 10

Nathaniel Thomas, Tess Smidt, Steven M. Kearnes, Lusann Yang, Li Li, Kai Kohlhoff, and Patrick Riley. Tensor field networks: Rotation- and translation-equivariant neural networks for 3d point clouds. *CoRR*, abs/1802.08219, 2018. 2

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 3

Clement Vignac, Igor Krawczuk, Antoine Siraudin, Bohan Wang, Volkan Cevher, and Pascal Frossard. Digress: Discrete denoising diffusion for graph generation. *arXiv preprint arXiv:2209.14734*, 2022. 1, 2, 3, 7

Zichao Wang, Weili Nie, Zhuoran Qiao, Chaowei Xiao, Richard Baraniuk, and Anima Anandkumar. Retrieval-based controllable molecule generation. *arXiv preprint arXiv:2208.11126*, 2022. 7

Joseph L Watson, David Juergens, Nathaniel R Bennett, Brian L Trippe, Jason Yim, Helen E Eisenach, Woody Ahern, Andrew J Borst, Robert J Ragotte, Lukas F Milles, et al. Broadly applicable and accurate protein design by integrating structure prediction networks and diffusion generative models. *bioRxiv*, 2022. 7

Kevin E Wu, Kevin K Yang, Rianne van den Berg, James Y Zou, Alex X Lu, and Ava P Amini. Protein structure generation via folding diffusion. *arXiv preprint arXiv:2209.15611*, 2022. 7

Zhenqin Wu, Bharath Ramsundar, Evan N. Feinberg, Joseph Gomes, Caleb Geniesse, Aneesh S. Pappu, Karl Leswing, and Vijay Pande. Moleculenet: a benchmark for molecular machine learning. *Chem. Sci.*, 9:513–530, 2018. doi: 10.1039/C7SC02664A. URL http://dx.doi.org/10.1039/C7SC02664A. 5

Minkai Xu, Lantao Yu, Yang Song, Chence Shi, Stefano Ermon, and Jian Tang. Geodiff: A geometric diffusion model for molecular conformation generation. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=PzcvxEMzvQC. 2, 3, 7

Lily H Zhang, Veronica Tozzo, John M. Higgins, and Rajesh Ranganath. Set norm and equivariant skip connections: Putting the deep in deep sets, 2022. URL https://openreview.net/forum?id=MDT30TEtaVY. 5

A Equations for the noise model

A.1 Gaussian noise

Gaussian diffusion processes are defined by

$$q(z_t|z_{t-1}) \sim \mathcal{N}(\alpha_t \boldsymbol{z}_t, \sigma_t^2 \boldsymbol{I}) \tag{3}$$

As Gaussian distributions are stable under composition, we have:

$$q(z_t|x) \sim \mathcal{N}(\bar{\alpha}_t \boldsymbol{z}_t, \bar{\sigma}_t^2 \boldsymbol{I}) \tag{4}$$

with $\bar{\alpha}_t = \prod_{s=1}^{t} \alpha_s$ or equivalently, $\alpha_t = \bar{\alpha}_t / \bar{\alpha}_{t-1}$. We also have $\bar{\sigma}_t^2 = \sigma_t^2 - \alpha_t^2$.

While any choice of the $(\bar{\alpha}_t)_{t \leq T}$ and $(\bar{\sigma}_t)_{+t \leq T}$ is in theory possible, variance preserving processes are most often used. They satisfy: $\forall 0 \leq t \leq T,\ \bar{\alpha}_t^2 + \bar{\sigma}_t^2 = 1$.

The posterior of the transitions conditioned on x is also normal:

$$q(z_{t-1}|z_t, x) \sim \mathcal{N}(\tilde{\alpha}_t \boldsymbol{x} + (1 - \tilde{\alpha}_t)\boldsymbol{z}_t, \tilde{\sigma}_t^2 \mathbf{I}) \tag{5}$$

with $\tilde{\alpha}_t = \frac{\bar{\alpha}_t \sigma_{t-1}^2}{\sigma_t^2}$ and $\tilde{\sigma}_t = \frac{\bar{\sigma}_t \sigma_s}{\sigma_t}$. Note that these formula can be extended to the computation of $q(z_s|z_t, x)$ for any $s < t$ (Song et al., 2020), which allows for faster sampling with none to little reduction to the sample quality.

A.2 Discrete diffusion

Discrete diffusion considers that data points x belong to one of d classes. We write $\boldsymbol{x} \in \mathbb{R}^{1 \times d}$ the one-hot encoding of x. The noise is represented by transition matrices $(\boldsymbol{Q}^1, ..., \boldsymbol{Q}^T)$ such that $[\boldsymbol{Q}^t]_{ij}$ represents the probability of jumping from state i to state j: $q(z^t|z^{t-1}) = \boldsymbol{z}^{t-1}\boldsymbol{Q}^t$.

Since the process used 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. It gives:

$$q(z^{t-1}|z^t, x) \propto \boldsymbol{z}^t\ (\boldsymbol{Q}^t)' \odot \boldsymbol{x}\ \bar{\boldsymbol{Q}}^{t-1} \tag{6}$$

where $\odot$ denotes a pointwise product and $\boldsymbol{Q}'$ is the transpose of $\boldsymbol{Q}$.

B METRICS

- Validity is measured by the success rate of RdKit sanitization over 10,000 molecules.
- Uniqueness corresponds to the proportion of valid molecules that have different canonical SMILES.
- Atom stability measures the proportion of atoms that have the right valency. Contrary to RdKit sanitization, implicit hydrogens cannot be added to satisfy the valency constraints, so that atom stability is in general a stringer metric. Note that our list of allowed bonds does not exactly match the one of Hoogeboom et al. (2022), in part because our model also handles formal charges. This explains why the results for EDM do not match the original paper.
- Molecule stability: proportion of molecules for which all atoms are stable.
- The atom total variation (AtomTV) measure the distance between the distribution of generated atom types in the generate and test sets:

$$\text{AtomTV}(\text{generated}, \text{target}) = \sum_{x \in \text{atom types}} |\hat{p}(x) - p(x)|$$

- The bond total variation similarly measures the distance between bond types:

$$\text{BondTV}(\text{generated}, \text{target}) = \sum_{e \in \text{bond types}} |\hat{p}(e) - p(e)|$$

- ValencyW1 measures the Wasserstein distance between the histograms of valencies for each atom types, and then computes a weighted sum of these values.

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

  where $D_{\text{val}}(x)$ is the distribution of valencies for atoms with type x.
- BondLengthsW1 computes the Wasserstein distance between the distribution of bond length for each bond type, and then computes a weighted sum of these values.

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

  where $D_{\text{dist}}(e)$ is the distribution of bond lengths for bonds with type x.
- BondAnglesW1 computes the distribution of bond angles (in degrees) for each atom types. We compute a weighted sum of these values using the proportion of each atom type in the dataset, restricted to the atoms that have two neighbors or more (so 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 $\tilde{p}(x)$ denotes the proportion of atoms of type x among the atoms with two neighbors or more, and $D_{\text{angles}}(x)$ is the distribution of geometric angles $\angle(p_k - p_i, p_j - p_i)$, where i is an atom of type x.
- #CC denotes the average number of connected components in the generated graphs. Is it obtained using the GetMolFrags function of RdKit.

C ADDITIONAL RESULTS

QM9 with implicit hydrogens

D SAMPLES FROM THE MODEL

Table 4: Unconditional generation on QM9 with implicit hydrogens. On this simple dataset, the basic lookup table used by EDM for bond prediction achieves good results, but sometimes fail to produce valid molecules.

Metric	*2D metrics*					*3D*
Method	Valid	Unique	AtomTV	BondTV	Valency $\mathcal{W}_1$	Bond Length $\mathcal{W}_1$
Data	99.5	99.9	0.001	0.001	0.006	0.30
EDM (3D)	96.8	**96.6**	0.029	0.009	0.043	**0.19**
EDM + O. Babel	**99.9**	96.1	0.029	**0.003**	0.038	**0.15**
MiDi (2D+3D)	99.4	**96.6**	**0.008**	0.007	**0.024**	1.36

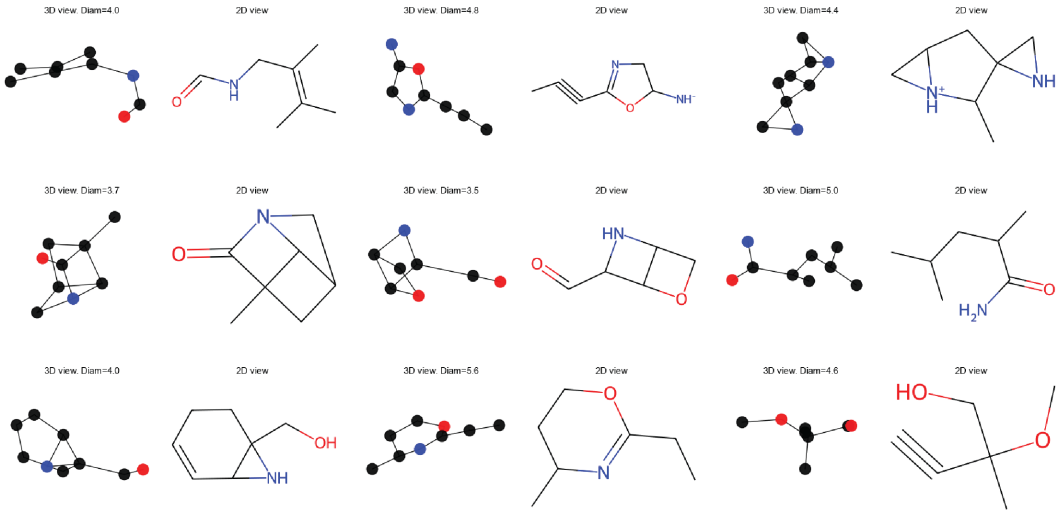


Figure 3: Non-curated samples on QM9 with implicit hydrogens.

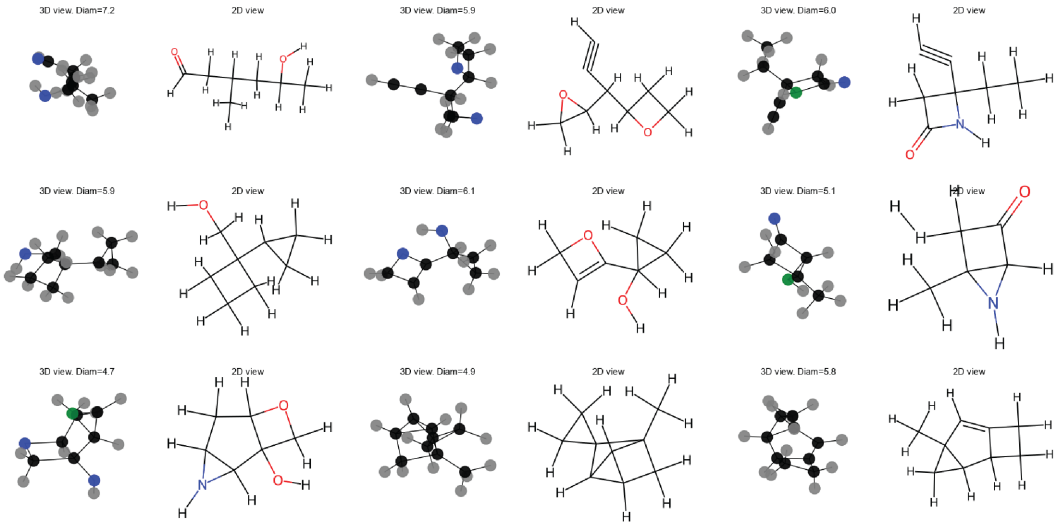


Figure 4: Non-curated samples on QM9 with explicit hydrogens.