MolHIT: Advancing Molecular-Graph Generation with Hierarchical Discrete Diffusion Models

Anonymous authors
Paper under double-blind review

Abstract

Molecular generation with diffusion models has emerged as a promising direction for AI-driven drug discovery and materials science. While graph diffusion models have been widely adopted due to the discrete nature of 2D molecular graphs, existing models suffer from low chemical validity and struggle to meet the desired properties compared to 1D modeling. In this work, we introduce **MolHIT**, a powerful molecular graph generation framework that overcomes long-standing performance limitations in existing methods. MolHIT is based on the Hierarchical Discrete Diffusion Model, which generalizes discrete diffusion to additional categories that encode chemical priors, and decoupled atom encoding that splits the atom types according to their chemical roles. Overall, MolHIT achieves new state-of-the-art performance on the MOSES dataset with near-perfect validity for the first time in graph diffusion, surpassing strong 1D baselines across multiple metrics. We further demonstrate strong performance in downstream tasks, including multi-property guided generation and scaffold extension.

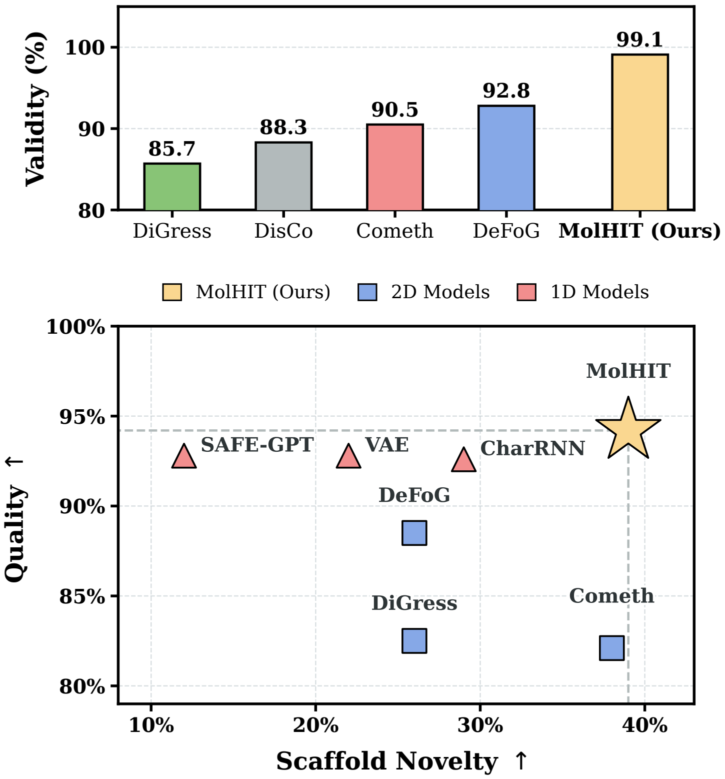

Figure 1: MolHIT achieves SOTA result on MOSES dataset. (Top) Near-perfect validity, outperforming existing graph diffusion models. (Bottom) Pareto-optimal in quality-novelty trade-off.

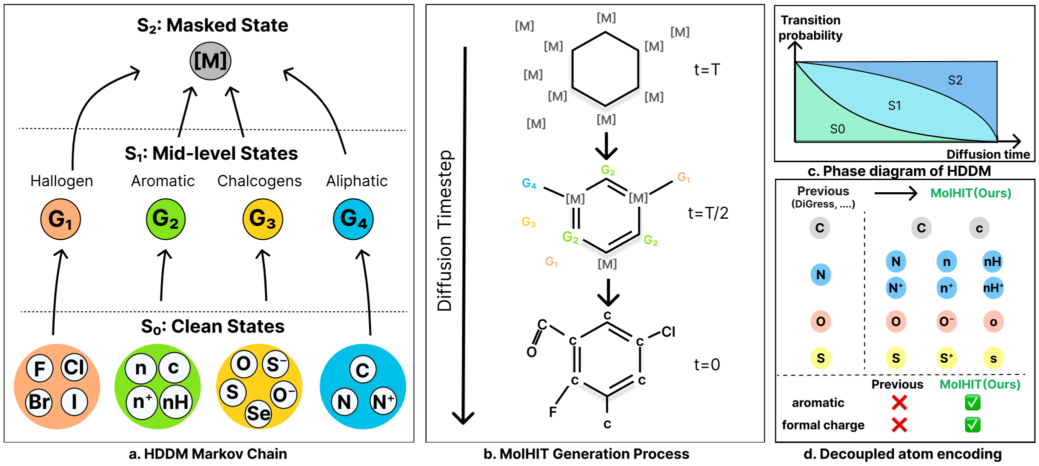

Figure 2: Overview of MolHIT. (a) Markov chain of Hierarchical Discrete Diffusion Model (HDDM). Clean states (S_0) are transited to the mid-level states (S_1) and finally to the masked state (S_2). (b) Generation process of MolHIT. From the masked prior, atoms are denoised into mid-level states and then to atomic tokens in a coarse-to-fine manner. (c) Phase diagram of HDDM showing the transition probability of the forward process. (d) Decoupled atom encoding scheme, separately encoding the aromatic and charged atom types.

1 Introduction

Molecular generation with AI has the potential to significantly speed up materials design (Sanchez-Lengeling and Aspuru-Guzik, 2018) and drug discovery (Zhang et al., 2025). While this promise has led to many different modeling strategies, generating valid and novel molecules is challenging due to the vast combinatorial search space (Dobson, 2004). Here, the primary challenge is not generating novel structures, but ensuring the structures remain chemically valid and relevant. Even a minor atom-level error can produce a structure that is chemically impossible or synthetically inaccessible. Consequently, it is necessary to develop generative models that efficiently explore this immense chemical space while generating valid and synthesizable molecules.

One common approach is to treat molecules as 1D sequences, most commonly through the SMILES representation (Weininger, 1988). By representing molecular graphs into strings, these models can leverage powerful natural language processing techniques to learn patterns of characters. While this simpler learning objective results in generating valid molecules, they suffer from memorization, often reproducing patterns or common subsequences in the training set. This limited exploration capability creates a performance plateau as shown in Fig. 1, where high validity is achieved at the expense of a reduced number of new structures.

To overcome the exploration limits of sequence-based approaches, graph generative models (Jo et al., 2022; Liu et al., 2023) treat molecules as interconnected systems of atoms and bonds. Unlike 1D models that often overfit to specific textual patterns, graph-based architectures are designed to internalize the underlying topological principles of chemical structures, allowing them to generalize beyond the training set and discover novel structures. In particular, discrete diffusion models (Austin et al., 2021) have been widely studied for molecular graph generation as they naturally align with the categorical nature of atoms and bonds (Vignac et al., 2022; Xu et al., 2024; Qin et al., 2024; Seo et al., 2025).

While these models excel at structural exploration, they are prone to generating invalid or chemically unstable samples compared to well-optimized 1D models. This creates a performance gap that raises a fundamental research question: **Can we leverage the inductive biases of graph modeling to match the validity of sequence models while maintaining their superior capacity for structural novelty?**

We identify two critical limitations in existing molecular graph generation with discrete diffusion. **(1)** First, current uniform or absorbing transition treats each atom category as an independent category, even though there is well known chemical relationship that some atoms are easier to be replaced with another. Neglecting well-established domain priors often makes the learning unnecessarily

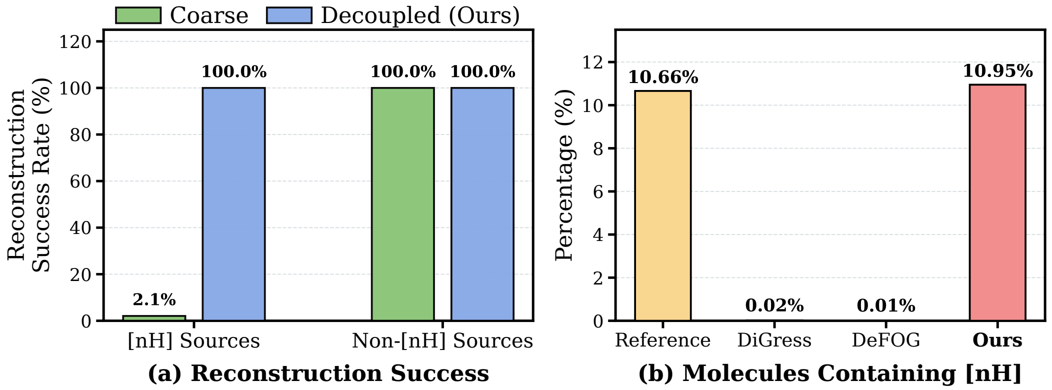

Figure 3: Existing atom encoding for molecular graph is ill-posed. (Left) Reconstruction success rate on the Moses dataset with previous encoding and our decoupled atom encoding. (Right) Proportion of generated molecules containing pyrrolic nitrogen $[nH]$.

hard, especially in molecular settings where high-quality molecule data is scarce. **(2)** Second, existing graph models rely on naive atom encodings, ignoring the fact that a single atom can have different characteristics when it has a formal charge or consists of a ring (aromaticity). We reveal that this makes molecular graph generation tasks ill-posed and unnecessarily challenging, which we demonstrate in the reconstruction experiments in Fig. 3 where previous atom encoding fails.

In light of these observations, we introduce MolHIT, a hierarchical discrete diffusion framework designed to bridge the gap between structural innovation and chemical validity. Our framework is built upon the Hierarchical Discrete Diffusion Model (HDDM), where additional categories are added to represent natural chemical groups into the diffusion process. This coarse-to-fine approach allows the model to establish high-level chemical identities before refining them into specific atom types, thereby capturing the meaningful dependencies of molecular structure that uniform or absorbing kernels often overlook. Furthermore, we introduce Decoupled Atom Encoding (DAE) to resolve the information loss found in naive representations by explicitly split atoms based on their specific chemical roles, such as formal charge and aromaticity. By providing a chemical role into each token, DAE not only resolves the reconstruction problem in original atom encoding, but also reduces the burden of differentiating atom roles solely with the $(O(n^2))$ bond features. Combined together, **MolHIT** reaches a new Pareto frontier in generating novel structures with high quality, surpassing both existing 1D and 2D models (Fig. 1).

We extensively evaluate MolHIT with experiments on large molecular benchmarks, including unconditional generation tasks on MOSES (Polykovskiy et al., 2020) and GuacaMol (Brown et al., 2019) benchmarks and conditional generation tasks, including scaffold extension and multi-property guided generation tasks. Across all benchmarks and tasks, MolHIT shows significant improvements over previous graph diffusion models, resulting in a new state-of-the-art that surpass 1D models.

Our contributions can be summarized as follows:

- We introduce **MolHIT**, a molecular graph diffusion model built upon a novel **Hierarchical Discrete Diffusion Model (HDDM)** framework with a mathematically guaranteed ELBO.
- We identify a critical limitation in the prior graph generative models' atom encoding and propose a simple solution: **Decoupled Atom Encoding (DAE)**. By representing atoms based on their specific chemical roles, we find DAE enhances both the model's generative expressiveness and chemical reliability.
- We achieve the SOTA performance on the MOSES benchmarks in multiple metrics, significantly outperforming both existing graph diffusion models and 1D sequence-level baselines.
- We test our algorithm on practical downstream tasks including multi-property guided generation and scaffold extension, achieving the highest performance compared to the previous graph diffusion approach.

2 PRELIMINARIES

2.1 DISCRETE DIFFUSION MODELS

Given a discrete state space S with K categories, discrete diffusion models define a noising and denoising process within a discrete space. Specifically, for $\mathbf{x} \in S$, the noising process is described by a Markov chain as follows:

$$q(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathrm{Cat}(\mathbf{x}_t; \mathbf{x}_{t-1}Q_t). \tag{1}$$

Here, marginal probability of $\mathbf{x}_t$ in timestep t, given clean data $\mathbf{x}_0$ can be calculated with

$$q(\mathbf{x}_t|\mathbf{x}_0) = \mathrm{Cat}(\mathbf{x}_t; \mathbf{x}_0\bar{Q}_t), \;\; \bar{Q}_t = Q_tQ_{t-1}\cdots Q_1. \tag{2}$$

As shown by Austin et al. (2021), one can design multiple types of diffusion process, where two types of processes are widely used because of the closed-form calculation of the forward process and natural noising process.

Uniform transition Uniform transition assumes uniform prior $p_T(\mathbf{x}_T = c) = \frac{1}{K}$ for all $c \in \{1, \ldots, K\}$. Then one could define a forward noising process by interpolating clean data $\mathbf{x}_0$ with the prior in the following way:

$$q(\mathbf{x}_t|\mathbf{x}_0) = \mathrm{Cat}(\mathbf{x}_t; (1-\bar{\alpha}_t)\frac{1}{K}\mathbf{1}\mathbf{1}^T + \bar{\alpha}_t\mathbf{x}_0), \tag{3}$$

where $\bar{\alpha}_t$ is monotonic decreasing function with $\bar{\alpha}_0] = 1, \bar{\alpha}_T = 0$, which we call diffusion scheduler.

Marginal transition To facilitate the diffusion learning, marginal transition assumes data prior π to be an optimal probability distribution that approximates the empirical data distribution from the training set. This has been primarily adopted for graph diffusion models DiGress (Vignac et al., 2022; Siraudin et al., 2024), where further details are in Appendix C.2.

Absorbing transition Unlike uniform and marginal transition where diffusion process operates on the given K categories, one can introduce an additional masked (absorbing) state $\mathbf{m}$ with prior $\mathbf{e_m}$ being a one-hot vector of the masked state. Then, one can naturally define a diffusion process as an absorbing process in a Markov chain, which results in the following forward form:

$$q(\mathbf{x}_t|\mathbf{x}_0) = \mathrm{Cat}(\mathbf{x}_t; \bar{\alpha}_t\mathbf{x}_0 + (1-\bar{\alpha}_t)\mathbf{e_m}). \tag{4}$$

Given $q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) = \mathrm{Cat}(\mathbf{x}_{t-1}; \frac{\mathbf{x}_t Q_{t|s}^\top \odot \mathbf{x}_0\bar{Q}_{t-1}}{\mathbf{x}_0\bar{Q}_t\mathbf{x}_t^T})$, one could estimate posterior $p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)$ by learning to estimate the clean data $\hat{\mathbf{x}}_0$ given the noisy data $\mathbf{x}_t$. This enables training the diffusion models with simple cross-entropy loss, where the loss function becomes directly linked to the negative evidence lower bound (NELBO) (Austin et al., 2021; Sahoo et al., 2024).

2.2 MOLECULAR GRAPH GENERATION WITH DISCRETE DIFFUSION

Given molecular graph $G = (X, E)$, denote $X \in \mathbb{R}^{n \times d_X}$, $E \in \mathbb{R}^{n \times n \times d_E}$ for the atom matrix and adjacency matrix (bond matrix) where n is the number of atoms and d_X, d_E are feature dimensions of atoms and edges. The forward process of discrete diffusion operates independently on the atom and bond matrices:

$$G_t = (\mathbf{X}_t, \mathbf{E}_t) \;\; : \; \mathbf{X}_t = \mathbf{X}_0\bar{\mathbf{Q}}_{\mathbf{X},t}, \; \mathbf{E}_t = \mathbf{E}_0\bar{\mathbf{Q}}_{\mathbf{E},t}. \tag{5}$$

where, we define $\bar{Q}_{X,t} = Q_{X,t}\cdots Q_{X,1}, \bar{Q}_{E,t} = Q_{E,t}\cdots Q_{E,1}$ are forward transition matrix usually calculated in a closed form for efficiency.

Given a noisy graph G_t, a neural network is trained to estimate a clean graph $G_0 = (X_0, E_0)$ through predicting clean atoms and bonds independently, which in practice results in the following cross-entropy (CE) loss:

$$\mathcal{L}_\theta = \mathbb{E}_{t,\, G_t \sim q(\cdot|G_0)}\left[\sum_{i=1}^{n} -\log p_\theta^X(X_{0,i} \mid G_t, t) + \lambda \sum_{1 \le i < j \le n} -\log p_\theta^E(E_{0,ij} \mid G_t, t)\right], \tag{6}$$

where $\lambda > 0$ is a weighting factor that balances the relative contribution of node and edge loss.

3 MOLHIT FRAMEWORK

3.1 HIERARCHICAL DISCRETE DIFFUSION MODELS

We introduce *Hierarchical Discrete Diffusion Models* (HDDM), which generalize the discrete diffusion framework into a multi-stage setting. Unlike standard discrete diffusion (Austin et al., 2021), where the forward transitions operate either within the clean vocabulary space or toward an absorbing (masked) state, HDDM introduces additional mid-level states that bridge the corruption process.

To design a discrete diffusion in this augmented space, we first show that there exists a simple forward process that admits a tractable closed-form transition kernel. Specifically, for clean state space $\mathcal{S}_0$ with K categories, suppose we add additional $G+1$ categories such that we have an augmented state space $\mathcal{T}$ with cardinality $D = K + G + 1$. As illustrated in Figure 2-(a), we partition $\mathcal{T}$ into three disjoint subsets: $\mathcal{S}_0$, mid-level states $\mathcal{S}_1$ with G categories, and the masked state $\mathcal{S}_2 = \{m\}$.

Now, we define the transition kernel via a row-stochastic matrix $\mathbf{\Phi} \in [0,1]^{K\times G}$, where $\mathbf{\Phi}_{ij}$ represents the probability of mapping any element $i \in \mathcal{S}_0$ to a mid-level element $j \in \mathcal{S}_1$. This operator induces a transition matrix $Q^{(1)} \in [0,1]^{D\times D}$ on the full space $\mathcal{T}$, structured as a block matrix relative to the partition $(\mathcal{S}_0, \mathcal{S}_1, \mathcal{S}_2)$:

$$Q^{(1)} = \begin{bmatrix} \mathbf{0}_{K\times K} & \mathbf{\Phi} & \mathbf{0}_{K\times 1} \\ \mathbf{0}_{G\times K} & \mathbf{I}_G & \mathbf{0}_{G\times 1} \\ \mathbf{0}_{1\times K} & \mathbf{0}_{1\times G} & 1 \end{bmatrix}$$

Here, $\mathbf{I}_G$ is the identity matrix, indicating that states in $\mathcal{S}_1$ are absorbing the clean states in $\mathcal{S}_0$ under $Q^{(1)}$. Similarly, we define the masking operation via the transition matrix $Q^{(2)}$, which maps all states in $\mathcal{S}_0 \cup \mathcal{S}_1$ to the unique absorbing state in $\mathcal{S}_2$:

$$Q^{(2)} = \begin{bmatrix} \mathbf{0}_{(K+G)\times(K+G)} & \mathbf{1}_{(K+G)\times 1} \\ \mathbf{0}_{1\times(K+G)} & 1 \end{bmatrix}$$

These transition matrices form the basis of the HDDM forward process as in the following lemma:

Lemma 3.1. *Define diffusion schedules α_t, β_t to be monotonically decreasing functions satisfying the boundary conditions $\alpha_0 = \beta_0 = 1$ and $\alpha_1 = \beta_1 = 0$, such that $\alpha_t \leq \beta_t$ for all t. We define the forward diffusion process of the hierarchical Markov chain via the transition kernel $Q_{t|s}$ from timestep s to t as:*

$$Q_{t|s} = \alpha_{t|s}\mathrm{I} + (\beta_{t|s} - \alpha_{t|s})Q^{(1)} + (1 - \beta_{t|s})Q^{(2)}, \tag{7}$$

where $\alpha_{t|s} := \alpha_t/\alpha_s, \beta_{t|s} := \beta_t/\beta_s$. Then, the transition kernels satisfy the Chapman–Kolmogorov equation, such that $Q_{t|s}Q_{s|r} = Q_{t|r}$ for any $r < s < t$. Consequently, the cumulative forward transition from the initial state to timestep t is given by:

$$Q_t = \alpha_t\mathrm{I} + (\beta_t - \alpha_t)Q^{(1)} + (1 - \beta_t)Q^{(2)}, \tag{8}$$

where I denotes the identity matrix in $\mathcal{T}$.

Note that the above forward transition operators can be naturally extended to arbitrary hierarchies in state space. We provide a proof of the above lemma with a generalized forward process in Appendix D.1.

Now for training guarantee, one can derive negative ELBO (NELBO), which we prove in Theorem D.2 in Appendix. In practice, one can define $\mathbf{\Phi}$ as a deterministic projection that clusters clean atom categories into meaningful groups. We show in this special case, NEBO of HDDM can be further simplified as in the following.

Theorem 3.2. *If the forward transition kernels Q_t in Eq. 8 is induced from the deterministic projection $\mathbf{\Phi}$, the NELBO of HDDM is given as:*

$$\mathcal{L}^{\infty}_{NELBO}(\theta) = \mathbb{E}_{Q,t} \frac{(-\alpha_s \beta_{t|s} + \alpha_t)}{\beta_t - \alpha_t} \log \langle \mathbf{x}_\theta, \mathbf{x} \rangle \cdot \mathbb{I}[\mathbf{z}_t \in \mathcal{S}_1] + \frac{(1-\beta_{t|s})(\beta_s - \alpha_s)}{1-\beta_t} \log \langle Q^{(1)} \mathbf{x}_\theta, Q^{(1)} \mathbf{x} \rangle \rangle \cdot \mathbb{I}[\mathbf{z}_t = \mathbf{m}] + \frac{\alpha_s(1-\beta_{t|s})}{1-\beta_t} \log \langle \mathbf{x}_\theta, \mathbf{x} \rangle \cdot \mathbb{I}[\mathbf{z}_t = \mathbf{m}] + C, \tag{9}$$

for some constant C and the denoiser $\mathbf{x}_\theta(\mathbf{z}_t, t)$.

We provide a proof in Appendix D.2. For a sanity check, one can observe that Eq. 9 reduces to the NELBO of the original masked diffusion models when $\beta_t = \alpha_t$ (i.e, no $\mathcal{S}_1$). With Theorem 3.2, we can design a simple cross-entropy loss for HDDM training in a principled way. We empirically find that regularization loss in Eq. 9 does not improve the performance, so we take the original loss in Eq. 6.

3.2 Decoupled Atom Encoding

Existing graph diffusion frameworks (Vignac et al., 2022; Xu et al., 2024; Qin et al., 2024) typically rely on a coarse atom encoding scheme, where node identities are determined solely by their atomic numbers. While this simplifies the encoding, we identify that this one-to-many mapping between atomic tokens and their physical states (e.g., protonation or aromaticity) causes the generative task to be ill-posed. As illustrated in Fig. 3 (Left), this leads to a systematic reconstruction failure in molecules requiring fine-grained atomic descriptors, such as specific nitrogen motifs found in drug-like scaffolds. Consequently, models using these coarse encodings suffer from a representational bias, struggling to generate essential motifs that are statistically prevalent in the training distribution (Fig. 3, Right).

To resolve these representational gaps and ensure the model can generalize across diverse chemical spaces, we introduce Decoupled Atom Encoding (DAE). DAE expands atomic state space by explicitly encoding aromaticity and formal charge as primary node attributes. This results in a near-perfect reconstruction ratio both on the MOSES and GuacaMol dataset. Furthermore, by providing the model with necessary structural priors, **MolHIT** successfully recovers the distribution of complex motifs such as pyrrolic nitrogen ([nH]), which baselines using coarse encoding struggle to capture (Fig. 3, Right). Further details are in Appendix E.1.

3.3 Forward and Reverse Process of MolHIT

Forward process of MolHIT Since the atom and bond are perturbed independently throughout the forward process, we decouple their transition dynamics in graph diffusion. This flexibility is particularly advantageous for molecular graph modeling. We empirically observe that a uniform transition kernel is essential for edge generation, whereas HDDM yields superior performance for atom types compared to a uniform approach. Therefore, we employ an HDDM process for atoms and a uniform transition for edges, resulting in the following forward process dynamics:

$$\begin{aligned} Q_{X,t} &= \alpha_{X,t} \mathrm{I} + (\beta_{X,t} - \alpha_{X,t}) Q^{(1)}_{X,t} + (1-\beta_{X,t}) Q^{(2)}_{X,t}, \\ Q_{E,t} &= \alpha_{E,t} \mathrm{I} + (1-\alpha_{E,t}) \mathbf{1}_{d_E} \mathbf{1}^T_{d_E}, \end{aligned} \tag{10}$$

Our preliminary experiments show robustness on the HDDM scheduler $\alpha_{X,t}$, $\beta_{X,t}$, and therefore we simply opt for linear schedule for $\alpha_{X,t} = \alpha_{E,t} = 1 - t$ and $\beta_{X,t} = 1 - t^2$ for the experiments.

Grouping strategy Given the mid-level states $\mathcal{S}_1$, HDDM allows for the design of arbitrary transition kernels from $\mathcal{S}_0$ to $\mathcal{S}_1$. We implement a deterministic grouping kernel that clusters atom elements based on their intrinsic chemical properties and aromaticity. For instance, in the MOSES dataset, we partition 12 atom types into four semantic groups: $\{C\}$, $\{N, O, S\}$, $\{F, Cl, Br\}$, and $\{c, o, n, nH, s\}$. This hierarchical structure simplifies the initial stages of diffusion by focusing on broad chemical categories before refining specific identities. We extend this strategy to other datasets,

Table 1: Comprehensive MOSES benchmark results. Scaffold Novelty (Scaf-Novel) measures the ratio of novel scaffold molecules to the number of generated molecules, while Scaffold Retrieval (Scaf-Ret.) quantifies test scaffold retrievals. All of the results are the averaged value over 3 runs of 25,000 samples. Bold denotes the best in each category, and underline indicates SOTA performance within the 2D Graph models. Empty values are due to the absence of publicly available checkpoints or samples.

Category	Model	Quality ↑	Scaf-Novel ↑	Scaf-Ret. ↑	Valid ↑	Unique ↑	Novel ↑	Filters ↑	FCD ↓	SNN ↑	Scaf ↑
-	Training set	95.4	—	—	100.0	100.0	—	100.0	0.48	0.59	-
1D Sequence	VAE[23]	92.8	0.22	0.031	97.7	99.7	69.5	**99.7**	0.57	0.58	5.9
	CharRNN [39]	92.6	0.29	**0.035**	97.5	99.9	84.2	99.4	**0.52**	0.56	11.0
	SAFE-GPT [31]	92.8	0.12	0.015	**99.8**	98.9	43.7	97.7	0.72	0.57	6.3
	GenMol [25]	62.1	0.05	0.012	99.7	64.0	68.9	98.1	16.4	**0.64**	1.6
2D Graph	DiGress [44]	82.5	0.26	0.031	87.1	**100.0**	94.2	97.5	1.25	0.53	12.8
	DisCo [48]	-	-	-	88.3	100.0	**97.7**	95.6	1.44	0.50	15.1
	Cometh [42]	82.1	0.36	0.023	87.2	100.0	96.4	97.3	1.44	0.51	**16.8**
	DeFoG [35]	88.5	0.26	0.031	92.8	99.9	92.1	98.9	1.95	0.55	14.4
	MolHIT	**94.2**	**0.39**	0.033	99.1	99.8	91.4	98.0	1.03	0.55	14.4

such as GuacaMol, by adapting the groupings to their respective atom vocabularies. Full details of these partitions are provided in Appendix E.2

Project and Noise (PN-sampler) Due to the standard ELBO guarantee as we prove in Theorem 3.2, one can sample from the original posterior update as in prior works (Austin et al., 2021). While standard posterior updates follow the transition $q(G_{t-\Delta t}|G_t, G_0)$ as justified by the ELBO guarantee in Theorem 3.2, we empirically find that this approach often restricts the structural exploration necessary for complex molecular generation. To address this, we design a Project-and-Noise (PN) sampler. PN sampler projects model's denoising prediction $p_\theta(G_0|G_t)$ onto the clean manifold $\mathcal{M}$ (one-hot vector) via categorical sampling to obtain a discrete candidate $\hat{G}_0$. This candidate is then directly re-noised to the preceding timestep $s = t - \Delta t$ using the cumulative transition kernel Q_t, effectively bypassing posterior constraints of G_t to encourage greater diversity in the generated graph. The overall algorithm is illustrated in Alg. 1 in Appendix E.10.

Temperature sampling While temperature and top-p sampling have become standard techniques for managing the quality-diversity trade-off in generative domains (Holtzman et al., 2019; Ficler and Goldberg, 2017; Hashimoto et al., 2019), their application to molecular graph generation remains largely unexplored. We evaluate the impact of these sampling strategies and demonstrate that our PN-sampler effectively controls this trade-off. We empirically find that temperature sampling can be naturally adopted for PN-sampler, where doing temperature sampling only for the atom prediction (line 7 in Alg. 1 in Appendix E.10.) results in the best performance.

3.4 Conditional Modeling

To enable conditional modeling, we train a conditional model by modifying the original graph transformer architecture in DiGress (Vignac et al., 2022) by adding adaptive layer normalization (adaLN) for node attention only. For sampling, we adopt classifier-free guidance (CFG) (Ho and Salimans, 2022). We provide the details in Appendix E.8.

4 Experiments

We evaluate **MolHIT** on two large-scale molecular datasets: MOSES (Polykovskiy et al., 2020) and Guacamol (Brown et al., 2019). The MOSES dataset consists of 1.9M molecules containing 7 heavy atom types, which we augment into 12 tokens using DAE (Sec. 3.2). Similarly, the GuacaMol (Brown et al., 2019) dataset, which originally contains 12 heavy atom types, is decoupled into 56 tokens via DAE. For the model architecture, we utilize the original graph transformer from DiGress Vignac et al. (2022), maintaining the same model size. All reported results represent the average of three independent runs, and standard deviations are provided in Appendix E.3.

4.1 Unconditional Generation on MOSES

Evaluation Following previous graph diffusion works (Vignac et al., 2022; Qin et al., 2024), we measure with official benchmarks for Moses (Polykovskiy et al., 2020) which includes 7 metrics:

Validity (%), Uniqueness (%), Novelty (%), Filters (%), FCD, SNN, Scaf. We also measure Quality Lee et al. (2025), which is defined by the proportion of molecules that are valid, unique, synthetic accessibility (SA (Bickerton et al., 2012) ≤ 4), and drug-like (QED (Ertl and Schuffenhauer, 2009) ≥ 0.6). Formal definitions of the metrics are provided in Appendix E.5.

Scaffold novelty metrics While the standard MOSES benchmark provides a foundation for evaluating molecular generative models, simple metrics like novelty may not reflect the capability for generating new molecules. For instance, a high novelty score itself can come from merely generating novel-looking noise outside the manifold of drug-like molecules, while high uniqueness may not reflect true structural diversity if the model is trapped in a narrow chemical subspace. To address this, we introduce two metrics given n_{total} generated molecules: (1) Scaffold Novelty $= |\mathcal{S}_{\text{gen}} \setminus \mathcal{S}_{\text{train}}|/n_{\text{total}}$, which quantifies the efficiency of structural extrapolation; and (2) Scaffold Retrieval $= |\mathcal{S}_{\text{gen}} \cap \mathcal{S}_{\text{test}}|/n_{\text{total}}$, which measures distributional fidelity. Further details are in Appendix E.6.

Baselines For graph generative models, we compare with DiGress (Vignac et al., 2022), DisCo (Xu et al., 2024), Cometh (Siraudin et al., 2024), DeFoG (Qin et al., 2024), which are previous SOTA in atom-level graph diffusion. We also compare with 1D baselines; VAE (Kingma and Welling, 2013), Char-RNN (Segler et al., 2018), SAFE-GPT (Noutahi et al., 2024), GenMol (Lee et al., 2025).

Result As shown in Table 1, MolHIT significantly outperforms previous graph-based baselines across nearly all key metrics, including Quality, Validity, FCD, and Scaffold Novelty. While 1D sequence-based models (SAFE-GPT, GenMol) excel in Validity, they exhibit a clear tendency toward memorization, evidenced by their lower Scaf-Novelty and novelty scores. On the other hand, MolHIT achieves a new state-of-the-art both for Quality (94.2%) and Scaffold Novelty (0.39) while achieving near perfect validity score (99.1). The above results validate that MolHIT effectively navigates the valid drug-like manifold without sacrificing its ability to explore novel chemical space.

4.2 Unconditional Generation on GuacaMol

Setup Compared to MOSES where molecules contain charged atoms are filtered, GuacaMol benchmark (Brown et al., 2019) contains a broader chemical space, including compounds with formal charges that are not eliminated by neutralization. Previous atom encoding (Vignac et al., 2022) fails to reconstruct these properties (Fig. 4), and they train model only with a manually filtered dataset which are failed to be reconstructed. This helps improve the validity measure, but making models learn from the imperfect, biased distribution. In contrast, we utilize the full GuacaMol dataset for training to evaluate the robustness of our model. We run 3 run of generating 10,000 samples for each experiment.

Results We put the result in Table 12 in Appendix E.7. The result shows that MolHIT achieves the highest performance among all metrics except FCD. For FCD, the strong performance of original DiGress without DAE indicates that using DAE does not always lead to the generative task being easier since it can be hard to model with differentiate extended atom vocabulary. However, as in Appendix E.1, we find that using DAE substantially increase the amount of molecules having charged or special atoms, which is not rare in the GuacaMol. Note that the original DiGress is trained for 1,000 epochs, while our results are from training only with 40 epochs, so further training will improve the metrics.

4.3 Multi-property guided generation

Generating molecules with targeted chemical properties is important for practical applications in materials science and drug discovery. For this, we evaluate the capacity of MolHIT under the multi-conditional generation scenario.

Setup We train a conditional graph transformer (Sec. 3.4) on the MOSES dataset, labeled with four key chemical properties: Quantitative Estimate of Drug-likeness (QED), Synthetic Accessibility (SA), Molecular Weight (MW), and the lipophilicity (logP). We utilize RDKit (Landrum, 2006), an open-source cheminformatics toolkit, for all property labeling and condition evaluation.

Evaluation For inference, we generate 10,000 samples conditioned on target properties of the molecules that are randomly sampled from the test split. We measure Mean Absolute Error (MAE)

Table 2: Scaffold extension results on the MOSES dataset.

Model	Val. (%) ↑	Div. ↑	Hit@1 ↑	Hit@5 ↑
DiGress	50.8	44.8	2.07	6.41
+ DAE	64.8	**58.0**	1.67	6.37
MolHIT	**83.9**	57.4	**3.92**	**9.79**

Table 3: Incremental performance gains on MOSES by integrating components into DiGress.

Method	Quality ↑	FCD ↓	Val. (%) ↑
DiGress	82.5	1.25	87.1
+ DAE	87.6	**0.89**	96.2
+ PN Sampler	92.9	1.65	**99.4**
+ HDDM (**MolHIT**)	**94.2**	1.03	99.1

and the Pearson correlation coefficient (r) for conditioning and validity for the structural fidelity of the samples. We compare MolHIT against two baselines: (1) Marginal transition (effectively a DiGress without a geometric prior) and (2) Marginal transition with a DAE (incorporating decoupled atom encoding into the marginal transition baseline).

Results Table 11 shows that MolHIT significantly outperforms all baselines across every metric. For conditioning precision, MolHIT achieves a macro-averaged MAE of 0.058, a 52.4% reduction compared to the Marginal+DAE baseline. MolHIT also exhibits high reliability, reaching a Pearson r of 0.807 on average, including a near-perfect 0.950 for $\log P$ and 0.804 for QED. The results also show this improved conditioning does not come with the cost of lower validity, where MolHIT achieves validity higher than 95%, outperforming baselines with a large gap. We provide more experimental details of the multi-property guided generation in Appendix E.8.

4.4 Scaffold Extension

Setup We evaluate pretrained unconditional model's generative capability when conditioned on a given substructure. For this, we use Bemis-Murcko scaffold (Bemis and Murcko, 1996) of the molecules in the test split and treat fix this part during the diffusion sampling. For each experiment, we utilize 10,000 unique target scaffolds, generating multiple candidates per target to assess the model's distributional coverage. Specifically, we measure the Hit@1 and Hit@5 ratios, which are the probability that the ground-truth extension is recovered within the top k samples along with standard metrics of validity and diversity. Further experimental setup is provided in Appendix E.9.

Result Table 2 shows that MolHIT significantly outperforming DiGress in all metrics. Interestingly, applying DAE to DiGress improves validity and diversity while reducing in Hit@1, which may due to the extended expressivity of the model. However, DAE results in higher diversity, which results in matched Hit@5 ratio for original DiGress.

4.5 Ablation Studies

Component analysis To show the contribution of each component on MolHIT's performance, we conduct an ablation study by testing on the Moses dataset. The result in Table 3 shows that our atom encoding method (DAE), sampler (PN sampler), and diffusion algorithm (HDDM) all contributes to get to the highest value of Quality, FCD, and Validity among graph diffusion models.

Effect of temperature sampling We put the ablation on temperature sampling in Appendix E.10.

5 Conclusion

In this work, we present MolHIT, a novel molecular diffusion model with a hierarchical discrete diffusion framework. Our algorithm results in state-of-the-art performance in large molecular datasets. It unlocks new capacity for end-to-end atom-level molecular generation, directly generating atoms with formal charges or explicit nH for the first time, moving us towards more realistic molecule generation.

Impact Statement

This paper presents work whose goal is to advance the field of molecule generation. We hope our work accelerates the discovery of useful drugs and materials, improving human lives. However, one might maliciously use our model to generate harmful substances to humans and environments.

References

Jacob Austin, Daniel D Johnson, Jonathan Ho, Daniel Tarlow, and Rianne Van Den Berg. Structured denoising diffusion models in discrete state-spaces. *Advances in neural information processing systems*, 34:17981–17993, 2021.

Guy W Bemis and Mark A Murcko. The properties of known drugs. 1. molecular frameworks. *Journal of medicinal chemistry*, 39(15):2887–2893, 1996.

G Richard Bickerton, Gaia V Paolini, Jérémy Besnard, Sorel Muresan, and Andrew L Hopkins. Quantifying the chemical beauty of drugs. *Nature chemistry*, 4(2):90–98, 2012.

Tiwei Bie, Maosong Cao, Kun Chen, Lun Du, Mingliang Gong, Zhuochen Gong, Yanmei Gu, Jiaqi Hu, Zenan Huang, Zhenzhong Lan, et al. Llada2. 0: Scaling up diffusion language models to 100b. *arXiv preprint arXiv:2512.15745*, 2025.

Nathan Brown, Marco Fiscato, Marwin HS Segler, and Alain C Vaucher. Guacamol: benchmarking models for de novo molecular design. *Journal of chemical information and modeling*, 59(3): 1096–1108, 2019.

Andrew Campbell, Joe Benton, Valentin De Bortoli, Thomas Rainforth, George Deligiannidis, and Arnaud Doucet. A continuous time framework for discrete denoising models. *Advances in Neural Information Processing Systems*, 35:28266–28279, 2022.

Huiwen Chang, Han Zhang, Lu Jiang, Ce Liu, and William T Freeman. Maskgit: Masked generative image transformer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11315–11325, 2022.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pages 4171–4186, 2019.

Christopher M Dobson. Chemical space and biology. *Nature*, 432(7019), 2004.

Rick Durrett. *Probability: theory and examples*, volume 49. Cambridge university press, 2019.

Peter Ertl and Ansgar Schuffenhauer. Estimation of synthetic accessibility score of drug-like molecules based on molecular complexity and fragment contributions. *Journal of cheminformatics*, 1(1):8, 2009.

Jessica Ficler and Yoav Goldberg. Controlling linguistic style aspects in neural language generation. *arXiv preprint arXiv:1707.02633*, 2017.

Nate Gruver, Samuel Stanton, Nathan Frey, Tim GJ Rudner, Isidro Hotzel, Julien Lafrance-Vanasse, Arvind Rajpal, Kyunghyun Cho, and Andrew G Wilson. Protein design with guided discrete diffusion. *Advances in neural information processing systems*, 36:12489–12517, 2023.

Tatsunori B Hashimoto, Hugh Zhang, and Percy Liang. Unifying human and statistical evaluation for natural language generation. *arXiv preprint arXiv:1904.02792*, 2019.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. The curious case of neural text degeneration. *arXiv preprint arXiv:1904.09751*, 2019.

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*, 34:12454–12465, 2021.

Emiel Hoogeboom, Vıctor Garcia Satorras, Clément Vignac, and Max Welling. Equivariant diffusion for molecule generation in 3d. In *International conference on machine learning*, pages 8867–8887. PMLR, 2022.

John J Irwin and Brian K Shoichet. Zinc- a free database of commercially available compounds for virtual screening. *Journal of chemical information and modeling*, 45(1):177–182, 2005.

Jaehyeong Jo, Seul Lee, and Sung Ju Hwang. Score-based generative modeling of graphs via the system of stochastic differential equations. In *International conference on machine learning*, pages 10362–10383. PMLR, 2022.

Seo Hyun Kim, Sunwoo Hong, Hojung Jung, Youngrok Park, and Se-Young Yun. Klass: Kl-guided fast inference in masked diffusion models. *arXiv preprint arXiv:2511.05664*, 2025.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Greg Landrum. RDKit: Open-source cheminformatics. `https://www.rdkit.org`, 2006.

Seul Lee, Karsten Kreis, Srimukh Prasad Veccham, Meng Liu, Danny Reidenbach, Yuxing Peng, Saee Paliwal, Weili Nie, and Arash Vahdat. Genmol: A drug discovery generalist with discrete diffusion. *arXiv preprint arXiv:2501.06158*, 2025.

Chengyi Liu, Wenqi Fan, Yunqing Liu, Jiatong Li, Hang Li, Hui Liu, Jiliang Tang, and Qing Li. Generative diffusion models on graphs: Methods and applications. *arXiv preprint arXiv:2302.02591*, 2023.

Gang Liu, Jiaxin Xu, Tengfei Luo, and Meng Jiang. Graph diffusion transformers for multi-conditional molecular generation. *Advances in Neural Information Processing Systems*, 37:8065–8092, 2024.

Aaron Lou, Chenlin Meng, and Stefano Ermon. Discrete diffusion modeling by estimating the ratios of the data distribution. *arXiv preprint arXiv:2310.16834*, 2023.

David Mendez, Anna Gaulton, A Patrícia Bento, Jon Chambers, Marleen De Veij, Eloy Félix, María Paula Magariños, Juan F Mosquera, Prudence Mutowo, Michał Nowotka, et al. Chembl: towards direct deposition of bioassay data. *Nucleic acids research*, 47(D1):D930–D940, 2019.

Shen Nie, Fengqi Zhu, Chao Du, Tianyu Pang, Qian Liu, Guangtao Zeng, Min Lin, and Chongxuan Li. Scaling up masked diffusion models on text. *arXiv preprint arXiv:2410.18514*, 2024.

Emmanuel Noutahi, Cristian Gabellini, Michael Craig, Jonathan SC Lim, and Prudencio Tossou. Gotta be safe: a new framework for molecular design. *Digital Discovery*, 3(4):796–804, 2024.

Jingyang Ou, Shen Nie, Kaiwen Xue, Fengqi Zhu, Jiacheng Sun, Zhenguo Li, and Chongxuan Li. Your absorbing discrete diffusion secretly models the conditional distributions of clean data. *arXiv preprint arXiv:2406.03736*, 2024.

Daniil Polykovskiy, Alexander Zhebrak, Benjamin Sanchez-Lengeling, Sergey Golovanov, Oktai Tatanov, Stanislav Belyaev, Rauf Kurbanov, Aleksey Artamonov, Vladimir Aladinskiy, Mark Veselov, et al. Molecular sets (moses): a benchmarking platform for molecular generation models. *Frontiers in pharmacology*, 11:565644, 2020.

Kristina Preuer, Philipp Renz, Thomas Unterthiner, Sepp Hochreiter, and Gunter Klambauer. Fréchet chemnet distance: a metric for generative models for molecules in drug discovery. *Journal of chemical information and modeling*, 58(9):1736–1741, 2018.

Yiming Qin, Manuel Madeira, Dorina Thanou, and Pascal Frossard. Defog: Discrete flow matching for graph generation. *arXiv preprint arXiv:2410.04263*, 2024.

Subham Sahoo, Marianne Arriola, Yair Schiff, Aaron Gokaslan, Edgar Marroquin, Justin Chiu, Alexander Rush, and Volodymyr Kuleshov. Simple and effective masked diffusion language models. *Advances in Neural Information Processing Systems*, 37:130136–130184, 2024.

Benjamin Sanchez-Lengeling and Alán Aspuru-Guzik. Inverse molecular design using machine learning: Generative models for matter engineering. *Science*, 361(6400):360–365, 2018.

Yair Schiff, Subham Sekhar Sahoo, Hao Phung, Guanghan Wang, Sam Boshar, Hugo Dalla-torre, Bernardo P de Almeida, Alexander Rush, Thomas Pierrot, and Volodymyr Kuleshov. Simple guidance mechanisms for discrete diffusion models. *arXiv preprint arXiv:2412.10193*, 2024.

Marwin HS Segler, Thierry Kogej, Christian Tyrchan, and Mark P Waller. Generating focused molecule libraries for drug discovery with recurrent neural networks. *ACS central science*, 4(1): 120–131, 2018.

Hyunjin Seo, Taewon Kim, Sihyun Yu, and SungSoo Ahn. Learning flexible forward trajectories for masked molecular diffusion. *arXiv preprint arXiv:2505.16790*, 2025.

Jiaxin Shi, Kehang Han, Zhe Wang, Arnaud Doucet, and Michalis Titsias. Simplified and generalized masked diffusion for discrete data. *Advances in neural information processing systems*, 37: 103131–103167, 2024.

Antoine Siraudin, Fragkiskos D Malliaros, and Christopher Morris. Cometh: A continuous-time discrete-state graph diffusion model. *arXiv preprint arXiv:2406.06449*, 2024.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

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.

David Weininger. Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules. *Journal of chemical information and computer sciences*, 28(1):31–36, 1988.

Chengyue Wu, Hao Zhang, Shuchen Xue, Zhijian Liu, Shizhe Diao, Ligeng Zhu, Ping Luo, Song Han, and Enze Xie. Fast-dllm: Training-free acceleration of diffusion llm by enabling kv cache and parallel decoding. *arXiv preprint arXiv:2505.22618*, 2025.

Minkai Xu, Alexander S Powers, Ron O Dror, Stefano Ermon, and Jure Leskovec. Geometric latent diffusion models for 3d molecule generation. In *International Conference on Machine Learning*, pages 38592–38610. PMLR, 2023.

Zhe Xu, Ruizhong Qiu, Yuzhong Chen, Huiyuan Chen, Xiran Fan, Menghai Pan, Zhichen Zeng, Mahashweta Das, and Hanghang Tong. Discrete-state continuous-time diffusion for graph generation. *Advances in Neural Information Processing Systems*, 37:79704–79740, 2024.

Ling Yang, Ye Tian, Bowen Li, Xinchen Zhang, Ke Shen, Yunhai Tong, and Mengdi Wang. Mmada: Multimodal large diffusion language models. *arXiv preprint arXiv:2505.15809*, 2025.

Kang Zhang, Xin Yang, Yifei Wang, Yunfang Yu, Niu Huang, Gen Li, Xiaokun Li, Joseph C Wu, and Shengyong Yang. Artificial intelligence in drug development. *Nature medicine*, 31(1):45–59, 2025.

Cai Zhou, Chenyu Wang, Dinghuai Zhang, Shangyuan Tong, Yifei Wang, Stephen Bates, and Tommi Jaakkola. Next semantic scale prediction via hierarchical diffusion language models. *arXiv preprint arXiv:2510.08632*, 2025.

TABLE OF CONTENTS

A Limitation and future directions

Limitations While our models improve the traditional diffusion based molecular generation, we have not tested with the model size increase or architectural improvement, in which we believe have further room for better performance. Moreover, we have not fully trained the model until performance saturation on GuacaMol dataset and we believe performance improvement with further training.

Future directions There are many interesting future directions. One is to apply Hierarchical Discrete Diffusion Models to the language domains (Sahoo et al., 2024), or image models (Chang et al., 2022) and combining with different sampling schemes (Wu et al., 2025; Kim et al., 2025). Another direction is to further improve MolHIT's framework with more advanced tokenization incorporating motifs or functional groups and apply into the 3D molecular generation (Hoogeboom et al., 2022; Xu et al., 2023) and proteins (Gruver et al., 2023).

B Related Works

Discrete diffusion models Along with the success of continuous diffusion models (Ho et al., 2020; Song et al., 2020), discrete diffusion models formulate a noise process within a discrete state space. Hoogeboom et al. (2021) investigate uniform transition of the discrete diffusion models, while D3PM Austin et al. (2021) explore different types of transition mechanism which include absorbing transition. Recently and independently developed alongside our work, Zhou et al. (2025) propose a hierarchical discrete diffusion approach to language modeling. While similar in spirit, our HDDM is derived from a semigroup-consistent family of closed-form transition kernels $Q^{(1)}, Q^{(2)}$ parameterized by explicit diffusion scheduler α_t, β_t while Zhou et al. (2025) is developed in the CTMC framework (Campbell et al., 2022). Moreover, HDDM supports an arbitrary row-stochastic projection $\mathbf{\Phi}$, which generalizes the deterministic hierarchical mapping used in Zhou et al. (2025).

Diffusion models for molecular generation Various diffusion models and its techniques have been applied for molecular graph generation. GDSS (Jo et al., 2022) formulate continuous diffusion modeling through the system of SDE with a score matching objective. DiGress (Vignac et al., 2022) utilize primary form of discrete diffusion models with uniform-style transition with data dependent prior. Siraudin et al. (2024); Xu et al. (2024); Qin et al. (2024) apply CTMC framework as in Campbell et al. (2022) to further boost the performance. Another axis for molecular generation is to model 1D sequence. SAFE-GPT (Noutahi et al., 2024) trains an Autoregressive model with their unique representation of molecule while GenMol (Lee et al., 2025) adopts masked diffusion framework in a wide range of drug discovery tasks. We defer further related works in Appendix C.1.

C Further background

C.1 Further related works

Further backgrounds on discrete diffusion models Recently, Masked Language Models are actively studied due to its simple form and potential of bi-directional modeling (Devlin et al., 2019). Campbell et al. (2022) formulate the discrete diffusion using Continuous Time Markov Chain (CTMC) framework and propose correction sampler leveraging tau-leaping. SEDD (Lou et al., 2023) introduce score entropy loss in analogous to the score matching loss in the continuous diffusion models and show scalability in language modeling. Recently, masked diffusion models is further simplified (Sahoo et al., 2024; Ou et al., 2024; Shi et al., 2024) and reaching to the level that is comparable to the standard AR modeling in large scale (Nie et al., 2024; Bie et al., 2025) and even in multi-modal setting (Yang et al., 2025).

Conditional generation with discrete diffusion For conditional generation, Liu et al. (2024) propose graph diffusion transformer for multi-conditional generation on polymer dataset and shows the effectiveness of the classifier-free guidance. Schiff et al. (2024) propose simple mechanism for conditional sampling which is in analogous with CFG in continuous diffusion.

C.2 Details of masked diffusion models

Marginal transition Let $X = (X_1, \ldots, X_N)$ be a discrete random vector with $X_k \in \{1, \ldots, K\}$, and let P denote the empirical data distribution on $\{1, \ldots, K\}^N$. Define the family of product of distributions:

$$\mathcal{C} \;=\; \left\{ q : \; q(x) = \prod_{k=1}^{N} q_k(x_k), \;\; q_k \in \Delta^{K-1} \right\}, \tag{11}$$

where Δ^{K-1} is the $(K-1)$-simplex.

Then, one can define the marginal terminal distribution π as the product of the data marginals:

$$\pi(x) \;=\; \prod_{k=1}^{N} \pi_k(x_k), \qquad \pi_k(a) \;=\; \mathbb{P}_{X\sim P}[X_k = a]\,, \;\; a \in \{1, \ldots, K\}. \tag{12}$$

Equivalently, π is the (unique) KL projection of P onto $\mathcal{C}$:

$$\pi \;=\; \arg\min_{q \in \mathcal{C}} \mathrm{KL}(P \,\|\, q)\,. \tag{13}$$

Then, marginal transition defines forward process of discrete diffusion as follows:

$$q(\mathbf{x}_t | \mathbf{x}_0) = \mathrm{Cat}(\mathbf{x}_t; \bar{\alpha}_t \mathbf{x}_0 + (1 - \bar{\alpha}_t)\pi). \tag{14}$$

D Mathematical derivations

D.1 Generalized HDDM forward process

We now derive a generalized forward process that incorporates arbitrary multi-level hierarchies. Let $\mathcal{T}$ be the total discrete state space with dimension $D = \sum_{k=0}^{n} K_k$. We partition $\mathcal{T}$ into $n+1$ disjoint subsets $\mathcal{S}_0, \mathcal{S}_1, \ldots, \mathcal{S}_n$, where $\mathcal{S}_0$ represents the clean atomic states (with $|\mathcal{S}_0| = K_0$) and $\mathcal{S}_k$ represents the k-th level of intermediate hierarchical states (with $|\mathcal{S}_k| = K_k$). We further define the cumulative subspace up to level i as $\mathcal{T}_i := \bigcup_{k=0}^{i} \mathcal{S}_k$. For each hierarchical stage $i \in \{1, \ldots, n\}$, we define the transition kernel as a row-stochastic matrix $\mathbf{\Phi}_i \in [0,1]^{|\mathcal{T}_{i-1}| \times K_i}$. This kernel encodes the probabilistic mapping from the cumulative lower-level states in $\mathcal{T}_{i-1}$ to the specific higher-level states in $\mathcal{S}_i$. To characterize the evolution in full space $\mathcal{T}$, we induce a global transition matrix $Q^{(i)} \in [0,1]^{D \times D}$ on $\mathcal{T}$ which embeds the local kernel $\mathbf{\Phi}_i$ into the full space as follows:

$$Q^{(i)}(\mathbf{x}_{\text{next}} \mid \mathbf{x}) = \begin{cases} \mathbf{\Phi}_i(\mathbf{x}_{\text{next}} \mid \mathbf{x}) & \text{if } \mathbf{x} \in \mathcal{T}_{i-1} \text{ and } \mathbf{x}_{\text{next}} \in \mathcal{S}_i, \\ 1 & \text{if } \mathbf{x} \in \mathcal{T} \setminus \mathcal{T}_{i-1} \text{ and } \mathbf{x}_{\text{next}} = \mathbf{x}, \\ 0 & \text{otherwise.} \end{cases} \tag{15}$$

In matrix notation, $Q^{(i)}$ forms a block structure where the transitions from $\mathcal{T}_{i-1}$ are governed by $\mathbf{\Phi}_i$, while the remaining diagonal blocks form an identity matrix. Under this formulation, each $Q^{(i)}$ represents the probabilistic projection onto the i-th hierarchical level, enabling us to design the diffusion forward process with the following lemma:

Proposition D.1. *Suppose monotonically decreasing functions $\alpha_t^{(i)} := \alpha_i(t)$ $(i = 1, 2, ..., n)$ defined in $0 \leq t \leq 1$ are satisfying $0 \leq \alpha_t^{(1)} \leq \cdots \leq \alpha_t^{(n)} \leq 1$ and boundary conditions $\alpha_0^{(i)} = 1, \alpha_1^{(i)} = 0$ for all i. We define the transition matrix from timestep s to timestep t $(s \leq t)$ as:*

$$Q_{t|s} := \alpha_{t|s}^{(1)} I + (\alpha_{t|s}^{(2)} - \alpha_{t|s}^{(1)}) Q^{(0)} + \cdots + (1 - \alpha_{t|s}^{(n)}) Q^{(n)}, \tag{16}$$

where $\alpha_{t|s}^{(i)} = \frac{\alpha_t^{(i)}}{\alpha_s^{(i)}}$ *for every* i*. Then, transition kernel defined by Eq. 16 satisfies Chapman–Kolmogorov consistency (Durrett, 2019) as follows:*

$$Q_{t|s}Q_{s|r} = Q_{t|r} \;\; \forall 0 \leq r \leq s \leq t \leq 1. \tag{17}$$

Moreover, one could represent cumulative forward transition from initial timestep 0 *to* t *in the following form:*

$$Q_t = \alpha_t^{(1)}\mathrm{I} + (\alpha_t^{(2)} - \alpha_t^{(1)})Q^{(1)} + \cdots (1 - \alpha_t^{(n)})Q^{(n)}. \tag{18}$$

Proof. First, we can observe $Q^{(i)}$ is a projection operator; i.e, ${Q^{(i)}}^2 = Q^{(i)}$ for all i by definition. In fact, this can be generalized as $Q^{(i)}Q^{(j)} = Q^{max(i,j)}$ for any $1 \leq i, j \leq n$ by the definition of the $Q^{(i)}$ and φ_i. Now, suppose Eq. 16 holds for some $j \in \mathbb{N}$. Then, one can observe:

$$\begin{aligned}
&Q_{t|s}Q_{s|r} \\
&= \left(\alpha_{t|s}^{(1)} I + (\alpha_{t|s}^{(2)} - \alpha_{t|s}^{(1)})Q^{(0)} + \cdots + (1 - \alpha_{t|s}^{(j+1)})Q^{(j+1)}\right)\left(\alpha_{s|r}^{(1)} I + (\alpha_{s|r}^{(2)} - \alpha_{s|r}^{(1)})Q^{(0)} + \cdots + (1 - \alpha_{s|r}^{(j+1)})Q^{(j+1)}\right) \\
&= \left(\alpha_{t|s}^{(1)} I + (\alpha_{t|s}^{(2)} - \alpha_{t|s}^{(1)})Q^{(0)} + \cdots + (\alpha_{t|s}^{(j+1)} - \alpha_{t|s}^{(j)})Q^{(j)}\right)\left(\alpha_{s|r}^{(1)} I + (\alpha_{s|r}^{(2)} - \alpha_{s|r}^{(1)})Q^{(0)} + \cdots + (\alpha_{s|r}^{(j+1)} - \alpha_{s|r}^{(j)})Q^{(j)}\right) \\
&+ \left(\alpha_{t|s}^{(1)} + \cdots + (1 - \alpha_{t|s}^{(j+1)})\right)\left(1 - \alpha_{s|r}^{(j+1)}\right)Q^{(j+1)} + \left(1 - \alpha_{t|s}^{(j+1)}\right)\left(\alpha_{s|r} + \cdots (1 - \alpha_{s|r}^{(j+1)})\right)Q^{(j+1)} \\
&+ \left(1 - \alpha_{t|s}^{(j+1)})(1 - \alpha_{s|r}^{(j+1)})\right)(Q^{(j+1)})^2 \\
&= \alpha_{t|s}^{(j+1)}\left(\frac{\alpha_{t|s}^{(1)}}{\alpha_{t|s}^{(j+1)}} I + \cdots + \frac{(\alpha_{t|s}^{(j+1)} - \alpha_{t|s}^{(j)})}{\alpha_{t|s}^{(j+1)}}Q^{(j)}\right) \cdot \alpha_{s|r}^{(j+1)}\left(\frac{\alpha_{s|r}^{(1)}}{\alpha_{s|r}^{(j+1)}} I + \cdots + \frac{(\alpha_{s|r}^{(j+1)} - \alpha_{s|r}^{(j)})}{\alpha_{s|r}^{(j+1)}}Q^{(j)}\right) \\
&+ \left(\alpha_{t|s}^{(1)} + \cdots + (1 - \alpha_{t|s}^{(j+1)})\right)\left(1 - \alpha_{s|r}^{(j+1)}\right)Q^{(j+1)} + \left(1 - \alpha_{t|s}^{(j+1)}\right)\left(\alpha_{s|r} + \cdots (1 - \alpha_{s|r}^{(j+1)})\right)Q^{(j+1)} \\
&+ \left(1 - \alpha_{t|s}^{(j+1)})(1 - \alpha_{s|r}^{(j+1)})\right)(Q^{(j+1)})^2 \\
&= \left(\alpha_{t|r}^{(1)} I + (\alpha_{t|r}^{(2)} - \alpha_{t|r}^{(1)})Q^{(0)} + \cdots + (1 - \alpha_{t|r}^{(j)})Q^{(j)}\right) + \left(\alpha_{t|s}^{(1)} + \cdots + (1 - \alpha_{t|s}^{(j+1)})\right)\left(1 - \alpha_{s|r}^{(j+1)}\right)Q^{(j+1)} \\
&+ \left(1 - \alpha_{t|s}^{(j+1)}\right)\left(\alpha_{s|r} + \cdots (1 - \alpha_{s|r}^{(j+1)})\right)Q^{(j+1)} + \left(1 - \alpha_{t|s}^{(j+1)})(1 - \alpha_{s|r}^{(j+1)})\right)(Q^{(j+1)})^2 \\
&= \alpha_{t|r}^{(1)} I + (\alpha_{t|r}^{(2)} - \alpha_{t|r}^{(1)}) + \cdots + (1 - \alpha_{t|r}^{(j+1)})Q^{(j+1)},
\end{aligned} \tag{19}$$

where the second to the last equation comes from the inductive assumption on j. Since $j = 1$ case is trivial, the result follows by mathematical induction on j. □

Proof of Lemma 3.1 Lemma 3.1 is now just a special case of above generalized formula in Proposition D.1.

D.2 Proof of Theorem 3.2

Let $\mathbf{m}$ be the one-hot representation of the masked state which we take as a prior, and let $\mathbf{x} \in S_0$ denote an element in a clean state. For a forward transition kernel $Q_{t|s}$ defined induced by the row stochastic matrix $\mathbf{\Phi} \in [0, 1]^{K \times G}$ and the masking operation as in Lemma 3.1, the conditional transition is defined as $q(\mathbf{z}_t|\mathbf{z}_s) = \mathrm{Cat}(\mathbf{z}_t; \mathbf{z}_s Q_{t|s})$. By applying the closed-form transition from Lemma 3.1 and Bayes' rule, the posterior distribution $q(\mathbf{z}_s|\mathbf{z}_t, \mathbf{x})$ can be derived as follows:

$$q(\mathbf{z}_s|\mathbf{z}_t, \mathbf{x}) = \mathrm{Cat}\left(\mathbf{z}_s; \frac{\left[\alpha_{t|s}\mathrm{I} + (\beta_{t|s} - \alpha_{t|s})Q^{(1)} + (1 - \beta_{t|s})Q^{(2)}\right]^{\mathsf{T}}\mathbf{z}_t \;\odot\; \left[\alpha_s\mathbf{x} + (\beta_s - \alpha_s)Q^{(1)}\mathbf{x} + (1 - \beta_s)\mathbf{m}\right]}{\mathbf{z}_t^T[\alpha_t\mathrm{I} + (\beta_t - \alpha_t)Q^{(1)} + (1 - \beta_t)Q^{(2)}]^T\mathbf{x}}\right) \tag{20}$$

We can divide into the following 3 cases depending on which state $\mathbf{z}_t$ belongs to.

Case 1. $\mathbf{z}_t \in S_0$

$$q(\mathbf{z}_s|\mathbf{z}_t, \mathbf{x}) = \mathrm{Cat}(\mathbf{z}_s; \mathbf{x}). \tag{21}$$

Case 2. $\mathbf{z}_t \in S_1$ When the observed state at time t belongs to the mid-level space S_1, the posterior depends on the state of $\mathbf{z}_t$ under the stochastic transition. Using the block structure of the transition kernels, we can obtain:

$$q(\mathbf{z}_s|\mathbf{z}_t, \mathbf{x}) = \mathrm{Cat}\left(\mathbf{z}_s; \frac{(\alpha_s \beta_{t|s} - \alpha_t)\mathbf{x} + (\beta_t - \beta_{t|s}\alpha_s)\mathbf{z}_t}{\beta_t - \alpha_t}\right), \tag{22}$$

Case 3. $\mathbf{z}_t \in S_2 = \{\mathbf{m}\}$ When the observed state is the mask token $\mathbf{m} \in S_2$, the posterior distribution $q(\mathbf{z}_s|\mathbf{z}_t, \mathbf{x})$ becomes a weighted combination of the clean state, its stochastic projection, and the mask prior. Using the normalization constant $1 - \beta_t$, the posterior is given by:

$$q(\mathbf{z}_s|\mathbf{z}_t, \mathbf{x}) = \mathrm{Cat}\left(\mathbf{z}_s; \frac{\alpha_s(1 - \beta_{t|s})\mathbf{x} + (1 - \beta_{t|s})(\beta_s - \alpha_s)Q^{(1)}\mathbf{x} + (1 - \beta_s)\mathbf{m}}{1 - \beta_t}\right). \tag{23}$$

Parameterization Inspired by the masked diffusion literature (Sahoo et al., 2024; Shi et al., 2024; Ou et al., 2024), we derive simplified loss form through the parameterizing a neural network θ to estimate only the probability in clean final state in $\mathcal{S}_0$. This leads to the posterior $p_\theta(\mathbf{z}_s|\mathbf{z}_t)$ in following closed forms depending on the current state $\mathbf{z}_t$.

Case 1. $\mathbf{z}_t \in S_0$

$$p_\theta(\mathbf{z}_s|\mathbf{z}_t) = \mathrm{Cat}(\mathbf{z_s}; \mathbf{z_t}). \tag{24}$$

Case 2. $\mathbf{z}_t \in S_1$

$$p_\theta(\mathbf{z}_s|\mathbf{z}_t) = \mathrm{Cat}\left(\mathbf{z_s}; \frac{(\alpha_s \beta_{t|s} - \alpha_t){Q^{(1)}}^\intercal \mathbf{z}_t \odot \mathbf{x}_\theta(\mathbf{z}_t, t) + (\beta_t - \beta_{t|s}\alpha_s)\mathbf{z}_t \odot Q^{(1)}\mathbf{x}_\theta}{(\beta_t - \alpha_t)\mathbf{z}_t^T Q^{(1)}\mathbf{x}_\theta}\right). \tag{25}$$

Case 3. $\mathbf{z}_t \in S_2 = \{\mathbf{m}\}$

$$\begin{aligned}
&p_\theta(\mathbf{z}_s|\mathbf{z}_t)\\
&= \mathrm{Cat}\left(\mathbf{z_s}; \frac{[\alpha_{t|s}\mathbf{m} + (\beta_{t|s} - \alpha_{t|s}){Q^{(1)}}^\intercal\mathbf{m} + (1 - \beta_{t|s}){Q^{(2)}}^\intercal\mathbf{m}] \odot [\alpha_s \mathbf{x}_\theta + (\beta_s - \alpha_s)Q^{(1)}\mathbf{x}_\theta + (1 - \beta_s)\mathbf{m}]}{1 - \beta_t}\right).\\
&= \mathrm{Cat}\left(\mathbf{z_s}; \frac{\alpha_s(1 - \beta_{t|s})\mathbf{x}_\theta + (1 - \beta_{t|s})(\beta_s - \alpha_s)Q^{(1)}\mathbf{x}_\theta + (1 - \beta_s)\mathbf{m}}{1 - \beta_t}\right).
\end{aligned} \tag{26}$$

ELBO analysis Now, we start with analyzing the ELBO of the Hierarchical Discrete diffusion models in discrete timesteps.

$$L_T = \mathbb{E}_{t \in \{\frac{1}{T}, \frac{2}{T}, \ldots, 1\}} \mathbb{E}_{q(\mathbf{z}_t|\mathbf{x})}\left[T \cdot \mathrm{D}_{\mathrm{KL}}\left(q(\mathbf{z}_s|\mathbf{z}_t, \mathbf{x}) \,\|\, p_\theta(\mathbf{z}_s|\mathbf{z}_t)\right)\right] \tag{27}$$

Case 1. $\mathbf{z}_t \in S_0$

$$\mathrm{D}_{\mathrm{KL}}\left(q(\mathbf{z}_s|\mathbf{z}_t, \mathbf{x}) \,\|\, p_\theta(\mathbf{z}_s|\mathbf{z}_t)\right) = 0. \tag{28}$$

Case 2. $\mathbf{z}_t \in S_1$

$$\begin{aligned}
&\mathrm{D}_{\mathrm{KL}}\left(q(\mathbf{z}_s|\mathbf{z}_t, \mathbf{x}) \,\|\, p_\theta(\mathbf{z}_s|\mathbf{z}_t)\right)\\
&= \sum_{\mathbf{z}_s \in \{\mathbf{x}, \varphi(\mathbf{x})\}} q(\mathbf{z}_s|\mathbf{z}_t, \mathbf{x}) \log \frac{q(\mathbf{z}_s|\mathbf{z}_t, \mathbf{x})}{p_\theta(\mathbf{z}_s|\mathbf{z}_t)}\\
&= q(\mathbf{z}_s = \mathbf{x}|\mathbf{z}_t, \mathbf{x}) \log \frac{q(\mathbf{z}_s = \mathbf{x}|\mathbf{z}_t, \mathbf{x})}{p_\theta(\mathbf{z}_s = \mathbf{x}|\mathbf{z}_t)} + q(\mathbf{z}_s = \varphi(\mathbf{x})|\mathbf{z}_t, \mathbf{x}) \log \frac{q(\mathbf{z}_s = Q^{(1)}\mathbf{x}|\mathbf{z}_t, \mathbf{x})}{p_\theta(\mathbf{z}_s = Q^{(1)}\mathbf{x}|\mathbf{z}_t)}\\
&= \frac{(\alpha_s \beta_{t|s} - \alpha_t)}{\beta_t - \alpha_t} \log \frac{\mathbf{z}_t^\intercal Q^{(1)}\mathbf{x}_\theta}{\langle \mathbf{x}, {Q^{(1)}}^\intercal \mathbf{z}_t\rangle \cdot \langle \mathbf{x}_\theta, \mathbf{x}\rangle} + 0.
\end{aligned} \tag{29}$$

Case 3. $\mathbf{z}_t \in S_2 = \{\mathbf{m}\}$

$$
\begin{aligned}
&\mathrm{D}_{\mathrm{KL}}\left(q(\mathbf{z}_s|\mathbf{z}_t, \mathbf{x}) \,\|\, p_\theta(\mathbf{z}_s|\mathbf{z}_t)\right) \\
&= \sum_{\mathbf{z}_s \in \{\mathbf{x}, \varphi(\mathbf{x}), \mathbf{m}\}} q(\mathbf{z}_s|\mathbf{z}_t, \mathbf{x}) \log \frac{q(\mathbf{z}_s|\mathbf{z}_t, \mathbf{x})}{p_\theta(\mathbf{z}_s|\mathbf{z}_t)} \\
&= q(\mathbf{z}_s = \mathbf{x}|\mathbf{z}_t = \mathbf{m}, \mathbf{x}) \log \frac{q(\mathbf{z}_s = \mathbf{x}|\mathbf{z}_t = \mathbf{m}, \mathbf{x})}{p_\theta(\mathbf{z}_s|\mathbf{z}_t)} + q(\mathbf{z}_s = Q^{(1)}\mathbf{x}|\mathbf{z}_t = \mathbf{m}, \mathbf{x}) \log \frac{q(\mathbf{z}_s = Q^{(1)}\mathbf{x}|\mathbf{z}_t = \mathbf{m}, \mathbf{x})}{p_\theta(\mathbf{z}_s|\mathbf{z}_t)} \\
&+ q(\mathbf{z}_s = \mathbf{m}|\mathbf{z}_t = \mathbf{m}, \mathbf{x}) \log \frac{q(\mathbf{z}_s = \mathbf{m}|\mathbf{z}_t = \mathbf{m}, \mathbf{x})}{p_\theta(\mathbf{z}_s|\mathbf{z}_t)} \\
&= \frac{(\alpha_s - \alpha_s\beta_{t|s})}{1 - \beta_t} \log \frac{1}{\langle \mathbf{x}_\theta, \mathbf{x}\rangle} + \frac{(1 - \beta_{t|s})(\beta_s - \alpha_s)}{1 - \beta_t} D_{\mathrm{KL}}\left(Q^{(1)}\mathbf{x} \| Q^{(1)}\mathbf{x}_\theta(\mathbf{z}_t, t)\right) + 0.
\end{aligned} \tag{30}
$$

Combined together, each term in Eq. 27 can be expressed as follows:

$$
\begin{aligned}
&\mathrm{D}_{\mathrm{KL}}\left(q(\mathbf{z}_s|\mathbf{z}_t, \mathbf{x}) \,\|\, p_\theta(\mathbf{z}_s|\mathbf{z}_t)\right) \\
&= \frac{(\alpha_s\beta_{t|s} - \alpha_t)}{\beta_t - \alpha_t} \left(\log\langle \mathbf{z}_t, Q^{(1)}\mathbf{x}_\theta\rangle - \log\langle \mathbf{x}_\theta, \mathbf{x}\rangle\right) \langle \mathbf{z}_t, Q^{(1)}\mathbf{x}\rangle \\
&- \left(\frac{(1 - \beta_{t|s})(\beta_s - \alpha_s)}{1 - \beta_t} \log\langle Q^{(1)}\mathbf{x}, Q^{(1)}\mathbf{x}_\theta(\mathbf{z}_t, t)\rangle + \frac{(\alpha_s - \alpha_s\beta_{t|s})}{1 - \beta_t} \log\langle \mathbf{x}_\theta, \mathbf{x}\rangle\right) \langle \mathbf{z}_t, \mathbf{m}\rangle + C_{s,t},
\end{aligned} \tag{31}
$$

for some constant $C_{s,t}$, which leads into following continuous time NELBO of HDDM:

General NELBO in HDDM

Theorem D.2. *Suppose forward process of two-level HDDM Q is defined with stochastic operator $Q^{(1)}, Q^{(2)}$, which are induced from the $\mathbf{\Phi}$ and masking operator, respectively as in Lemma 3.1. Then, for some constant C_1, the negative evidence lower bound (NELBO) can be expressed as follows:*

$$
\begin{aligned}
&\mathcal{L}^{\infty}_{\mathit{NELBO}}(\theta) = \mathbb{E}_{Q,t} \frac{(\alpha_s\beta_{t|s} - \alpha_t)}{\beta_t - \alpha_t} \left(\log\langle \mathbf{x}_\theta, \mathbf{x}\rangle - \log\langle \mathbf{z}_t, Q^{(1)}\mathbf{x}_\theta\rangle\right) \langle \mathbf{z}_t, Q^{(1)}\mathbf{x}\rangle \\
&+ \left(\frac{(1 - \beta_{t|s})(\beta_s - \alpha_s)}{1 - \beta_t} \log\langle Q^{(1)}\mathbf{x}, Q^{(1)}\mathbf{x}_\theta(\mathbf{z}_t, t)\rangle + \frac{(\alpha_s - \alpha_s\beta_{t|s})}{1 - \beta_t} \log\langle \mathbf{x}_\theta, \mathbf{x}\rangle\right) \langle \mathbf{z}_t, \mathbf{m}\rangle \\
&+ C_1.
\end{aligned} \tag{32}
$$

Note that above theorem holds for arbitrary stochastic row matrix $\mathbf{\Phi}$, which means we can design any stochastic mapping from $\mathcal{S}_0$ to $\mathcal{S}_1$.

When $Q^{(1)}$ is deterministic (i.e, its rows are composed of one-hot vectors), we can parameterize model to estimate the categories that are in the same mid-level state. This parameterization leads to further simplified form of Theorem D.2 as follows:

NELBO in HDDM with deterministic grouping

Corollary D.3.

$$
\begin{aligned}
&\mathcal{L}^{\infty}_{\mathit{NELBO}}(\theta) = \mathbb{E}_{Q,t} \frac{(-\alpha_s\beta_{t|s} + \alpha_t)}{\beta_t - \alpha_t} \log\langle \mathbf{x}_\theta(\mathbf{z}_t, t), \mathbf{x}\rangle \cdot \mathbb{I}\left[\mathbf{z}_t \in \mathcal{S}_1\right] \\
&+ \frac{1 - \beta_{t|s}}{1 - \beta_t} \left((\beta_s - \alpha_s) \log\langle Q^{(1)}\mathbf{x}, Q^{(1)}\mathbf{x}_\theta(\mathbf{z}_t, t)\rangle + \alpha_s \log\langle \mathbf{x}_\theta(\mathbf{z}_t, t), \mathbf{x}\rangle\right) \mathbb{I}\left[\mathbf{z}_t = \mathbf{m}\right] + C_2,
\end{aligned} \tag{33}
$$

for some constant C_2.

E EXPERIMENT DETAILS

E.1 DECOUPLED ATOM ENCODING (DAE)

While standard graph-based diffusion models typically adopt a coarse node encoding based solely on atomic numbers (Z), decoupled atom encoding (DAE) expands the original token vocabulary by explicitly decoupling three standards: aromaticity, hydrogen saturation, and formal charge magnitude. Decoupled Atom Encoding (DAE) expands the token vocabulary by explicitly decoupling three critical chemical descriptors: aromaticity, hydrogen saturation, and formal charge magnitude. Unlike previous methods that rely on implicit hydrogen estimation (e.g., via RDKit's valence rules), DAE treats these attributes as primary node features to be explicitly encoded and decoded. This approach resolves the one-to-many mapping problem between atomic tokens and their physical states, enabling the near-perfect reconstruction of drug-like scaffolds from the MOSES and GuacaMol datasets. Furthermore, this extended vocabulary facilitates the reliable generation of complex heteroaromatics and zwitterionic species which are extremely rare for baselines using coarse tokenization. Specifically, we emphasize that tokenizing $[nH]$ as a distinct state is fundamentally different from modeling explicit hydrogen atoms as separate nodes. While the latter can significantly increases graph complexity and computational overhead, DAE preserves graph sparsity while maintaining chemical precision.

Table 4: Comparison of Atom Vocabularies on the MOSES Dataset. DAE resolves structural ambiguities by decoupling elements into specific aromatic and hydrogen-locked states.

Method	Elemental Basis	Unique Tokens (Vocabulary)	Size
Standard Encoding	{C, N, S, O, F, Cl, Br}	C, N, S, O, F, Cl, Br	7
DAE (Ours)	{C, N, S, O, F, Cl, Br}	**Aliphatic:** C, N, S, O, F, Cl, Br **Aromatic:** c, n, nH, s, o	12

DAE in MOSES The MOSES dataset consists primarily of stable, neutral drug-like molecules which is clean lead filtered from the ZINC dataset (Irwin and Shoichet, 2005). In this context, the reconstruction bottleneck is primarily structural. Previous coarse-grained encodings fail to resolve the placement of pyrrolic hydrogens ($[nH]$), a critical motif in heteroaromatic rings like indole or imidazole. By explicitly decoupling aromaticity and hydrogen counts, DAE enables model can explicitly distinguish these motifs, resulting in improved generation quality.

Table 5: Vocabulary expansion for the Guacamol dataset. DAE scales from 12 elemental types to 56 semantic tokens including aromatic and charged atoms.

Category	Standard Encoding (Size: 12)	DAE Tokens (Size: 56)
Neutral Aliphatic	{C, N, O, F, B, Br, Cl, I, P, S, Se, Si}	C, N, O, F, B, Br, Cl, I, P, S, Se, Si
Aromatic States	(None / Implicit)	c, c+, c-, n, nH, n+, nH+, n-, s, s+, o, o+, se, se+, p
Charged & Hypervalent	(None / Implicit)	C+, C-, N+, NH+, NH2+, NH3+, N-, NH-, O+, O-, F+, F-, B-, Br+2, Br-, Cl+, Cl+2, Cl+3, Cl-, I+, I+2, I+3, P+, P-, S+, S-, Se+, Se-, Si-

DAE in GuacaMol GuacaMol (Brown et al., 2019) is constructed from a standardized subset of ChEMBL (Mendez et al., 2019), restricted to common medicinal-chemistry elements. In this unconstrained space, previous models suffer from a fundamental reconstruction failure; for instance, standard coarse-grained encoding achieves only a 1.88% success rate on the [nH] group, with a negligible 0.09% identity preservation rate. While previous models implicitly rely on the relaxation technique which can improve the success rates (e.g., increasing charged group success from 80.43% to 96.54%), this it only preserves 80.07% of total molecules, indicating a failure to maintain the

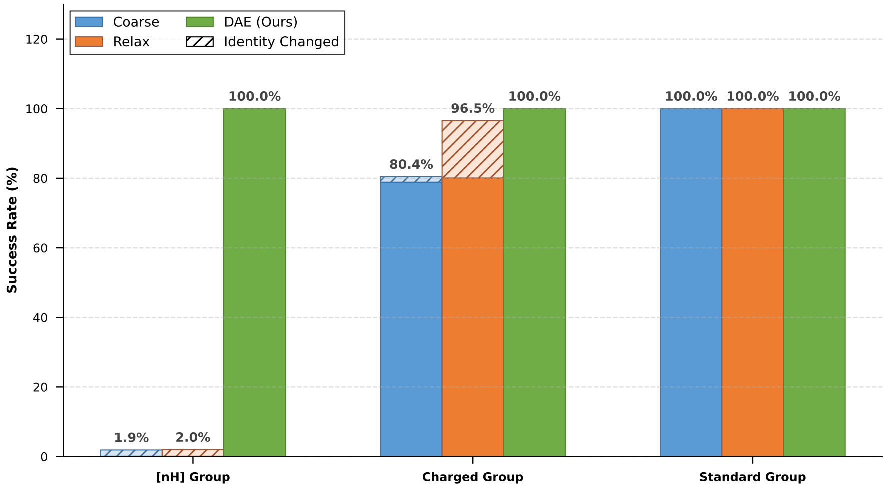

Figure 4: The ratios of generated molecules having formal charge. **MolHIT** can reach to the training level proportion, while models with previous coarse encoding (left two) barely generate the charged atoms.

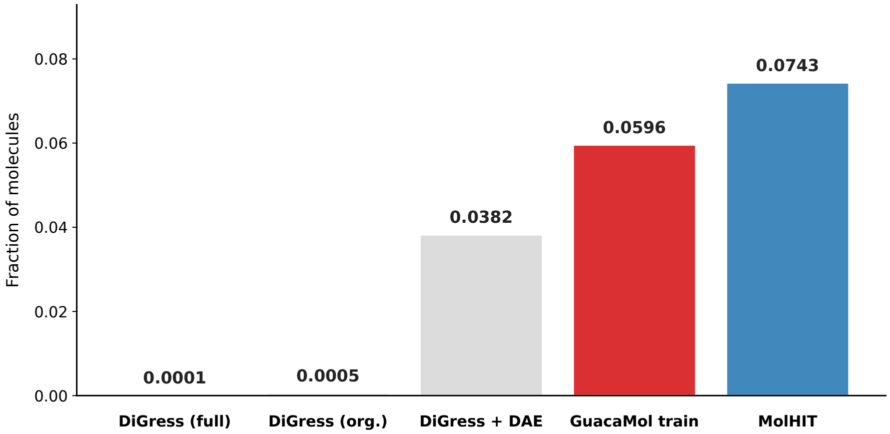

Figure 5: Reconstruction Fidelity and Identity Preservation. We measure the proportion of generated molecules that have at least one atom with formal charge.

original chemical identity. As illustrated in Figure 4, **MolHIT** addresses this through Decoupled Atom Encoding (DAE), which expands the vocabulary to 56 tokens by encode-decode the atoms with extended vocabulary space, resulting in 100 % success rate and over 99.98% in identity preservation rate. Moreover, as illustrated in Fig. 5, the effect of DAE also happens in generative performance, where it enables generating molecules with formal charge which consist of about 6% in GuacaMol dataset.

E.2 Grouping in HDDM

Grouping Details for MOSES and GuacaMol We employ dataset-specific grouping strategies to align the intermediate state space $\mathcal{S}_1$ with the underlying chemical distribution of each corpus. Table 6 summarizes these partitions.

E.3 Full experimental results with standard deviations

For statistical significance, we run 3 experiments for every experiment. We put the result including standard deviation of unconditional MOSES generation in Table 7, GuacaMol experiment in Table 8, multi-property guided generation result in Table 9, and scaffold extension result in Table 10.

Table 6: Deterministic grouping kernels for node state space partitioning in MOSES and GuacaMol.

Dataset	Group ID	Atom Elements ($\mathcal{S}_0$)
MOSES	Group 1	{C}
	Group 2	{N, O, S}
	Group 3	{F, Cl, Br}
	Group 4	{c, o, n, [nH], s}
GuacaMol	Group 1	{F, Cl, Br, I, F^-, Cl^-, Br^-}
	Group 2	{C, N, O, P, S, Se}
	Group 3	{c, n, [nH], o, s, se, p}
	Group 4	{N^+, n^+, $[nH]^+$, P^+, $[NH]^+$, $[NH_2]^+$, $[NH_3]^+$, Br^{+2}, Cl^{+2}, Cl^{+3}, I^{+2}, I^{+3}}
	Group 5	{O^-, N^-, $[NH]^-$, O^+, S^+, B^-, C^+, C^-, c^+, c^-, n^-, s^+, o^+, se^+, F^+, Cl^+, I^+, P^-, S^-, Se^+, Se^-, Si^-}
	Group 6	{B, Si}

Table 7: Unconditional generation on MOSES dataset with full statistics. We bring the reported value from Cometh and DeFoG from their work.

Category	Model	Quality ↑	Scaf-Novel ↑	Scaf-Ret. ↑	Valid ↑	Unique ↑	Novel ↑	Filters ↑	FCD ↓	SNN ↑	Scaf ↑
-	Training set	95.4	—	—	100.0	100.0	—	100.0	0.48	0.59	-
1D Sequence	VAE[23]	92.8 ± 0.2	0.22 ± 0.01	0.031 ± 0.003	97.7 ± 0.1	99.7 ± 0.0	69.5 ± 0.6	**99.7 ± 0.0**	0.57 ± 0.00	0.58 ± 0.01	5.9 ±1.0
	CharRNN [39]	92.6 ± 2.5	0.29 ± 0.04	**0.035 ± 0.003**	97.5 ± 2.6	99.9 ± 0.0	84.2 ± 5.1	99.4 ± 0.3	**0.52 ± 0.03**	0.56 ± 0.01	11.0 ± 0.8
	SAFE-GPT [31]	92.8 ± 0.0	0.12 ± 0.00	0.015 ± 0.000	**99.8 ± 0.0**	98.9 ± 0.0	43.7 ± 0.3	97.7 ± 0.0	0.72 ± 0.02	0.57 ± 0.01	6.3 ± 0.7
	GenMol [25]	62.1 ± 0.0	0.05 ± 0.00	0.012 ± 0.001	99.7 ± 0.1	64.0 ± 0.5	68.9 ± 0.4	98.1 ± 0.1	16.36 ± 0.07	**0.64 ± 0.01**	1.6 ± 0.1
2D Graph	DiGress [44]	82.5 ± 0.7	0.26 ± 0.00	0.031 ± 0.000	87.1 ± 0.9	**100.0 ± 0.0**	94.2 ± 0.2	97.5 ± 0.0	1.25 ± 0.03	0.53 ± 0.00	12.8 ± 1.4
	DisCo [48]	-	-	-	88.3	100.0	**97.7**	95.6	1.44	0.50	15.1
	Cometh [42]	82.1 ± 0.1	0.36 ± 0.00	0.023 ± 0.000	87.2 ± 0.0	100.0 ± 0.0	96.4 ± 0.1	97.3 ± 0.0	1.44 ± 0.02	0.51 ± 0.00	**16.8 ± 0.7**
	DeFoG [35]	88.5 ± 0.0	0.26 ± 0.00	0.031 ± 0.000	92.8 ± 0.0	99.9 ± 0.0	92.1 ± 0.0	98.9 ± 0.0	1.95 ± 0.00	0.55 ± 0.00	14.4 ± 0.0
	MolHIT	**94.2 ± 0.2**	**0.39 ± 0.00**	0.033 ± 0.001	99.1 ± 0.0	99.8 ± 0.0	91.4 ± 0.2	98.0 ± 0.00	1.03 ± 0.02	0.55 ± 0.00	14.4 ± 1.0

E.4 Implementation of baselines

For all baselines, we use released checkpoints when available. Otherwise, we train the models using their official codebase, following the training hyperparameters reported in the paper or provided in the codebase. Note that the original GenMol (Lee et al., 2025) model was trained on a much larger molecule dataset, so we train the model on the MOSES dataset for a fair comparison.

E.5 Unconditional generation with MOSES and GuacaMol

Training details For our model backbone, we adopt the graph transformer proposed by Vignac et al. (2022), which simultaneously predicts node and edge features. To ensure a fair comparison across all experimental settings, we maintain a consistent architecture of 12 transformer blocks without altering any internal dimensional configurations. The total trainable parameter count is approximately 16.2M. The introduction of additional token indices (DAE and HDDM) adds negligible overhead where representing a variance of less than 0.01% in total parameters. For training stability, we employ gradient clipping with a threshold of 2.0 and an Exponential Moving Average (EMA) rate of 0.999. We early stop with 100 epoch training with MOSES and 50 epochs with GuacaMol, compared to the original 300 epoch training of other graph diffusion baselines (Vignac et al., 2022; Siraudin et al., 2024; Qin et al., 2024). We also remove calculating geometric prior originally used in Vignac et al. (2022), where they use extra graph features as conditional information. In our experiments, this has negligible effects on the performance.

Evaluation of MOSES The following metrics are utilized to evaluate the generative performance on the MOSES dataset, following the standardized protocols established by Polykovskiy et al. (2020).

- **Validity (↑):** The fraction of generated molecules that pass RDKit's sanitization checks and basic chemical valency rules. High validity is a primary indicator that the **DAE** system successfully constrains the sampling process to the chemically feasible manifold.

Table 8: Full statistics of GuacaMol benchmark results (unfiltered). Val.: Validity, V.U.: Unique, V.U.N.: Novel. All results are averaged over 3 runs.

Model	Val. ↑	V.U. ↑	V.U.N. ↑	KL ↑	FCD ↓
Training set	100.0	100.0	—	99.9	92.8
DiGress (org.)	85.2	85.2	85.1	92.9	**68.0**
DiGress (full)	74.7 ± 0.4	74.6 ± 0.5	74.0 ± 0.4	92.4 ± 0.5	61.1 ± 0.2
DiGress+DAE	65.2 ± 0.4	65.2 ± 0.4	64.9 ± 0.4	87.0 ± 0.4	49.2 ± 0.6
MolHIT (Ours)	**87.1** ± **0.5**	**87.1** ± **0.3**	**86.0** ± **0.5**	**96.7** ± **0.1**	54.9 ± 0.2

Table 9: Full statistics of multi-property guided generation on MOSES with 4 different conditions. We report mean absolute error (MAE; ↓), Pearson correlation (r; ↑), and validity. Avg. is the macro-average across properties. **Bold** denotes best values.

	MAE ↓					Pearson r ↑					Validity (%) ↑
Method	QED	SA	LogP	MW	**Avg.**	QED	SA	LogP	MW	**Avg.**	
Marginal	0.117 (±0.002)	0.115 (±0.003)	0.067 (±0.001)	0.272 (±0.009)	0.143 (±0.004)	0.489 (±0.003)	0.570 (±0.012)	0.802 (±0.003)	0.396 (±0.001)	0.564 (±0.005)	75.03 (±0.74)
Marginal + DAE	0.107 (±0.001)	0.094 (±0.001)	0.061 (±0.000)	0.227 (±0.004)	0.122 (±0.001)	0.565 (±0.005)	0.559 (±0.009)	0.836 (±0.005)	0.437 (±0.015)	0.599 (±0.002)	87.85 (±0.46)
MolHIT	**0.061 (±0.001)**	**0.040 (±0.001)**	**0.049 (±0.001)**	**0.081 (±0.005)**	**0.058 (±0.002)**	**0.804 (±0.009)**	**0.790 (±0.011)**	**0.950 (±0.004)**	**0.685 (±0.024)**	**0.807 (±0.011)**	**96.31 (±0.23)**

Table 10: Full statistics of scaffold extension results on MOSES. Results are averaged over 3 runs of 10,000 targets. **Hit@k** denotes the recovery of ground-truth within k samples.

Model	Valid (%) ↑	Diversity ↑	Hit@1 ↑	Hit@5 ↑
DiGress	50.8 ± 0.5	44.8 ± 1.8	2.07 ± 0.09	6.41 ± 0.21
Marginal + DAE	64.8 ± 0.2	**58.0** ± **0.1**	1.67 ± 0.10	6.37 ± 0.24
MolHIT (Ours)	**83.9** ± **0.4**	57.4 ± 0.6	**3.92** ± **0.23**	**9.79** ± **0.09**

- **Uniqueness (↑):** The proportion of valid molecules that are not duplicates. This measures the model's ability to avoid mode collapse and explore a diverse structural space.
- **Novelty (↑):** The fraction of valid, unique molecules that were not present in the training set. This differentiates between a model that has memorized the data and one that has learned the underlying generative distribution.
- **Filters (↑):** The percentage of generated molecules that pass common medicinal chemistry filters (e.g., MCULE, BRENK, and PAINS). This evaluates the drug-likeness and synthetic viability of the generated samples.
- **Fréchet ChemNet Distance (FCD (Preuer et al., 2018), ↓):** A measure of the distance between the multivariate distributions of generated and test molecules in the feature space of ChemNet. FCD captures both chemical and biological similarity, serving as the most rigorous metric for distributional fidelity.
- **Similarity to Nearest Neighbor (SNN, ↑):** The average Tanimoto similarity between a generated molecule and its closest neighbor in the test set. High SNN indicates that the model has captured the specific structural motifs and chemical space of the benchmark.
- **Scaffold Similarity (Scaf, ↑):** The cosine similarity between the frequencies of Bemis–Murcko scaffolds (Bemis and Murcko, 1996) in the generated and test sets. This assesses whether the model's learned distribution of backbone structure matches the architectural diversity of real-world leads.

Baselines

E.6 Structure novelty metric

- **Scaffold Novelty:** We evaluate the model's ability to innovate beyond the training distribution using Bemis–Murcko scaffolds (Bemis and Murcko, 1996). The absolute number of unique generated scaffolds absent from the training set:

$$\text{Scaf-Novel} = \frac{|\mathcal{S}_{\text{gen}} \setminus \mathcal{S}_{\text{train}}|}{n_{\text{total}}}. \tag{34}$$

This metric quantifies the model's capacity for structural extrapolation, measuring its efficiency in exploring the beyond the molecular structure of the given dataset.

Table 11: Multi-property guided generation on MOSES with four different conditions. We report mean absolute error (MAE), Pearson correlation (Pearson r), and validity. Avg. denotes the macro-average across four properties. Bold denotes best performances. All results are the averaged value over 3 runs of 10,000 samples.

	MAE ↓					Pearson r ↑					Validity (%) ↑
Method	QED	SA	logP	MW	**Avg.**	QED	SA	logP	MW	**Avg.**	
Marginal	0.117	0.115	0.067	0.272	0.143	0.489	0.570	0.802	0.396	0.564	75.03
Marginal + DAE	0.107	0.094	0.061	0.227	0.122	0.565	0.559	0.836	0.437	0.599	87.85
MolHIT (Ours)	**0.061**	**0.040**	**0.049**	**0.081**	**0.058**	**0.804**	**0.790**	**0.950**	**0.685**	**0.807**	**96.31**

Table 12: Full GuacaMol benchmark results using unfiltered dataset.

Model	Val.	V.U.	V.U.N.	KL Div.	FCD
Training set	100.0	100.0	—	99.9	92.8
DiGress [44] (org.)	85.2	85.2	85.1	92.9	**68.0**
DiGress [44] (full)	74.7	74.6	74.0	92.4	61.1
DiGress + DAE	65.2	65.2	64.9	87.0	49.2
MolHIT (Ours)	**87.1**	**87.1**	**86.0**	**96.7**	54.9

- **Scaffold Retrieval:** This assesses the model's ability to rediscover known, high-quality frameworks from the held-out test set. This is defined as the absolute number of unique test-set scaffolds successfully generated: from the training set:

$$\text{Scaf-Ret} = \frac{|\mathcal{S}_{\text{gen}} \cap \mathcal{S}_{\text{test}}|}{n_{\text{total}}}. \tag{35}$$

  Scaffold retrieval serves as a rigorous test of distributional accuracy. A high retrieval density demonstrates that the model has not merely learned to generate novel-looking noise, but has accurately captured the underlying manifold of valid, drug-like molecules defined by the test distribution.

E.7 Unconditional generation with GuacaMol

GuacaMol experiment For experiment on GuacaMol, we test our algorithm on the unfiletered, full dataset. Previous graph diffusion model baselines (Vignac et al., 2022; Siraudin et al., 2024; Qin et al., 2024) train the model on the filtered dataset, where they filter out the molecules that are failed to be reconstructed back. This can bias the training data distribution. In contrast, we use full, unfiltered dataset for the experiment and since there is no graph diffusion baseline, we compare with the original DiGress trained on a full GuacaMol dataset with coarse atom encoding, Discrete Diffusion using marginal transition with DAE, and compare them with MolHIT.

E.8 Multi-property guided generation

Data construction For conditional generation, we augment the MOSES dataset (Polykovskiy et al., 2020) with four continuous molecular descriptors: Quantitative Estimate of Drug-likeness (QED), Synthetic Accessibility (SA) score, Octanol-Water Partition Coefficient (logP), and Molecular Weight (MW).

- **Quantitative Estimate of Drug-likeness (QED, ↑):** A widely used composite score that summarizes multiple molecular properties (e.g., lipophilicity, polarity, and molecular size) into a single measure of drug-likeness; higher values indicate more drug-like compounds.
- **Synthetic Accessibility (SA, ↓):** An empirical estimate of synthetic difficulty that combines fragment-based contributions with a complexity penalty; lower values indicate molecules that are easier to synthesize.
- **Octanol–Water Partition Coefficient (logP):** A measure of lipophilicity that is informative of solubility and membrane permeability; excessively high logP is typically associated with poor solubility and unfavorable ADMET profiles.
- **Molecular Weight (MW):** The molecular mass in Daltons. Consistency with the training distribution (e.g., MOSES) helps ensure generated molecules remain within a drug-like regime.

All properties are calculated using the RDKit library and the sascorer module (Ertl and Schuffenhauer, 2009). To ensure stable convergence of the conditioning vector $\mathbf{C}$ within our AdaLayerNorm layers, we perform min-max normalization on these values using the global statistics of the training split, which are in Table 13.

Table 13: Min and max values for molecular property conditioning in MOSES training / test split.

Split	Statistic	QED	SA Score	logP	MW (Da)
Training	Min	0.1912	1.2694	-5.3940	250.017
	Max	0.9484	7.4831	5.5533	349.999
Test	Min	0.2265	1.3339	-4.2894	250.042
	Max	0.9484	6.6916	5.7255	349.990

Conditional graph transformer While we maintain the core node-edge attention mechanism of the original graph transformer (Vignac et al., 2022), we introduce several key modifications to enable conditional modeling. First, we remove the persistent global feature vector y—which in the original framework is updated at every layer—and replace it with a centralized conditioning vector $\mathbf{C}$. This vector is composed of a sinusoidal timestep embedding (Ho et al., 2020) and an optional MLP-encoded external property condition $\mathbf{c}$. Second, to integrate $\mathbf{C}$ into the denoising process, we replace standard Layer Normalization with Adaptive Layer Normalization (AdaLayerNorm) for node features. Specifically, for a node embedding $\mathbf{x}$, the normalization is defined as:

$$\text{AdaLN}(\mathbf{x}, \mathbf{C}) = (1 + \gamma(\mathbf{C})) \cdot \text{LayerNorm}(\mathbf{x}) + \beta(\mathbf{C})$$

where γ and β are affine transformations of the conditioning vector. This allows the global context (time and properties) to directly modulate the scale and shift of node representations. Finally, we implement Classifier-Free Guidance (CFG) support by incorporating a dropout mechanism on the property embedding during training, while ensuring the temporal signal remains persistent to maintain denoising stability. Our conditional graph transformer naturally inherits permutation equivariance, which is different from the Liu et al. (2024).

Evaluation details For sampling, we employ Classifier-Free Guidance (CFG) (Ho and Salimans, 2022) with a guidance scale of $w = 1.0$. We observe that in our discrete graph-diffusion framework, increasing the guidance weight beyond unity did not consistently yield better property alignment. We leave the better design the sampler or models to be effective in higher guidance strength w as a promising avenue for future research.

E.9 Scaffold extension

Task Formulation Given a ground-truth molecule $\mathcal{G}$ from the test split, we use RDKit to extract its scaffold $\mathcal{S} \subset \mathcal{G}$. The task is to generate a completed molecule $\mathcal{M}$ such that $\mathcal{S} \subseteq \mathcal{M}$. To isolate the generative capability from size prediction errors, we bound the generation size (number of atoms) to match $|\mathcal{G}|$.

Sampling Protocol At each reverse timestep t, the region corresponding to $\mathcal{S}$ is forced to be the same (i.e, $q(\mathbf{x}_t|\mathbf{x}_{\text{scaffold}}) = \mathbf{x}_{\text{scaffold}}$, ensuring the scaffold region is strictly fixed during the generation. The extension region is initialized from the limit distribution (i.e, prior) p_{prior} and evolved via the standard reverse process. We generate $K = 1, 5$ independent samples per scaffold to capture the model's exploration capability.

Metric Definitions Metrics are computed per scaffold and then averaged across the test set. Let $\mathcal{M}_1, \ldots, \mathcal{M}_K$ be the generated graphs for a single scaffold.

- **Validity:** The fraction of $\mathcal{M}_i$ that are chemically valid according to RDKit sanitization.
- **Diversity:** Calculated on the unique valid set $\mathcal{U}$. We define diversity as $1 - \frac{1}{|\mathcal{U}|^2} \sum_{u,v \in \mathcal{U}} \text{Sim}(u, v)$, where Sim is the Tanimoto similarity using Morgan fingerprints (r = 2, 2048 bits).

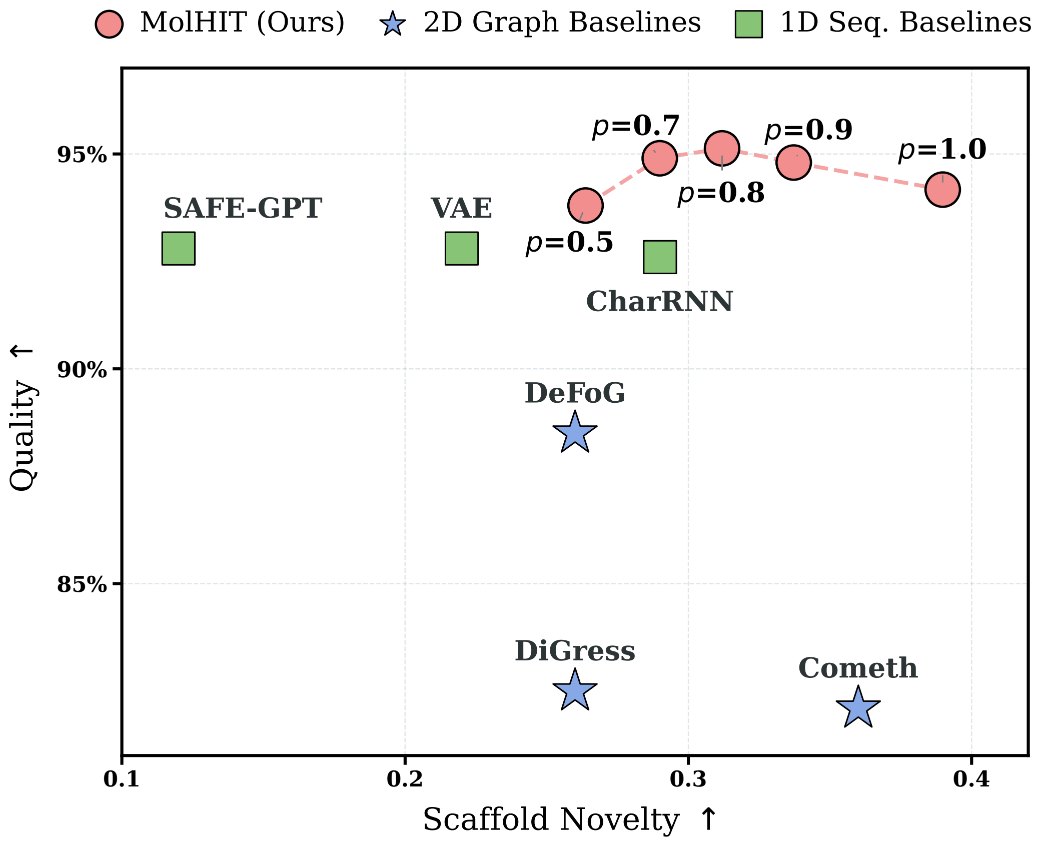

Figure 6: Effect of top-p sampling in MolHIT.

Algorithm 1 PN-sampler with temperature sampling

```
Input: Sample size $S$, Timesteps $T$, Temperature $\tau$, Nucleus threshold $p$
for $i = 1$ to $S$ do
    Sample $N \sim P_{\text{train}}(N)$
    $G_T \sim p_T(G_T)$ $\{G = (\mathbf{X}, \mathbf{E})\}$
    for $t = T$ down to $\Delta t$ with step $\Delta t$ do
        $\hat{p}_0(\mathbf{X}), \hat{p}_0(\mathbf{E}) \leftarrow f_\theta(G_t, t)$
        $\hat{p}'_0(\mathbf{X}) \leftarrow \text{TopP}(\text{Softmax}(\hat{p}_0(\mathbf{X})/\tau), p)$
        $\hat{\mathbf{X}}_0 \sim \text{Categorical}(\hat{p}'_0(\mathbf{X}))$
        $\hat{\mathbf{E}}_0 \sim \text{Categorical}(\hat{p}_0(\mathbf{E}))$
        $G_{t-\Delta t} \sim q(G_{t-\Delta t}|\hat{G}_0)$ where $\hat{G}_0 = (\hat{\mathbf{X}}_0, \hat{\mathbf{E}}_0)$
    end for
    Store $G_{\text{final}}$
end for
```

- **Exact Match (Hit@K):** A binary indicator, set to 1 if the ground truth $\mathcal{G}$ (canonical SMILES) is present in the generated set $\{\mathcal{M}_1, \ldots, \mathcal{M}_K\}$, and 0 otherwise.

E.10 Temperature sampling

The algorithm overview of the PNJ sampler is illustrated in Alg. 1. In Fig. 6, we analyze MolHIT trained on the MOSES dataset across a range of top-p values. Our results demonstrate that as top-p decreases, a clear trade-off emerges between sample quality and scaffold novelty. Specifically, lowering the top-p value from 1.0 down to 0.8 consistently improves the quality and validity of generated structures, while further reducing the p threshold leads to a sharp decline in both chemical metrics and structural diversity. Notably, when sampling with top-p, **MolHIT** achieves a high validity of 99.4% and a quality score of 95.1%, demonstrating the effectiveness of the nucleus sampling in **MolHIT**.