MOLGAN: FEW-STEP 3D MOLECULE GENERATION WITH HIERARCHICAL REASONING*

Anonymous authors
Paper under double-blind review

ABSTRACT

Diffusion models generate high-quality three-dimensional molecules but typically require long iterative sampling trajectories. We introduce **MolGAN**, a few-step molecular generator that combines energy-based adversarial training with the multi-timescale computation of the Hierarchical Reasoning Model. This organization allows substantial latent computation without exposing a long sequence of molecular states. On QM9 and GEOM-DRUG, MolGAN achieves state-of-the-art few-step performance across molecular-quality and sampling-efficiency metrics. Its learned energy model additionally enables test-time scaling, improving molecular-property scores on GEOM-DRUG as the compute budget increases.

1 INTRODUCTION

Three-dimensional molecular generation jointly models atom types and coordinates, enabling direct reasoning about molecular geometry and interactions. Diffusion models produce strong structures but often require tens to hundreds of network evaluations per sample, limiting large-scale virtual screening and lead optimization (Hoogeboom et al., 2022; Xu et al., 2023; Guan et al., 2023).

Symmetry-aware architectures have improved both unconditional generation and protein-pocket-conditioned design by representing categorical atom features together with continuous three-dimensional coordinates (Hoogeboom et al., 2022; Xu et al., 2023; Guan et al., 2023; Qu et al., 2024). Their iterative sampling procedures nevertheless expose a persistent quality–cost trade-off: additional refinement can improve molecular geometry, but increases the computation required for every generated candidate.

Recent work therefore studies high-quality generation with only a few steps. FlashMol distills a 1000-step GeoLDM teacher into a four-step student, demonstrating aggressive acceleration while highlighting the limited opportunities for correcting invalid structures, unrealistic geometry, or reduced diversity (Wei et al., 2026).

We introduce **MolGAN**, a generative adversarial architecture built on the Hierarchical Reasoning Model (HRM) (Wang et al., 2025). A slowly updated high-level state coordinates multiple rapid low-level computations before each molecular output, enabling deep latent refinement over a short external trajectory.

2 RELATED WORK

2.1 GENERATIVE MODELS FOR 3D MOLECULES

Three-dimensional molecular generators must jointly model atom identities and coordinates while respecting molecular symmetries. EDM introduced E(3)-equivariant diffusion, GeoLDM moved generation into a geometric latent space, and TargetDiff and MolCRAFT incorporated protein-pocket conditioning (Hoogeboom et al., 2022; Xu et al., 2023; Guan et al., 2023; Qu et al., 2024). These symmetry-aware approaches are effective but often require many network evaluations.

Sampling cost can be reduced with accelerated solvers, flow matching, progressive distillation, and consistency objectives (Song et al., 2021; Lu et al., 2022; Lipman et al., 2023; Salimans & Ho, 2022;

*Official code will be released on GitHub soon.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

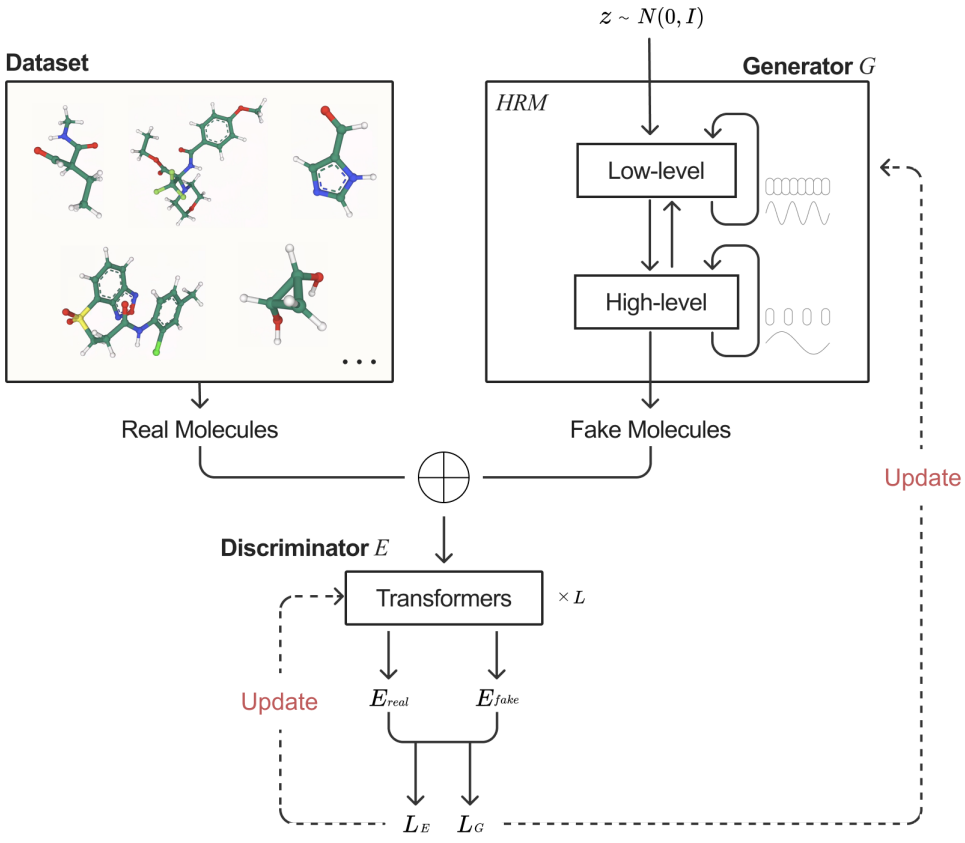

Figure 1: Overview of MolGAN training. The diagram abbreviates G_θ and E_ϕ as G and E, respectively. The incoming state $\mathbf{x}^{(0)} \sim p_{\text{init}}$ denotes the random molecular initialization. Within each external refinement pass, the HRM generator performs n high-level cycles with m low-level updates per cycle. Real and generated molecules are jointly processed by an energy Transformer with L layers, yielding $E_{\text{real}} \equiv E_\phi(\mathbf{x}_{\text{real}})$ and $E_{\text{fake}} \equiv E_\phi(\mathbf{x}_{\text{fake}})$. The relativistic losses $\mathcal{L}_E$ and $\mathcal{L}_G$ update the energy model and generator, respectively.

Song et al., 2023a). Implicit and high-order solvers shorten the numerical integration of a trained diffusion model (Song et al., 2021; Lu et al., 2022), whereas progressive distillation and consistency training modify the learned sampling process to reduce the required number of steps (Salimans & Ho, 2022; Song et al., 2023a). Flow matching instead learns a continuous transport path that can be integrated using flexible numerical solvers (Lipman et al., 2023). FlashMol distills a pretrained GeoLDM teacher into a few-step student (Wei et al., 2026); MolGAN instead uses teacher-free hierarchical recurrent computation and adversarial energy training.

2.2 Hierarchical Reasoning Models

The Hierarchical Reasoning Model couples a slowly evolving high-level state with rapid low-level computation, enabling substantial latent reasoning without explicit long-form traces (Wang et al., 2025). Repeated low-level updates followed by high-level revision allocate computation within the latent state rather than through a long sequence of externally visible outputs. MolGAN transfers this multi-timescale organization to molecular refinement.

Our main contributions are:

- We adapt hierarchical reasoning to jointly generate atom types and three-dimensional coordinates through coordinated high- and low-level recurrent computation.
- We introduce a teacher-free training approach that combines stochastic recurrent refinement, a relativistic pairwise energy objective, and persistent latent carries.
- We evaluate MolGAN under strict few-step budgets on QM9 and GEOM-DRUG, demonstrating a state-of-the-art quality–efficiency trade-off against diffusion, flow, and distilled molecular generators.

3 PRELIMINARIES

3.1 THREE-DIMENSIONAL MOLECULAR STATES

We represent a discrete molecule with q atoms as $\mathbf{x} = (\mathbf{a}, \mathbf{r})$, where $\mathbf{a} \in \mathcal{A}^q$ contains categorical atom attributes and $\mathbf{r} \in \mathbb{R}^{q\times 3}$ contains Cartesian coordinates. During refinement, we relax the categorical component to atom-type probabilities and write $\mathbf{x}^{(k)} = (\mathbf{p}^{(k)}, \mathbf{r}^{(k)})$; hard labels are recovered by taking the row-wise argmax of $\mathbf{p}^{(k)}$. During preprocessing, physical-atom coordinates are centered by subtracting their unweighted mean. We initialize the coordinates of every slot as independent noise, including `PAD` elements. The Transformer processes batched atom tokens without positional encoding, making its outputs permutation-equivariant with respect to atom-token ordering. The architecture is not strictly equivariant under rotations or translations; we therefore apply an independent random rotation to each molecule during energy-model training to reduce orientation sensitivity.

3.2 FEW-STEP MOLECULAR GENERATION

Let $\mathbf{x}^{(0)}$ denote the initial relaxed state, whose coordinates are initialized as noise, and let $\mathbf{x}^{(K)}$ denote the final state after K refinement passes. At each pass, the generator applies the same learned transition and updates its high- and low-level latent states:

$$\left(\mathbf{x}^{(k+1)}, \mathbf{z}_H^{(k+1)}, \mathbf{z}_L^{(k+1)}\right) = G_\theta\left(\mathbf{x}^{(k)}, \mathbf{z}_H^{(k)}, \mathbf{z}_L^{(k)}\right), \qquad k = 0, \ldots, K-1. \tag{1}$$

When each transition requires one generator evaluation, the total number of function evaluations is K. Few-step generation aims to minimize K while preserving chemical validity, three-dimensional stability, and distributional diversity.

3.3 HIERARCHICAL REASONING

For a fixed refinement pass k, HRM maintains a high-level state $\mathbf{h}_H$ and a low-level state $\mathbf{h}_L$. For each high-level cycle $i = 0, \ldots, n-1$, the low-level module performs m recurrent updates,

$$\mathbf{h}_L^{i,t+1} = F_{\text{low},\theta}\left(\mathbf{h}_L^{i,t}, \mathbf{h}_H^{i}, \mathbf{e}^{(k)}\right), \qquad t = 0, \ldots, m-1, \tag{2}$$

where $\mathbf{e}^{(k)}$ is the molecular embedding at pass k. The high-level state is then updated using the converged low-level representation,

$$\mathbf{h}_H^{i+1} = F_{\text{high},\theta}\left(\mathbf{h}_H^{i}, \mathbf{h}_L^{i,m}\right). \tag{3}$$

Repeating this interaction for n cycles performs $n\,m$ low-level updates and n high-level updates, totaling $n(m+1)$ HRM module evaluations while retaining a compact external computation graph.

4 METHOD

Figure 1 summarizes the core MolGAN training architecture and expands the hierarchical reasoning module used by the generator. We first describe the molecular representation and initialization, then the generator and energy model, and finally the training objectives.

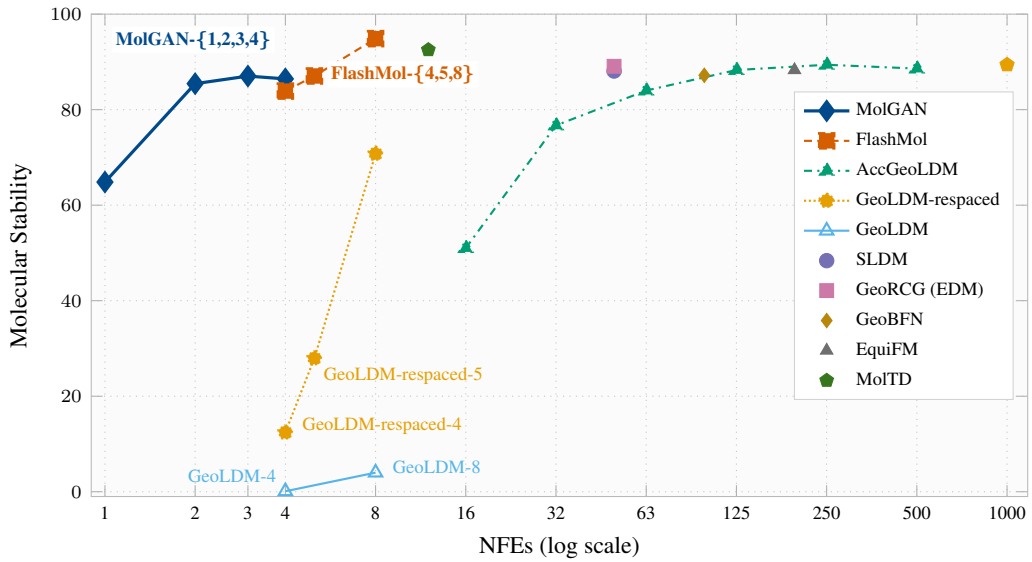

Figure 2: QM9 molecular stability as a function of network evaluations. Curves and isolated points compare molecular generators across a wide range of sampling budgets. MolGAN lies on the quality–efficiency frontier in the few-step regime, achieving high molecular stability with only one to four external refinement passes.

Packed molecular representation. MolGAN represents a batch of B variable-size molecules as flat atom-type and coordinate tensors. For $b = 1, \ldots, B$, let molecule b contain q_b physical atoms and q_b^{PAD} additional `PAD` elements, giving the packed length $\widetilde{q}_b = q_b + q_b^{\mathrm{PAD}}$. With $Q = \sum_{b=1}^{B} \widetilde{q}_b$ total slots, the packed representation is

$$\mathbf{a} \in \{0, \ldots, C-1\}^Q, \qquad \mathbf{r} \in \mathbb{R}^{Q \times 3}. \tag{4}$$

Here, $C = |\mathcal{A}|+1$: the first $C-1$ classes are the chemical elements in $\mathcal{A}$, and class $C-1$ is the special `PAD` type. Molecules are concatenated without padding every example to the largest molecule in the minibatch. A segment map $\beta : \{1, \ldots, Q\} \to \{1, \ldots, B\}$ identifies the molecule containing each packed slot, and every Transformer layer (Vaswani et al., 2017) uses the block-diagonal attention mask

$$\mathcal{M}_{ij} = \begin{cases} 0, & \beta(i) = \beta(j), \\ -\infty, & \beta(i) \neq \beta(j), \end{cases} \tag{5}$$

so a slot can attend only to slots from the same molecule.

Coordinate scaling. We observed that the generator consistently struggled to recover accurate bond lengths, typically missing the target distances by a few picometers. We attribute this behavior to the comparatively small numerical scale of Cartesian coordinates relative to the other activations processed by the network. To place the coordinates on a more suitable numerical scale, we multiply them by a fixed factor γ during preprocessing. The generator operates entirely in this scaled coordinate space, and its outputs are divided by γ when converted back to physical units. In our experiments, this transformation eliminated the systematic bond-length offset and substantially improved geometric stability.

The same coordinate scale is used to construct the initial relaxed state $\mathbf{x}^{(0)} = (\mathbf{p}^{(0)}, \mathbf{r}^{(0)})$ during training. We initialize $\mathbf{p}^{(0)}$ with shuffled one-hot atom labels and draw each coordinate from

$$\mathbf{r}_j^{(0)} \sim \mathcal{N}\left(\mathbf{0}, \gamma^2 \mathbf{I}_3\right), \qquad j = 1, \ldots, Q, \tag{6}$$

where γ is the coordinate scale and $\mathbf{I}_3$ is the 3×3 identity matrix. This initialization removes the original atom-to-molecule assignment and molecular geometry before refinement begins.

Learned `PAD` atom type. The `PAD` class is a semantic output of the model rather than conventional minibatch padding. During training, we append a random number of `PAD` elements, up to a configured maximum, to each molecule. Their coordinates are sampled independently as

$$\mathbf{r}_{b,j}^{\mathrm{PAD}} \sim \mathcal{N}\big(\mathbf{0}, \gamma^2 \mathbf{I}_3\big), \qquad b = 1, \ldots, B, \quad j = 1, \ldots, q_b^{\mathrm{PAD}}, \tag{7}$$

and the physical atoms and `PAD` elements are randomly permuted before packing. Consequently, `PAD` participates in the same categorical prediction space and adversarial distribution-matching objective as the chemical elements. The generator can therefore assign `PAD` to coordinate slots that should not contain a physical atom. During generation, the initial slot-count and atom-type distributions include the expected number of `PAD` elements; after the final refinement pass, all slots predicted as `PAD` are removed before molecule reconstruction. The number of retained atoms is thus determined by the generated labels rather than fixed by the initial number of coordinate slots, giving MolGAN additional flexibility to generate molecules of different sizes.

Generator G_θ and energy model E_ϕ. With the packed state and initialization defined, we now describe the two trainable components. The generator G_θ maintains token-level high- and low-level carries, $\mathbf{z}_H, \mathbf{z}_L \in \mathbb{R}^{Q \times d}$, where d is the hidden width; these carries implement the abstract states $\mathbf{h}_H$ and $\mathbf{h}_L$ introduced above. At refinement pass k, the relaxed molecular state is $\mathbf{x}^{(k)} = (\mathbf{p}^{(k)}, \mathbf{r}^{(k)})$, where $\mathbf{p}^{(k)} \in [0,1]^{Q \times C}$ contains categorical atom-type probabilities and $\mathbf{r}^{(k)} \in \mathbb{R}^{Q \times 3}$ contains coordinates. Atom labels are embedded with a learned lookup table Emb_a and coordinates with a linear projection Emb_r. We first obtain the hard labels

$$\widehat{a}_j^{(k)} = \arg \max_{c \in \{0, \ldots, C-1\}} p_{j,c}^{(k)}, \qquad j = 1, \ldots, Q, \tag{8}$$

and then form the molecular input embedding

$$\mathbf{e}^{(k)} = \mathrm{Emb}_a\Big(\widehat{\mathbf{a}}^{(k)}\Big) + \mathrm{Emb}_r\Big(\mathbf{r}^{(k)}\Big). \tag{9}$$

Both HRM levels are non-causal Transformer stacks with multi-head self-attention, RMS pre-normalization (Zhang & Sennrich, 2019), and SwiGLU feed-forward layers (Shazeer, 2020). The stacks $\mathcal{T}_{\mathrm{low}}$ and $\mathcal{T}_{\mathrm{high}}$ instantiate the recurrent maps $F_{\mathrm{low},\theta}$ and $F_{\mathrm{high},\theta}$, respectively. The low-level stack receives the sum of its current state, the high-level state, and the molecular input,

$$\mathbf{z}_L \leftarrow \mathcal{T}_{\mathrm{low}}\Big(\mathbf{z}_L + \mathbf{z}_H + \mathbf{e}^{(k)}\Big), \tag{10}$$

whereas the high-level stack is updated from the low-level representation,

$$\mathbf{z}_H \leftarrow \mathcal{T}_{\mathrm{high}}(\mathbf{z}_H + \mathbf{z}_L). \tag{11}$$

For every high-level cycle, MolGAN applies m low-level updates followed by one high-level update. The output head uses $\mathbf{W}_o \in \mathbb{R}^{(C+3) \times d}$ and $\mathbf{b}_o \in \mathbb{R}^{C+3}$ to map the final high-level representation directly to C atom logits and three Cartesian coordinates, with $\mathbf{b}_o$ broadcast across the Q slots:

$$\mathbf{o}^{(k)} = \mathbf{z}_H \mathbf{W}_o^\top + \mathbf{b}_o \in \mathbb{R}^{Q \times (C+3)}, \qquad \mathbf{p}^{(k+1)} = \mathrm{softmax}(\mathbf{o}_{:,1:C}^{(k)}), \qquad \mathbf{r}^{(k+1)} = \mathbf{o}_{:,C+1:C+3}^{(k)}. \tag{12}$$

The model therefore predicts the next coordinates directly rather than predicting a coordinate residual.

The energy model E_ϕ serves as the discriminator. Let L denote its number of Transformer layers. Atom-type vectors, Cartesian coordinates, and expanded carry embeddings are projected into a common hidden space and processed with a non-causal Transformer using the same packed block-diagonal attention scheme. During training, each molecule is independently rotated by a uniformly sampled $\mathrm{SO}(3)$ transformation before coordinate embedding. The resulting atom features are mean-pooled within each molecule and mapped to a scalar energy $E_\phi(\mathbf{x})$.

Stochastic refinement and persistent carries. Generation is organized into repeated refinement passes. With a configured maximum of K passes, training samples a realized training pass count

$$\widetilde{K} \sim \mathrm{Uniform}\{1, \ldots, K\}. \tag{13}$$

The first $\widetilde{K}-1$ passes run under `no_grad`; only the final pass contributes gradients. Within that final pass, gradients are tracked only through the last low- and high-level update pair, while the earlier recurrent updates are evaluated without gradient tracking. We found empirically that this restriction performs better than backpropagating through all recurrent updates. The returned carries are detached before being reused, yielding the one-step-gradient approximation used by HRM. Each pass feeds the previous softmax atom predictions and coordinates back into G_θ, with both inputs detached before re-embedding. At inference, all K passes are executed and every intermediate molecular state is retained for NFE-aware evaluation.

The HRM states are supplied by a persistent first-in–first-out carry queue. The queue stores detached $\mathbf{z}_H$ and $\mathbf{z}_L$ vectors across minibatches, consumes the oldest Q entries for the current packed batch, and appends the newly produced states. This mechanism gives G_θ a persistent latent population without propagating gradients across training iterations. The energy model E_ϕ uses an analogous per-molecule persistent carry $\mathbf{z}_{E,b} \in \mathbb{R}^d$ for $b = 1, \ldots, B$. After each forward pass, the updated atom-level energy features are mean-pooled within each molecule to form $\mathbf{z}_{E,b}$, and the resulting carries are randomly permuted before reuse.

Relativistic pairwise loss. Real and generated molecules are concatenated into a single packed forward pass through E_ϕ. Its parameters are updated using a relativistic pairwise objective (Jolicoeur-Martineau, 2019),

$$\mathcal{L}_E = \mathbb{E}\left[\mathrm{softplus}(E_\phi(\mathbf{x}_{\mathrm{real}}) - E_\phi(\mathbf{x}_{\mathrm{fake}}))\right], \tag{14}$$

which encourages real molecules to receive lower energy than generated molecules. During the generator update, the energy model E_ϕ is frozen and the generator G_θ is optimized using

$$\mathcal{L}_G = \mathbb{E}\left[\mathrm{softplus}(E_\phi(\mathbf{x}_{\mathrm{fake}}) - \mathrm{stopgrad}(E_\phi(\mathbf{x}_{\mathrm{real}})))\right]. \tag{15}$$

Only the final stochastic refinement output is used for the generator loss.

VISReg representation regularization. To regularize the molecule-level features learned by the generator model, MolGAN incorporates Variance–Invariance–Sketching Regularization (VISReg) (Wu et al., 2026). For a batch of B molecules, two independently rotated views are processed by the same energy model and projector g_ψ, producing $\mathbf{Z}^{(1)}, \mathbf{Z}^{(2)} \in \mathbb{R}^{B\times D}$. Their per-molecule mean across views is

$$\overline{\mathbf{Z}} = \frac{1}{2}\left(\mathbf{Z}^{(1)} + \mathbf{Z}^{(2)}\right). \tag{16}$$

The two-view specialization of the VISReg prediction term is

$$\mathcal{L}_{\mathrm{inv}} = \frac{1}{2BD}\sum_{r=1}^{2}\left\|\mathbf{Z}^{(r)} - \overline{\mathbf{Z}}\right\|_F^2. \tag{17}$$

For each view $r \in \{1, 2\}$, define its batch mean, centered representation, and per-dimension population standard deviation as

$$\begin{aligned}
\mu_d^{(r)} &= \frac{1}{B}\sum_{b=1}^{B} Z_{bd}^{(r)}, \\
\widehat{Z}_{bd}^{(r)} &= Z_{bd}^{(r)} - \mu_d^{(r)}, \\
\sigma_d^{(r)} &= \sqrt{\frac{1}{B}\sum_{b=1}^{B}\left(\widehat{Z}_{bd}^{(r)}\right)^2}, \\
\sigma_{\epsilon,d}^{(r)} &= \max\left(\sigma_d^{(r)}, \epsilon\right).
\end{aligned} \tag{18}$$

The centering and variance terms are

$$\mathcal{L}_{\mathrm{center}} = \frac{1}{2D}\sum_{r=1}^{2}\sum_{d=1}^{D}\left(\mu_d^{(r)}\right)^2, \qquad \mathcal{L}_{\mathrm{scale}} = \frac{1}{2D}\sum_{r=1}^{2}\sum_{d=1}^{D}\left(1 - \sigma_{\epsilon,d}^{(r)}\right)^2. \tag{19}$$

The centered features are then standardized without propagating gradients through the scale,

$$\widetilde{Z}_{bd}^{(r)} = \frac{\widehat{Z}_{bd}^{(r)}}{\mathrm{stopgrad}\left(\sigma_{\epsilon,d}^{(r)}\right)}, \tag{20}$$

where $\epsilon > 0$ is a numerical stabilizer. For J independently sampled unit directions $\mathbf{u}_j \in \mathbb{R}^D$, let sort arrange a vector in ascending order and define the standard-normal quantile vector

$$\mathbf{q}_{\mathcal{N}} = \left[\Phi^{-1}\left(\frac{b}{B+1}\right)\right]_{b=1}^{B}. \tag{21}$$

The sketching term matches each one-dimensional projection to these target quantiles:

$$\mathcal{L}_{\mathrm{shape}} = \frac{1}{2BJ}\sum_{r=1}^{2}\sum_{j=1}^{J}\left\|\mathrm{sort}\left(\widetilde{\mathbf{Z}}^{(r)}\mathbf{u}_j\right) - \mathbf{q}_{\mathcal{N}}\right\|_2^2. \tag{22}$$

Combining the three distributional penalties gives

$$\mathcal{L}_{\mathrm{reg}} = \lambda_{\mathrm{scale}}\mathcal{L}_{\mathrm{scale}} + \lambda_{\mathrm{shape}}\mathcal{L}_{\mathrm{shape}} + \lambda_{\mathrm{center}}\mathcal{L}_{\mathrm{center}}, \qquad \mathcal{L}_{\mathrm{VIS}} = (1-\lambda)\mathcal{L}_{\mathrm{inv}} + \lambda\mathcal{L}_{\mathrm{reg}}. \tag{23}$$

During the energy-model update, VISReg is computed from two rotated views of the real molecules; during the generator update, it is computed from two rotated views of the generated molecules while the energy-model parameters are frozen. The augmented objectives are

$$\mathcal{L}_E^{\mathrm{total}} = \mathcal{L}_E + \eta_E\mathcal{L}_{\mathrm{VIS}}^{\mathrm{real}}, \qquad \mathcal{L}_G^{\mathrm{total}} = \mathcal{L}_G + \eta_G\mathcal{L}_{\mathrm{VIS}}^{\mathrm{fake}}. \tag{24}$$

Overall, VISReg improves the diversity of valid QM9 generations: in our experiments, it increased both uniqueness and the valid-and-unique rate (V×U) by approximately 20%.

5 Experiments

5.1 Experimental Settings

Datasets. We evaluate MolGAN on QM9 (Ramakrishnan et al., 2014) and GEOM-DRUG (Axelrod & Gómez-Bombarelli, 2022). QM9 contains small organic molecules with up to nine heavy atoms, whereas GEOM-DRUG contains larger and more structurally diverse drug-like molecules.

Metrics. Following established 3D molecular-generation evaluations (Hoogeboom et al., 2022; Xu et al., 2023), we report atom stability, molecule stability, validity, and valid-and-unique rate for QM9. For GEOM-DRUG, we report atom stability and validity. We use the number of refinement passes as the reported NFE budget.

Baselines. We compare with published diffusion, flow, bridge, latent-space, and accelerated generators (Hoogeboom et al., 2022; Wu et al., 2022; Xu et al., 2023; Song et al., 2023b; 2024; Hong et al., 2025; Ni et al., 2025; Li et al., 2024; Lacombe & Vaidya, 2024; Zhang et al., 2025; Wei et al., 2026).

5.2 Generation on QM9 and GEOM-DRUG

Table 1 compares MolGAN with previous molecular-generation methods under their reported NFE budgets. MolGAN achieves state-of-the-art results in the few-step regime.

At four NFEs on QM9, MolGAN achieves 98.30% atom stability, 86.44% molecular stability, and 92.52% validity. Reducing the budget to two or three NFEs preserves most of this stability and validity, while the one-step setting exhibits a clear degradation. On GEOM-DRUG, MolGAN reaches 88.84% atom stability and 99.64% validity at four NFEs, with validity remaining above 97% at two and three NFEs. These results demonstrate that MolGAN retains strong molecular quality across short external refinement trajectories.

Table 1: Generation performance on QM9 and GEOM-DRUG across NFE budgets. MolGAN achieves state-of-the-art few-step performance using one to four refinement passes.

Method	NFE	QM9				GEOM-DRUG	
		Atom Sta. (%)	Mol Sta. (%)	Valid (%)	V × U (%)	Atom Sta. (%)	Valid (%)
Data	–	99.0	95.2	97.7	97.7	86.5	99.9
GDM-aug	1000	97.6	71.6	90.4	89.5	77.7	91.8
EDM	1000	98.7	82.0	91.9	90.7	81.3	92.6
EDM-Bridge	1000	98.8	84.6	92.0	90.7	82.4	92.8
GeoLDM	1000	98.9	89.4	93.8	92.7	84.4	99.3
EquiFM	200	98.9	88.3	94.7	93.5	84.1	98.9
GeoBFN	100	98.6	87.2	93.0	91.5	78.9	93.1
GOAT	90	99.2	–	92.9	92.0	84.8	96.2
SLDM	50	98.70	88.09	92.84	89.53	–	–
GeoRCG (EDM)	50	98.75	89.08	95.05	–	81.44	95.70
AccGeoLDM	16	92.08	51.02	73.22	72.39	–	–
MOLTD	12	99.40	92.53	96.04	92.0	86.88	95.33
FlashMol	8	99.55	94.87	96.97	87.51	89.98	99.86
FlashMol	5	98.73	87.05	92.67	78.83	87.69	99.69
FlashMol	4	97.70	83.89	90.70	72.70	84.72	99.72
MolGAN *(ours)*	4	98.30	86.44	92.52	34.71	88.84	99.64
MolGAN *(ours)*	3	98.31	87.03	92.77	35.11	88.20	99.24
MolGAN *(ours)*	2	98.11	85.44	91.59	35.51	85.41	97.94
MolGAN *(ours)*	1	94.80	64.84	78.34	37.17	72.54	83.88

5.3 Sampling Efficiency

We use one NFE to denote one externally visible molecular refinement pass. Because each pass contains multiple internal high- and low-level HRM updates, this convention measures the length of the molecular trajectory rather than the total depth of hierarchical latent computation.

Figure 2 visualizes the trade-off between molecular stability and external network evaluations on QM9 using published baseline results (Wei et al., 2026). MolGAN lies on the quality–efficiency frontier at the smallest sampling budgets, achieving strong molecular stability with only a few external refinement passes. This demonstrates the effectiveness of MolGAN in combining deep hierarchical latent computation with a short and efficient molecular refinement trajectory.

5.4 Test-Time Scaling with Energy-Based Selection

Test-time scaling improves model outputs by allocating additional computation during inference without updating the trained parameters. One common approach generates multiple candidates and uses a learned verifier or scoring model to select the strongest outputs (Snell et al., 2024). MolGAN applies this principle by generating multiple candidate molecules and using the frozen energy model to select a low-energy subset. Because n already denotes the number of high-level HRM cycles, we use v for the candidate-to-output ratio. To retain U outputs, we generate a global pool of Uv candidates,

$$\mathcal{C}_v = \left\{ \mathbf{x}_j^{(K)} \right\}_{j=1}^{Uv}, \qquad \varepsilon_j = E_\phi\left(\mathbf{x}_j^{(K)} \right), \tag{25}$$

using the same K-pass generation schedule for every candidate. If π orders the candidate indices by increasing energy, the selected set is

$$\mathcal{S}_v^{\text{best}} = \left\{ \mathbf{x}_{\pi(j)}^{(K)} \right\}_{j=1}^{U}, \qquad \varepsilon_{\pi(1)} \le \cdots \le \varepsilon_{\pi(Uv)}, \tag{26}$$

where lower energy indicates greater similarity to the real-molecule distribution according to the learned energy model. Selection requires no gradients and does not modify either G_θ or E_ϕ.

We compare best-of-v with a matched baseline that selects U candidates uniformly without replacement from the same pool, so only the selection rule differs.

We additionally evaluate early stop generation, which scores every intermediate state along a trajectory and selects the state immediately preceding the first increase in energy. If the energy does not increase, the final state is retained. This procedure uses the same learned energy model but allocates computation across refinement depth rather than across independent candidates.

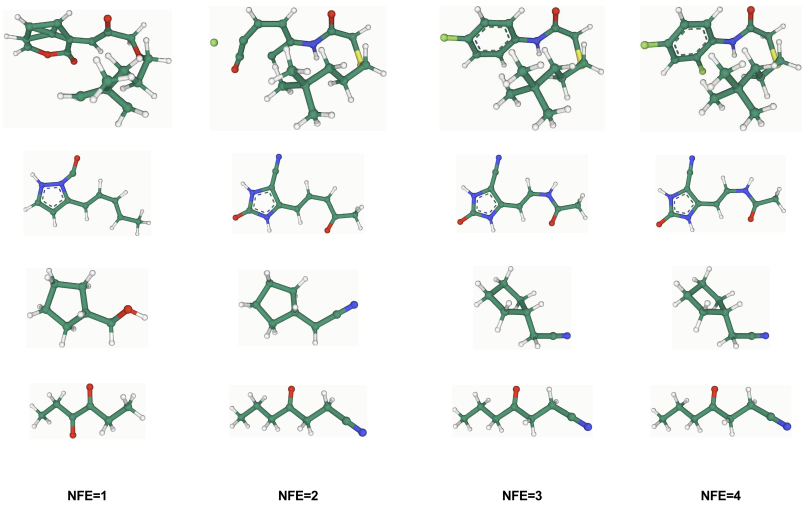

Figure 3: Four-pass MolGAN trajectories. Rows show GEOM-DRUG (top) and QM9 (bottom); columns show NFEs 1–4.

For GEOM-DRUG, Table 2 reports QED (Bickerton et al., 2012), normalized synthetic accessibility (SA) (Ertl & Schuffenhauer, 2009), the mean number of Lipinski rules satisfied (Lipinski et al., 2001), mean LogP (Wildman & Crippen, 1999), and the PAINS alert rate (Baell & Holloway, 2010). All best-of-v results are computed using candidates from the final, K-th NFE step. This comparison tests whether additional sampling computation improves molecular quality or merely concentrates the output distribution. Best-of-v requires vK generator evaluations and v energy evaluations per retained molecule.

Table 2: Test-time scaling on GEOM-DRUG. Early stopping selects an intermediate trajectory state, whereas best-of-v selects the U lowest-energy final states from Uv candidates.

Selection	v	QED ↑	SA ↑	Lipinski ↑	LogP	PAINS (%) ↓
Single sample	1	0.6110	0.7062	4.8145	3.0321	0.59
Early stop	1	0.5960	0.7002	4.7667	3.1481	0.77
Best-of-v	2	0.6345	0.7458	4.9020	2.4738	0.63
	4	0.6522	0.7714	4.9500	2.1913	0.83
	8	0.6627	0.7900	4.9746	2.0208	0.89

Early stop generation chooses states at a mean refinement index of 3.01 while maintaining performance similar to standard final-step generation. This result suggests that an online stopping rule could, in theory, achieve comparable performance with fewer NFEs by terminating trajectories once the energy ceases to improve. The best-of-v results show a clear test-time scaling trend in molecular property quality. Increasing v from 2 to 8 consistently improves QED from 0.6345 to 0.6627, normalized SA from 0.7458 to 0.7900, and the mean Lipinski score from 4.9020 to 4.9746. These gains indicate that discriminator-based selection increasingly favors drug-like and synthetically accessible molecules.

6 Conclusion

We introduced MolGAN, a few-step three-dimensional molecular generator that combines energy-based adversarial training with hierarchical recurrent reasoning. By coordinating slowly updated high-level states with rapid low-level computation, MolGAN retains strong molecular stability and validity across short external trajectories on QM9 and GEOM-DRUG. The learned energy model further supports early stopping and best-of-v selection, providing complementary ways to trade additional test-time computation for improved molecular properties. The results also identify QM9 diversity as an important remaining limitation.

LLM USAGE

Large language models were used to assist with drafting and polishing portions of the manuscript. They also supported grammar correction, improvements to clarity and presentation, and the refinement of LaTeX notation. Research ideation, the design of the methodology, model implementation, the execution of experiments, and the analysis and interpretation of results were developed and conducted by the authors. All technical claims, equations, citations, experimental results, and final wording were reviewed and verified by the authors, who retain full responsibility for the content of the paper.

REFERENCES

Simon Axelrod and Rafael Gómez-Bombarelli. GEOM, energy-annotated molecular conformations for property prediction and molecular generation. *Scientific Data*, 9:185, 2022. doi: 10.1038/s41597-022-01288-4.

Jonathan B. Baell and Georgina A. Holloway. New substructure filters for removal of pan assay interference compounds (PAINS) from screening libraries and for their exclusion in bioassays. *Journal of Medicinal Chemistry*, 53(7):2719–2740, 2010. doi: 10.1021/jm901137j.

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. doi: 10.1038/nchem.1243.

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. doi: 10.1186/1758-2946-1-8.

Jiaqi Guan, Wesley Wei Qian, Xingang Peng, Yufeng Su, Jian Peng, and Jianzhu Ma. 3d equivariant diffusion for target-aware molecule generation and affinity prediction. In *International Conference on Learning Representations*, 2023.

Haokai Hong, Wanyu Lin, and Kay Chen Tan. Accelerating 3d molecule generation via jointly geometric optimal transport. In *International Conference on Learning Representations*, 2025.

Emiel Hoogeboom, Víctor Garcia Satorras, Clément Vignac, and Max Welling. Equivariant diffusion for molecule generation in 3d. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 8867–8887. PMLR, 2022.

Alexia Jolicoeur-Martineau. The relativistic discriminator: A key element missing from standard GAN. In *International Conference on Learning Representations*, 2019.

Romain Lacombe and Neal Vaidya. Accelerating the generation of molecular conformations with progressive distillation of equivariant latent diffusion models. *arXiv preprint arXiv:2404.13491*, 2024.

Zian Li, Cai Zhou, Xiyuan Wang, Xingang Peng, and Muhan Zhang. Geometric representation condition improves equivariant molecule generation. *arXiv preprint arXiv:2410.03655*, 2024.

Christopher A. Lipinski, Franco Lombardo, Beryl W. Dominy, and Paul J. Feeney. Experimental and computational approaches to estimate solubility and permeability in drug discovery and development settings. *Advanced Drug Delivery Reviews*, 46(1–3):3–26, 2001. doi: 10.1016/S0169-409X(00)00129-0.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *International Conference on Learning Representations*, 2023.

Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. DPM-Solver: A fast ODE solver for diffusion probabilistic model sampling in around 10 steps. In *Advances in Neural Information Processing Systems*, volume 35, pp. 5775–5787, 2022.

Yuyan Ni, Shikun Feng, Haohan Chi, Bowen Zheng, Huan-ang Gao, Wei-Ying Ma, Zhi-Ming Ma, and Yanyan Lan. Straight-line diffusion model for efficient 3d molecular generation. *arXiv preprint arXiv:2503.02918*, 2025.

Yanru Qu, Keyue Qiu, Yuxuan Song, Jingjing Gong, Jiawei Han, Mingyue Zheng, Hao Zhou, and Wei-Ying Ma. MolCRAFT: Structure-based drug design in continuous parameter space. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 41749–41768. PMLR, 2024.

Raghunathan Ramakrishnan, Pavlo O. Dral, Matthias Rupp, and O. Anatole von Lilienfeld. Quantum chemistry structures and properties of 134 kilo molecules. *Scientific Data*, 1:140022, 2014. doi: 10.1038/sdata.2014.22.

Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. In *International Conference on Learning Representations*, 2022.

Noam Shazeer. GLU variants improve transformer. *arXiv preprint arXiv:2002.05202*, 2020.

Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling LLM test-time compute optimally can be more effective than scaling model parameters. *arXiv preprint arXiv:2408.03314*, 2024.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021.

Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 32211–32252. PMLR, 2023a.

Yuxuan Song, Jingjing Gong, Minkai Xu, Ziyao Cao, Yanyan Lan, Stefano Ermon, Hao Zhou, and Wei-Ying Ma. Equivariant flow matching with hybrid probability transport for 3d molecule generation. In *Advances in Neural Information Processing Systems*, volume 36, 2023b.

Yuxuan Song, Jingjing Gong, Yanru Qu, Hao Zhou, Mingyue Zheng, Jingjing Liu, and Wei-Ying Ma. Unified generative modeling of 3d molecules via bayesian flow networks. In *International Conference on Learning Representations*, 2024.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Guan Wang, Jin Li, Yuhao Sun, Xing Chen, Changling Liu, Yue Wu, Meng Lu, Sen Song, and Yasin Abbasi Yadkori. Hierarchical reasoning model. *arXiv preprint arXiv:2506.21734*, 2025.

Xinyuan Wei, Zian Li, Shaoheng Yan, Cai Zhou, and Muhan Zhang. FlashMol: High-quality molecule generation in as few as four steps. *arXiv preprint arXiv:2605.07020*, 2026.

Scott A. Wildman and Gordon M. Crippen. Prediction of physicochemical parameters by atomic contributions. *Journal of Chemical Information and Computer Sciences*, 39(5):868–873, 1999. doi: 10.1021/ci9903071.

Haiyu Wu, Randall Balestriero, and Morgan Levine. Visreg: Variance–invariance–sketching regularization for JEPA training. *arXiv preprint arXiv:2606.02572*, 2026.

Lemeng Wu, Chengyue Gong, Xingchao Liu, Mao Ye, and Qiang Liu. Diffusion-based molecule generation with informative prior bridges. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

Minkai Xu, Alexander S. Powers, Ron O. Dror, Stefano Ermon, and Jure Leskovec. Geometric latent diffusion models for 3d molecule generation. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 38592–38610. PMLR, 2023.

Biao Zhang and Rico Sennrich. Root mean square layer normalization. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Zhilong Zhang, Yuxuan Song, Yichun Wang, Jingjing Gong, Hanlin Wu, Dongzhan Zhou, Hao Zhou, and Wei-Ying Ma. Accelerating 3d molecule generative models with trajectory diagnosis. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-1168.