GRAPH FORWARD DISTRIBUTION MATCHING FOR MOLECULAR INVERSE DESIGN

Anonymous authors
Paper under double-blind review

ABSTRACT

Achieving precise control over multiple properties without sacrificing chemical validity remains a central challenge in molecular inverse design. Existing reinforcement learning (RL) methods fine-tune graph diffusion models by treating **reverse** sampling as a sequential policy, using a single terminal reward to optimize hundreds of coupled decisions. They often suffer from instability, validity collapse, and limited property gains. We introduce GRAPHFDM (Graph Forward Distribution Matching), a new online RL paradigm for graph diffusion that performs optimization through the **forward** process. GRAPHFDM uses valid generations to define a reward-tilted target distribution jointly optimized over graph size and molecular structure for each property condition, incorporating reinforcement signals into supervised learning without storing reverse trajectories. We derive the unique optimal target, prove a condition-wise improvement guarantee, and show that the fixed graph-size prior of standard graph diffusion leaves an irreducible matching gap. In multi-conditional polymer and small-molecule generation, GRAPHFDM achieves the lowest MAE on every target property, with reductions of up to 53.0% relative to the strongest baselines and chemical validity above 0.99. It further generalizes to out-of-distribution property combinations.

1 INTRODUCTION

Graph diffusion models have emerged as a powerful approach for molecular inverse design, learning from large-scale unlabeled molecular data to generate chemically valid structures by denoising discrete atom and bond types (Vignac et al., 2023). However, they offer limited controllability over target properties under scarce labeled supervision (Li et al., 2026b; Liu et al., 2026). Practical inverse design in drug and materials discovery requires precise, simultaneous control over task-specific objectives such as synthetic accessibility and gas permeability, while preserving chemical validity.

Reinforcement learning (RL) provides a natural post-training paradigm to improve controllability and exploration (Black et al., 2024). Existing RL approaches fine-tune graph diffusion models by treating reverse sampling as a sequential policy, using a single terminal reward to optimize hundreds of coupled decisions along each sampling trajectory, as shown in Figure 1 (a) (Zhu et al., 2026b). Under multi-property objectives, modifying a functional group, ring, or linker can shift properties simultaneously, and a single inconsistent bond can invalidate the entire structure. These coupled effects make reverse-policy updates prone to disrupting the structural priors learned during pretraining. For example, GDPO (Liu et al., 2024b) and GraphGRPO (Zhu et al., 2026a) drive validity close to zero while providing very limited property improvement in Table 1.

We propose a new online RL framework for graph diffusion named Graph Forward Distribution Matching (**GRAPHFDM**). GRAPHFDM shifts policy optimization from reverse sampling trajectories to the forward process, incorporating terminal rewards into the denoising objective used in pretraining. As illustrated in Figure 1 (b), GRAPHFDM constructs a reward-tilted target by reweighting valid generations and matches it through denoising under the same sample weights.

Standard graph diffusion methods denoise node and edge types at a fixed graph size (the number of active nodes), sampled from a condition-independent prior (Vignac et al., 2023; Liu et al., 2024a). However, molecular size is itself an important conditioning variable that influences property control and chemical validity (Ninniri et al., 2024; 2025). Holding the size marginal fixed leaves an irreducible gap to the target distribution, which we formalize in Proposition 3.2. We introduce *Structural*

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.

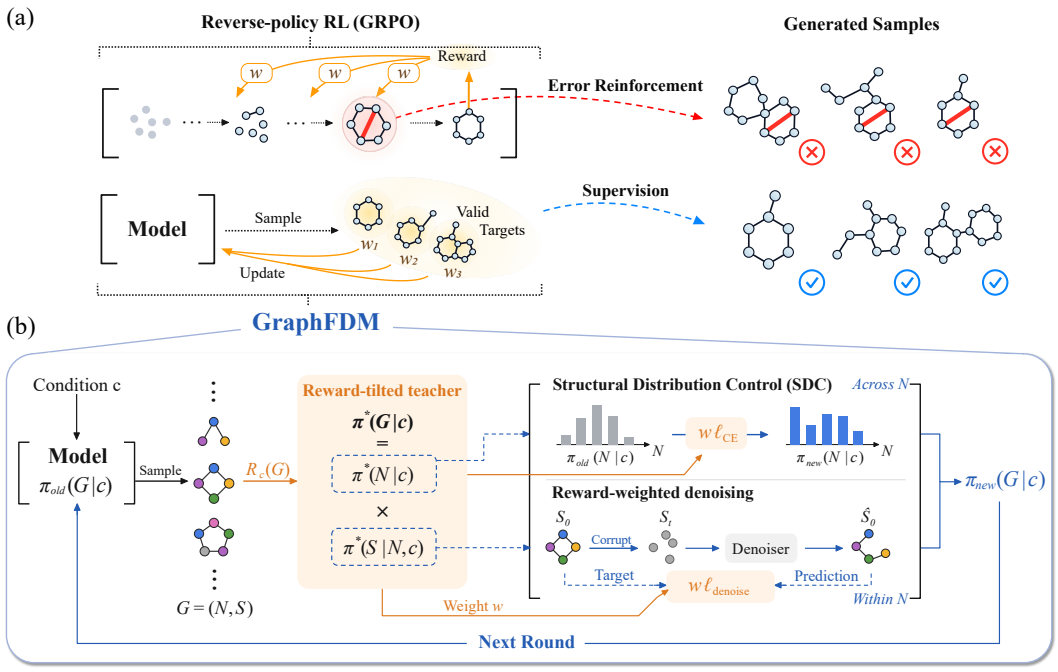

Figure 1: **Overview of GRAPHFDM.** (a) Reverse-policy RL (e.g., GRPO) assigns terminal rewards across coupled reverse transitions which can reinforce structural errors. (b) At each online round, GRAPHFDM constructs a reward-tilted teacher $\pi^*(G \mid c)$ over valid graphs $G = (N, S)$. SDC reallocates probability across the number of active nodes N while reward-weighted denoising refines molecular structures S at each N.

Distribution Control (SDC) to match the complete terminal graph distribution, jointly optimizing the graph-size distribution and the conditional distribution of molecular structures at each size under separate KL constraints.

We derive the unique optimal target distribution and prove its condition-wise improvement guarantee (Section 3.3). We formalize the limitation of a fixed graph-size prior as an irreducible gap and show how SDC overcomes it (Section 3.2). In experiments, across a six-property molecular and a four-property polymer benchmark (Tables 1 and 2), GRAPHFDM outperforms the strongest baseline on every property, reducing MAE by up to 53.0% and maintaining chemical validity above 0.99. We show that GRAPHFDM generalizes to unseen, out-of-distribution property combinations (Figure 2). Together, these results establish GRAPHFDM as a principled and effective post-training framework for controllable molecular inverse design.

2 PRELIMINARIES

Motif-Aware Graph Diffusion Let $\mathbf{c} \in \mathcal{C} \subseteq \mathbb{R}^d$ specify the desired values of d molecular properties. Let $G \in \mathcal{G}$ denote a candidate molecular graph and $\mathcal{G}_{\text{val}} \subseteq \mathcal{G}$ the set of valid molecular graphs. Molecular inverse design seeks a conditional distribution that assigns high probability to valid graphs whose properties match $\mathbf{c}$. We use the motif-aware discrete graph diffusion backbone of CoMole (Zhu et al., 2026b). Under its losslessly decodable representation, a clean molecular graph is written as $G = (N, S)$, with $S = (X, E, P)$, where N denotes the number of active graph units, each represented by either an atom or motif in the vocabulary. X, E, and P denote the node, edge, and attachment-position labels, respectively. We set $S_0 = S$ and represent N by an active-node mask m satisfying $N = \|m\|_1$. The mask remains fixed throughout diffusion. Let $q_t(S_t \mid S_0, m)$ denote the forward corruption distribution at timestep t. Given $(S_t, m, t, \mathbf{c})$, the conditional denoiser predicts the clean structure S_0 with the standard objective

$$\mathcal{L}_{\text{den}}(\theta) = \mathbb{E}_{\substack{(N,S_0,\mathbf{c})\sim p_{\text{data}},\ t\sim\text{Unif}(\{1,\dots,T\}),\\ S_t\sim q_t(\cdot|S_0,m)}} \left[\ell^{\theta}_{\text{den}}(S_0, S_t, m, t, \mathbf{c})\right], \tag{1}$$

where $\ell^{\theta}_{\text{den}}$ is the sum of masked categorical cross-entropies for X, E, and P. During generation, $N \sim p_0(N)$ is sampled from a fixed, condition-independent empirical size prior. Starting from $S_T \sim p_T(\cdot \mid m)$, the reverse kernels $p_\theta(S_{t-1} \mid S_t, m, t, \mathbf{c})$ then induce the clean conditional distribution $\pi_\theta(S \mid N, \mathbf{c})$. Hence,

$$\pi_\theta(G \mid \mathbf{c}) = p_0(N)\, \pi_\theta(S \mid N, \mathbf{c}). \tag{2}$$

Reward-Based Post-Training Reward-based post-training uses feedback from model-generated molecules to improve conditional control. For a target condition $\mathbf{c}$, let $R_{\mathbf{c}} : \mathcal{G}_{\text{val}} \to \mathbb{R}$ denote a terminal reward that assigns larger values to valid molecules whose evaluated properties better match $\mathbf{c}$. Reverse-policy optimization treats the denoising trajectory $S_T \to S_{T-1} \to \cdots \to S_0$ as a finite-horizon policy and propagates the terminal reward through updates proportional to $R_{\mathbf{c}}(G) \sum_{t=1}^{T} \nabla_\theta \log p_\theta(S_{t-1} \mid S_t, m, t, \mathbf{c})$. The same terminal molecular feedback weights the transitions along the sampled reverse trajectory.

3 GRAPH FORWARD DISTRIBUTION MATCHING

At online iteration k, GRAPHFDM turns terminal molecular feedback into a single structured teacher over the clean molecular graph $G = (N, S)$. This teacher is then projected onto the native learning objectives for the structural marginal over N and the conditional molecular distribution over S. The current clean-graph generator factorizes as

$$\pi_k(G \mid \mathbf{c}) = p_k(N \mid \mathbf{c})\, \pi_{\theta_k}(S \mid N, \mathbf{c}), \qquad p_k(\cdot \mid \mathbf{c}) \ll p_0. \tag{3}$$

We initialize $p_k(N \mid \mathbf{c})$ with the empirical prior $p_0(N)$ and subsequently use the learned controller $p_k(N \mid \mathbf{c}) = p_{\phi_k}(N \mid \mathbf{c})$. Here, $\ll$ restricts the controller to the structural support of the empirical prior. Chemically valid graphs define the admissible support of the terminal teacher. We define the valid-conditioned reference as

$$\bar{\pi}_k(G \mid \mathbf{c}) := \frac{\pi_k(G \mid \mathbf{c})\, \mathbf{1}\{G \in \mathcal{G}_{\text{val}}\}}{\pi_k(\mathcal{G}_{\text{val}} \mid \mathbf{c})}, \qquad \pi_k(\mathcal{G}_{\text{val}} \mid \mathbf{c}) > 0. \tag{4}$$

Writing $\bar{\pi}_k(N, S \mid \mathbf{c}) = \bar{p}_k(N \mid \mathbf{c})\, \bar{\pi}_k(S \mid N, \mathbf{c})$, p_k is the structural marginal used to sample N, whereas $\bar{p}_k$ is its valid-conditioned counterpart. They generally differ because molecular validity can depend on N (see Section A.2.1).

For any terminal teacher $\mu_k(\cdot \mid \mathbf{c}) \ll \bar{\pi}_k(\cdot \mid \mathbf{c})$, GRAPHFDM defines the density ratio

$$w^{\mu}_k(G, \mathbf{c}) := \frac{\mu_k(G \mid \mathbf{c})}{\bar{\pi}_k(G \mid \mathbf{c})}. \tag{5}$$

GRAPHFDM then re-corrupts clean graphs sampled from $\bar{\pi}_k$ and uses w^{μ}_k to express the teacher's denoising and structural prediction risks as weighted expectations under the current reference. Here, *forward* refers to constructing supervision by reapplying the fixed forward-corruption process. We first establish this teacher-agnostic projection, then identify the structural-marginal gap of fixed-mask denoising, and finally instantiate $\mu_k = \pi^{\star}_k$ through SDC.

3.1 FORWARD-DENOISING MATCHING

Factor the terminal teacher introduced above as

$$\mu_k(N, S \mid \mathbf{c}) = \mu_{N,k}(N \mid \mathbf{c})\, \mu_{S,k}(S \mid N, \mathbf{c}).$$

GRAPHFDM transports both factors to their respective native learning objectives using the single density ratio w^{μ}_k defined in Eq. (5).

Proposition 3.1 (Forward-denoising matching)**.** *For any target condition* $\mathbf{c}$*, assume that the losses below are integrable under the corresponding teacher-induced distributions. Then the weighted forward-denoising objective satisfies*

$$\begin{aligned} \mathcal{L}^{\mu}_{S,k}(\theta; \mathbf{c}) &:= \mathbb{E}_{\substack{G=(N,S_0)\sim\bar{\pi}_k(\cdot\mid\mathbf{c}),\\ t\sim\text{Unif}(\{1,\dots,T\}),\\ S_t\sim q_t(\cdot\mid S_0,m)}} \left[w^{\mu}_k(G, \mathbf{c}) \ell^{\theta}_{\text{den}}(S_0, S_t, m, t, \mathbf{c}) \right] \\ &= \mathbb{E}_{\substack{G=(N,S_0)\sim\mu_k(\cdot\mid\mathbf{c}),\\ t\sim\text{Unif}(\{1,\dots,T\}),\\ S_t\sim q_t(\cdot\mid S_0,m)}} \left[\ell^{\theta}_{\text{den}}(S_0, S_t, m, t, \mathbf{c}) \right], \end{aligned} \tag{6}$$

where m denotes the fixed structural mask associated with N.

The same density ratio projects the structural marginal:

$$\begin{aligned}\mathcal{L}^{\mu}_{N,k}(\phi;\mathbf{c}) &:= \mathbb{E}_{G\sim\bar{\pi}_k(\cdot|\mathbf{c})}\left[w^{\mu}_k(G,\mathbf{c})\big(-\log p_\phi(N\mid\mathbf{c})\big)\right]\\ &= \mathbb{E}_{N\sim\mu_{N,k}(\cdot|\mathbf{c})}\left[-\log p_\phi(N\mid\mathbf{c})\right].\end{aligned} \tag{7}$$

Eq. (6) is an exact condition-wise change of measure. It transports terminal teacher preferences to every diffusion noise level by reapplying the backbone's fixed forward-corruption process. Eq. (7) transports the marginal of the same teacher to the structural controller.

Let $\mathbb{E}_{\mathbf{c}}$ denote the empirical average over target conditions. Because the denoiser and structural controller share parameters across conditions, their population objectives are

$$\mathcal{L}^{\mu}_{S,k}(\theta) := \mathbb{E}_{\mathbf{c}}\left[\mathcal{L}^{\mu}_{S,k}(\theta;\mathbf{c})\right], \qquad \mathcal{L}^{\mu}_{N,k}(\phi) := \mathbb{E}_{\mathbf{c}}\left[\mathcal{L}^{\mu}_{N,k}(\phi;\mathbf{c})\right].$$

Under simultaneous realizability, the population minimizers recover the teacher-induced denoising posterior marginals at every noise level and the structural marginal $\mu_{N,k}$. This is the precise population-level sense in which GRAPHFDM projects and matches the teacher, see Sections A.2.2 and A.2.3.

3.2 BEYOND FIXED-MASK DENOISING: STRUCTURAL-MARGINAL GAP

In traditional graph diffusion models, the structural variable N is sampled before the reverse process and its mask remains fixed throughout denoising. Any denoiser-only update is restricted to the form $p_k(N\mid\mathbf{c})\,\widetilde{\pi}(S\mid N,\mathbf{c})$, leaving the structural marginal $p_k(N\mid\mathbf{c})$ unchanged. For the terminal teacher introduced above, $\mu_k \ll \bar{\pi}_k \ll \pi_k$, which implies $\mu_{N,k}(\cdot\mid\mathbf{c}) \ll p_k(\cdot\mid\mathbf{c})$.

Proposition 3.2 (Structural-marginal gap)**.** *For any conditional molecular distribution $\widetilde{\pi}(S\mid N,\mathbf{c})$ for which the divergence below is finite, the KL chain rule gives*

$$\begin{aligned}&D_{\mathrm{KL}}\left(\mu_k(\cdot\mid\mathbf{c})\|p_k(N\mid\mathbf{c})\,\widetilde{\pi}(S\mid N,\mathbf{c})\right)\\ &= D_{\mathrm{KL}}\left(\mu_{N,k}(\cdot\mid\mathbf{c})\|p_k(\cdot\mid\mathbf{c})\right)\\ &+ \mathbb{E}_{N\sim\mu_{N,k}(\cdot|\mathbf{c})} D_{\mathrm{KL}}\left(\mu_{S,k}(\cdot\mid N,\mathbf{c})\|\widetilde{\pi}(\cdot\mid N,\mathbf{c})\right).\end{aligned} \tag{8}$$

Consequently,

$$\inf_{\widetilde{\pi}(S|N,\mathbf{c})} D_{\mathrm{KL}}\left(\mu_k(\cdot\mid\mathbf{c})\|p_k(N\mid\mathbf{c})\,\widetilde{\pi}(S\mid N,\mathbf{c})\right) = D_{\mathrm{KL}}\left(\mu_{N,k}(\cdot\mid\mathbf{c})\|p_k(\cdot\mid\mathbf{c})\right), \tag{9}$$

where the infimum is attained by $\widetilde{\pi}(\cdot\mid N,\mathbf{c}) = \mu_{S,k}(\cdot\mid N,\mathbf{c})$ for $\mu_{N,k}(\cdot\mid\mathbf{c})$-almost every N.

Thus, even an oracle denoiser that exactly recovers the teacher conditional $\mu_{S,k}$ leaves a nonzero mismatch whenever $\mu_{N,k}(\cdot\mid\mathbf{c}) \neq p_k(\cdot\mid\mathbf{c})$. This result motivates the structural projection in Eq. (7): forward denoising learns teacher-aligned molecular structures within each N. SDC, introduced next, unlocks reward-driven probability allocation across N. The proof is provided in Section A.2.4.

3.3 CLOSING THE GAP: STRUCTURAL DISTRIBUTION CONTROL

Structural Distribution Control (SDC) addresses the gap established in Prop. 3.2 by making the active-node-count marginal part of the reward-based graph target. It separately controls teacher movement across N and conditional molecular refinement within each N.

Let $\rho(N,S\mid\mathbf{c}) = \rho_N(N\mid\mathbf{c})\,\rho_S(S\mid N,\mathbf{c})$ be a candidate graph distribution satisfying $\rho(\cdot\mid\mathbf{c}) \ll \bar{\pi}_k(\cdot\mid\mathbf{c})$ for every target condition. Define

$$\begin{aligned}\mathcal{C}_N(\rho) &:= \mathbb{E}_{\mathbf{c}} D_{\mathrm{KL}}\left(\rho_N(\cdot\mid\mathbf{c})\|\bar{p}_k(\cdot\mid\mathbf{c})\right),\\ \mathcal{C}_S(\rho) &:= \mathbb{E}_{\mathbf{c}}\mathbb{E}_{N\sim\rho_N(\cdot|\mathbf{c})} D_{\mathrm{KL}}\left(\rho_S(\cdot\mid N,\mathbf{c})\|\bar{\pi}_k(\cdot\mid N,\mathbf{c})\right).\end{aligned} \tag{10}$$

By the KL chain rule,

$$\mathbb{E}_{\mathbf{c}} D_{\mathrm{KL}}\left(\rho(\cdot\mid\mathbf{c})\|\bar{\pi}_k(\cdot\mid\mathbf{c})\right) = \mathcal{C}_N(\rho) + \mathcal{C}_S(\rho). \tag{11}$$

For budgets $\epsilon_N, \epsilon_S > 0$, SDC instantiates the generic GRAPHFDM teacher as the solution of

$$\pi_k^\star \in \arg\max_{\rho \ll \bar{\pi}_k} \quad \mathbb{E}_{\mathbf{c}} \mathbb{E}_{G\sim\rho(\cdot|\mathbf{c})} \left[R_{\mathbf{c}}(G)\right] \quad \text{subject to} \qquad \mathcal{C}_N(\rho) \le \epsilon_N, \quad \mathcal{C}_S(\rho) \le \epsilon_S. \tag{12}$$

The two budgets separately constrain reward-driven reallocation across N and conditional refinement within each N. For $\tau_S > 0$ and $n \in \operatorname{supp} \bar{p}_k(\cdot \mid \mathbf{c})$, define the conditional partition function

$$Z_{S,k}(n, \mathbf{c}; \tau_S) := \mathbb{E}_{S\sim\bar{\pi}_k(\cdot|n,\mathbf{c})} \left[\exp\left(\frac{R_{\mathbf{c}}(n, S)}{\tau_S}\right)\right]. \tag{13}$$

Theorem 3.3 (Unique structured reward-tilted teacher)**.** *Assume that the relevant graph and target-condition supports are finite and that both constraints in Eq.* (12) *are active at the optimum, with positive optimal dual variables τ_N, τ_S. Then the optimizer is unique and factorizes, for every target condition, as*

$$\pi_k^\star(G \mid \mathbf{c}) = \pi_k^\star(N \mid \mathbf{c})\, \pi_k^\star(S \mid N, \mathbf{c}),$$

where

$$\pi_k^\star(S \mid n, \mathbf{c}) = \frac{\bar{\pi}_k(S \mid n, \mathbf{c}) \exp\left(R_{\mathbf{c}}(n, S)/\tau_S\right)}{Z_{S,k}(n, \mathbf{c}; \tau_S)}, \tag{14}$$

and

$$\pi_k^\star(N = n \mid \mathbf{c}) = \frac{\bar{p}_k(n \mid \mathbf{c}) Z_{S,k}(n, \mathbf{c}; \tau_S)^{\tau_S/\tau_N}}{Z_{N,k}(\mathbf{c}; \tau_N, \tau_S)}. \tag{15}$$

Here,

$$Z_{N,k}(\mathbf{c}; \tau_N, \tau_S) := \sum_{n \in \operatorname{supp} \bar{p}_k(\cdot|\mathbf{c})} \bar{p}_k(n \mid \mathbf{c}) Z_{S,k}(n, \mathbf{c}; \tau_S)^{\tau_S/\tau_N} \tag{16}$$

normalizes the structural marginal.

The conditional factor in Eq. (14) tilts molecular realizations within each N, whereas Eq. (15) reallocates the valid-conditioned reference mass $\bar{p}_k(n \mid \mathbf{c})$ according to the aggregate reward desirability $Z_{S,k}(n, \mathbf{c}; \tau_S)^{\tau_S/\tau_N}$. The temperature τ_S controls reward concentration within each N, while τ_S/τ_N directly controls the strength of cross-N reallocation. The proof is provided in Section A.2.5.

SDC projection weight. Substituting $\mu_k = \pi_k^\star$ into Eq. (5) gives

$$w_k(G, \mathbf{c}) := w_k^{\pi_k^\star}(G, \mathbf{c}) = \frac{\pi_k^\star(G \mid \mathbf{c})}{\bar{\pi}_k(G \mid \mathbf{c})} = \frac{\exp\left(R_{\mathbf{c}}(G)/\tau_S\right) Z_{S,k}(N, \mathbf{c}; \tau_S)^{\tau_S/\tau_N - 1}}{Z_{N,k}(\mathbf{c}; \tau_N, \tau_S)}. \tag{17}$$

By Prop. 3.1, the same weight projects the SDC teacher onto both native learning objectives. We henceforth write

$$\mathcal{L}_{S,k} := \mathcal{L}_{S,k}^{\pi_k^\star}, \qquad \mathcal{L}_{N,k} := \mathcal{L}_{N,k}^{\pi_k^\star}.$$

Corollary 3.4 (Condition-wise teacher improvement)**.** *Under the assumptions of theorem 3.3, define*

$$\Delta_k(\mathbf{c}) := \mathbb{E}_{G\sim\pi_k^\star(\cdot|\mathbf{c})} \left[R_{\mathbf{c}}(G)\right] - \mathbb{E}_{G\sim\bar{\pi}_k(\cdot|\mathbf{c})} \left[R_{\mathbf{c}}(G)\right].$$

Then, for every target condition $\mathbf{c}$*,*

$$\Delta_k(\mathbf{c}) \ge \tau_N D_{\mathrm{KL}}\left(\pi_k^\star(N \mid \mathbf{c}) \| \bar{p}_k(N \mid \mathbf{c})\right) + \tau_S \mathbb{E}_{N\sim\pi_k^\star(N|\mathbf{c})} D_{\mathrm{KL}}\left(\pi_k^\star(S \mid N, \mathbf{c}) \| \bar{\pi}_k(S \mid N, \mathbf{c})\right) \ge 0, \tag{18}$$

with strict reward improvement whenever $\pi_k^\star(\cdot \mid \mathbf{c}) \ne \bar{\pi}_k(\cdot \mid \mathbf{c})$. Moreover, because both constraints are active,

$$\mathbb{E}_{\mathbf{c}}\left[\Delta_k(\mathbf{c})\right] \ge \tau_N \epsilon_N + \tau_S \epsilon_S \ge 0. \tag{19}$$

Since $\pi_k^\star \ll \bar{\pi}_k$, the exact teacher also satisfies $\pi_k^\star(\mathcal{G}_{\mathrm{val}} \mid \mathbf{c}) = 1$ for every target condition. These reward and support guarantees apply to the exact teacher. Its population projections are characterized by Prop. 3.1. The proof and the corresponding improvement identity are provided in Section A.2.6.

Shared-temperature special case. If $\tau_N = \tau_S = \tau$, multiplying Eqs. (14) and (15) recovers the standard joint reward tilt:

$$\pi_k^\star(G \mid \mathbf{c}) = \frac{\bar{\pi}_k(G \mid \mathbf{c}) \exp\left(R_{\mathbf{c}}(G)/\tau\right)}{\mathbb{E}_{G'\sim\bar{\pi}_k(\cdot|\mathbf{c})} \left[\exp\left(R_{\mathbf{c}}(G')/\tau\right)\right]}. \tag{20}$$

Thus, ordinary reward tilting is the shared-temperature special case of SDC.

Online update. At iteration k, GRAPHFDM samples candidate graphs from π_k, retains chemically valid candidates, evaluates their terminal rewards, and estimates the SDC teacher weights w_k. It then updates θ and ϕ using empirical estimates of $\mathcal{L}_{S,k}$ and $\mathcal{L}_{N,k}$, respectively. The resulting generator factorizes as

$$\pi_{k+1}(G \mid \mathbf{c}) = p_{\phi_{k+1}}(N \mid \mathbf{c})\, \pi_{\theta_{k+1}}(S \mid N, \mathbf{c}),$$

and supplies the candidate distribution for the next iteration. Finite-sample normalization, proposal correction, and dual estimation are described in Section A.2.7.

4 EXPERIMENTS

RQ1: We evaluate the controllable generation performance of GRAPHFDM against molecular optimization, graph generation, and diffusion post-training baselines in Section 4.2. **RQ2**: We examine its out-of-distribution generalization and structural KL allocation in Section 4.3.

4.1 EXPERIMENTAL SETUP

We evaluate GRAPHFDM on two multi-property inverse-design benchmarks spanning small molecules and polymers. We additionally report **SFT (Shared Init.)**, the common supervised fine-tuned checkpoint used to initialize CoMole (Zhu et al., 2026b) and both GRAPHFDM variants. **GRAPHFDM$_{\text{w/o SDC}}$** uses the same forward-denoising updates as GRAPHFDM, but without SDC, isolating the contribution of structural distribution control. Model performance is validated across up to eleven metrics, including validity, distribution coverage, and conditional accuracy. Full dataset statistics and implementation details are provided in Sections A.1 and A.3.

Datasets For materials, we adopt the polymer dataset introduced in Liu et al. (2024a), with four target properties: three gas permeabilities (O_2Perm, CO_2Perm, N_2Perm) and synthetic accessibility score (SAscore) (Ertl & Schuffenhauer, 2009). For molecules, we use QM9 (Ramakrishnan et al., 2014) and consider six target properties: dipole moment (μ), isotropic polarizability (α), the energies of the highest occupied and lowest unoccupied molecular orbitals ($\varepsilon_{\mathrm{HOMO}}$ and $\varepsilon_{\mathrm{LUMO}}$), heat capacity at constant volume (C_v), and SAscore.

Evaluation We use a fixed 6:2:2 split of each conditional benchmark into train/val/test sets. Evaluations are conducted on 10,000 generated examples and report: (1) post-processed validity (*Valid.*), with raw validity measured before any rule checking or repair shown in parentheses; (2) internal diversity (Div.); (3) fragment-based similarity with the reference set (Sim.); (4) Fréchet ChemNet Distance with the reference set (Dis.) (Preuer et al., 2018); and MAE between conditioning targets and evaluated properties: (5) synthetic accessibility (Synth.); (6)–(8)/(10) task-specific properties (Property). The oracle is a random forest trained on all task-related molecules (Gao et al., 2022), while SAscore is computed directly from molecular structure. Oracle fitting accuracy and robustness to alternative evaluators are analyzed in Section A.3.3.

Baselines We compare against a broad set of baselines spanning molecular optimization (GraphGA (Jensen, 2019), LSTM-HC (Brown et al., 2019), MARS (Xie et al., 2021)), diffusion-based generators (DiGress (Vignac et al., 2023), GDSS (Jo et al., 2022), GraphDiT (Liu et al., 2024a), DeFoG (Qin et al., 2024), CSGD (Qiao et al., 2025), MELD (Seo et al., 2026)), and methods with post-training (GDPO (Liu et al., 2024b), GraphGRPO (Zhu et al., 2026a), VIDD (Su et al., 2026), CoMole (Zhu et al., 2026b)) and an inference-time method (TreeDiff (Zhao et al., 2026)).

4.2 RQ1: CONDITIONAL MOLECULAR GENERATION

As shown in Tables 1 and 2, GRAPHFDM **achieves the best controllability on all target properties and maintains validity over 0.99**.

Chemical Validity. Rule-filtered validity can obscure the intrinsic feasibility of generated structures. For example, in Table 1, GDSS and VIDD both report 1.0 validity, yet their raw validity collapses to 0.003 and 0.008, respectively. The post-training baselines GDPO and GraphGRPO remain weak under both evaluations, attaining only $0.354(0.077)$ and $0.251(0.020)$ validity. In contrast,

Table 1: **Multi-Conditional Generation of 10K Polymers:** Results on four properties (synthetic score, gas permeability for O_2, N_2, CO_2). MAE is computed between the input conditions and evaluated properties of the generated polymers. Best results are in **red**.

Model	Validity	Distribution Learning			Condition Control				
	Valid. (raw) ↑	Div. ↑	Sim. ↑	Dis. ↓	Synth. ↓	O_2Perm ↓	N_2Perm ↓	CO_2Perm ↓	Avg. MAE ↓
GraphGA	**1.000** (N.A.)	0.883	0.927	9.188	1.331	1.984	2.290	1.949	1.888
LSTM-HC	0.991 (N.A.)	0.894	0.932	12.122	1.413	1.381	1.641	1.356	1.448
MARS	**1.000** (N.A.)	0.838	0.928	7.562	1.166	1.576	1.833	1.607	1.546
DiGress	0.839 (0.117)	0.918	0.515	20.256	2.104	1.543	1.861	1.477	1.746
GDSS	**1.000** (0.003)	0.857	0.004	37.627	1.164	1.260	1.477	1.285	1.297
DeFoG	0.326 (0.003)	0.934	0.168	24.859	1.548	1.178	1.366	1.186	1.320
GraphDiT	0.826 (0.855)	0.890	0.891	6.643	1.297	0.763	0.891	0.771	0.931
MELD	0.135 (0.042)	0.897	0.406	28.876	1.768	1.305	1.517	1.280	1.468
CSGD	0.282 (0.212)	0.862	0.651	20.397	1.171	1.154	1.342	1.148	1.204
TreeDiff	0.985 (0.005)	**0.935**	0.072	30.719	2.411	1.232	1.476	1.221	1.585
VIDD	**1.000** (0.008)	0.920	0.284	26.819	2.291	1.513	1.802	1.461	1.767
GDPO	0.354 (0.077)	0.921	0.489	24.422	1.424	1.357	1.606	1.325	1.428
GraphGRPO	0.251 (0.020)	0.871	0.347	26.221	2.972	1.818	2.229	1.730	2.187
CoMole	0.918 (0.838)	0.845	**0.944**	9.410	0.389	0.816	0.946	0.819	0.743
SFT (Shared Init.)	0.921 (0.898)	0.863	0.930	**6.328**	0.370	0.901	1.013	0.899	0.796
GRAPHFDM$_{\text{w/o SDC}}$ (ours)	0.946 (0.909)	0.845	0.902	9.427	0.231	0.668	0.766	0.691	0.589
GRAPHFDM (ours)	0.991 (**0.986**)	0.834	0.923	9.642	**0.183**	**0.594**	**0.664**	**0.597**	**0.509**

Table 2: **Multi-Conditional Generation of 10K QM9 Molecules:** Results on six properties (synthetic score and five QM9 properties). MAE is computed between the input conditions and evaluated properties of the generated molecules. Overall Rank aggregates property-wise MAE ranks across units and scales. Best results are in **red**.

Model	Validity	Distribution Learning			Condition Control						
	Valid. (raw) ↑	Div. ↑	Sim. ↑	Dis. ↓	Synth. ↓	μ ↓	α ↓	$\varepsilon_{\text{HOMO}}$ ↓	$\varepsilon_{\text{LUMO}}$ ↓	C_v ↓	Overall Rank ↓
GraphGA	**1.000** (N.A.)	0.927	0.922	0.777	0.946	0.702	4.903	0.019	0.020	2.121	8
LSTM-HC	0.990 (N.A.)	0.916	**0.957**	1.400	1.016	1.478	8.489	0.022	0.052	4.300	16
MARS	**1.000** (N.A.)	0.931	0.770	5.731	1.037	0.844	7.597	0.014	0.021	2.961	10
DiGress	0.915 (0.849)	0.918	0.941	4.696	1.612	0.912	3.897	0.012	0.018	1.900	7
GDSS	**1.000** (0.737)	0.908	0.733	6.559	1.340	1.804	10.371	0.021	0.052	3.586	16
DeFoG	0.903 (0.578)	**0.932**	0.890	1.521	0.823	1.038	7.319	0.016	0.024	3.762	12
GraphDiT	0.913 (0.720)	0.924	0.875	2.407	0.701	1.063	5.306	0.015	0.025	2.529	9
MELD	0.823 (0.645)	0.916	0.905	1.377	0.893	1.152	7.077	0.020	0.030	3.112	13
CSGD	0.971 (0.875)	0.907	0.936	2.147	0.823	1.119	5.664	0.017	0.031	2.647	11
TreeDiff	0.974 (0.088)	0.911	0.662	4.394	1.264	1.477	5.480	0.019	0.054	4.252	14
VIDD	**1.000** (0.550)	0.921	0.790	6.157	1.804	1.395	8.223	0.020	0.057	3.700	15
GDPO	0.910 (0.860)	0.917	0.947	0.710	0.542	0.908	3.748	0.012	0.018	1.886	4
GraphGRPO	0.976 (0.841)	0.911	0.904	3.114	0.572	0.862	3.528	0.011	0.018	1.992	3
CoMole	0.962 (0.866)	0.916	0.927	4.231	0.477	0.882	4.322	0.012	0.021	1.929	5
SFT (Shared Init.)	0.895 (0.753)	0.917	0.942	**0.550**	0.537	0.866	4.570	0.013	0.022	2.073	6
GRAPHFDM$_{\text{w/o SDC}}$ (ours)	0.973 (0.921)	0.917	0.907	0.689	0.400	0.782	3.411	0.010	0.015	1.622	2
GRAPHFDM (ours)	0.994 (**0.960**)	0.917	0.902	0.615	**0.359**	**0.697**	**3.307**	**0.009**	**0.015**	**1.473**	**1**

GRAPHFDM$_{\text{w/o SDC}}$ achieves raw validity of 0.909 on Gas and 0.921 on QM9, which GRAPHFDM further improves to 0.986 and 0.960. Forward distribution matching preserves intrinsic chemical feasibility under stronger property optimization, with SDC providing a further consistent gain.

Distribution Learning. A central concern in property optimization is whether improved conditional accuracy comes at the cost of drifting away from the learned data distribution. On the polymer benchmark, SFT (Shared Init.) and GraphDiT achieve low distribution distances of 6.328 and 6.643, while several optimization baselines exhibit substantial distribution shift, with fragment similarity below 0.5 and distribution distance above 20. GRAPHFDM maintains high fragment similarity (0.923) with a moderate distribution distance (9.642). On QM9, GRAPHFDM retains a low distribution distance of 0.615, close to the shared SFT initialization (0.550), while maintaining high diversity (0.917). These results suggest that GRAPHFDM improves controllability without severe distributional drift or mode collapse.

Condition Controllability. GRAPHFDM provides the strongest overall condition control on both benchmarks. In Table 1, starting from the shared SFT initialization (0.796), GRAPHFDM$_{\text{w/o SDC}}$ reduces Avg. MAE to 0.589, while GRAPHFDM further reduces it to 0.509, outperforming the strongest baseline CoMole (0.743) by 31.5%. GRAPHFDM achieves the lowest MAE for all four properties. In Table 2, the same progression improves Overall Rank from 6 for SFT to 2 and 1 for GRAPHFDM$_{\text{w/o SDC}}$ and GRAPHFDM, ahead of GraphGRPO and GDPO. GRAPHFDM obtains the

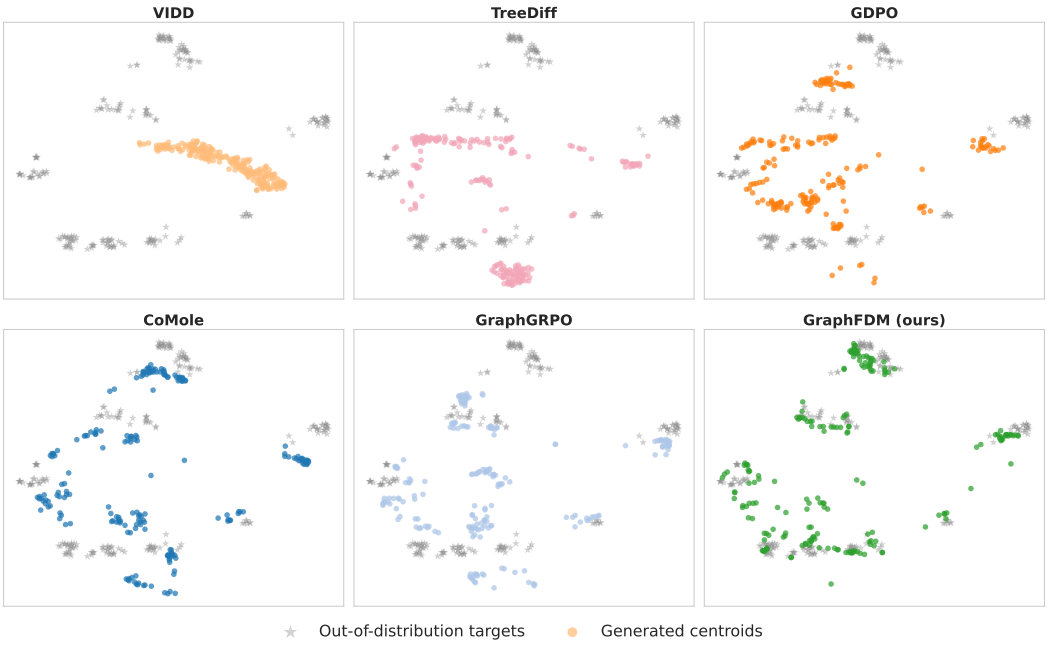

Figure 2: **Out-of-distribution multi-property alignment.** Joint t-SNE visualization of normalized target vectors (gray stars) and per-target centroids of valid generated molecules (colored circles). GRAPHFDM more consistently follows the target modes than competing post-training methods.

Table 3: **Out-of-distribution Multi-Conditional Generation of 10K QM9 Molecules:** We evaluate post-training methods on 200 unseen targets with rare property combinations. Best results are in **red**.

Model	Validity	Distribution Learning			Condition Control					
	Valid. (raw) ↑	Div. ↑	Sim. ↑	Dis. ↓	Synth. ↓	$\mu\downarrow$	$\alpha\downarrow$	$\varepsilon_{\text{HOMO}}\downarrow$	$\varepsilon_{\text{LUMO}}\downarrow$	$C_v\downarrow$
VIDD	**1.000** (0.535)	0.923	0.664	10.985	2.002	1.262	10.708	0.032	0.059	4.426
TreeDiff	0.944 (0.038)	**0.934**	0.687	5.488	0.996	1.229	9.879	0.021	0.031	3.372
GDPO	0.804 (0.717)	0.931	**0.829**	3.496	0.574	1.227	6.875	0.019	0.030	2.438
GraphGRPO	0.955 (0.705)	0.927	0.787	9.873	0.692	1.163	7.127	0.029	0.026	2.681
CoMole	0.875 (0.716)	0.926	0.744	**3.207**	0.642	1.210	6.688	0.019	0.030	2.792
GRAPHFDM (ours)	0.973 (**0.892**)	0.920	0.701	3.349	**0.449**	**1.059**	**5.275**	**0.015**	**0.023**	**1.870**

best MAE on all six properties, including SAS (0.359 vs. CoMole's 0.477), α (3.307 vs. GraphGRPO's 3.528), and C_v (1.473 vs. GDPO's 1.886). Together, these results isolate the gains from forward distribution matching and the additional contribution of SDC. SDC further improves all four polymer errors and five of six QM9 errors while matching the remaining one.

4.3 RQ2: Ablation Studies and Model Analysis

Generalization. We evaluate compositional out-of-distribution generalization on 200 unseen QM9 property targets. We partition each of the five quantum properties into low, medium, and high bins using training-set quantiles. We retain joint bin combinations occurring in fewer than 0.1% of training molecules and select the eight rarest eligible signatures. We randomly sample 25 targets from each combination and include SAScore as the sixth condition (see Section A.1). Without further training, GRAPHFDM achieves the lowest MAE on every property while maintaining high raw validity (Table 3). Figure 2 jointly embeds target vectors and per-target centroids of generated properties. GRAPHFDM more consistently follows the target modes, whereas competing methods exhibit larger deviations or incomplete coverage of the requested property space. These results suggest that GRAPHFDM can compose learned property controls to generalize to rare or unseen joint property combinations.

Table 4: **KL budget allocation in SDC:** $\mathcal{C}_N : \mathcal{C}_S$ denotes the ratio of structural-marginal to conditional-structure KL change. Avg. Gas averages the three gas-permeability MAEs. Best results are in **bold**.

Constraint	$\mathcal{C}_N : \mathcal{C}_S$	Validity	Distribution		Condition Control		
		Valid. (raw) ↑	Div. ↑	Dis. ↓	Synth. ↓	Avg. Gas ↓	Avg. MAE↓
Joint KL	Adaptive (≈ 2.5:1)	0.988 (0.979)	0.834	**8.764**	0.184	0.725	0.590
Separate 1:1	1:1	**0.991 (0.986)**	0.834	9.642	**0.183**	**0.618**	**0.509**
Separate 1:2	1:2	0.986 (0.965)	**0.840**	9.418	0.189	0.648	0.534

KL Budget Allocation in SDC. Following Eqs. (10) and (11), we study how to divide a fixed KL budget between graph-size reallocation and molecular refinement at each size. Here, $\mathcal{C}_N : \mathcal{C}_S$ denotes the ratio of KL changes in the structural marginal over graph sizes and the conditional molecular distributions. We compare a joint constraint that allows an adaptive split with separate constraints assigning budgets in ratios of 1:1 and 1:2. All variants use the same total KL radius and training configuration. As shown in Table 4, the joint constraint yields $\mathcal{C}_N : \mathcal{C}_S \approx 2.5$:1, allocating more distributional change to the structural marginal. Balancing the allocation at 1:1 reduces Avg. MAE from 0.590 to 0.509 (13.7%), with slightly improved validity and Avg. Gas MAE dropping from 0.725 to 0.618. The 1:2 allocation also reduces Avg. MAE relative to the joint constraint but is less effective than 1:1. We therefore use the 1:1 allocation in our main experiments.

5 Related Work

Molecular Graph Generation. Molecular inverse design has been approached through evolutionary search, autoregressive generation, and latent-space optimization (Jensen, 2019; Xie et al., 2021; Brown et al., 2019; Jin et al., 2018). Discrete graph diffusion models generate categorical node and edge variables (Vignac et al., 2023). GraphDiT (Liu et al., 2024a) incorporates multiple property conditions into graph denoising, while GrIDDD (Ninniri et al., 2025) supports node insertion and deletion during diffusion. DeFoG (Qin et al., 2024) formulates graph generation through discrete flow matching. Reward-guided post-training adapts pretrained graph generators to task-specific objectives. GDPO (Liu et al., 2024b) and CoMole (Zhu et al., 2026b) optimize reverse policies, with CoMole operating in a motif-aware graph space. GraphGRPO (Zhu et al., 2026a) applies group-relative policy optimization through analytic graph-flow transitions.

Diffusion Post-Training. Recent diffusion alignment methods incorporate terminal feedback at different levels of the generative process. For discrete sequence models, DDPP (Rector-Brooks et al., 2024) approximates reward-conditioned posteriors across masking levels, while DMPO (Zhu et al., 2025) formulates diffusion-language reasoning as policy-distribution matching. For continuous visual generators, DiffusionNFT (Zheng et al., 2026) contrasts positive and negative generations through forward flow matching, and AWM (Xue et al., 2025) derives a policy-gradient-consistent surrogate from score- or flow-matching losses. TMPO (Li et al., 2026a) matches sampled reverse-trajectory probabilities to a reward-induced Boltzmann distribution. GraphFDM targets categorical graphs and jointly matches the graph-size marginal and conditional molecular distributions of a reward-tilted target under separate KL constraints.

6 Conclusion

We introduced GraphFDM, a forward distribution matching framework for reward-based post-training of molecular graph diffusion models. GraphFDM matches a target distribution obtained by reward reweighting of valid terminal graphs, using the same sample weights in the denoising objective. SDC jointly optimizes the structural marginal over graph sizes and the conditional molecular distributions at each size. We derive the unique optimal teacher, prove condition-wise policy improvement, and establish an irreducible matching gap under a fixed graph-size prior. Across molecular and polymer benchmarks, GraphFDM achieves the strongest multi-property control with high chemical validity and generalizes to unseen targets from rare property combinations, demonstrating forward distribution matching as an effective alternative to reverse-policy optimization.

AI USE STATEMENT

We used generative AI tools to assist with literature search, checking mathematical proofs, and reviewing manuscript drafts for clarity and consistency. The authors reviewed the AI-assisted content and take responsibility for the final manuscript.

REPRODUCIBILITY STATEMENT

Section 3 specifies the training objectives, online optimization procedure, and Structural Distribution Control. Sections A.1 to A.3 provide complete theory proofs, together with dataset preparation and splits, property oracles, reward definitions, evaluation protocols, and training configurations. We provide reproducible code, including the tokenizer and relevant data, in the supplementary materials. We will publicly release the source code and the full set of trained model checkpoints to further support reproducibility.

ETHICS STATEMENT

GRAPHFDM supports molecular discovery through property-controlled generation. Computational property evaluations and chemical validity checks do not establish synthesizability, experimental performance, or safety. Generated candidates require independent validation before practical use. Molecular generation also carries potential dual-use risks, motivating appropriate screening for hazardous compounds in downstream applications.

REFERENCES

Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. Optuna: A next-generation hyperparameter optimization framework, 2019. URL https://arxiv.org/abs/1907.10902.

Kevin Black, Michael Janner, Yilun Du, Ilya Kostrikov, and Sergey Levine. Training diffusion models with reinforcement learning, 2024. URL https://arxiv.org/abs/2305.13301.

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.

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.

Wenhao Gao, Tianfan Fu, Jimeng Sun, and Connor Coley. Sample efficiency matters: a benchmark for practical molecular optimization. *Advances in neural information processing systems*, 35: 21342–21357, 2022.

Jan H Jensen. A graph-based genetic algorithm and generative model/monte carlo tree search for the exploration of chemical space. *Chemical science*, 10(12):3567–3572, 2019.

Jirui Jin, Cheng Zeng, Pawan Prakash, Ellad B. Tadmor, Adrian Roitberg, Richard G. Hennig, Stefano Martiniani, and Mingjie Liu. Molguidance: Advanced guidance strategies for conditional molecular generation with flow matching, 2025. URL https://arxiv.org/abs/2512.12198.

Wengong Jin, Regina Barzilay, and Tommi Jaakkola. Junction tree variational autoencoder for molecular graph generation. In *International conference on machine learning*, pp. 2323–2332. PMLR, 2018.

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*, pp. 10362–10383. PMLR, 2022.

Jiaming Li, Chenyu Zhu, Nanxi Yi, Youjun Bao, Li Sun, Quanying Lv, Xiang Fang, Daizong Liu, Jianjun Li, Kun He, Bowen Zhou, and Zhiyuan Ma. Tmpo: Trajectory matching policy optimization for diverse and efficient diffusion alignment, 2026a. URL https://arxiv.org/abs/2605.10983.

Tianxiao Li, Haoran Liu, Hongyu Guo, Mark Gerstein, and Martin Renqiang Min. Disentangled autoencoding equivariant diffusion model for controlled generation of 3d molecules. *Nature Communications*, 17:4824, 2026b. doi: 10.1038/s41467-026-71441-9.

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, 2024a.

Gang Liu, Jie Chen, Yihan Zhu, Michael Sun, Tengfei Luo, Nitesh Chawla, and Meng Jiang. Graph diffusion transformers are in-context molecular designers. In *International Conference on Learning Representations*, volume 2026, pp. 101812–101846, 2026.

Yijing Liu, Chao Du, Tianyu Pang, Chongxuan Li, Min Lin, and Wei Chen. Graph diffusion policy optimization, 2024b. URL https://arxiv.org/abs/2402.16302.

Matteo Ninniri, Marco Podda, and Davide Bacciu. Classifier-free graph diffusion for molecular property targeting, 2024. URL https://arxiv.org/abs/2312.17397.

Matteo Ninniri, Marco Podda, and Davide Bacciu. Graph diffusion that can insert and delete, 2025. URL https://arxiv.org/abs/2506.15725.

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.

Anjie Qiao, Zhen Wang, Chuan Chen, DeFu Lian, and Enhong Chen. Composable score-based graph diffusion model for multi-conditional molecular generation, 2025. URL https://arxiv.org/abs/2509.09451.

Yiming Qin, Manuel Madeira, Dorina Thanou, and Pascal Frossard. Defog: Discrete flow matching for graph generation. *arXiv preprint arXiv:2410.04263*, 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(1):140022, 2014.

Jarrid Rector-Brooks, Mohsin Hasan, Zhangzhi Peng, Zachary Quinn, Chenghao Liu, Sarthak Mittal, Nouha Dziri, Michael Bronstein, Yoshua Bengio, Pranam Chatterjee, Alexander Tong, and Avishek Joey Bose. Steering masked discrete diffusion models via discrete denoising posterior prediction, 2024. URL https://arxiv.org/abs/2410.08134.

Hyunjin Seo, Taewon Kim, Sihyun Yu, and SungSoo Ahn. Learning flexible forward trajectories for masked molecular diffusion. In *International Conference on Learning Representations*, volume 2026, pp. 95424–95448, 2026.

Xingyu Su, Xiner Li, Masatoshi Uehara, Sunwoo Kim, Yulai Zhao, Gabriele Scalia, Ehsan Hajiramezanali, Tommaso Biancalani, Degui Zhi, and Shuiwang Ji. Iterative distillation for reward-guided fine-tuning of diffusion models in biomolecular design. In *International Conference on Learning Representations*, volume 2026, pp. 139443–139469, 2026.

Clement Vignac, Igor Krawczuk, Antoine Siraudin, Bohan Wang, Volkan Cevher, and Pascal Frossard. Digress: Discrete denoising diffusion for graph generation, 2023. URL https://arxiv.org/abs/2209.14734.

Yutong Xie, Chence Shi, Hao Zhou, Yuwei Yang, Weinan Zhang, Yong Yu, and Lei Li. Mars: Markov molecular sampling for multi-objective drug discovery. *arXiv preprint arXiv:2103.10432*, 2021.

Shuchen Xue, Chongjian Ge, Shilong Zhang, Yichen Li, and Zhi-Ming Ma. Advantage weighted matching: Aligning rl with pretraining in diffusion models, 2025. URL https://arxiv.org/abs/2509.25050.

Jiachi Zhao, Zehong Wang, Yamei Liao, Chuxu Zhang, and Yanfang Ye. Controllable graph generation with diffusion models via inference-time tree search guidance. In *Proceedings of the ACM Web Conference 2026*, pp. 1195–1205, 2026.

Kaiwen Zheng, Huayu Chen, Haotian Ye, Haoxiang Wang, Qinsheng Zhang, Kai Jiang, Hang Su, Stefano Ermon, Jun Zhu, and Ming-Yu Liu. Diffusionnft: Online diffusion reinforcement with forward process. In *International Conference on Learning Representations*, volume 2026, pp. 134129–134150, 2026.

Baoheng Zhu, Deyu Bo, Delvin Ce Zhang, and Xiao Wang. Graph-grpo: Training graph flow models with reinforcement learning, 2026a. URL https://arxiv.org/abs/2603.10395.

Yihan Zhu, Yuhan Liu, Weijiang Li, Tengfei Luo, and Meng Jiang. Controllable molecular generative foundation models, 2026b. URL https://arxiv.org/abs/2605.15354.

Yuchen Zhu, Wei Guo, Jaemoo Choi, Petr Molodyk, Bo Yuan, Molei Tao, and Yongxin Chen. Enhancing reasoning for diffusion llms via distribution matching policy optimization. *arXiv preprint arXiv:2510.08233*, 2025.

A APPENDIX

A.1 ADDITIONAL DETAILS ON DATASET

We evaluate GRAPHFDM on a six-property small-molecule benchmark and a four-property polymer benchmark. Table 5 summarizes the data used for unconditional pretraining and conditional post-training. All random subset construction and splitting use seed 42. The same frozen splits and train-derived normalization statistics are used by all compared methods.

Table 5: Overview of the datasets used by GRAPHFDM. PT uses molecular structures only and does not access target labels or conditional split identities.

Benchmark	Unlabeled PT	Conditional Training	Train/Val/Test	Motif tokenizer
Small molecule	100,000	10,000	6,000/2,000/2,000	V300-R80
Polymer	12,792	553	337/107/109	V1000-R80

A.1.1 SMALL-MOLECULE BENCHMARK

The small-molecule benchmark is constructed from rQM9 v1 (Jin et al., 2025), a corrected version of QM9 with revised molecular graphs (Ramakrishnan et al., 2014). Starting from 133,885 source entries, we remove 303 officially identified problematic structures, 88 structures that fail RDKit sanitization, 742 structures containing formal charges, and 370 entries belonging to duplicate canonical-SMILES groups. This produces 132,382 unique neutral molecular structures.

We remove explicit hydrogens and stereochemical annotations and represent each molecule by a connected, canonical, non-isomeric SMILES. From the cleaned collection, we sample 100,000 structures without replacement using seed 42 for unconditional pretraining. The 10,000-molecule conditional benchmark is a fixed subset of this structural pretraining corpus. Importantly, property labels and conditional split identities are not used during pretraining.

OOD targets. We additionally construct 200 compositional-OOD target conditions from the 90,000 pretraining molecules outside the conditional benchmark. For each of μ, α, HOMO, LUMO, and C_v, the 20th and 80th percentiles of the conditional training split define low, medium, and high regions. Their Cartesian combinations define joint property signatures. We retain signatures represented by at most 0.1% training molecules and select the eight rarest eligible signatures. The selected signatures occur only two to five times in the 6,000-molecule conditional training split. We randomly sample 25 targets from each signature using a deterministic seed schedule beginning at 42. Every selected property value remains within its corresponding training-set marginal range. SA is computed after selection and included as the sixth target property.

A.1.2 POLYMER BENCHMARK

We follow the polymer gas-permeability benchmark, preprocessing, and evaluation protocol of GraphDiT (Liu et al., 2024a). The benchmark contains 553 polymer repeat-unit records with jointly observed O_2, N_2, and CO_2 permeability and an SA score. Repeat-unit SMILES use wildcard atoms to mark polymerization sites.

Permeabilities are stored in Barrer, and model conditioning uses train-split z-score normalization. Gas-property errors are evaluated in $\log_{10}$ Barrer space. The polymer backbone is initialized from CoMole's 12,792-polymer unconditional pretraining corpus.

A.1.3 TOKENIZER

We use domain-specific motif tokenizers based on the node-pair encoding procedure of CoMole (Zhu et al., 2026b), where V denotes the motif vocabulary size and R the ring-token budget. For polymers, we directly reuse CoMole's V1000-R80 tokenizer learned from its pretraining corpus.

For the rQM9 graphs, we select a smaller V300-R80 tokenizer using the 100k unconditional pre-training corpus. We evaluate the fixed grid $V \in \{300, 500, 1000\}$ and $R \in \{30, 50, 80\}$ using representation-level compression and reconstruction statistics. Increasing the ring budget from 30 to

Table 6: Target ranges and train-split normalization statistics. Gas-permeability statistics are computed in the raw Barrer scale.

Benchmark	Target	Unit	Full range	Train mean/std
Small molecule	SA	-	1.111-7.750	4.236 / 0.924
	μ	D	0-9.071	2.665 / 1.433
	α	a_0^3	13.21-127.46	75.274 / 8.226
	HOMO	Ha	-0.3963--0.1538	-0.2404 / 0.0219
	LUMO	Ha	-0.1466-0.1171	0.0128 / 0.0469
	C_v	cal mol^{-1} K^{-1}	6.469-46.306	31.663 / 4.118
Polymer	SA	-	2.48-7.62	4.036 / 0.990
	O_2	Barrer	2.8×10^{-4}-1.87×10^{4}	387.180 / 1,678.333
	N_2	Barrer	1.6×10^{-4}-1.66×10^{4}	253.367 / 1,281.755
	CO_2	Barrer	1.8×10^{-3}-4.70×10^{4}	1,281.308 / 4,771.168

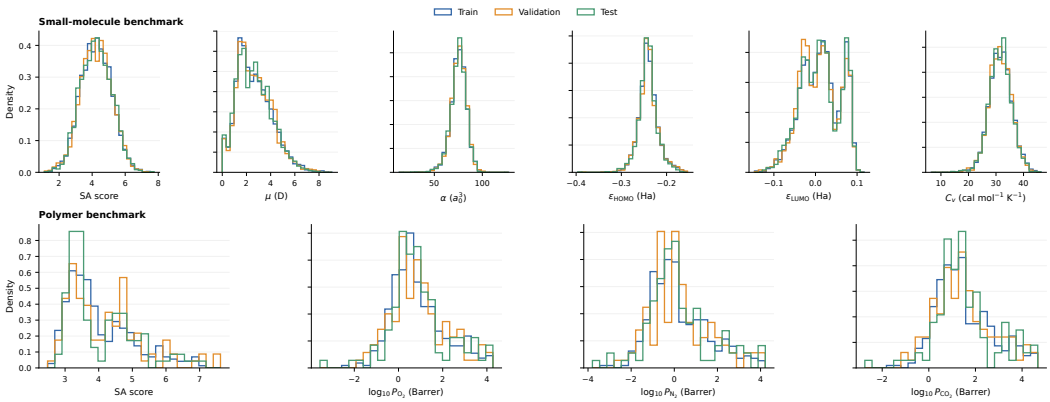

Figure 3: Target distributions for the small-molecule and polymer benchmarks, shown separately for the training, validation, and test splits. Polymer gas permeabilities are visualized in $\log_{10}$ Barrer, consistent with the evaluation metric.

80 consistently shortens the motif sequences. At $R = 80$, increasing V from 300 to 1000 reduces the mean motif count by only 1.6% and leaves its 95th percentile unchanged, while adding 700 prediction classes. We therefore choose V300-R80 as the smallest tokenizer at this compression plateau. No conditional labels or downstream validation performance are used in this selection.

A.1.4 Target Statistics and Normalization

For each property y, the model receives the standardized condition

$$\widetilde{y} = \frac{y - \mu_y^{\text{train}}}{\sigma_y^{\text{train}}},$$

where both statistics are computed exclusively from the corresponding training split. Table 6 reports the raw range over all splits and the train-split normalization statistics. Figure 3 shows that the fixed splits cover similar regions of each target distribution.

A.1.5 Structural Statistics

Table 7 summarizes the molecular scale and motif-graph scale of the two conditional benchmarks. Real atom counts exclude polymer wildcard atoms. Active motif units are the structural variable N used by the motif-aware diffusion backbone and controlled by SDC.

As shown in Figure 4, motif tokenization induces a substantially different active-unit distribution from the real-atom count. This distinction is directly relevant to GraphFDM because the active-unit count is sampled before denoising and forms the structural marginal controlled by SDC.

Table 7: Structural statistics of the conditional benchmarks. Values are reported as mean/maximum.

Benchmark	Real atoms	Hetero atoms	Rings	Mol. weight	Active motif units
Small molecule	8.80/9	2.44/7	1.74/8	122.72/152.04	4.84/9
Polymer	27.97/48	5.28/17	3.68/11	391.78/866.20	6.66/46

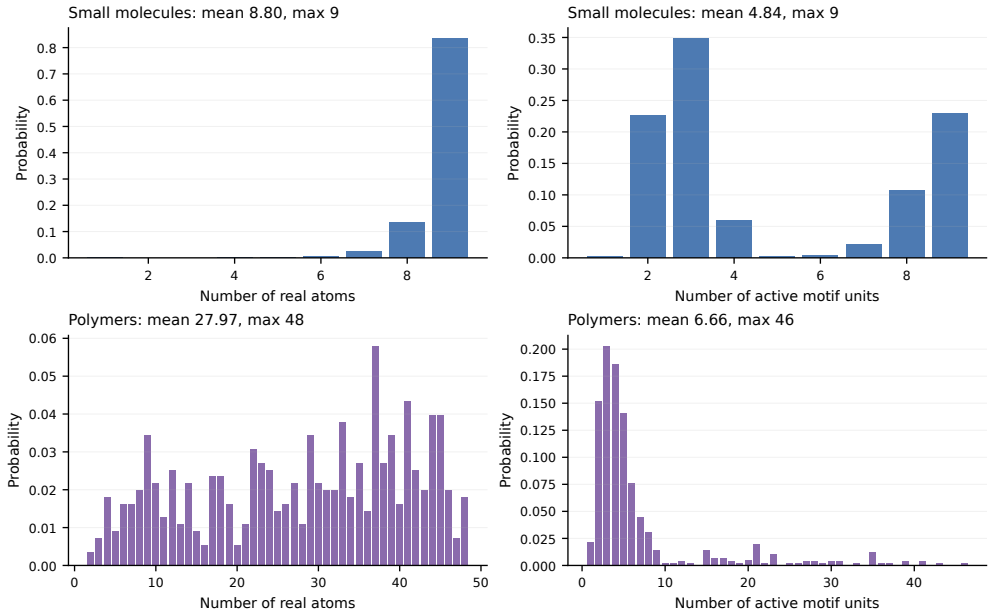

Figure 4: Distributions of real-atom counts and active motif-unit counts in the two conditional benchmarks. Polymer wildcard atoms are excluded from the real-atom count. The active motif-unit count corresponds to the structural variable N used by GRAPHFDM.

A.2 ADDITIONAL THEORY

Throughout this appendix, $\mathbb{E}_{\mathbf{c}}$ denotes the empirical average over target conditions. In condition-wise derivations, we suppress conditioning on $\mathbf{c}$ when no ambiguity arises.

A.2.1 VALIDITY CONDITIONING AND STRUCTURAL SUPPORT

Define the count-conditional validity probability

$$v_k(n, \mathbf{c}) := \Pr_{S \sim \pi_{\theta_k}(\cdot \mid n, \mathbf{c})} \left((n, S) \in \mathcal{G}_{\mathrm{val}} \right).$$

From Eqs. (3) and (4),

$$\bar{p}_k(n \mid \mathbf{c}) = \frac{p_k(n \mid \mathbf{c})\, v_k(n, \mathbf{c})}{\sum_{n'} p_k(n' \mid \mathbf{c})\, v_k(n', \mathbf{c})}. \tag{21}$$

Thus, validity conditioning reweights p_k according to the count-dependent validity probability v_k. The two marginals coincide if and only if $v_k(n, \mathbf{c})$ is constant over the support of $p_k(\cdot \mid \mathbf{c})$. Moreover,

$$\bar{p}_k(\cdot \mid \mathbf{c}) \ll p_k(\cdot \mid \mathbf{c}) \ll p_0.$$

A.2.2 PROOF OF FORWARD DISTRIBUTION MATCHING

Proof. Fix a target condition $\mathbf{c}$ and define

$$H_\theta(G, \mathbf{c}) := \mathbb{E}_{\substack{t \sim \mathrm{Unif}(\{1, \ldots, T\}), \\ S_t \sim q_t(\cdot \mid S_0, m)}} \left[\ell^\theta_{\mathrm{den}}(S_0, S_t, m, t, \mathbf{c}) \right].$$

Using

$$w_k^\mu(G, \mathbf{c}) = \frac{\mu_k(G \mid \mathbf{c})}{\bar{\pi}_k(G \mid \mathbf{c})},$$

the change-of-measure identity gives

$$\begin{aligned}&\mathbb{E}_{G\sim\bar{\pi}_k(\cdot|\mathbf{c})}\left[w_k^\mu(G, \mathbf{c})H_\theta(G, \mathbf{c})\right]\\&= \mathbb{E}_{G\sim\mu_k(\cdot|\mathbf{c})}\left[H_\theta(G, \mathbf{c})\right],\end{aligned}$$

which proves Eq. (6).

Applying the same identity to the structural log loss gives

$$\begin{aligned}&\mathbb{E}_{G\sim\bar{\pi}_k(\cdot|\mathbf{c})}\left[w_k^\mu(G, \mathbf{c})\big(-\log p_\phi(N \mid \mathbf{c})\big)\right]\\&= \mathbb{E}_{N\sim\mu_{N,k}(\cdot|\mathbf{c})}\left[-\log p_\phi(N \mid \mathbf{c})\right],\end{aligned}$$

which proves Eq. (7). Averaging over target conditions preserves both identities. □

A.2.3 Proper-Scoring Interpretation

Consider the teacher-corruption sampling scheme

$$\mathbf{c} \sim \text{the empirical target-condition distribution}, \qquad G = (N, S_0) \sim \mu_k(\cdot \mid \mathbf{c}),$$

$$t \sim \mathrm{Unif}(\{1, \ldots, T\}), \qquad S_t \sim q_t(\cdot \mid S_0, m).$$

Let $U = (S_t, m, t, \mathbf{c})$. For an active entry a of $Y \in \{X, E, P\}$, let $d_{\theta,Y,a}(\cdot \mid U)$ denote the corresponding categorical prediction and define the teacher-induced posterior marginal

$$r_{Y,a}^\mu(\cdot \mid U) := \Pr(Y_{0,a} = \cdot \mid U)$$

under the sampling scheme above. Its expected cross-entropy decomposes as

$$\begin{aligned}&\mathbb{E}\left[-\log d_{\theta,Y,a}(Y_{0,a} \mid U)\right]\\&= \mathbb{E}_U\left[H\left(r_{Y,a}^\mu(\cdot \mid U)\right) + D_{\mathrm{KL}}\left(r_{Y,a}^\mu(\cdot \mid U) \| d_{\theta,Y,a}(\cdot \mid U)\right)\right].\end{aligned}$$

Summing over the active entries of X, E, and P shows that, under simultaneous realizability across target conditions, every population minimizer of $\mathcal{L}_{S,k}^\mu(\theta)$ recovers the corresponding teacher-induced posterior marginals almost surely.

Likewise, the structural objective satisfies

$$\begin{aligned}\mathcal{L}_{N,k}^\mu(\phi) = \mathbb{E}_{\mathbf{c}}\Big[&H\left(\mu_{N,k}(\cdot \mid \mathbf{c})\right)\\&+ D_{\mathrm{KL}}\left(\mu_{N,k}(\cdot \mid \mathbf{c}) \| p_\phi(\cdot \mid \mathbf{c})\right)\Big].\end{aligned}$$

Hence, under simultaneous realizability, every population minimizer of $\mathcal{L}_{N,k}^\mu(\phi)$ recovers $\mu_{N,k}(\cdot \mid \mathbf{c})$ for every target condition with positive empirical mass.

A.2.4 PROOF OF PROPOSITION 3.2

Proof. Fix $\mathbf{c}$ and suppress it from the notation. For any conditional distribution $\widetilde{\pi}(S \mid N)$ for which the divergence is finite, the hierarchical factorizations give

$$\log \frac{\mu_k(N,S)}{p_k(N)\widetilde{\pi}(S \mid N)} = \log \frac{\mu_{N,k}(N)}{p_k(N)} + \log \frac{\mu_{S,k}(S \mid N)}{\widetilde{\pi}(S \mid N)}.$$

Taking expectation under $\mu_k(N,S)$ yields Eq. (8). The conditional KL term is nonnegative and becomes zero when

$$\widetilde{\pi}(\cdot \mid N) = \mu_{S,k}(\cdot \mid N)$$

for $\mu_{N,k}$-almost every N. Therefore, taking the infimum over conditional molecular distributions gives Eq. (9). □

A.2.5 PROOF OF THEOREM 3.3

Proof. The Lagrangian of Eq. (12) is

$$\mathfrak{L}(\rho, \tau_N, \tau_S) = \mathbb{E}_{\mathbf{c}}\mathbb{E}_{G\sim\rho(\cdot|\mathbf{c})}\left[R_{\mathbf{c}}(G)\right] - \tau_N(\mathcal{C}_N(\rho) - \epsilon_N) - \tau_S(\mathcal{C}_S(\rho) - \epsilon_S)\,. \tag{22}$$

Fix $\tau_N, \tau_S > 0$. The terms $\tau_N\epsilon_N + \tau_S\epsilon_S$ do not depend on ρ, and the remaining maximization separates across target conditions. All condition-wise statements below concern conditions in the support of the law defining $\mathbb{E}_{\mathbf{c}}$. For a fixed such $\mathbf{c}$, write

$$\begin{aligned}\mathcal{J}_{\mathbf{c}}(\rho_N, \rho_S) := \mathbb{E}_{N\sim\rho_N(\cdot|\mathbf{c})}\Big[&\mathbb{E}_{S\sim\rho_S(\cdot|N,\mathbf{c})}[R_{\mathbf{c}}(N,S)] \\ &- \tau_S D_{\mathrm{KL}}(\rho_S(\cdot \mid N,\mathbf{c})\|\bar{\pi}_k(\cdot \mid N,\mathbf{c}))\Big] \\ &- \tau_N D_{\mathrm{KL}}(\rho_N(\cdot \mid \mathbf{c})\|\bar{p}_k(\cdot \mid \mathbf{c}))\,.\end{aligned} \tag{23}$$

For $n \in \operatorname{supp}\bar{p}_k(\cdot \mid \mathbf{c})$, define

$$\widetilde{\pi}_{S,k}(S \mid n, \mathbf{c}) := \frac{\bar{\pi}_k(S \mid n, \mathbf{c})\exp(R_{\mathbf{c}}(n,S)/\tau_S)}{Z_{S,k}(n, \mathbf{c}; \tau_S)}.$$

The Gibbs variational identity gives, for every $\rho_S(\cdot \mid n, \mathbf{c}) \ll \bar{\pi}_k(\cdot \mid n, \mathbf{c})$,

$$\begin{aligned}&\mathbb{E}_{\rho_S(\cdot|n,\mathbf{c})}[R_{\mathbf{c}}(n,S)] - \tau_S D_{\mathrm{KL}}(\rho_S(\cdot \mid n,\mathbf{c})\|\bar{\pi}_k(\cdot \mid n,\mathbf{c})) \\ &\quad = \tau_S \log Z_{S,k}(n,\mathbf{c};\tau_S) - \tau_S D_{\mathrm{KL}}(\rho_S(\cdot \mid n,\mathbf{c})\|\widetilde{\pi}_{S,k}(\cdot \mid n,\mathbf{c}))\,.\end{aligned} \tag{24}$$

Thus the conditional maximizer is unique and equals $\widetilde{\pi}_{S,k}(\cdot \mid n, \mathbf{c})$.

Now define

$$\widetilde{\pi}_{N,k}(n \mid \mathbf{c}) := \frac{\bar{p}_k(n \mid \mathbf{c})Z_{S,k}(n,\mathbf{c};\tau_S)^{\tau_S/\tau_N}}{Z_{N,k}(\mathbf{c};\tau_N,\tau_S)}.$$

A second application of the Gibbs variational identity yields

$$\begin{aligned}&\mathbb{E}_{N\sim\rho_N(\cdot|\mathbf{c})}[\tau_S \log Z_{S,k}(N,\mathbf{c};\tau_S)] - \tau_N D_{\mathrm{KL}}(\rho_N(\cdot \mid \mathbf{c})\|\bar{p}_k(\cdot \mid \mathbf{c})) \\ &\quad = \tau_N \log Z_{N,k}(\mathbf{c};\tau_N,\tau_S) - \tau_N D_{\mathrm{KL}}(\rho_N(\cdot \mid \mathbf{c})\|\widetilde{\pi}_{N,k}(\cdot \mid \mathbf{c}))\,.\end{aligned} \tag{25}$$

Combining Eqs. (24) and (25) gives

$$\begin{aligned}\mathcal{J}_{\mathbf{c}}(\rho_N,\rho_S) = \tau_N \log Z_{N,k}(\mathbf{c};\tau_N,\tau_S) &- \tau_N D_{\mathrm{KL}}(\rho_N(\cdot \mid \mathbf{c})\|\widetilde{\pi}_{N,k}(\cdot \mid \mathbf{c})) \\ &- \tau_S \mathbb{E}_{N\sim\rho_N(\cdot|\mathbf{c})} D_{\mathrm{KL}}(\rho_S(\cdot \mid N,\mathbf{c})\|\widetilde{\pi}_{S,k}(\cdot \mid N,\mathbf{c}))\,.\end{aligned} \tag{26}$$

Both divergence terms are nonnegative. Moreover, $\widetilde{\pi}_{N,k}$ is positive on $\operatorname{supp}\bar{p}_k(\cdot \mid \mathbf{c})$. Consequently, the unique maximizing joint distribution is

$$\widetilde{\pi}_k(N,S \mid \mathbf{c}) = \widetilde{\pi}_{N,k}(N \mid \mathbf{c})\widetilde{\pi}_{S,k}(S \mid N,\mathbf{c}).$$

It remains to connect this Lagrangian maximizer to the constrained problem. Viewed as an optimization over the joint probabilities on the finite reference support, the reward objective is linear and both KL constraints are convex; convexity of $\mathcal{C}_S$ follows from the perspective form of conditional relative entropy. Furthermore, $\rho = \bar{\pi}_k$ satisfies $\mathcal{C}_N(\bar{\pi}_k) = \mathcal{C}_S(\bar{\pi}_k) = 0$, and is therefore strictly feasible because $\epsilon_N, \epsilon_S > 0$. Slater's condition and the KKT conditions therefore apply. At the assumed positive optimal dual variables, the primal optimizer must equal the unique Lagrangian maximizer above. Identifying $\widetilde{\pi}_k = \pi_k^\star$ proves Eqs. (14) and (15) and uniqueness. □

Projection ratio and shared-temperature case. On the reference support, multiplying the two factorwise ratios gives

$$\frac{\pi_k^\star(G \mid \mathbf{c})}{\bar{\pi}_k(G \mid \mathbf{c})} = \frac{\exp(R_{\mathbf{c}}(G)/\tau_S)\, Z_{S,k}(N, \mathbf{c}; \tau_S)^{\tau_S/\tau_N - 1}}{Z_{N,k}(\mathbf{c}; \tau_N, \tau_S)},$$

which is Eq. (17). If $\tau_N = \tau_S = \tau$, then

$$Z_{N,k}(\mathbf{c}; \tau, \tau) = \sum_n \bar{p}_k(n \mid \mathbf{c}) Z_{S,k}(n, \mathbf{c}; \tau) = \mathbb{E}_{G \sim \bar{\pi}_k(\cdot \mid \mathbf{c})}[\exp(R_{\mathbf{c}}(G)/\tau)]\,.$$

The factor $Z_{S,k}(N, \mathbf{c}; \tau)$ cancels between the structural and conditional teacher factors, yielding Eq. (20).

A.2.6 Proof of Corollary 3.4

Proof. Fix $\mathbf{c}$ and abbreviate

$$P_N := \pi_k^\star(N \mid \mathbf{c}), \qquad Q_N := \bar{p}_k(N \mid \mathbf{c}),$$

and, for each n in their common support,

$$P_S^n := \pi_k^\star(S \mid n, \mathbf{c}), \qquad Q_S^n := \bar{\pi}_k(S \mid n, \mathbf{c}).$$

Because the reward is real-valued on the finite reference support, the teacher and reference factors have the same supports. Combining Eqs. (14) and (15) therefore gives, on that support,

$$R_{\mathbf{c}}(n, s) = \tau_N \log \frac{P_N(n)}{Q_N(n)} + \tau_S \log \frac{P_S^n(s)}{Q_S^n(s)} + \tau_N \log Z_{N,k}(\mathbf{c}; \tau_N, \tau_S). \tag{27}$$

Taking the expectation of Eq. (27) first under $P_N P_S^N$, then under $Q_N Q_S^N$, and subtracting cancels the last term and yields the exact identity

$$\begin{aligned} \Delta_k(\mathbf{c}) = \tau_N\, &[D_{\mathrm{KL}}(P_N \| Q_N) + D_{\mathrm{KL}}(Q_N \| P_N)] \\ &+ \tau_S \mathbb{E}_{N \sim P_N} D_{\mathrm{KL}}(P_S^N \| Q_S^N) \\ &+ \tau_S \mathbb{E}_{N \sim Q_N} D_{\mathrm{KL}}(Q_S^N \| P_S^N). \end{aligned} \tag{28}$$

Dropping the two reverse-KL terms proves Eq. (18). If $\pi_k^\star(\cdot \mid \mathbf{c}) \neq \bar{\pi}_k(\cdot \mid \mathbf{c})$, then the KL chain rule implies

$$D_{\mathrm{KL}}(P_N \| Q_N) + \mathbb{E}_{N \sim P_N} D_{\mathrm{KL}}(P_S^N \| Q_S^N) > 0.$$

Since $\tau_N, \tau_S > 0$, the lower bound in Eq. (18) is then strictly positive.

Averaging Eq. (28) over target conditions and retaining the two forward-KL terms gives

$$\mathbb{E}_{\mathbf{c}}[\Delta_k(\mathbf{c})] \geq \tau_N \mathcal{C}_N(\pi_k^\star) + \tau_S \mathcal{C}_S(\pi_k^\star).$$

Both constraints are active by assumption, so $\mathcal{C}_N(\pi_k^\star) = \epsilon_N$ and $\mathcal{C}_S(\pi_k^\star) = \epsilon_S$, proving Eq. (19).

Finally, $\pi_k^\star \ll \bar{\pi}_k$ and $\bar{\pi}_k(\mathcal{G}_{\mathrm{val}} \mid \mathbf{c}) = 1$ imply $\pi_k^\star(\mathcal{G}_{\mathrm{val}} \mid \mathbf{c}) = 1$. □

A.2.7 Finite-Sample Structured Projection

We describe the estimator for one target condition $\mathbf{c}$; the training objectives and KL constraints are averaged over the sampled target conditions. To include active-node-count exploration, let

$$b_k(G \mid \mathbf{c}) = b_{N,k}(N \mid \mathbf{c})\, \pi_{\theta_k}(S \mid N, \mathbf{c})$$

be a behavior distribution satisfying $p_k(\cdot \mid \mathbf{c}) \ll b_{N,k}(\cdot \mid \mathbf{c})$. The online update in the main text is the special case $b_{N,k} = p_k$. Draw B independent candidates

$$N_i \sim b_{N,k}(\cdot \mid \mathbf{c}), \qquad S_{0,i} \sim \pi_{\theta_k}(\cdot \mid N_i, \mathbf{c}), \qquad G_i = (N_i, S_{0,i}).$$

Define

$$a_i(\mathbf{c}) := \mathbf{1}\{G_i \in \mathcal{G}_{\mathrm{val}}\} \frac{p_k(N_i \mid \mathbf{c})}{b_{N,k}(N_i \mid \mathbf{c})}.$$

Whenever $A(\mathbf{c}) := \sum_{j=1}^{B} a_j(\mathbf{c}) > 0$, set

$$\widehat{r}_{k,i}(\mathbf{c}) := \frac{a_i(\mathbf{c})}{A(\mathbf{c})}.$$

These self-normalized proposal weights form the empirical valid-conditioned reference distribution. In particular, if $b_{N,k} = p_k$, they are uniform over the valid candidates.

Let

$$\mathcal{I}_n(\mathbf{c}) := \{i : a_i(\mathbf{c}) > 0,\ N_i = n\}, \qquad \widehat{\widehat{p}}_k(n \mid \mathbf{c}) := \sum_{i \in \mathcal{I}_n(\mathbf{c})} \widehat{r}_{k,i}(\mathbf{c}).$$

For every empirically represented n, define the empirical reference conditional mass and conditional partition function by

$$\widehat{\widehat{\pi}}_k(i \mid n, \mathbf{c}) := \frac{\widehat{r}_{k,i}(\mathbf{c})}{\widehat{\widehat{p}}_k(n \mid \mathbf{c})}, \qquad i \in \mathcal{I}_n(\mathbf{c}), \tag{29}$$

$$\widehat{Z}_{S,k}(n, \mathbf{c}; \tau_S) := \sum_{i \in \mathcal{I}_n(\mathbf{c})} \widehat{\widehat{\pi}}_k(i \mid n, \mathbf{c}) \exp\left(\frac{R_{\mathbf{c}}(G_i)}{\tau_S}\right). \tag{30}$$

Here sampled indices are treated as empirical atoms; repeated graphs may equivalently be aggregated. The empirical structural partition function is

$$\widehat{Z}_{N,k}(\mathbf{c}; \tau_N, \tau_S) := \sum_{n : \widehat{\widehat{p}}_k(n \mid \mathbf{c}) > 0} \widehat{\widehat{p}}_k(n \mid \mathbf{c}) \widehat{Z}_{S,k}(n, \mathbf{c}; \tau_S)^{\tau_S/\tau_N}. \tag{31}$$

The plug-in estimate of the density ratio in Eq. (17) is

$$\widehat{w}_k(G_i, \mathbf{c}) := \frac{\exp(R_{\mathbf{c}}(G_i)/\tau_S)\, \widehat{Z}_{S,k}(N_i, \mathbf{c}; \tau_S)^{\tau_S/\tau_N - 1}}{\widehat{Z}_{N,k}(\mathbf{c}; \tau_N, \tau_S)}, \qquad a_i(\mathbf{c}) > 0. \tag{32}$$

The corresponding empirical teacher mass assigned to candidate i is

$$\widehat{\omega}_{k,i}(\mathbf{c}) := \widehat{r}_{k,i}(\mathbf{c})\, \widehat{w}_k(G_i, \mathbf{c}). \tag{33}$$

Indeed,

$$\begin{aligned} &\sum_{i : a_i(\mathbf{c}) > 0} \widehat{r}_{k,i}(\mathbf{c}) \exp(R_{\mathbf{c}}(G_i)/\tau_S)\, \widehat{Z}_{S,k}(N_i, \mathbf{c}; \tau_S)^{\tau_S/\tau_N - 1} \\ &\quad = \sum_n \widehat{\widehat{p}}_k(n \mid \mathbf{c}) \widehat{Z}_{S,k}(n, \mathbf{c}; \tau_S)^{\tau_S/\tau_N} = \widehat{Z}_{N,k}(\mathbf{c}; \tau_N, \tau_S), \end{aligned}$$

so $\sum_i \widehat{\omega}_{k,i}(\mathbf{c}) = 1$. The induced empirical teacher factors are

$$\widehat{\pi}^{\star}_{N,k}(n \mid \mathbf{c}) = \frac{\widehat{\widehat{p}}_k(n \mid \mathbf{c}) \widehat{Z}_{S,k}(n, \mathbf{c}; \tau_S)^{\tau_S/\tau_N}}{\widehat{Z}_{N,k}(\mathbf{c}; \tau_N, \tau_S)}, \tag{34}$$

$$\widehat{\pi}^{\star}_{k}(i \mid n, \mathbf{c}) = \frac{\widehat{\widehat{\pi}}_k(i \mid n, \mathbf{c}) \exp(R_{\mathbf{c}}(G_i)/\tau_S)}{\widehat{Z}_{S,k}(n, \mathbf{c}; \tau_S)}, \qquad i \in \mathcal{I}_n(\mathbf{c}). \tag{35}$$

Equivalently, $\widehat{\omega}_{k,i}(\mathbf{c}) = \widehat{\pi}^{\star}_{N,k}(N_i \mid \mathbf{c}) \widehat{\pi}^{\star}_{k}(i \mid N_i, \mathbf{c})$.

Empirical projection objectives. For each valid candidate, sample

$$t_i \sim \mathrm{Unif}\{1, \dots, T\}, \qquad S_{t_i} \sim q_{t_i}(\cdot \mid S_{0,i}, m_i).$$

Using the same empirical teacher mass for both projections gives

$$\widehat{\mathcal{L}}_{S,k}(\theta; \mathbf{c}) = \sum_{i : a_i(\mathbf{c}) > 0} \widehat{\omega}_{k,i}(\mathbf{c})\, \ell^{\theta}_{\mathrm{den}}(S_{0,i}, S_{t_i}, m_i, t_i, \mathbf{c}), \tag{36}$$

$$\widehat{\mathcal{L}}_{N,k}(\phi; \mathbf{c}) = -\sum_{i : a_i(\mathbf{c}) > 0} \widehat{\omega}_{k,i}(\mathbf{c}) \log p_{\phi}(N_i \mid \mathbf{c}). \tag{37}$$

The minibatch objectives are obtained by averaging these expressions over target conditions.

Dual estimation. Suppose a minibatch contains target conditions $\mathbf{c}_1, \ldots, \mathbf{c}_M$, each with a nonempty valid candidate bank. For a fixed positive pair (τ_N, τ_S), define

$$\widehat{\mathcal{C}}_N(\tau_N, \tau_S) := \frac{1}{M} \sum_{m=1}^{M} D_{\mathrm{KL}}\big(\widehat{\pi}^{\star}_{N,k}(\cdot \mid \mathbf{c}_m) \| \widehat{\overline{p}}_k(\cdot \mid \mathbf{c}_m)\big)\,, \tag{38}$$

$$\widehat{\mathcal{C}}_S(\tau_N, \tau_S) := \frac{1}{M} \sum_{m=1}^{M} \mathbb{E}_{N \sim \widehat{\pi}^{\star}_{N,k}(\cdot|\mathbf{c}_m)} D_{\mathrm{KL}}\big(\widehat{\pi}^{\star}_{k}(\cdot \mid N, \mathbf{c}_m) \| \widehat{\overline{\pi}}_k(\cdot \mid N, \mathbf{c}_m)\big)\,. \tag{39}$$

Applying the two Gibbs variational identities to the empirical reference distribution gives the empirical dual objective

$$\widehat{d}_k(\tau_N, \tau_S) := \tau_N \epsilon_N + \tau_S \epsilon_S + \frac{1}{M} \sum_{m=1}^{M} \tau_N \log \widehat{Z}_{N,k}(\mathbf{c}_m; \tau_N, \tau_S). \tag{40}$$

Thus the shared temperatures are estimated by minimizing $\widehat{d}_k$ over $\tau_N, \tau_S > 0$. By the envelope theorem,

$$\frac{\partial \widehat{d}_k}{\partial \tau_N} = \epsilon_N - \widehat{\mathcal{C}}_N, \qquad \frac{\partial \widehat{d}_k}{\partial \tau_S} = \epsilon_S - \widehat{\mathcal{C}}_S. \tag{41}$$

Consequently, an interior dual optimum enforces the two empirical KL budgets. The same pair of temperatures is shared across conditions because the constraints in Eq. (12) are condition-averaged.

If $A(\mathbf{c}) = 0$, that condition contributes no update in the current iteration. Counts absent from the valid candidate bank receive zero empirical teacher mass. Under the finite-support assumptions of theorem 3.3, the stated proposal-support condition, and integrability of the projection losses, the self-normalized partition and objective estimates converge almost surely to their population counterparts for fixed positive temperatures as the candidate-bank size tends to infinity.

A.3 Additional Details on Experiments

A.3.1 Experimental Setup

Each training and evaluation job is run on a single NVIDIA H100 GPU. We use the graph diffusion transformer backbone with 12 transformer layers, hidden dimension 1024, 16 attention heads, and an MLP expansion ratio of 4. The diffusion process uses 500 steps, a cosine noise schedule, and marginal transition kernels. PT and SFT use learning rate 2×10^{-5}. Each SFT model is initialized from its corresponding PT checkpoint, and GraphFDM is initialized from the resulting SFT checkpoint.

The forward distribution matching post-training runs for 10 online rounds. Each round collects $K = 32$ candidates per training condition and uses the symmetric trust-region budgets $\epsilon_N = \epsilon_S = 0.035$. The denoiser and structural controller are each updated for 20 epochs per round using learning rates 2×10^{-6} and 10^{-4}, respectively. The controller is a two-hidden-layer MLP of width 64. Dataset-specific epoch and batch configurations are reported in Table 8. Finite-sample teacher construction and proposal correction are described in Section A.2.7.

Table 8: Dataset-specific training configurations.

Configuration	Molecule	Polymer
PT epochs	200	400
PT batch size	256	64
SFT epochs	200	1200
SFT batch size	256	128
Online rounds	10	10
Candidates per condition	32	32
Graph-update batch size	256	64
Controller batch size	1024	64
Test samples	10,000	10,000

Table 9: **Multi-seed results of the top-5 models on QM9 multi-conditional generation.** Results on six properties (synthetic score and five QM9 properties) are reported as mean±std. Overall Rank aggregates property-wise MAE ranks across different units and scales. Best results are in **red**.

(a) Validity and Distribution Learning

Model	Validity	Distribution Learning		
	Valid. (raw) ↑	Div. ↑	Sim. ↑	Dis. ↓
DiGress	0.914±0.004 (0.827±0.020)	**0.919±0.002**	0.943±0.002	4.742±0.142
CoMole	0.937±0.037 (0.831±0.067)	0.916±0.001	0.930±0.010	4.253±0.019
GDPO	0.914±0.003 (0.849±0.010)	0.917±0.001	**0.950±0.003**	0.795±0.087
GraphGRPO	0.973±0.003 (0.836±0.015)	0.910±0.001	0.914±0.016	2.953±0.158
GRAPHFDM	**0.994±0.000 (0.959±0.001)**	0.916±0.001	0.907±0.007	**0.612±0.004**

(b) Condition Control

Model	Synth. ↓	μ ↓	α ↓	$\varepsilon_{\text{HOMO}}$ ↓	$\varepsilon_{\text{LUMO}}$ ↓	C_v ↓	Overall Rank ↓
DiGress	1.618±0.016	0.915±0.023	4.108±0.199	0.012±0.000	0.018±0.000	1.998±0.086	5
CoMole	0.497±0.035	0.874±0.008	4.375±0.175	0.012±0.000	0.021±0.001	1.967±0.093	4
GDPO	0.547±0.006	0.909±0.005	3.804±0.065	0.012±0.000	0.018±0.001	1.955±0.063	3
GraphGRPO	0.577±0.007	0.855±0.014	3.569±0.118	0.011±0.001	0.018±0.000	2.010±0.040	2
GRAPHFDM	**0.361±0.002**	**0.704±0.009**	**3.422±0.162**	**0.009±0.000**	**0.015±0.000**	**1.486±0.019**	**1**

Reward definition. Let $\widehat{y}_j(G)$ and c_j denote the evaluated and target values of property j, respectively. Define the property-specific evaluation transform

$$h_j(y) := \begin{cases} \log_{10} y, & j \text{ is a polymer gas-permeability property,} \\ y, & \text{otherwise.} \end{cases} \tag{42}$$

For both benchmarks, GraphFDM uses the normalized absolute error

$$e_j(G, \mathbf{c}) := \frac{|h_j(\widehat{y}_j(G)) - h_j(c_j)|}{\sigma_j^{\text{tr}}}, \tag{43}$$

where σ_j^{tr} is the standard deviation of $h_j(y_j)$ on the corresponding conditional training split. The terminal reward is

$$R_{\mathbf{c}}(G) = -\frac{1}{d}\sum_{j=1}^{d} e_j(G, \mathbf{c}), \tag{44}$$

where $d = 6$ for small molecules and $d = 4$ for polymers. Thus, all target properties contribute equally after property-wise normalization.

Baselines. We use author-released implementations and recommended configurations whenever available. Otherwise, we reproduce the reported architecture and training protocol and tune unspecified hyperparameters on the validation set using Optuna (Akiba et al., 2019). Implementations are adapted to support our data and evaluation interfaces. All methods use the same splits, target conditions, property normalization, validity criteria, and property evaluators.

A.3.2 ROBUSTNESS ACROSS SEEDS

To evaluate robustness to training stochasticity, we repeat the five leading methods from the main comparison (Table 2): GRAPHFDM, GraphGRPO, GDPO, CoMole, and DiGress, in order of their single-seed Avg. Rank. We use three training seeds, $0, 1, 2$, where seed 0 corresponds to the main-table run. The data split, property oracle, and test targets are fixed across all runs. For post-training methods, we keep the same reference checkpoint and rerun only the online post-training stage.

As shown in Table 9, the Overall Rank remains stable across seeds, with GRAPHFDM achieving the best overall performance and low variance.

Table 10: **Training Performance of Oracle Methods:** we train the models on all polymers or small molecules in a task to simulate the Oracle. Results from the random forest model are in **bold** because it has the lowest training MAE.

Oracle	Metric	O_2	N_2	CO_2	μ	α	$\varepsilon_{\text{HOMO}}$	$\varepsilon_{\text{LUMO}}$	C_v
RF	MAE ↓	**0.412**	**0.456**	**0.421**	**0.194**	**1.343**	**0.002**	**0.002**	**0.540**
SVR	MAE ↓	0.480	0.539	0.488	0.862	4.476	0.010	0.019	2.056
GP	MAE ↓	0.872	0.989	0.874	1.153	6.304	0.015	0.039	3.148

Table 11: **Oracles for Generation Evaluation:** Models are ranked from 1 to 9 across the polymer benchmark by Avg.MAE under each evaluator. Underlined entries preserve the same relative ordering across all evaluators.

Rank	Random Forest	Support Vector Regression	Gaussian Process
1	**GRAPHFDM**	**GRAPHFDM**	**GRAPHFDM**
2	CoMole	CoMole	CoMole
3	GraphDiT	GDSS	GDSS
4	CSGD	GDPO	GDPO
5	GDSS	GraphDiT	GraphDiT
6	DeFoG	LSTM	LSTM
7	GDPO	CSGD	DeFoG
8	LSTM	DeFoG	CSGD
9	MELD	MELD	MELD

A.3.3 ROBUSTNESS ACROSS ORACLES

We fit Random Forest (RF), Support Vector Regression (SVR), and Gaussian Process Regression (GP) property oracles using all labeled molecules available in each task, with the goal of approximating the underlying property functions as accurately as possible. The resulting predictors are treated as fixed surrogate property evaluators. Table 10 reports their fitting accuracy. RF achieves the lowest fitting MAE and is therefore used as the reward oracle and primary evaluator throughout our experiments.

To assess robustness to evaluator choice, we fix the generated molecules and rescore the same samples using SVR and GP oracles trained on identical data. As shown in Table 11, GRAPHFDM ranks first under all three evaluators, while a consistent partial ordering GRAPHFDM $\succ$ CoMole $\succ$ GraphDiT $\succ$ LSTM $\succ$ MELD is preserved despite changes in absolute ranks. These results indicate that the relative controllability gains are robust to the choice of learned property oracle.

A.3.4 COMPUTATIONAL COST

We report online post-training wall-clock costs on a single NVIDIA H100, including candidate generation, oracle evaluation, optimization, and validation, but excluding base-model training and final evaluation. TreeDiff is reported separately as an inference-time search method.

As shown in Table 12, CoMole and GRAPHFDM share the same SFT initialization and backbone, providing a matched comparison of post-training cost. GRAPHFDM requires 16.65 vs. 30.37 hours on polymers and 49.18 vs. 50.30 hours on molecules. Its endpoint-based updates reuse each clean sample through forward re-noising without retaining the sampled reverse trajectory. VIDD (Su et al., 2026) and GDPO (Liu et al., 2024b) have lower total costs under their native configurations, reflecting differences in backbone cost and online budget. For example, VIDD operates on GDSS (Jo et al., 2022) score/SDE trajectories, so replacing its backbone would require reformulating its optimization objective.

Oracle overhead. Property rewards for all models are evaluated by CPU-based random-forest oracles, with SAscore computed directly from molecular structure. Oracle evaluation takes 1.12 ms per polymer and 0.79 ms per QM9 molecule, corresponding to approximately 12 and 152 seconds per GraphFDM round (overall 20 rounds), respectively. This constitutes only a small fraction of the total online cost.

Table 12: **Online post-training and inference-time search cost.** We report wall-clock cost on one H100, including sampling, oracle evaluation, optimization, and validation. Evaluations denote the dominant model calls for each method's generative dynamics. CoMole and GRAPHFDM use the same SFT initialization and backbone.

Method	Online unit	Evaluations/unit	Seconds/unit	ms/evaluation	Total hours
Polymer					
GDPO	reverse trajectory	500	1.312	2.625	12.28
GraphGRPO	group trajectory	500	1.940	3.884	19.30
VIDD	score trajectory	2,400	9.291	3.871	3.55
CoMole	reverse trajectory	500	0.291	0.582	30.37
GRAPHFDM	clean endpoint	500	0.556	1.112	16.65
TreeDiff[†]	search output	26,327	49.314	1.873	136.98
Molecule					
GDPO	reverse trajectory	500	0.083	0.165	13.79
GraphGRPO	group trajectory	500	2.191	4.380	59.21
VIDD	score trajectory	2,400	2.112	0.880	10.59
CoMole	reverse trajectory	500	0.302	0.604	50.30
GRAPHFDM	clean endpoint	500	0.092	0.184	49.18
TreeDiff[†]	search output	26,294	53.605	2.039	148.90

A reverse trajectory is a complete generative path ending in one molecule. A group trajectory is evaluated relative to other trajectories sampled for the same condition. A score trajectory is used for score-based distillation. A clean endpoint retains only the terminal molecule. [†]TreeDiff performs inference-time search, requiring approximately 26K model forwards per output.

A.4 LIMITATIONS AND FUTURE WORK

GRAPHFDM is designed as a post-training framework for pretrained graph diffusion models and therefore operates within the structural support provided by the underlying backbone. In particular, SDC reallocates probability across graph sizes represented by the reference model. Extending the framework to support-expanding structural updates, such as adaptive graph growth or vocabulary expansion, is an interesting direction for future work. Our experiments cover both small molecules and polymers, and extending GRAPHFDM to broader chemical spaces and other structured generative domains would further examine its generality.

The current benchmarks use learned property oracles to enable scalable reward evaluation during online post-training. High-fidelity simulations and experimental measurements incur substantial computational or experimental costs, limiting their use for evaluation of large candidate populations. Future work could investigate sample-efficient integration of physics-based simulators and experimental feedback. Finally, we use a normalized multi-property reward to study the core distribution-matching mechanism. Incorporating task-specific preferences, constraints, or Pareto-aware objectives would further broaden the range of inverse-design settings addressed by GRAPHFDM.