CONTROL-GUIDED MOLECULAR DIFFUSION FOR TRAINING-FREE CONDITIONAL 3D MOLECULE GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Training-free diffusion guidance has attracted increasing attention in conditional 3D molecule generation for its flexibility across new design objectives. However, existing methods largely rely on gradients, while many important molecular constraints are non-differentiable. To handle these, prior works typically approximate gradients using zeroth-order estimation or differentiable surrogates, introducing errors during diffusion and consequently limiting their effectiveness. In this paper, we tackle this challenge from a new perspective. We propose Control-Guided Molecular Diffusion (CGMD), a framework that reformulates training-free molecular guidance as a stochastic optimal control problem and derives an expression for the guidance term that requires only forward evaluations from non-differentiable evaluators, fundamentally bypassing the additional errors brought by gradient recovery. Beyond non-differentiable evaluators, we find that CGMD's control formulation also naturally accommodates differentiable constraints, and further design a hierarchy-preserving control mechanism to handle different constraints. Experiments in non-differentiable and mixed-guidance settings demonstrate the efficacy of our framework.

1 INTRODUCTION

Conditional 3D molecule de novo generation aims to generate three-dimensional molecular structures that satisfy desired properties and constraints (Bao et al., 2023; Xu et al., 2023), which is of central importance in applications such as drug discovery (Hajduk & Greer, 2007; Mandal et al., 2009) and material design (Kang et al., 2006; Pyzer-Knapp et al., 2015). Recently, diffusion models have become widely adopted for 3D molecule generation, showing strong performance in modeling complex molecular distributions and generating high-quality 3D molecular structures (Hoogeboom et al., 2022; Xu et al., 2023; Shen et al., 2025). To further enable diffusion models to generate molecules with desired properties, several methods incorporate property labels or target conditions during training or fine-tuning (Hoogeboom et al., 2022; Xu et al., 2023). Yet, as molecular design objectives often change, these conditional 3D molecule generation models require retraining or fine-tuning for each new objective, which demands substantial effort and costly data collection (Han et al., 2024). This has motivated growing interest in training-free conditional molecular diffusion, which directly adopts off-the-shelf unconditional diffusion models and incorporates new design constraints at inference time, avoiding the need to retrain or fine-tune models for each new target (Shen et al., 2025; Ye et al., 2024; Han et al., 2024).

Despite this progress, existing training-free conditional 3D molecule generation models predominantly rely on gradients and focus on differentiable constraints (Ye et al., 2024; Han et al., 2024). However, in practice, key molecular design constraints are often examined via non-differentiable chemistry evaluators, such as quantum-chemistry-based force calculations (Bannwarth et al., 2019; Froitzheim et al., 2025; Shen et al., 2025), which return only function values and do not provide gradients. To incorporate such evaluators into diffusion guidance, existing methods usually try to recover gradients first, such as by applying zeroth-order methods to estimate gradients, or using a differentiable surrogate model to approximate the evaluator and adopting its gradients as guidance (Shen et al., 2025; Anstine et al., 2025). Yet,

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.

as zeroth-order estimation often suffers from high variance (Duchi et al., 2015; Liu et al., 2020), and surrogate models can be misaligned with the true evaluator (Tossou et al., 2024; Persson, 2012), the effectiveness of these methods is often limited (as shown in Fig. 1). Essentially, this limitation arises from forcing all guidance through gradients, even when no such gradients are naturally available. This introduces extra gradient-estimation errors into the diffusion process and can ultimately degrade guidance quality. Therefore, in this paper, we aim to circumvent this issue and investigate *whether we can bypass the gradient-estimation detour and construct guidance directly from non-differentiable evaluator queries*.

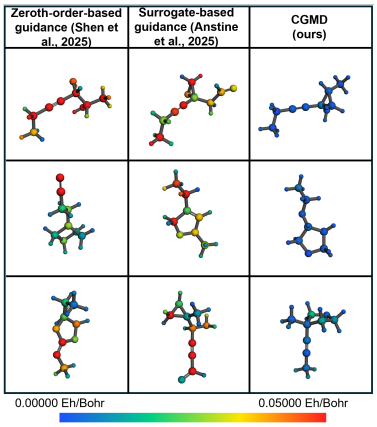


Figure 1: Qualitative comparison between gradient-estimation-based methods, including zeroth-order methods (Shen et al., 2025), surrogate-based methods (Anstine et al., 2025), and CGMD (Ours). Atom colors represent xTB force magnitude in Eh/Bohr, with blue indicating lower and red higher residual forces. As shown, CGMD-generated molecules have smaller forces and geometries closer to local equilibrium, which suggests more stable molecular structures.

To answer this question, we revisit training-free guidance for 3D molecular diffusion from a different angle. At each denoising step, guidance nudges the partially generated 3D molecular structure slightly toward the target property; these small nudges accumulate over the diffusion process and eventually steer the generation to a molecule that satisfies the design objective. This is, structurally, a control problem: the diffusion generation process corresponds to a stochastic dynamical system, and the guidance acts as an external control signal that steers it toward the target at each step. Based on this observation, diffusion guidance can be reformulated as a control problem, and motivated by this, we propose the **Control-Guided Molecular Diffusion (CGMD)** framework. This control-centric perspective opens a new possibility: it allows us to leverage optimal control theory (Merton, 1975; Theodorou et al., 2010) to analyze the guidance signal. By further exploiting the properties of the diffusion process itself, we find that non-differentiable guidance no longer needs to be routed through gradients; instead, it admits an analytical solution in the form of an expectation, which depends *only* on evaluator queries. This thus makes it applicable to non-differentiable molecular evaluators. However, realizing this expectation in practice remains non-trivial: naively evaluating it can require substantial evaluation queries, and each query invokes a non-trivial quantum-chemistry calculation (Bannwarth et al., 2019). This cost is further amplified in practice, where molecular design tasks typically require generating hundreds to thousands of molecule candidates per task to support downstream screening (Bilodeau et al., 2022; Gorgulla et al., 2020), making a naive realization computationally prohibitive. To address this, we exploit the Markov structure of the reverse diffusion to develop an evaluator-informed sampling strategy, allowing evaluator queries made in earlier steps to inform the current sampling step, providing a practical realization of non-differentiable guidance for 3D molecule generation.

Having established non-differentiable guidance as control inputs, a natural follow-up arises: *Can CGMD also handle the design scenarios where different design objectives, including non-differentiable and differentiable ones, are imposed simultaneously?* Specifically, in real-world molecular design, these objectives typically follow a natural ordering: Molecular stability constraints (often examined through non-differentiable evaluators) are first preserved to provide stable molecular geometries, upon which property constraints (typically evaluated via differentiable models) are further refined toward target values (Schneider et al., 2020; Keserü & Makara, 2009; Shen et al., 2025). To address this, we first show that the same control-centric formulation naturally extends to differentiable guidance, which can also be interpreted as control inputs applied to the same diffusion dynamics. Then, with both types of guidance now expressed in a unified control space, we can reason about how to combine them in a principled manner. Rather than simply summing them, we treat them as control inputs that are allocated and composed according to the natural hierarchy between stability and property objectives. Based on this insight, we further design a hierarchy-preserving control allocation mechanism tailored to 3D molecule design. In this formulation, stability control is applied in full to establish the foundational layer of control, while the property control is assigned the subspace orthogonal to it, refining the molecular design within the residual freedom that the foundational layer leaves unaffected. This way, multiple constraints under mixed-guidance settings can be coherently combined to achieve the desired design objectives.

Overall, our contributions are threefold. (1) From a new perspective, we reformulate training-free guidance for 3D molecular diffusion as a control problem, and based on careful theoretical derivation, we design CGMD, a novel framework that enables guidance from non-differentiable evaluators in conditional 3D molecule generation without requiring any gradient estimation. (2) To practically realize this framework, we exploit the structural properties of the molecule diffusion process to enable effective realization of the resulting non-differentiable guidance. Moreover, we find that the same control formulation also accommodates differentiable constraints, on top of which we design a hierarchy-preserving control allocation mechanism that coordinates different constraints according to their natural hierarchy in molecular design. (3) Experiments demonstrate that CGMD achieves superior performance in both non-differentiable and mixed-guidance settings.

2 Related Work

Conditional 3D molecule generation aims to directly generate three-dimensional molecular structures that satisfy desired properties. Recently, diffusion-based methods (Hoogeboom et al., 2022; Xu et al., 2023; Bao et al., 2023; Ye et al., 2024; Han et al., 2024; Shen et al., 2025; Sun et al., 2025; Li et al., 2026; Ben-Hamu et al., 2024; Zhou et al., 2025; Veljković et al., 2026) have become the dominant approach to this problem. Given a molecular diffusion model, a naive strategy for conditional generation is to first generate molecules and then post-optimize them. However, because many constraints, such as molecular properties and stability, are coupled through atomic geometry, enforcing one constraint post hoc may disrupt the original molecular structure and violate other constraints (Shen et al., 2025; Han et al., 2024). To mitigate this, numerous recent works bias the generative process toward satisfying constraints (Shen et al., 2025; Zhou et al., 2025; Li et al., 2026). Among existing approaches, many incorporate conditioning as part of the optimization process in 3D molecular diffusion (Hoogeboom et al., 2022; Xu et al., 2023; Sun et al., 2025), while some steer the diffusion process directly at inference time without additional optimization (Ye et al., 2024; Han et al., 2024; Shen et al., 2025). Among these training-free methods, most (Ye et al., 2024; Han et al., 2024) focus on differentiable guidance, yet many practically important molecular constraints are evaluated through non-differentiable evaluators such as xTB-based force calculations (Froitzheim et al., 2025; Shen et al., 2025). To incorporate such evaluators into diffusion guidance, existing methods still attempt to recover a gradient first (Shen et al., 2025; Anstine et al., 2025). One direct approach is to employ a differentiable surrogate to approximate the non-differentiable evaluator and use gradients of this surrogate as guidance (Cho et al., 2023; Shen et al., 2025), while a more recent approach applies zeroth-order gradient estimation (Shen et al., 2025), which approximates the gradient through random perturbations of the evaluator input. Yet, their effectiveness can be essentially limited by errors introduced in this extra gradient-recovery step: zeroth-order estimates often suffer from high variance (Duchi et al., 2015; Liu et al., 2020), while surrogate models may deviate from the true evaluator (Tossou et al., 2024; Persson, 2012), causing misaligned guidance. Motivated by this limitation, we propose CGMD, a framework that bypasses the gradient-recovery step entirely and handles non-differentiable guidance using only forward evaluator queries.

Control systems (Kirk, 2004; Theodorou et al., 2010) originated in automatic regulation of dynamical systems, and have since been studied in different areas (Pandey et al., 2025; Huang et al., 2024b; Havens et al., 2025; Domingo i Enrich et al., 2025; Hui et al., 2026; Yong & Zhou, 1999; Fleming & Rishel, 2012; Kappen, 2005). As diffusion models define stochastic dynamical processes, recent works have explored control-theoretic formulations for diffusion-based generation, such as super-resolution (Pandey et al., 2025), music generation (Huang et al., 2024b), and preference alignment (Domingo i Enrich et al., 2025). However, these methods are not well suited to training-free conditional 3D molecule generation, where non-differentiable chemistry evaluators can be very costly to query and heterogeneous molecular constraints need to be incorporated and coordinated according to their distinct roles in molecular design. Unlike these methods, we develop a novel optimal-control-based framework tailored to training-free conditional 3D molecule generation, which for the first time enables practical and effective non-differentiable step-wise guidance without gradient estimation, while preserving the natural hierarchy among heterogeneous molecular constraints in 3D molecule generation.

3 BACKGROUND

Diffusion process for 3D molecule generation. Diffusion models for 3D molecule generation (Hoogeboom et al., 2022; Xu et al., 2023) typically represent an N-atom molecule as a point cloud $\mathcal{G} = (\mathbf{x}, \mathbf{h}) \in \mathbb{R}^{N\times(3+d)}$, where $\mathbf{x} \in \mathbb{R}^{N\times 3}$ corresponds to the N atoms' 3D coordinates and $\mathbf{h} \in \mathbb{R}^{N\times d}$ are the d-dimensional atom features, respectively. In the generation process of a 3D molecule, the diffusion model gradually recovers the molecular state from pure noise via the reverse-time diffusion process. Formally, following Song et al. (2021) and Bao et al. (2023), this reverse process can be modeled via stochastic differential equation (SDE):

$$\mathrm{d}\mathcal{G}_t = \left[-\tfrac{1}{2}\beta(t)\mathcal{G}_t - \beta(t)\nabla_{\mathcal{G}_t}\log p_t(\mathcal{G}_t)\right]\mathrm{d}t + \sqrt{\beta(t)}\,\mathrm{d}\bar{\mathbf{B}}_t, \tag{1}$$

where $\nabla_{\mathcal{G}_t}\log p_t(\mathcal{G}_t)$ is the score function at time t estimated by the diffusion model, $\beta(t)$ is the diffusion noise schedule, $\bar{\mathbf{B}}_t$ represents a reverse-time Brownian motion, and t runs from T to 0, starting from pure noise $\mathcal{G}_T$ and recovering the clean molecule $\mathcal{G}_0$.

Training-free guidance for conditional 3D molecule generation. Given a molecular design target $\mathbf{y}$ and its constraint evaluator $L(\cdot)$ (with lower evaluation values indicating better satisfaction of the design objective), conditional 3D molecule generation aims to sample molecular structures $\mathcal{G}$ from a target distribution favoring molecules that satisfy the target. Following common practice (Han et al., 2024; LeCun et al., 2006; Song et al., 2023), the conditional likelihood can be modeled as $p(\mathbf{y} \mid \mathcal{G}) \propto \exp(-L(\mathcal{G}))$, and the target distribution is:

$$p(\mathcal{G} \mid \mathbf{y}) \propto p(\mathcal{G})\exp\big(-L(\mathcal{G})\big), \tag{2}$$

where $p(\mathcal{G})$ is the unconditional molecular distribution from the off-the-shelf unconditional 3D molecule diffusion model, and the exponential weight biases generation toward molecular configurations with lower constraint values. Instead of retraining the diffusion model for every new design objective, training-free guidance gradually steers the unconditional reverse process in Eq. 1 toward the desired target by injecting a condition-dependent guidance term $\mathbf{s}_t$ at each guidance step t:

$$\mathrm{d}\mathcal{G}_t = \left[-\tfrac{1}{2}\beta(t)\mathcal{G}_t - \beta(t)\big(\nabla_{\mathcal{G}_t}\log p_t(\mathcal{G}_t) + \mathbf{s}_t\big)\right]\mathrm{d}t + \sqrt{\beta(t)}\,\mathrm{d}\bar{\mathbf{B}}_t. \tag{3}$$

In existing training-free 3D molecule generation methods, $\mathbf{s}_t$ is typically constructed via the gradient of L evaluated on the denoised prediction (Ye et al., 2024; Han et al., 2024). When L is non-differentiable, this gradient is instead approximated through zeroth-order estimation or surrogate model (Shen et al., 2025; Anstine et al., 2025). However, as shown in Fig. 1, forcing non-differentiable feedback into a gradient-based guidance form can be unreliable, which introduces an additional source of error into the already stochastic reverse process.

4 METHOD

In this paper, we present CGMD, a training-free framework for conditional 3D molecule generation that constructs the guidance term directly from non-differentiable evaluator queries, without requiring gradients. Our framework builds on a reformulation of diffusion guidance as a control problem on the reverse dynamics from a novel perspective, where the guidance can be interpreted as a control input. Building on this reformulation, we can derive the analytical solution for non-differentiable control that depends only on evaluator queries, and further leverage the structural property of molecule diffusion to turn this analytical expression into a practical realization (Sec. 4.1). Beyond non-differentiable guidance, we show that differentiable properties can also be admitted to the control formulation, naturally unifying the two types of guidance under a single framework. This unification, in turn, opens up a structured way of composing multiple guidance signals, based on which we develop a hierarchy-preserving control mechanism that handles constraints while respecting the layered design hierarchy (Sec. 4.2).

4.1 TRAINING-FREE NON-DIFFERENTIABLE GUIDANCE VIA CONTROL REFORMULATION

Reformulating training-free molecular guidance as an optimal control problem. To effectively incorporate non-differentiable constraints into 3D molecule generation at inference time, we seek a solution that bypasses the intermediate step of gradient estimation and instead directly uses the evaluator feedback. Achieving this, however, is non-trivial. In training-free 3D molecule generation, steering the denoising process at each step with non-differentiable evaluator feedback has relied on

surrogates or gradient estimation. In this work, we make a key observation: the denoising process of 3D molecule generation is actually a time-evolving stochastic dynamical system, which fits naturally within the framework of optimal control. This connection inspires us to explore whether formulating non-differentiable guidance as an optimal control problem could provide a principled route to bypass gradient estimation entirely.

To make the correspondence more obvious, we rewrite the reverse process of conditional 3D molecule generation in Eq. 3 as a standard controlled stochastic dynamical system (Theodorou et al., 2010). Specifically, by defining $\mathbf{z}_t = \mathcal{G}_{T-t}$ (so that t now runs forward from 0 to T, and $\mathbf{z}_T$ denotes the terminal clean molecular state), we can rewrite Eq. 3 (with details shown in Appendix F.1) as:

$$\mathrm{d}\mathbf{z}_t = \mathbf{b}(\mathbf{z}_t, t)\,\mathrm{d}t + g(t)\mathbf{c}_t\,\mathrm{d}t + g(t)\mathrm{d}\mathbf{B}_t, \tag{4}$$

where $\mathrm{d}\mathbf{B}_t$ is a standard Brownian motion, $\mathbf{b}(\mathbf{z}_t, t) = \frac{1}{2}\beta(T-t)\mathbf{z}_t + \beta(T-t)\nabla_{\mathbf{z}_t}\log p_{T-t}(\mathbf{z}_t)$, $g(t) = \sqrt{\beta(T-t)}$, $\mathbf{c}_t = \sqrt{\beta(T-t)}\mathbf{s}_{T-t}$, and $\mathrm{d}t$ is an infinitesimal timestep. Based on this reformulation, the diffusion guidance term is modeled as the control input to the dynamical system. We can then solve for the step-wise optimal control input $\mathbf{c}_t$ (and hence the guidance term $\mathbf{s}_t$) leveraging tools from optimal control theory (Theodorou et al., 2010; Kirk, 2004). Following the standard formulation (Theodorou et al., 2010; Kirk, 2004), the optimal control term $\mathbf{c}_t^*$ can be defined as the minimizer of the cost function $\mathcal{C}_{\mathbf{c}}(\mathbf{z}_t, t)$:

$$\mathcal{C}_{\mathbf{c}}(\mathbf{z}_t, t) = \mathbb{E}\left[L(\mathbf{z}_T) + \int_t^T \tfrac{1}{2}\|\mathbf{c}_s\|_2^2\,\mathrm{d}s \,\middle|\, \mathbf{z}_t\right], \tag{5}$$

where the terminal cost $L(\mathbf{z}_T)$ encourages generated molecules to satisfy the constraint, and the quadratic running cost $\int_t^T \frac{1}{2}\|\mathbf{c}_s\|_2^2\,\mathrm{d}s$ regularizes control inputs, preventing the control input from deviating excessively from the pretrained molecular diffusion dynamics.

Deriving analytical solution for optimal control. Having formulated the molecule diffusion guidance as an optimal control problem, our next goal is to derive the optimal control $\mathbf{c}_t^*$ in a manner directly applicable to non-differentiable molecular evaluators. Specifically, leveraging the Hamilton-Jacobi-Bellman (HJB) equation (Yong & Zhou, 1999) and the Feynman–Kac formula (Oksendal, 2013) from stochastic optimal control theory, we can derive the analytical form of $\mathbf{c}_t^*$ as (see Appendix F.2 for detailed derivation):

$$\mathbf{c}_t^*\mathrm{d}t = g(t)\nabla_{\mathbf{z}_t}\log\mathbb{E}_{\mathcal{P}_0}\big[\exp\big(-L(\mathbf{z}_T)\big) \mid \mathbf{z}_t\big]\,\mathrm{d}t, \tag{6}$$

where $\mathcal{P}_0$ represents the distribution induced by the uncontrolled diffusion process (i.e., setting $\mathbf{c}_t = 0$ in Eq. 4). Directly applying Eq. 6, however, still requires computing the gradient with respect to $\mathbf{z}_t$. To address this, we examine the structure of Eq. 6 and the diffusion process's intrinsic properties. The derivation in Appendix F.3 shows that, by leveraging reverse diffusion's Markov property and Stein's lemma (Stein, 1981), we can obtain $\mathbf{c}_t^*$ without requiring gradient computation as:

$$\mathbf{c}_t^*\,\mathrm{d}t = \frac{\mathbb{E}_{\mathcal{P}_0}\big[\exp\big(-L(\mathbf{z}_T)\big)\,\mathrm{d}\mathbf{B}_t \mid \mathbf{z}_t\big]}{\mathbb{E}_{\mathcal{P}_0}\big[\exp\big(-L(\mathbf{z}_T)\big) \mid \mathbf{z}_t\big]}. \tag{7}$$

Compared to Eq. 6, Eq. 7 eliminates the need to compute any gradient with respect to $\mathbf{z}_t$, leaving the computation of $\mathbf{c}_t^*$ dependent only on forward evaluations using the non-differentiable evaluator. As a result, the uncertainty in obtaining the control term (and hence the guidance term for non-differentiable evaluators) is now confined to the inherent stochasticity of the diffusion process, and is no longer compounded by the additional, inevitable error introduced by estimating gradients for non-differentiable evaluators. This removes a persistent source of extra noise and bias that has previously limited non-differentiable guidance for 3D molecule generation. Moreover, we prove in Theorem 1 that steering the controlled dynamics in Eq. 4 with the optimal control $\mathbf{c}_t^*$ in Eq. 7 still produces samples from the target distribution in Eq. 2 (see Appendix F.4).

Exploiting Markov structure for practical realization. While Eq. 7 provides an analytical expression for $\mathbf{c}_t^*$ without requiring gradients, it still involves a conditional expectation over the diffusion process. A straightforward realization is Monte Carlo sampling: at each guidance step, one samples Brownian increments $\mathrm{d}\mathbf{B}_t$, obtains the corresponding clean estimates via Tweedie's formula (Efron, 2011), queries the evaluator, and aggregates the weighted increments (see Appendix D.1 for details). However, this naive realization can be prohibitive in 3D molecule generation. Physically valid molecular geometries are constrained by chemistry rules such as bond lengths, bond angles, and valence, and can occupy only a small subset of the structure space. Blindly sampled perturbations

can easily move the clean estimates away from this valid region, producing invalid configurations with poor evaluator results that have little effect in computing $\mathbf{c}_t^*$. As a result, obtaining a reliable realization may require many evaluator queries (see Fig. 11), which is expensive given the cost of quantum-chemistry evaluation and the large number of molecule candidates typically needed for downstream screening.

The above challenge motivates a natural question: instead of drawing samples blindly from the original distribution of $\mathrm{d}\mathbf{B}_t$, can we bias regions that may contribute more to the expectation in Eq. 7? In principle, this requires knowing which regions of the current sampling space are likely to yield large exponential weights $\exp(-L(\mathbf{z}_T))$ in Eq. 7. However, naively learning such a sampling distribution would again require many evaluator queries per guidance step, defeating the purpose.

Constructing evaluator-informed sampling distribution. Fortunately, the diffusion process itself provides a way to reveal such a sampling distribution. The key observation is that the structure of the reverse diffusion process allows evaluator queries already made at earlier denoising steps to be reused in the current step, and thus inform sampling distribution. Suppose that at the current step t, we have already conducted evaluator queries from earlier denoising steps. From these earlier queries, we retain a collection of cached perturbations together with their associated quantities: for the j-th perturbation, we have the terminal clean-state estimate $\hat{\mathbf{z}}_T^{(j)}$ and the corresponding evaluator result $L^{(j)} = L(\hat{\mathbf{z}}_T^{(j)})$. To reuse these cached samples in the current step t, we seek, for each $\hat{\mathbf{z}}_T^{(j)}$, a state $\bar{\mathbf{z}}_t^{(j)}$ at step t whose clean-state estimate is consistent with $\hat{\mathbf{z}}_T^{(j)}$. The corresponding evaluator weight $\exp(-L^{(j)})$ evaluated at earlier step can then carry over to $\bar{\mathbf{z}}_t^{(j)}$ at the current step, without any new evaluator query. By the Markov structure of the reverse diffusion, this is to solve an implicit function:

$$\bar{\mathbf{z}}_t^{(j)} = f\big(\bar{\mathbf{z}}_t^{(j)}; \hat{\mathbf{z}}_T^{(j)}\big) \;=\; \sqrt{\bar{\alpha}_{T-t}}\,\hat{\mathbf{z}}_T^{(j)} + \sigma_t\,\hat{\boldsymbol{\epsilon}}_\theta\big(\bar{\mathbf{z}}_t^{(j)},\, T-t\big), \tag{8}$$

where $\hat{\boldsymbol{\epsilon}}_\theta$ denotes the noise prediction induced by the diffusion backbone, $\bar{\alpha}_{T-t} = \exp(-\int_0^{T-t}\beta(s)\,\mathrm{d}s)$, and $\sigma_t = \sqrt{1-\bar{\alpha}_{T-t}}$. We address Eq. 8 in Appendix D.2. In this way, we obtain a set of evaluator-informed anchors $\{\big(\bar{\mathbf{z}}_t^{(j)}, \exp(-L^{(j)})\big)\}_j$ at the current step, with known weights inherited from earlier steps and thus requiring no new evaluator queries. These anchors carry information about the sampling distribution: anchors with higher weights correspond to regions of the current sampling space that contribute more meaningfully to the conditional expectation, while anchors with lower weights indicate regions of smaller contribution. With these anchors, we can form the sampling distribution q_{ϕ_t} for the current step by fitting a parametric sampling distribution. We adopt a Student-t mixture (a mixture of Student-t components) as the parametric family (Zhou et al., 2014), due to its high expressiveness (Zeevi & Meir, 1997; Peel & McLachlan, 2000). Detailed fitting procedure is provided in Appendix D.3.

Realizing the control from the sampling distribution. Having obtained the evaluator-informed sampling distribution q_{ϕ_t} at the current step, we can then effectively instantiate Eq. 7 via Tweedie's formula. We draw K perturbation samples $\mathrm{d}\mathbf{B}_t^{(k)}$ independently from a proposal distribution $q_{\phi_t} = \sum_{a=1}^{M} \lambda_t^{(a)} t_\nu(\cdot; \boldsymbol{\mu}_t^{(a)}, \boldsymbol{\Sigma}_t)$ (Zhou et al., 2014), query the evaluator on their corresponding terminal clean-state estimates $\hat{\mathbf{z}}_T^{(k)} = \mathcal{T}_t(\mathbf{z}_t^{(k)})$, compute the weights according to Eq. 7, and realize the non-differentiable control increment as:

$$\widehat{\mathbf{c}}_t\,\mathrm{d}t = \frac{\sum_{k=1}^{K} r_t^{(k)}\,\mathrm{d}\mathbf{B}_t^{(k)}}{\sum_{k=1}^{K} r_t^{(k)}}, \quad r_t^{(k)} = \exp\Big(-L(\hat{\mathbf{z}}_T^{(k)})\Big)\,\frac{p_0(\mathrm{d}\mathbf{B}_t^{(k)})}{q_{\phi_t}(\mathrm{d}\mathbf{B}_t^{(k)})}, \tag{9}$$

where p_0 is the original Brownian-increment distribution and $p_0(\mathrm{d}\mathbf{B}_t^{(k)})/q_{\phi_t}(\mathrm{d}\mathbf{B}_t^{(k)})$ is the correction term for the change of sampling distribution. We further show in Appendix F.5 that the realization in Eq. 9 targets the same expectation as the naive Monte Carlo realization, and can more effectively yield stable computation of Eq. 7 as q_{ϕ_t} can better capture high-weight perturbations.

4.2 Multi-Constraint Guidance via Hierarchy-Preserving Control

Having developed guidance for non-differentiable evaluators from the optimal control perspective above, we now consider how CGMD extends to settings involving multiple different constraints. In realistic molecular design tasks, differentiable property objectives and non-differentiable chemistry evaluators are often imposed simultaneously (Shen et al., 2025; Han et al., 2024). These constraints

typically follow a natural hierarchy observable in real-world molecular design pipelines: molecular-stability constraints are typically first preserved to establish a chemically valid molecular foundation, on top of which property constraints are further refined toward target property (Schneider et al., 2020; Keserü & Makara, 2009). To extend CGMD to such settings, here we address two questions: (1) can differentiable constraints be handled within the same control formulation, and (2) how can multiple constraints be coordinated under this formulation while respecting their natural hierarchy?

Formulating differentiable guidance via control perspective. To answer the first question, we show that differentiable guidance actually also fits well within the same control formulation. Re-examining the optimal control solution in Eq. 6, we find that we can naturally obtain a gradient-based optimal control for differentiable conditions as:

$$\mathbf{c}_t^* \mathrm{d}t = g(t)\nabla_{\mathbf{z}_t} \log p_t(\mathbf{y} \mid \mathbf{z}_t)\mathrm{d}t = g(t)\nabla_{\mathbf{z}_t} \log \int p(\mathbf{y}|\mathbf{z}_T)p(\mathbf{z}_T|\mathbf{z}_t)\mathrm{d}\mathbf{z}_T \, \mathrm{d}t, \tag{10}$$

where $\mathbf{y}$ is the molecular design target. At this point, differentiable constraints have also been brought into the same theoretical control form as non-differentiable constraints. To obtain practical realization of Eq. 10, we use the Tweedie formula (Efron, 2011) to map $\mathbf{z}_t$ to the terminal clean-state estimate $\hat{\mathbf{z}}_T$ and approximate the marginal likelihood by $p(\mathbf{y} \mid \hat{\mathbf{z}}_T)$. We can then obtain a realization of the control term in gradient-based form as:

$$\mathbf{c}_t = g(t)\nabla_{\mathbf{z}_t} \log p(\mathbf{y} \mid \hat{\mathbf{z}}_T). \tag{11}$$

The detailed derivations for Eq. 10 and Eq. 11 are provided in Appendix F.6.

Coordinating mixed guidance via hierarchy-preserving control. With both differentiable and non-differentiable guidance now expressed as control inputs in the same control space, we can finally coordinate them according to their roles in molecule design. A naive strategy is to add them directly. However, this flat fusion ignores the layered hierarchy in molecule design: molecular stability guidance constitutes the foundational control, while property guidance provides refinement on top of it. Simply summing them can cause the refinement control to corrupt the foundational signal, breaking the layered hierarchy at every guidance step.

To address this, we draw inspiration from *hierarchical control* (Siciliano & Slotine, 1991; Nakamura et al., 1987), which coordinates different controls acting on shared dynamics by letting the foundation control act in its required direction and restricting the refinement to the redundancy left by foundation controls so that foundation effects are preserved. Following this principle, we allocate the two guidance controls in a hierarchy-preserving manner: the foundational stability control is applied in full, while the refinement property control is restricted to the null space of the foundational control, where it contributes to design target without altering the foundational signal. Specifically, the refinement property control $\mathbf{c}_t^r$ is modified as:

$$\tilde{\mathbf{c}}_t^r = \mathbf{P}_{\mathbf{c}_t^f}^{\perp} \mathbf{c}_t^r, \quad \mathbf{P}_{\mathbf{c}_t^f}^{\perp} = \mathbf{I} - \frac{\mathbf{c}_t^f (\mathbf{c}_t^f)^\top}{\|\mathbf{c}_t^f\|_2^2}, \tag{12}$$

where $\mathbf{c}_t^f$ and $\mathbf{c}_t^r$ denote foundation and refinement controls, and $\mathbf{P}_{\mathbf{c}_t^f}^{\perp}$ projects $\mathbf{c}_t^r$ onto the subspace orthogonal to $\mathbf{c}_t^f$. The foundational control $\mathbf{c}_t^f$ is kept unchanged, and the final control is obtained by combining $\mathbf{c}_t^f$ and $\tilde{\mathbf{c}}_t^r$. We provide more details about this projection in Appendix F.7. By this construction, property guidance refines generation while preserving the foundational stability control.

4.3 Overall Inference

During inference, given a pretrained 3D molecular diffusion model and user-specified molecular constraints, CGMD guides the reverse diffusion process to generate 3D molecules that satisfy the specified constraints by injecting the appropriate control term $\mathbf{c}_t$ at each guidance step (as illustrated in Fig. 2). When only non-differentiable constraints are required, $\mathbf{c}_t$ is obtained via Eq. 9. When both types of constraints are present, we obtain control terms via Eq. 9 and Eq. 11, and combine them via the hierarchy-preserving control. The complete step-wise procedure is given in Algorithm 1 in Appendix E.1.

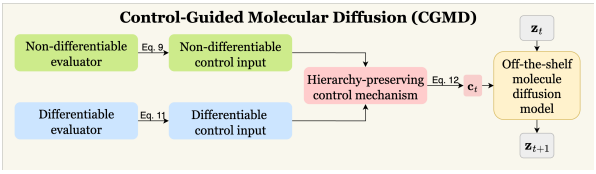


Figure 2: Overall inference pipeline of CGMD.

5 EXPERIMENTS

Datasets and evaluation settings. Following Shen et al. (2025), we evaluate CGMD on QM9 (Ramakrishnan et al., 2014) under both non-differentiable and mixed-guidance settings, and further explore its performance on larger molecules in GEOM (Axelrod & Gomez-Bombarelli, 2022) in the non-differentiable setting. The QM9 dataset covers 134K molecules, while the GEOM dataset scales to 450K molecules. Following ChemGuide (Shen et al., 2025), the non-differentiable setting considers xTB-based force guidance without accessing force gradients. Specifically, we first follow (Shen et al., 2025) to report results using GFN2-xTB (Bannwarth et al., 2019). We additionally evaluate our method using g-xTB (Froitzheim et al., 2025), the current state-of-the-art xTB-based force calculator. The guidance aims to reduce the atomic forces toward zero, thereby moving generated molecules closer to local equilibrium with lower energy and optimizing molecular stability (Shen et al., 2025). The mixed-guidance setting combines xTB-based force guidance with differentiable property guidance. We evaluate the differentiable guidance component on QM9 following ChemGuide (Shen et al., 2025). Six property objectives from QM9 are evaluated: static polarizability norm α, dipole moment norm μ, room-temperature heat capacity C_v, highest occupied molecular orbital energy $\varepsilon_{\mathrm{HOMO}}$, lowest unoccupied molecular orbital energy $\varepsilon_{\mathrm{LUMO}}$, and the HOMO–LUMO gap $\Delta\varepsilon$. More details are in Appendix E.2.

Evaluation metrics. For force guidance, we follow ChemGuide (Shen et al., 2025) and report the root mean square of atomic forces, the energy above the ground state, validity, uniqueness, atom stability on both QM9 and GEOM, and molecule stability on QM9. For mixed guidance, we report MAE, force RMS, and energy above the ground state. More details are in Appendix E.3.

Implementation details. Following Shen et al. (2025), we use GeoLDM (Xu et al., 2023) as an off-the-shelf 3D molecular diffusion backbone. To further evaluate the effectiveness of CGMD with a different backbone, we also test our framework with a widely used model, EDM (Hoogeboom et al., 2022). All methods requiring evaluator queries use the same per-step xTB query budget. More implementation details are in Appendix E.2.

Table 1: Results with GFN2-xTB force guidance on QM9.

Model	Method	Force RMS ↓	Energy above Ground State ↓	Validity ↑	Uniqueness ↑	Atom Stability ↑	Molecule Stability ↑
GeoLDM	Unconditional	0.0111	0.0050	89.80%	**100.00%**	98.93%	88.80%
	ChemGuide (w/ SPSA) (Shen et al., 2025; Spall, 1992)	0.0104	0.0042	91.40%	**100.00%**	99.02%	90.60%
	ChemGuide (w/ RGF) (Shen et al., 2025; Nesterov & Spokoiny, 2017)	0.0103	0.0049	90.80%	99.78%	99.01%	90.40%
	ChemGuide (w/ ES) (Shen et al., 2025; Salimans et al., 2017)	0.0105	0.0053	88.20%	**100.00%**	99.05%	90.60%
	AIMNet2 (Anstine et al., 2025)	0.0110	0.0055	89.00%	**100.00%**	98.75%	88.00%
	eSEN-sm-cons. (Levine et al., 2025)	0.0107	0.0055	88.80%	**100.00%**	98.97%	89.20%
	Ours	**0.0068**	**0.0021**	**93.20%**	**100.00%**	**99.19%**	**92.80%**
EDM	Unconditional	0.0114	0.0072	86.60%	**100.00%**	98.53%	83.60%
	ChemGuide (w/ SPSA) (Shen et al., 2025; Spall, 1992)	0.0105	0.0065	85.80%	**100.00%**	98.55%	83.60%
	ChemGuide (w/ RGF) (Shen et al., 2025; Nesterov & Spokoiny, 2017)	0.0108	0.0063	85.20%	**100.00%**	98.59%	83.40%
	ChemGuide (w/ ES) (Shen et al., 2025; Salimans et al., 2017)	0.0106	0.0065	85.40%	**100.00%**	98.52%	83.40%
	AIMNet2 (Anstine et al., 2025)	0.0113	0.0078	86.60%	**100.00%**	98.29%	81.60%
	eSEN-sm-cons. (Levine et al., 2025)	0.0111	0.0076	86.20%	**100.00%**	98.21%	81.60%
	Ours	**0.0096**	**0.0053**	**87.20%**	100.00%	**98.69%**	**86.80%**

Table 2: Results with GFN2-xTB force guidance on GEOM.

Model	Method	Force RMS ↓	Energy above Ground State ↓	Validity ↑	Uniqueness ↑	Atom Stability ↑
GeoLDM	Unconditional	0.0478	0.2248	49.00%	**100.00%**	84.53%
	ChemGuide (w/ SPSA) (Shen et al., 2025; Spall, 1992)	0.0411	0.1935	50.40%	**100.00%**	84.36%
	ChemGuide (w/ RGF) (Shen et al., 2025; Nesterov & Spokoiny, 2017)	0.0427	0.2041	49.60%	**100.00%**	84.32%
	ChemGuide (w/ ES) (Shen et al., 2025; Salimans et al., 2017)	0.0420	0.2085	48.20%	**100.00%**	84.45%
	AIMNet2 (Anstine et al., 2025)	0.0414	0.2283	49.80%	**100.00%**	85.13%
	eSEN-sm-cons. (Levine et al., 2025)	0.0398	0.2213	49.80%	**100.00%**	84.79%
	Ours	**0.0211**	**0.1003**	**59.60%**	**100.00%**	**85.96%**
EDM	Unconditional	0.0742	0.3742	46.40%	**100.00%**	81.22%
	ChemGuide (w/ SPSA) (Shen et al., 2025; Spall, 1992)	0.0557	0.3318	45.20%	**100.00%**	81.25%
	ChemGuide (w/ RGF) (Shen et al., 2025; Nesterov & Spokoiny, 2017)	0.0553	0.3401	43.40%	**100.00%**	81.31%
	ChemGuide (w/ ES) (Shen et al., 2025; Salimans et al., 2017)	0.0564	0.3320	43.60%	**100.00%**	81.33%
	AIMNet2 (Anstine et al., 2025)	0.0559	0.3492	43.80%	**100.00%**	81.35%
	eSEN-sm-cons. (Levine et al., 2025)	0.0535	0.3419	43.80%	**100.00%**	81.14%
	Ours	**0.0411**	**0.2074**	**50.40%**	**100.00%**	**85.08%**

5.1 MAIN RESULTS

We compare CGMD against zeroth-order gradient-estimation methods: ChemGuide (w/ SPSA) is the SPSA-based method (Shen et al., 2025; Spall, 1992), while ChemGuide (w/ RGF) and ChemGuide (w/ ES) replace the SPSA estimator with random-gradient-free (Nesterov & Spokoiny, 2017) and evolution-strategy (Salimans et al., 2017) estimators, respectively. We also compare with differentiable surrogate-based methods using AIMNet2 (Anstine et al., 2025) and eSEN-sm-cons. (Levine et al., 2025). Tables 1 and 2 report results in the non-differentiable setting using GFN2-xTB force guidance on QM9 and GEOM, respectively, while Table 3 reports mixed-guidance results on QM9. The corresponding g-xTB results are provided in Appendix C (Tables 6, 7, and 8). As shown, CGMD achieves the best overall performance.

Table 3: Mixed-guidance results on QM9 with GFN2-xTB force guidance

Model	Method	α			μ			C_v			$\varepsilon_{\text{HOMO}}$			$\varepsilon_{\text{LUMO}}$			$\Delta\varepsilon$		
		MAE↓	Force↓	Energy↓	MAE↓	Force↓	Energy↓	MAE↓	Force↓	Energy↓	MAE↓	Force↓	Energy↓	MAE↓	Force↓	Energy↓	MAE↓	Force↓	Energy↓
GeoLDM	ChemGuide (w/ SPSA) (Shen et al., 2025; Spall, 1992)	2.4860	0.0132	0.0052	0.7246	0.0117	0.0050	0.9369	0.0110	0.0060	0.4445	0.0101	0.0029	0.8099	0.0098	0.0024	0.7041	0.0104	0.0031
	ChemGuide (w/ RGF) (Shen et al., 2025; Nesterov & Spokoiny, 2017)	2.5948	0.0134	0.0064	0.7565	0.0120	0.0052	1.1791	0.0136	0.0065	0.5192	0.0128	0.0036	0.9065	0.0099	0.0024	0.8807	0.0106	0.0055
	ChemGuide (w/ ES) (Shen et al., 2025; Salimans et al., 2017)	2.5547	0.0135	0.0054	0.8363	0.0119	0.0051	1.1611	0.0245	0.0173	0.4856	0.0117	0.0033	0.9763	0.0098	0.0027	0.8669	0.0109	0.0047
	AIMNet2 (Anstine et al., 2025)	2.7666	0.0146	0.0065	0.8775	0.0128	0.0054	1.1623	0.0188	0.0110	0.5401	0.0125	0.0053	0.9568	0.0110	0.0043	0.9186	0.0109	0.0054
	eSEN-sm-cons. (Levine et al., 2025)	2.8124	0.0139	0.0068	0.9012	0.0131	0.0052	1.1895	0.0181	0.0106	0.5568	0.0119	0.0049	0.9347	0.0114	0.0041	0.9472	0.0105	0.0056
	Ours	**2.3475**	**0.0101**	**0.0032**	**0.6559**	**0.0084**	**0.0031**	**0.8585**	**0.0088**	**0.0042**	**0.3460**	**0.0082**	**0.0024**	**0.6757**	**0.0081**	**0.0021**	**0.6664**	**0.0086**	**0.0023**
EDM	ChemGuide (w/ SPSA) (Shen et al., 2025; Spall, 1992)	3.9873	0.0195	0.0093	1.3966	0.0176	0.0091	2.3291	0.0177	0.0114	0.6702	0.0174	0.0088	1.0938	0.0162	0.0073	1.1615	0.0175	0.0130
	ChemGuide (w/ RGF) (Shen et al., 2025; Nesterov & Spokoiny, 2017)	3.8041	0.0197	0.0098	1.4212	0.0177	0.0087	2.2303	0.0172	0.0117	0.7407	0.0168	0.0084	1.1357	0.0165	0.0080	1.3851	0.0167	0.0120
	ChemGuide (w/ ES) (Shen et al., 2025; Salimans et al., 2017)	3.5845	0.0196	0.0101	1.6305	0.0179	0.0090	2.3984	0.0169	0.0112	0.7209	0.0166	0.0087	1.0857	0.0159	0.0102	1.2530	0.0168	0.0128
	AIMNet2 (Anstine et al., 2025)	4.2925	0.0210	0.0102	1.6073	0.0203	0.0096	2.5674	0.0188	0.0126	0.7753	0.0193	0.0094	1.2388	0.0175	0.0098	1.3590	0.0190	0.0138
	eSEN-sm-cons. (Levine et al., 2025)	4.4178	0.0201	0.0107	1.5639	0.0210	0.0092	2.6315	0.0194	0.0121	0.8016	0.0187	0.0091	1.2714	0.0160	0.0094	1.3278	0.0186	0.0142
	Ours	**2.1777**	**0.0124**	**0.0069**	**0.7329**	**0.0114**	**0.0061**	**1.1742**	**0.0107**	**0.0071**	**0.3819**	**0.0104**	**0.0064**	**0.7313**	**0.0108**	**0.0062**	**0.8919**	**0.0112**	**0.0064**

5.2 Ablation Study

Following Shen et al. (2025), we conduct ablations on QM9 with GeoLDM. **More results, ablation studies, further analysis, and qualitative results are provided in Appendices B and C.**

Impact of evaluator-informed sampling. In non-differentiable guidance, CGMD uses evaluator-informed anchors to fit a Student-t mixture distribution, which is then used to sample informative perturbations for estimating the control signal. To validate this design, we evaluate three variants: *w/o evaluator-informed anchors*, which removes the evaluator-informed anchors and naively samples from Student-t mixture; *w/o Student-t mixture*, which replaces the Student-t mixture with a single Gaussian proposal; and *w/o evaluator-informed sampling*, which removes the evaluator-informed sampling strategy and employs the naive Monte Carlo sampling. As shown in Table 4, CGMD achieves superior performance across the metrics, supporting the effectiveness of evaluator-informed sampling.

Table 4: Impact of evaluator-informed sampling on QM9 with GeoLDM under GFN2-xTB force guidance.

Method	Force RMS ↓	Energy ↓	Validity ↑	Uniqueness ↑	Atom Stable. ↑	Mol. Stable. ↑
w/o evaluator-informed anchors	0.0091	0.0037	91.80%	**100.00%**	99.03%	91.00%
w/o Student-t mixture	0.0084	0.0032	92.00%	**100.00%**	98.98%	91.80%
w/o evaluator-informed sampling	0.0093	0.0039	91.80%	**100.00%**	98.99%	90.20%
Ours	**0.0068**	**0.0021**	**93.20%**	**100.00%**	**99.19%**	**92.80%**

Table 5: Impact of hierarchy-preserving control in mixed guidance

Method	α			μ			C_v			$\varepsilon_{\text{HOMO}}$			$\varepsilon_{\text{LUMO}}$			$\Delta\varepsilon$		
	MAE↓	Force↓	Energy↓	MAE↓	Force↓	Energy↓	MAE↓	Force↓	Energy↓	MAE↓	Force↓	Energy↓	MAE↓	Force↓	Energy↓	MAE↓	Force↓	Energy↓
w/o hierarchy-preserving control	2.5084	0.0114	0.0038	0.7660	0.0091	0.0037	0.9485	0.0092	0.0046	0.4487	0.0091	0.0028	0.8519	0.0093	0.0022	0.7283	0.0097	0.0027
Ours	**2.3475**	**0.0101**	**0.0032**	**0.6559**	**0.0084**	**0.0031**	**0.8585**	**0.0088**	**0.0042**	**0.3460**	**0.0082**	**0.0024**	**0.6757**	**0.0081**	**0.0021**	**0.6664**	**0.0086**	**0.0023**

Impact of hierarchy-preserving control. In mixed-guidance settings, CGMD treats force guidance as foundational control and property guidance as refinement control, and uses the hierarchy-preserving control mechanism to handle them. To validate this, we compare CGMD with the *w/o hierarchy-preserving control* variant on QM9 with GeoLDM and GFN2-xTB force guidance; this variant naively combines different control terms via a weighted sum following the ChemGuide evaluation protocol. As shown in Table 5, our method achieves superior performance across reported metrics, showing the effectiveness of the hierarchy-preserving mechanism.

Qualitative results. We qualitatively compare methods on QM9 using GeoLDM with GFN2-xTB force guidance. As shown in Fig. 3, samples from our method have lower atomic forces than those from the compared methods, further demonstrating the effectiveness of the proposed method.

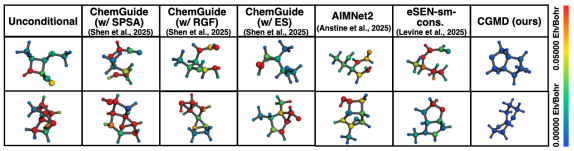


Figure 3: Qualitative comparison with baselines. Cooler colors indicate lower atomic forces.

6 Conclusion

In this paper, we present Control-Guided Molecular Diffusion (CGMD), a training-free framework that addresses the fundamental difficulty of guiding 3D molecular diffusion under non-differentiable constraints. CGMD reformulates training-free molecular guidance into a stochastic optimal control problem, derives an analytical, gradient-free expression for the molecular guidance term, and instantiates it leveraging the Markov structure of the reverse diffusion. The same control formulation also accommodates differentiable constraints, on top of which a hierarchy-preserving control allocation mechanism coordinates multiple constraints by their natural ordering. Experiment results show that CGMD achieves superior performance in both non-differentiable and mixed-guidance settings.

AI USE STATEMENT

In this work, generative AI tools were used for grammatical and typographical corrections, as well as improvements in sentence clarity. We have not used generative AI tools for research ideation, method design, or experiment design. All AI-assisted work was reviewed and verified by the authors.

REFERENCES

Suzanne Ackloo, Rima Al-Awar, Rommie E Amaro, Cheryl H Arrowsmith, Hatylas Azevedo, Robert A Batey, Yoshua Bengio, Ulrich AK Betz, Cristian G Bologa, John D Chodera, et al. Cache (critical assessment of computational hit-finding experiments): A public–private partnership benchmarking initiative to enable the development of computational methods for hit-finding. *Nature Reviews Chemistry*, 6(4):287–295, 2022.

Sergios Agapiou, Omiros Papaspiliopoulos, Daniel Sanz-Alonso, and Andrew M Stuart. Importance sampling: Intrinsic dimension and computational cost. *Statistical Science*, pp. 405–431, 2017.

Dylan M Anstine, Roman Zubatyuk, and Olexandr Isayev. AIMNet2: a neural network potential to meet your neutral, charged, organic, and elemental-organic needs. *Chemical Science*, 16(23): 10228–10244, 2025.

Simon Axelrod and Rafael Gomez-Bombarelli. Geom, energy-annotated molecular conformations for property prediction and molecular generation. *Scientific data*, 9(1):185, 2022.

Christoph Bannwarth, Sebastian Ehlert, and Stefan Grimme. Gfn2-xtb—an accurate and broadly parametrized self-consistent tight-binding quantum chemical method with multipole electrostatics and density-dependent dispersion contributions. *Journal of chemical theory and computation*, 15 (3):1652–1671, 2019.

Fan Bao, Min Zhao, Zhongkai Hao, Peiyao Li, Chongxuan Li, and Jun Zhu. Equivariant energy-guided SDE for inverse molecular design. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=r0otLtOwYW.

Heli Ben-Hamu, Omri Puny, Itai Gat, Brian Karrer, Uriel Singer, and Yaron Lipman. D-flow: Differentiating through flows for controlled generation. In *International Conference on Machine Learning*, pp. 3462–3483. PMLR, 2024.

Camille Bilodeau, Wengong Jin, Tommi Jaakkola, Regina Barzilay, and Klavs F Jensen. Generative models for molecular discovery: Recent advances and challenges. *Wiley Interdisciplinary Reviews: Computational Molecular Science*, 12(5):e1608, 2022.

Sourav Chatterjee and Persi Diaconis. The sample size required in importance sampling. *The Annals of Applied Probability*, 28(2):1099–1135, 2018.

Youngwoo Cho, Seunghoon Yi, Soo Kyung Kim, Hongkee Yoon, and Joonseok Lee. Hybrid diffusions for stable molecular structure generation via explicit energy-based model. In *1st Workshop on the Synergy of Scientific and Machine Learning Modeling@ ICML2023*, 2023.

Fabrizio Costa and Kurt De Grave. Fast neighborhood subgraph pairwise distance kernel. In *Proceedings of the 26th International Conference on Machine Learning*, pp. 255–262. Omnipress; Madison, WI, USA, 2010.

Carles Domingo i Enrich, Michal Drozdzal, Brian Karrer, and Ricky T. Q. Chen. Adjoint matching: Fine-tuning flow and diffusion generative models with memoryless stochastic optimal control. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 53791–53846, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/852f50969a9e523ec41d26f2f68bd456-Paper-Conference.pdf.

John C Duchi, Michael I Jordan, Martin J Wainwright, and Andre Wibisono. Optimal rates for zero-order convex optimization: The power of two function evaluations. *IEEE Transactions on Information Theory*, 61(5):2788–2806, 2015.

Bradley Efron. Tweedie's formula and selection bias. *Journal of the American Statistical Association*, 106(496):1602–1614, 2011.

Wendell H Fleming and Raymond W Rishel. *Deterministic and stochastic optimal control*. Springer Science & Business Media, 2012.

M. J. Frisch, G. W. Trucks, H. B. Schlegel, G. E. Scuseria, M. A. Robb, J. R. Cheeseman, G. Scalmani, V. Barone, G. A. Petersson, H. Nakatsuji, X. Li, M. Caricato, A. V. Marenich, J. Bloino, B. G. Janesko, R. Gomperts, B. Mennucci, H. P. Hratchian, J. V. Ortiz, A. F. Izmaylov, J. L. Sonnenberg, D. Williams-Young, F. Ding, F. Lipparini, F. Egidi, J. Goings, B. Peng, A. Petrone, T. Henderson, D. Ranasinghe, V. G. Zakrzewski, J. Gao, N. Rega, G. Zheng, W. Liang, M. Hada, M. Ehara, K. Toyota, R. Fukuda, J. Hasegawa, M. Ishida, T. Nakajima, Y. Honda, O. Kitao, H. Nakai, T. Vreven, K. Throssell, J. A. Montgomery, Jr., J. E. Peralta, F. Ogliaro, M. J. Bearpark, J. J. Heyd, E. N. Brothers, K. N. Kudin, V. N. Staroverov, T. A. Keith, R. Kobayashi, J. Normand, K. Raghavachari, A. P. Rendell, J. C. Burant, S. S. Iyengar, J. Tomasi, M. Cossi, J. M. Millam, M. Klene, C. Adamo, R. Cammi, J. W. Ochterski, R. L. Martin, K. Morokuma, O. Farkas, J. B. Foresman, and D. J. Fox. Gaussian˜16 Revision C.01, 2016. Gaussian Inc. Wallingford CT.

Thomas Froitzheim, Marcel Müller, Andreas Hansen, and Stefan Grimme. g-xtb: A general-purpose extended tight-binding electronic structure method for the elements h to lr (z= 1–103). *ChemRxiv*, 2025.

Christoph Gorgulla, Andras Boeszoermenyi, Zi-Fu Wang, Patrick D Fischer, Paul W Coote, Krishna M Padmanabha Das, Yehor S Malets, Dmytro S Radchenko, Yurii S Moroz, David A Scott, et al. An open-source drug discovery platform enables ultra-large virtual screens. *Nature*, 580 (7805):663–668, 2020.

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

Xu Han, Caihua Shan, Yifei Shen, Can Xu, Han Yang, Xiang Li, and Dongsheng Li. Training-free multi-objective diffusion model for 3d molecule generation. In *The Twelfth International Conference on Learning Representations*, 2024.

Aaron J Havens, Benjamin Kurt Miller, Bing Yan, Carles Domingo-Enrich, Anuroop Sriram, Daniel S Levine, Brandon M Wood, Bin Hu, Brandon Amos, Brian Karrer, et al. Adjoint sampling: Highly scalable diffusion samplers via adjoint matching. In *International Conference on Machine Learning*, pp. 22204–22237. PMLR, 2025.

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

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

Han Huang, Leilei Sun, Bowen Du, and Weifeng Lv. Learning joint 2-d and 3-d graph diffusion models for complete molecule generation. *IEEE transactions on neural networks and learning systems*, 35(9):11857–11871, 2024a.

Ruili Huang, Miao Xu, Hu Zhu, Catherine Z Chen, Wei Zhu, Emily M Lee, Shihua He, Li Zhang, Jinghua Zhao, Khalida Shamim, et al. Biological activity-based modeling identifies antiviral leads against sars-cov-2. *Nature biotechnology*, 39(6):747–753, 2021.

Yujia Huang, Adishree Ghatare, Yuanzhe Liu, Ziniu Hu, Qinsheng Zhang, Chandramouli Shama Sastry, Siddharth Gururani, Sageev Oore, and Yisong Yue. Symbolic music generation with non-differentiable rule guided diffusion. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 19772–19797. PMLR, 21–27 Jul 2024b. URL https://proceedings.mlr.press/v235/huang24g.html.

Xiaofei Hui, Bo Yan, Haoxuan Qu, Hossein Rahmani, and Jun Liu. Training-free controllable human motion generation under heterogeneous constraints. In *European Conference on Computer Vision*, pp. 276–296. Springer, 2026.

Kisuk Kang, Ying Shirley Meng, Julien Breger, Clare P Grey, and Gerbrand Ceder. Electrodes with high power and high capacity for rechargeable lithium batteries. *Science*, 311(5763):977–980, 2006.

Hilbert J Kappen. Path integrals and symmetry breaking for optimal control theory. *Journal of statistical mechanics: theory and experiment*, 2005(11):P11011–P11011, 2005.

György M Keserü and Gergely M Makara. The influence of lead discovery strategies on the properties of drug candidates. *nature reviews Drug Discovery*, 8(3):203–212, 2009.

Donald E Kirk. *Optimal control theory: an introduction*. Courier Corporation, 2004.

Levente Kocsis and Csaba Szepesvári. Bandit based monte-carlo planning. In *European conference on machine learning*, pp. 282–293. Springer, 2006.

Yann LeCun, Sumit Chopra, Raia Hadsell, M Ranzato, Fujie Huang, et al. A tutorial on energy-based learning. *Predicting structured data*, 1(0), 2006.

Daniel S Levine, Muhammed Shuaibi, Evan Walter Clark Spotte-Smith, Michael G Taylor, Muhammad R Hasyim, Kyle Michel, Ilyes Batatia, Gábor Csányi, Misko Dzamba, Peter Eastman, et al. The open molecules 2025 (omol25) dataset, evaluations, and models. *arXiv preprint arXiv:2505.08762*, 2025.

Yunyang Li, Lin Huang, Luojia Xia, Wenhe Zhang, and Mark Gerstein. Elign: Equivariant diffusion model alignment from foundational machine learning force fields. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=7u1EZZcwqX.

Sijia Liu, Pin-Yu Chen, Bhavya Kailkhura, Gaoyuan Zhang, Alfred O Hero III, and Pramod K Varshney. A primer on zeroth-order optimization in signal processing and machine learning: Principals, recent advances, and applications. *IEEE Signal Processing Magazine*, 37(5):43–54, 2020.

Youzhi Luo and Shuiwang Ji. An autoregressive flow model for 3d molecular geometry generation from scratch. In *International conference on learning representations*, 2022.

Soma Mandal, Mee'nal Moudgil, and Sanat K Mandal. Rational drug design. *European journal of pharmacology*, 625(1-3):90–100, 2009.

Robert C Merton. Optimum consumption and portfolio rules in a continuous-time model. In *Stochastic optimization models in finance*, pp. 621–661. Elsevier, 1975.

Christian A Naesseth, Fredrik Lindsten, and Thomas B Schön. Elements of sequential monte carlo. *Foundations and Trends® in Machine Learning*, 12(3):187–306, 2019.

Yoshihiko Nakamura, Hideo Hanafusa, and Tsuneo Yoshikawa. Task-priority based redundancy control of robot manipulators. *The International Journal of Robotics Research*, 6(2):3–15, 1987.

Yurii Nesterov and Vladimir Spokoiny. Random gradient-free minimization of convex functions. *Foundations of Computational Mathematics*, 17(2):527–566, 2017.

Filipp Nikitin, Ian Dunn, David Ryan Koes, and Olexandr Isayev. Geom-drugs revisited: toward more chemically accurate benchmarks for 3d molecule generation. *Digital Discovery*, 4(11):3282–3291, 2025.

Bernt Oksendal. *Stochastic differential equations: an introduction with applications*. Springer Science & Business Media, 2013.

Art B Owen. Monte carlo theory, methods and examples, 2013.

Kushagra Pandey, Farrin Marouf Sofian, Felix Draxler, Theofanis Karaletsos, and Stephan Mandt. Variational control for guidance in diffusion models. In *International Conference on Machine Learning*, pp. 47755–47780. PMLR, 2025.

David Peel and Geoffrey J McLachlan. Robust mixture modelling using the t distribution. *Statistics and computing*, 10(4):339–348, 2000.

Johan Persson. *Design and optimization under uncertainties a simulation and surrogate model based approach*. Linkopings Universitet (Sweden), 2012.

Edward O Pyzer-Knapp, Changwon Suh, Rafael Gómez-Bombarelli, Jorge Aguilera-Iparraguirre, and Alán Aspuru-Guzik. What is high-throughput virtual screening? a perspective from organic materials discovery. *Annual Review of Materials Research*, 45(1):195–216, 2015.

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):1–7, 2014.

Tim Salimans, Jonathan Ho, Xi Chen, Szymon Sidor, and Ilya Sutskever. Evolution strategies as a scalable alternative to reinforcement learning. *arXiv preprint arXiv:1703.03864*, 2017.

Petra Schneider, W Patrick Walters, Alleyn T Plowright, Norman Sieroka, Jennifer Listgarten, Robert A Goodnow Jr, Jasmin Fisher, Johanna M Jansen, José S Duca, Thomas S Rush, et al. Rethinking drug design in the artificial intelligence era. *Nature reviews drug discovery*, 19(5): 353–364, 2020.

Yuchen Shen, Chenhao Zhang, Sijie Fu, Chenghui Zhou, Newell Washburn, and Barnabas Poczos. Chemistry-inspired diffusion with non-differentiable guidance. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=4dAgG8ma3B.

B Siciliano and J-JE Slotine. A general framework for managing multiple tasks in highly redundant robotic systems. In *Fifth International Conference on Advanced Robotics' Robots in Unstructured Environments*, pp. 1211–1216. IEEE, 1991.

Jiaming Song, Qinsheng Zhang, Hongxu Yin, Morteza Mardani, Ming-Yu Liu, Jan Kautz, Yongxin Chen, and Arash Vahdat. Loss-guided diffusion models for plug-and-play controllable generation. In *International Conference on Machine Learning*, pp. 32483–32498. PMLR, 2023.

Yang Song, Jascha Sohl-dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. 2021. URL https://openreview.net/forum?id=PxTIG12RRHS.

James C Spall. Multivariate stochastic approximation using a simultaneous perturbation gradient approximation. *IEEE transactions on automatic control*, 37(3):332–341, 1992.

Charles M Stein. Estimation of the mean of a multivariate normal distribution. *The annals of Statistics*, pp. 1135–1151, 1981.

Ruiqing Sun, Dawei Feng, Sen Yang, Ronghang Wang, Huaiyuan Song, Bo Ding, Yijie Wang, and Huaimin Wang. Diffusion-based evolutionary optimization for 3d multi-objective molecular generation. *arXiv preprint arXiv:2505.11037*, 2025.

Evangelos Theodorou, Jonas Buchli, and Stefan Schaal. A generalized path integral control approach to reinforcement learning. *The Journal of Machine Learning Research*, 11:3137–3181, 2010.

Prudencio Tossou, Cas Wognum, Michael Craig, Hadrien Mary, and Emmanuel Noutahi. Real-world molecular out-of-distribution: Specification and investigation. *Journal of Chemical Information and Modeling*, 64(3):697–711, 2024.

Tim Van Erven and Peter Harremos. Rényi divergence and kullback-leibler divergence. *IEEE transactions on information theory*, 60(7):3797–3820, 2014.

Tin Hadži Veljković, Erik J Bekkers, Michael Tiemann, and Jan-Willem van de Meent. CORDS - continuous representations of discrete structures. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=RObkOKADBU.

Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3):229–256, 1992.

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

Haotian Ye, Haowei Lin, Jiaqi Han, Minkai Xu, Sheng Liu, Yitao Liang, Jianzhu Ma, James Zou, and Stefano Ermon. Tfg: Unified training-free guidance for diffusion models. *Advances in Neural Information Processing Systems*, 37:22370–22417, 2024.

Jiongmin Yong and Xun Yu Zhou. *Stochastic controls: Hamiltonian systems and HJB equations*, volume 43. Springer Science & Business Media, 1999.

Ziming Yu, Pan Zhou, Sike Wang, Jia Li, Mi Tian, and Hua Huang. Zeroth-order fine-tuning of llms in random subspaces. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4475–4485, 2025.

Assaf J Zeevi and Ronny Meir. Density estimation through convex combinations of densities: approximation and estimation bounds. *Neural Networks*, 10(1):99–109, 1997.

Zhijian Zhou, Junyi An, Zongkai Liu, Yunfei Shi, Xuan Zhang, Fenglei Cao, Chao Qu, and Yuan Qi. Guiding diffusion models with reinforcement learning for stable molecule generation. *arXiv preprint arXiv:2508.16521*, 2025.

Zhiyong Zhou, Jian Zheng, Yakang Dai, Zhe Zhou, and Shi Chen. Robust non-rigid point set registration using student's-t mixture model. *PloS one*, 9(3):e91381, 2014.

APPENDIX

A LIMITATIONS

Although our method has demonstrated strong performance across diverse molecular properties and molecule sizes on two datasets, our broader evaluation reveals a practical limitation when generating extremely large molecules. Our method still outperforms the compared approaches, including zeroth-order gradient-estimation methods and surrogate-based methods, but the chemistry software used during generation can require substantial memory and more capable hardware for such molecular systems. The chemistry software used for force evaluation in our method, as in many existing approaches, is xTB-based and can itself be resource-intensive for very large molecular systems. For example, GFN2-xTB can require around 16 GB of memory per thread for a 3,000-atom system. This can require machines with more than 16 GB of available memory to run reliably. This represents a broader limitation of current conditional molecular generation pipelines and methods and remains an open problem in this area.

B MORE QUALITATIVE RESULTS

In this section, we provide more detailed qualitative visualizations to better illustrate the generation behavior of CGMD under different guidance settings. Specifically, we consider three settings as in the main experiments: force guidance on QM9, force guidance on GEOM, and mixed guidance on QM9. In the mixed-guidance setting, we show the combinations of force guidance with each of the six property guidance on QM9. For each setting, we provide qualitative comparisons between CGMD and representative baseline methods in Figs. 4, 6, and 8, with the multi-view comparison further shown in Fig. 10. We additionally show more samples generated by CGMD in Figs. 5, 7, and 9. Atom colors represent force magnitudes calculated by GFN2-xTB, with cooler colors indicating lower forces.

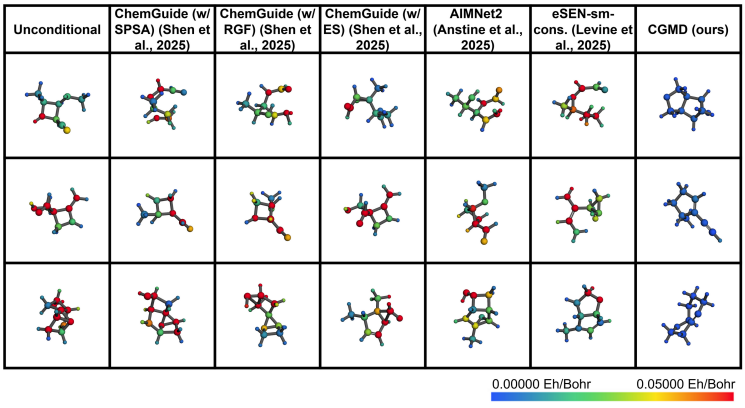


Figure 4: Qualitative comparison on QM9 with GeoLDM under GFN2-xTB force guidance. Columns from left to right show samples generated by the unconditional baseline, ChemGuide (w/ SPSA) (Shen et al., 2025; Spall, 1992), ChemGuide (w/ RGF) (Shen et al., 2025; Nesterov & Spokoiny, 2017), ChemGuide (w/ ES) (Shen et al., 2025; Salimans et al., 2017), surrogate-based guidance with AIMNet2 (Anstine et al., 2025) and eSEN-sm-cons. (Levine et al., 2025), and CGMD (Ours). Atoms are colored by their GFN2-xTB-computed atomic force magnitudes (Eh/Bohr), where cooler colors indicate smaller atomic forces.

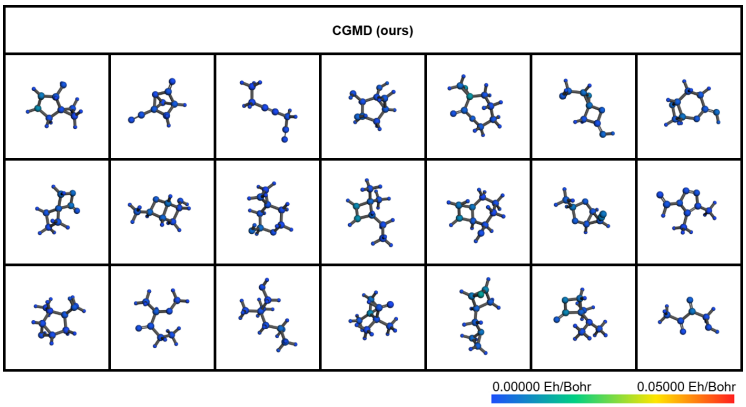


Figure 5: Qualitative results of CGMD (Ours) on QM9 with GeoLDM under GFN2-xTB force guidance. Atoms are colored by their GFN2-xTB-computed atomic force magnitudes (Eh/Bohr), where cooler colors indicate smaller atomic forces and thus geometries closer to local equilibrium.

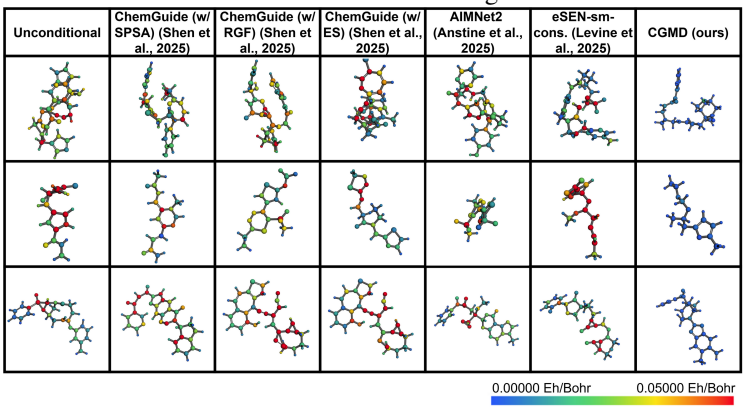


Figure 6: Qualitative comparison on GEOM with GeoLDM under GFN2-xTB force guidance. Columns from left to right show samples generated by the unconditional baseline, three zeroth-order guidance baselines including ChemGuide (w/ SPSA) (Shen et al., 2025; Spall, 1992), ChemGuide (w/ RGF) (Shen et al., 2025; Nesterov & Spokoiny, 2017), and ChemGuide (w/ ES) (Shen et al., 2025; Salimans et al., 2017), surrogate-based guidance baseline using AIMNet2 (Anstine et al., 2025) and eSEN-sm-cons. (Levine et al., 2025), and CGMD (Ours). Atoms are colored by their GFN2-xTB-computed atomic force magnitudes (Eh/Bohr), where cooler colors indicate smaller atomic forces and thus geometries closer to local equilibrium.

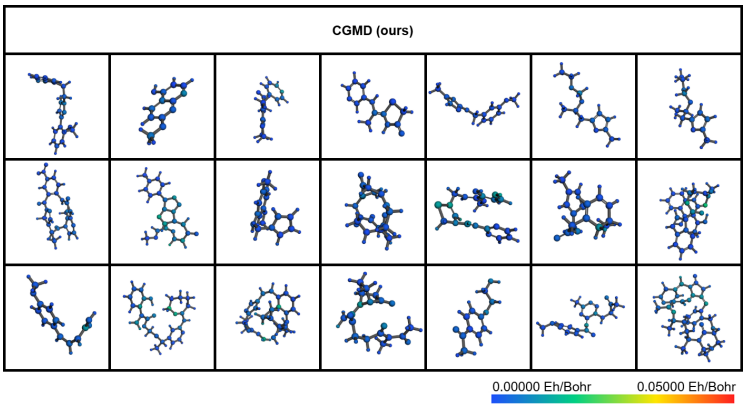


Figure 7: Qualitative results of CGMD (Ours) on GEOM with GeoLDM under GFN2-xTB force guidance. Atoms are colored by their GFN2-xTB-computed atomic force magnitudes (Eh/Bohr), where cooler colors indicate smaller atomic forces and thus geometries closer to local equilibrium.

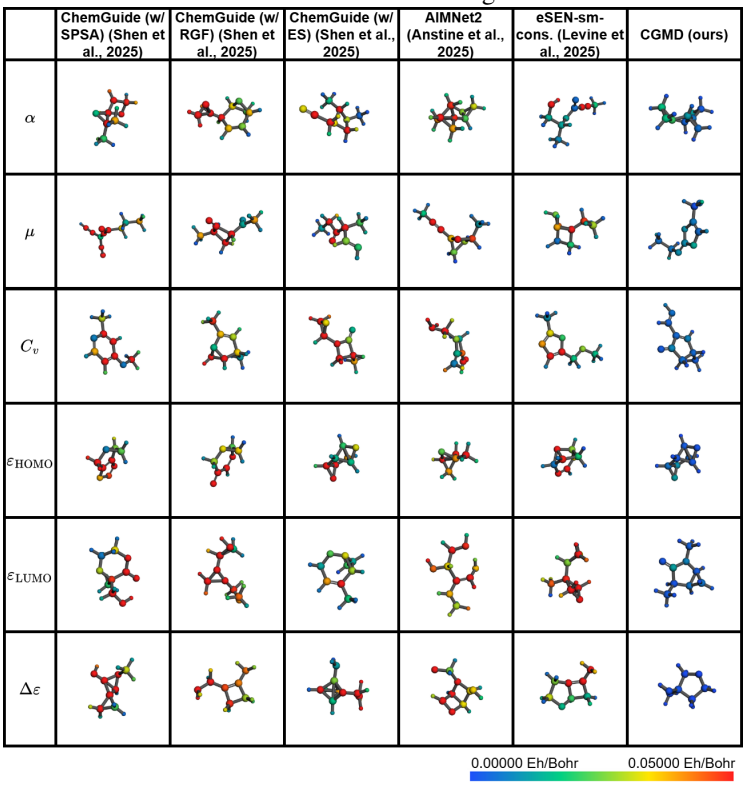


Figure 8: Qualitative comparison on QM9 with GeoLDM under mixed guidance using GFN2-xTB for force guidance across six property objectives: α, μ, C_v, $\varepsilon_{\mathrm{HOMO}}$, $\varepsilon_{\mathrm{LUMO}}$, and $\Delta\varepsilon$. The mixed-guidance setting combines force guidance with each property guidance objective. Columns from left to right show samples generated by ChemGuide (w/ SPSA) (Shen et al., 2025; Spall, 1992), ChemGuide (w/ RGF) (Shen et al., 2025; Nesterov & Spokoiny, 2017), ChemGuide (w/ ES) (Shen et al., 2025; Salimans et al., 2017), surrogate-based guidance with AIMNet2 (Anstine et al., 2025) and eSEN-sm-cons. (Levine et al., 2025), and CGMD. Rows from top to bottom correspond to static molecular polarizability (α), molecular dipole moment (μ), molecular heat capacity at room temperature (C_v), HOMO energy ($\varepsilon_{\mathrm{HOMO}}$), LUMO energy ($\varepsilon_{\mathrm{LUMO}}$), and HOMO–LUMO energy gap ($\Delta\varepsilon$). Atoms are colored by their GFN2-xTB-computed atomic force magnitudes (Eh/Bohr), where cooler colors indicate smaller atomic forces and thus geometries closer to local equilibrium.

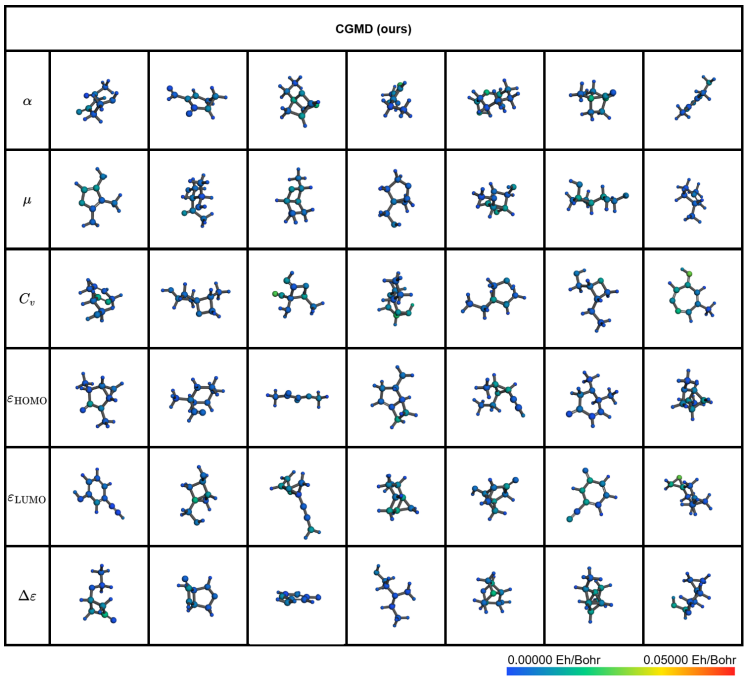


Figure 9: Qualitative results of CGMD on QM9 with GeoLDM under the mixed-guidance setting using GFN2-xTB for force guidance. Each row shows samples generated by CGMD when GFN2-xTB force guidance is combined with one property guidance objective. Rows from top to bottom correspond to static molecular polarizability (α), molecular dipole moment (μ), molecular heat capacity at room temperature (C_v), HOMO energy ($\varepsilon_{\mathrm{HOMO}}$), LUMO energy ($\varepsilon_{\mathrm{LUMO}}$), and HOMO–LUMO energy gap ($\Delta\varepsilon$). Atoms are colored by their GFN2-xTB-computed atomic force magnitudes (Eh/Bohr), where cooler colors indicate smaller atomic forces and thus geometries closer to local equilibrium.

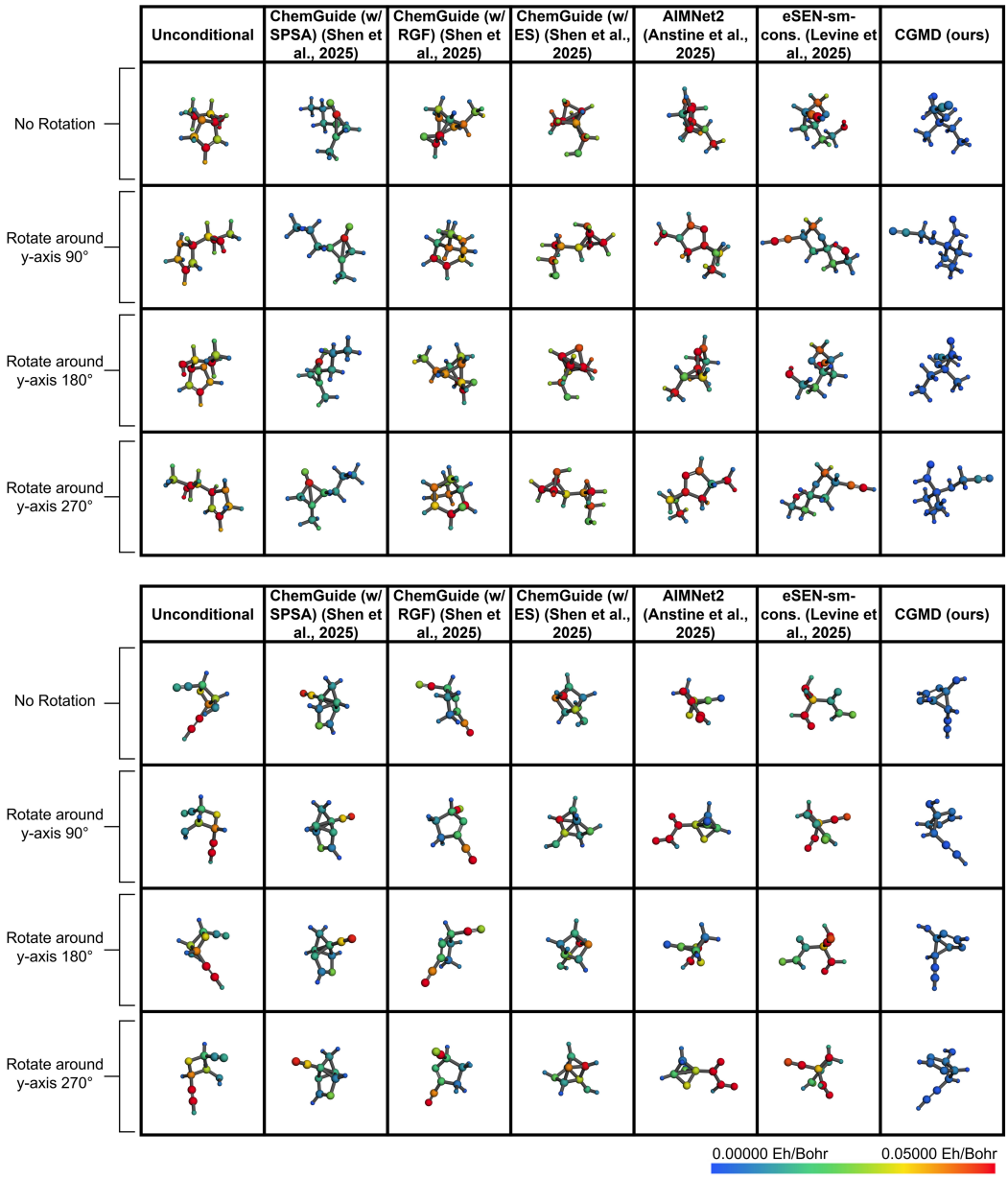


Figure 10: Multi-view qualitative comparison on QM9 with GeoLDM under GFN2-xTB force guidance. Atoms are colored by their GFN2-xTB-computed atomic force magnitudes (Eh/Bohr), where cooler colors indicate smaller atomic forces.

C More Experimental Results

In this section, to further complement the experimental results in the main text, we provide more ablation studies, further analysis, and results on more experimental settings.

C.1 Additional Results

Additional Results with force guidance. In addition to the GFN2-xTB results reported in Tables 1, 2, and 3, we report results using g-xTB in Tables 6, 7, and 8. As shown, CGMD also achieves superior performance with g-xTB. Specifically, Tables 1 and 6 show that, on QM9, CGMD reduces force RMS while improving validity, atom stability, and molecule stability. Although stability is

Table 6: Results with g-xTB force guidance on QM9 (Ramakrishnan et al., 2014).

Model	Method	Force RMS ↓	Energy above Ground State ↓	Validity ↑	Uniqueness ↑	Atom Stability ↑	Molecule Stability ↑
GeoLDM	Unconditional	0.0125	0.0061	89.80%	**100.00%**	98.93%	88.80%
	ChemGuide (w/ SPSA) (Shen et al., 2025; Spall, 1992)	0.0119	0.0053	91.20%	**100.00%**	99.11%	90.80%
	ChemGuide (w/ RGF) (Shen et al., 2025; Nesterov & Spokoiny, 2017)	0.0120	0.0057	90.60%	**100.00%**	99.07%	90.20%
	ChemGuide (w/ ES) (Shen et al., 2025; Salimans et al., 2017)	0.0119	0.0056	90.60%	**100.00%**	98.99%	90.20%
	AIMNet2 (Anstine et al., 2025)	0.0102	0.0050	89.00%	**100.00%**	98.75%	88.00%
	eSEN-sm-cons. (Levine et al., 2025)	0.0100	0.0052	88.80%	**100.00%**	98.97%	89.20%
	Ours	**0.0071**	**0.0026**	**93.00%**	**100.00%**	**99.23%**	**92.40%**
EDM	Unconditional	0.0124	0.0085	86.60%	**100.00%**	98.53%	83.60%
	ChemGuide (w/ SPSA) (Shen et al., 2025; Spall, 1992)	0.0117	0.0079	86.20%	**100.00%**	98.48%	81.40%
	ChemGuide (w/ RGF) (Shen et al., 2025; Nesterov & Spokoiny, 2017)	0.0115	0.0081	84.60%	**100.00%**	98.41%	81.40%
	ChemGuide (w/ ES) (Shen et al., 2025; Salimans et al., 2017)	0.0119	0.0078	86.00%	**100.00%**	98.43%	81.20%
	AIMNet2 (Anstine et al., 2025)	0.0111	0.0073	86.60%	**100.00%**	98.29%	81.60%
	eSEN-sm-cons. (Levine et al., 2025)	0.0109	0.0072	86.20%	**100.00%**	98.21%	81.60%
	Ours	**0.0094**	**0.0050**	**87.40%**	**100.00%**	**98.73%**	**86.40%**

Table 7: Results with g-xTB force guidance on GEOM (Axelrod & Gomez-Bombarelli, 2022).

Model	Method	Force RMS ↓	Energy above Ground State ↓	Validity ↑	Uniqueness ↑	Atom Stability ↑
GeoLDM	Unconditional	0.0498	0.2601	49.00%	**100.00%**	84.53%
	ChemGuide (w/ SPSA) (Shen et al., 2025; Spall, 1992)	0.0430	0.2221	50.20%	**100.00%**	84.41%
	ChemGuide (w/ RGF) (Shen et al., 2025; Nesterov & Spokoiny, 2017)	0.0443	0.2337	49.80%	**100.00%**	84.27%
	ChemGuide (w/ ES) (Shen et al., 2025; Salimans et al., 2017)	0.0439	0.2364	48.60%	**100.00%**	84.51%
	AIMNet2 (Anstine et al., 2025)	0.0397	0.2174	49.80%	**100.00%**	85.13%
	eSEN-sm-cons. (Levine et al., 2025)	0.0389	0.2154	49.80%	**100.00%**	84.79%
	Ours	**0.0220**	**0.1158**	**59.40%**	**100.00%**	**85.92%**
EDM	Unconditional	0.0734	0.4151	46.40%	**100.00%**	81.22%
	ChemGuide (w/ SPSA) (Shen et al., 2025; Spall, 1992)	0.0550	0.3654	45.60%	**100.00%**	81.16%
	ChemGuide (w/ RGF) (Shen et al., 2025; Nesterov & Spokoiny, 2017)	0.0548	0.3759	43.20%	**100.00%**	81.21%
	ChemGuide (w/ ES) (Shen et al., 2025; Salimans et al., 2017)	0.0556	0.3698	44.00%	**100.00%**	81.08%
	AIMNet2 (Anstine et al., 2025)	0.0549	0.3438	43.80%	**100.00%**	81.35%
	eSEN-sm-cons. (Levine et al., 2025)	0.0526	0.3367	43.80%	**100.00%**	81.14%
	Ours	**0.0427**	**0.2291**	**50.60%**	**100.00%**	**85.03%**

optimized through force rather than directly through energy, energy also decreases. This lower energy suggests that the generated molecules are closer to the ground state (Shen et al., 2025). Tables 2 and 7 show similar trends on GEOM. We also observe a large reduction in force RMS on GEOM, further demonstrating the effectiveness of the proposed framework.

Table 8: Results in the mixed-guidance setting on QM9 (Ramakrishnan et al., 2014) using g-xTB for force guidance

		α			μ			C_v		
Model	Method	MAE ↓	Force ↓	Energy ↓	MAE ↓	Force ↓	Energy ↓	MAE ↓	Force ↓	Energy ↓
GeoLDM	ChemGuide (w/ SPSA) (Shen et al., 2025; Spall, 1992)	2.5009	0.0135	0.0051	0.7895	0.0119	0.0051	0.9372	0.0112	0.0071
	ChemGuide (w/ RGF) (Shen et al., 2025; Nesterov & Spokoiny, 2017)	2.6676	0.0137	0.0065	0.7610	0.0123	0.0051	1.0673	0.0135	0.0066
	ChemGuide (w/ ES) (Shen et al., 2025; Salimans et al., 2017)	2.6015	0.0136	0.0055	0.8011	0.0120	0.0052	1.2822	0.0241	0.0175
	AIMNet2 (Anstine et al., 2025)	2.7666	0.0141	0.0064	0.8775	0.0123	0.0049	1.1623	0.0132	0.0083
	eSEN-sm-cons. (Levine et al., 2025)	2.8124	0.0133	0.0063	0.9012	0.0119	0.0048	1.1895	0.0143	0.0081
	Ours	**2.3187**	**0.0100**	**0.0031**	**0.6481**	**0.0082**	**0.0030**	**0.8535**	**0.0086**	**0.0043**
EDM	ChemGuide (w/ SPSA) (Shen et al., 2025; Spall, 1992)	4.0963	0.0198	0.0095	1.3965	0.0179	0.0090	2.3105	0.0181	0.0116
	ChemGuide (w/ RGF) (Shen et al., 2025; Nesterov & Spokoiny, 2017)	4.0065	0.0201	0.0098	1.4216	0.0179	0.0089	2.4504	0.0175	0.0101
	ChemGuide (w/ ES) (Shen et al., 2025; Salimans et al., 2017)	3.7252	0.0199	0.0103	1.6419	0.0178	0.0089	2.2864	0.0168	0.0094
	AIMNet2 (Anstine et al., 2025)	4.2925	0.0197	0.0098	1.6073	0.0183	0.0091	2.5674	0.0171	0.0105
	eSEN-sm-cons. (Levine et al., 2025)	4.4178	0.0189	0.0093	1.5639	0.0187	0.0078	2.6315	0.0183	0.0098
	Ours	**2.1337**	**0.0122**	**0.0068**	**0.7385**	**0.0113**	**0.0046**	**1.1646**	**0.0098**	**0.0067**
		$\varepsilon_{\text{HOMO}}$			$\varepsilon_{\text{LUMO}}$			$\Delta\varepsilon$		
Model	Method	MAE ↓	Force ↓	Energy ↓	MAE ↓	Force ↓	Energy ↓	MAE ↓	Force ↓	Energy ↓
GeoLDM	ChemGuide (w/ SPSA) (Shen et al., 2025; Spall, 1992)	0.4585	0.0107	0.0031	0.8510	0.0101	0.0026	0.7039	0.0107	0.0037
	ChemGuide (w/ RGF) (Shen et al., 2025; Nesterov & Spokoiny, 2017)	0.4950	0.0131	0.0036	0.8129	0.0101	0.0026	0.8728	0.0119	0.0056
	ChemGuide (w/ ES) (Shen et al., 2025; Salimans et al., 2017)	0.5018	0.0120	0.0035	0.9695	0.0107	0.0032	0.8766	0.0112	0.0053
	AIMNet2 (Anstine et al., 2025)	0.5401	0.0123	0.0039	0.9568	0.0108	0.0031	0.9186	0.0107	0.0046
	eSEN-sm-cons. (Levine et al., 2025)	0.5568	0.0109	0.0032	0.9347	0.0103	0.0025	0.9472	0.0103	0.0041
	Ours	**0.3422**	**0.0092**	**0.0025**	**0.6701**	**0.0090**	**0.0021**	**0.6577**	**0.0093**	**0.0031**
EDM	ChemGuide (w/ SPSA) (Shen et al., 2025; Spall, 1992)	0.7282	0.0177	0.0090	1.1015	0.0166	0.0075	1.2385	0.0178	0.0133
	ChemGuide (w/ RGF) (Shen et al., 2025; Nesterov & Spokoiny, 2017)	0.7366	0.0167	0.0083	1.1289	0.0168	0.0082	1.3447	0.0170	0.0128
	ChemGuide (w/ ES) (Shen et al., 2025; Salimans et al., 2017)	0.7159	0.0169	0.0068	1.1028	0.0161	0.0099	1.2618	0.0198	0.0143
	AIMNet2 (Anstine et al., 2025)	0.7753	0.0181	0.0080	1.2388	0.0162	0.0083	1.3590	0.0184	0.0131
	eSEN-sm-cons. (Levine et al., 2025)	0.8016	0.0173	0.0078	1.2714	0.0147	0.0081	1.3278	0.0172	0.0130
	Ours	**0.3877**	**0.0102**	**0.0049**	**0.7254**	**0.0093**	**0.0048**	**0.8839**	**0.0108**	**0.0059**

Guidance with a Non-Differentiable ZPVE Objective. To further evaluate CGMD beyond force guidance, we consider zero-point vibrational energy (ZPVE) as a non-differentiable quantum-chemical guidance objective on QM9. ZPVE characterizes the vibrational energy associated with a molecular structure and is related to molecular reaction energies and activation barriers. We use GeoLDM as the base model and sample target ZPVE values from the empirical distribution of QM9 following ChemGuide (Shen et al., 2025). We use g-xTB (Froitzheim et al., 2025) as the non-differentiable evaluator and compare CGMD with previous SoTA baselines (Shen et al., 2025) on QM9. We report the mean absolute error (MAE) between the target and evaluated ZPVE values, together with molecular stability metrics. As shown in Table 9, CGMD achieves the best

results, demonstrating its effectiveness on non-differentiable quantum-chemical objectives beyond force-based stability optimization.

Table 9: Results for non-differentiable ZPVE guidance using g-xTB with GeoLDM on QM9 over 500 generated molecules.

Method	ZPVE MAE (meV) ↓	Validity ↑	Uniqueness ↑	Atom Stability ↑	Molecule Stability ↑
Unconditional	134.68	89.80%	**100.00%**	98.93%	88.80%
ChemGuide (w/ SPSA) (Shen et al., 2025; Spall, 1992)	128.21	89.60%	**100.00%**	98.37%	88.20%
ChemGuide (w/ RGF) (Shen et al., 2025; Nesterov & Spokoiny, 2017)	121.23	89.60%	**100.00%**	98.43%	88.20%
ChemGuide (w/ ES) (Shen et al., 2025; Salimans et al., 2017)	127.49	89.40%	**100.00%**	98.29%	88.00%
Ours	**111.68**	**91.80%**	**100.00%**	**98.97%**	**90.00%**

C.2 Ablation Studies

To further validate the key designs of CGMD, we conduct additional ablation studies.

Ablation on the Sampling Distribution Family. CGMD fits a Student-t mixture sampling distribution to realize non-differentiable guidance. To further validate this design, we also conduct experiments on several alternative sampling distributions, including single Gaussian sampling distribution, single Student-t sampling distribution, and Gaussian-mixture sampling distribution. As shown in Tables 10 and 11, CGMD with the Student-t mixture achieves superior performance, validating the effectiveness of the chosen sampling distribution family.

Table 10: Ablation on the sampling distribution family in CGMD on QM9 with GeoLDM under GFN2-xTB force guidance.

Proposal family	Force RMS ↓	Energy above Ground State ↓	Validity ↑	Uniqueness ↑	Atom Stability ↑	Molecule Stability ↑
Single Gaussian	0.0084	0.0032	92.00%	**100.00%**	98.98%	91.80%
Single Student-t	0.0078	0.0029	92.40%	**100.00%**	99.09%	92.00%
Gaussian mixture	0.0081	0.0032	92.00%	**100.00%**	99.10%	92.40%
Ours	**0.0068**	**0.0021**	**93.20%**	**100.00%**	**99.19%**	**92.80%**

Table 11: Ablation on the sampling distribution family in CGMD on GEOM with GeoLDM under GFN2-xTB force guidance.

Proposal family	Force RMS ↓	Energy above Ground State ↓	Validity ↑	Uniqueness ↑	Atom Stability ↑
Single Gaussian	0.0301	0.1265	52.20%	**100.00%**	84.98%
Single Student-t	0.0265	0.1141	55.60%	**100.00%**	85.24%
Gaussian mixture	0.0295	0.1243	52.00%	**100.00%**	85.02%
Ours	**0.0211**	**0.1003**	**59.60%**	**100.00%**	**85.96%**

Ablation on the Hierarchy-Preserving Control Order. To further validate the control hierarchy in CGMD, we compare the hierarchy used in the main experiments against a reversed-hierarchy variant. In CGMD, the force control is treated as the foundational control, while the property control is treated as the refinement control. In the reversed-hierarchy variant, we instead treat the property control as the foundational control and restrict the force control to the remaining directions. As shown in Table 12, reversing the control hierarchy degrades the overall performance, indicating the effectiveness of the hierarchy used in the main experiments.

Table 12: Ablation on the hierarchy order in mixed-guidance settings on QM9 with GeoLDM using GFN2-xTB for force guidance

		α			μ			C_v	
Method	MAE ↓	Force ↓	Energy ↓	MAE ↓	Force ↓	Energy ↓	MAE ↓	Force ↓	Energy ↓
w/ reversed order	2.4362	0.0139	0.0057	0.7310	0.0101	0.0054	0.9386	0.0116	0.0063
Ours	**2.3475**	**0.0101**	**0.0032**	**0.6559**	**0.0084**	**0.0031**	**0.8585**	**0.0088**	**0.0042**
Method	MAE ↓	$\varepsilon_{\mathrm{HOMO}}$ Force ↓	Energy ↓	MAE ↓	$\varepsilon_{\mathrm{LUMO}}$ Force ↓	Energy ↓	MAE ↓	$\Delta\varepsilon$ Force ↓	Energy ↓
w/ reversed order	0.4132	0.0109	0.0037	0.7697	0.0103	0.0027	0.7436	0.0097	0.0035
Ours	**0.3460**	**0.0082**	**0.0024**	**0.6757**	**0.0081**	**0.0021**	**0.6664**	**0.0086**	**0.0023**

Ablation on the Number of Evaluator Queries. CGMD draws K perturbation samples, each requiring one evaluator query, at each guidance step to estimate the non-differentiable control term. To study the impact of the per-step query budget K, we run CGMD with $K \in \{2, 3, 5, 7, 15, 30\}$ on QM9 and GEOM while keeping other settings fixed. We further evaluate a substantially larger query budget of $K = 150$. As shown in Tables 13 and 14, CGMD achieves strong performance across different query budgets, even when the per-step budget is small. As K increases beyond a sufficient value, performance becomes stable. We set $K = 3$ in the main experiments.

Table 13: Analysis on the number of evaluator queries with GeoLDM on QM9 with GFN2-xTB force guidance.

Method	K	Force RMS ↓	Energy above Ground State ↓	Validity ↑	Uniqueness ↑	Atom Stability ↑	Molecule Stability ↑
Ours	2	0.0078	0.0026	92.60%	100.00%	99.06%	92.40%
Ours	3	0.0068	0.0021	93.20%	100.00%	99.19%	92.80%
Ours	5	0.0066	0.0021	93.20%	100.00%	99.20%	93.00%
Ours	7	0.0065	0.0020	93.00%	100.00%	99.20%	93.00%
Ours	15	0.0064	0.0019	93.20%	100.00%	99.20%	93.00%
Ours	30	0.0064	0.0019	93.20%	100.00%	99.24%	93.00%
Ours	150	0.0064	0.0019	93.20%	100.00%	99.24%	93.00%

Table 14: Analysis on the number of evaluator queries with GeoLDM on GEOM with GFN2-xTB force guidance.

Method	K	Force RMS ↓	Energy above Ground State ↓	Validity ↑	Uniqueness ↑	Atom Stability ↑
Ours	2	0.0230	0.1075	59.40%	100.00%	85.93%
Ours	3	0.0211	0.1003	59.60%	100.00%	85.96%
Ours	5	0.0209	0.1003	59.60%	100.00%	85.96%
Ours	7	0.0209	0.1004	59.60%	100.00%	85.97%
Ours	15	0.0206	0.1000	59.80%	100.00%	85.99%
Ours	30	0.0202	0.0998	59.80%	100.00%	85.98%
Ours	150	0.0202	0.0997	59.80%	100.00%	86.01%

Ablation on the Anchor Window. CGMD uses cached evaluator-informed anchors to fit the sampling distribution at each guidance step. We study the impact of the anchor window, i.e., the number of previous denoising steps from which cached anchors are used. As shown in Tables 15 and 16, even the 1-step window achieves strong performance and outperforms all baseline methods. The 5-step window already achieves strong performance on both datasets, so we use it in the main experiments.

Table 15: Parameter analysis on the anchor window on QM9 with GeoLDM under GFN2-xTB force guidance.

Anchor window	Force RMS ↓	Energy above Ground State ↓	Validity ↑	Uniqueness ↑	Atom Stability ↑	Molecule Stability ↑
1 step	0.0076	0.0029	92.40%	100.00%	99.10%	92.20%
5 steps	0.0068	0.0021	93.20%	100.00%	99.19%	92.80%
10 steps	0.0069	0.0022	92.80%	100.00%	99.15%	92.80%
20 steps	0.0070	0.0023	92.80%	100.00%	99.14%	92.60%

Table 16: Parameter analysis on the anchor window on GEOM with GeoLDM under GFN2-xTB force guidance.

Anchor window	Force RMS ↓	Energy above Ground State ↓	Validity ↑	Uniqueness ↑	Atom Stability ↑
1 step	0.0229	0.1058	59.20%	100.00%	85.88%
5 steps	0.0211	0.1003	59.60%	100.00%	85.96%
10 steps	0.0216	0.1036	59.60%	100.00%	85.95%
20 steps	0.0220	0.1037	59.60%	100.00%	85.94%

Ablation on the Number of Student-t Mixture Components. CGMD models the evaluator-informed sampling distribution as a Student-t mixture distribution, where M denotes the number of mixture components. We study the impact of M by varying $M \in \{1, 2, 3, 4\}$ while keeping other settings fixed. As shown in Tables 17 and 18, even $M = 1$ achieves the best overall performance among the baseline methods in Tables 1 and 2. With $M = 2$, CGMD already achieves strong performance on both datasets, so we use $M = 2$ in the main experiments.

Table 17: Impact of the number of Student-t mixture components on QM9 with GeoLDM under GFN2-xTB force guidance.

M	Force RMS ↓	Energy above Ground State ↓	Validity ↑	Uniqueness ↑	Atom Stability ↑	Molecule Stability ↑
1	0.0078	0.0029	92.40%	100.00%	99.09%	92.00%
2	0.0068	0.0021	93.20%	100.00%	99.19%	92.80%
3	0.0069	0.0022	93.00%	100.00%	99.18%	92.80%
4	0.0071	0.0023	93.00%	100.00%	99.16%	92.60%

Table 18: Impact of the number of Student-t mixture components on GEOM under GFN2-xTB force guidance with GeoLDM.

M	Force RMS ↓	Energy above Ground State ↓	Validity ↑	Uniqueness ↑	Atom Stability ↑
1	0.0265	0.1141	55.60%	100.00%	85.24%
2	0.0211	0.1003	59.60%	100.00%	85.96%
3	0.0215	0.1004	59.20%	100.00%	85.92%
4	0.0220	0.1006	58.80%	100.00%	85.83%

Ablation on the Normalization Factor κ. Following the scale normalization described in Appendix E.2, κ accounts for the numerical scale of the evaluator output and is set separately for QM9 and GEOM. We set $\kappa = 3 \times 10^{-6}$ on QM9 and $\kappa = 3 \times 10^{-3}$ on GEOM in the main experiments. We vary κ over a broad range around the dataset-specific default values while keeping other settings fixed. As shown in Tables 19 and 20, CGMD consistently achieves strong performance across a wide range of κ on both datasets, outperforming the baseline methods.

Table 19: Sensitivity to the normalization scale κ on QM9 with GeoLDM under GFN2-xTB force guidance.

κ	Force RMS ↓	Energy above Ground State ↓	Validity ↑	Uniqueness ↑	Atom Stability ↑	Molecule Stability ↑
3×10^{-7}	0.0070	0.0023	93.00%	100.00%	99.16%	92.60%
1×10^{-6}	0.0069	0.0022	93.00%	100.00%	99.17%	92.60%
3×10^{-6}	0.0068	0.0021	93.20%	100.00%	99.19%	92.80%
1×10^{-5}	0.0071	0.0023	93.00%	100.00%	99.15%	92.40%
3×10^{-5}	0.0076	0.0027	92.60%	100.00%	99.10%	92.00%

Table 20: Sensitivity to the normalization scale κ on GEOM with GeoLDM under GFN2-xTB force guidance.

κ	Force RMS ↓	Energy above Ground State ↓	Validity ↑	Uniqueness ↑	Atom Stability ↑
3×10^{-4}	0.0216	0.1022	59.40%	100.00%	85.92%
1×10^{-3}	0.0214	0.1014	59.60%	100.00%	85.94%
3×10^{-3}	0.0211	0.1003	59.60%	100.00%	85.96%
1×10^{-2}	0.0218	0.1031	59.40%	100.00%	85.89%
3×10^{-2}	0.0232	0.1094	59.00%	100.00%	85.76%

C.3 Further Analysis

To further analyze the key designs of our proposed CGMD framework, we provide further analysis below.

Further analysis on different diffusion samplers. We further explore CGMD with other widely used diffusion samplers, such as DDIM, while keeping other settings unchanged. As shown in Tables 21 and 22, we observe that CGMD also performs well under DDIM sampling on both QM9 and GEOM with GeoLDM.

Table 21: Further analysis on DDIM sampling with GeoLDM on QM9 under GFN2-xTB force guidance. Note that AIMNet2 and eSEN-sm-cons. (Levine et al., 2025) are surrogate models whose gradients serve as differentiable approximations of the non-differentiable evaluator

Method	Force RMS ↓	Energy above Ground State ↓	Validity ↑	Uniqueness ↑	Atom Stability ↑	Molecule Stability ↑
Unconditional (DDIM)	0.0643	0.0335	74.00%	**100.00%**	96.03%	69.40%
ChemGuide (w/ SPSA, DDIM) (Shen et al., 2025; Spall, 1992)	0.0576	0.0307	74.00%	**100.00%**	96.79%	72.60%
ChemGuide (w/ RGF, DDIM) (Shen et al., 2025; Nesterov & Spokoiny, 2017)	0.0591	0.0294	73.40%	**100.00%**	96.34%	69.40%
ChemGuide (w/ ES, DDIM) (Shen et al., 2025; Salimans et al., 2017)	0.0533	0.0303	73.80%	**100.00%**	96.71%	71.60%
AIMNet2 (DDIM) (Anstine et al., 2025)	0.0539	0.0302	74.00%	**100.00%**	96.33%	68.20%
eSEN-sm-cons. (DDIM) (Levine et al., 2025)	0.0587	0.0301	73.80%	**100.00%**	96.31%	68.60%
Ours (DDIM)	**0.0315**	**0.0243**	**81.20%**	**100.00%**	**97.43%**	**77.80%**

Table 22: Further analysis on DDIM sampling with GeoLDM on GEOM under GFN2-xTB force guidance. Note that AIMNet2 and eSEN-sm-cons. (Levine et al., 2025) are surrogate models whose gradients serve as differentiable approximations of the non-differentiable evaluator

Method	Force RMS ↓	Energy above Ground State ↓	Validity ↑	Uniqueness ↑	Atom Stability ↑
Unconditional(DDIM)	0.0785	0.3718	46.00%	**100.00%**	75.98%
ChemGuide (w/ SPSA, DDIM) (Shen et al., 2025; Spall, 1992)	0.0728	0.3328	45.40%	**100.00%**	77.32%
ChemGuide (w/ RGF, DDIM) (Shen et al., 2025; Nesterov & Spokoiny, 2017)	0.0724	0.3312	45.20%	**100.00%**	77.18%
ChemGuide (w/ ES, DDIM) (Shen et al., 2025; Salimans et al., 2017)	0.0722	0.3211	45.40%	**100.00%**	77.20%
AIMNet2 (DDIM) (Anstine et al., 2025)	0.0751	0.3504	44.20%	**100.00%**	76.93%
eSEN-sm-cons. (DDIM) (Levine et al., 2025)	0.0734	0.3534	45.80%	**100.00%**	76.62%
Ours (DDIM)	**0.0694**	**0.2966**	**49.20%**	**100.00%**	**83.13%**

Further Analysis on Non-Differentiable Guidance. In Section 4.1, we explored our design for non-differentiable guidance and theoretically showed that effective guidance can be obtained without gradient estimation under the optimal control framework. To further validate the effectiveness of this design, we compare CGMD with several alternative methods. The original **ChemGuide** (Shen et al., 2025) uses simultaneous perturbation stochastic approximation (**SPSA**) (Spall, 1992) to estimate gradients from non-differentiable evaluator queries. We additionally consider **RGF** (Nesterov & Spokoiny, 2017), **ES** (Salimans et al., 2017), **REINFORCE** (Williams, 1992), and **RS-ZO** (Yu et al., 2025). **RGF** (Nesterov & Spokoiny, 2017) provides guidance by estimating a gradient from non-differentiable evaluator queries. It samples random perturbation directions around the current clean estimate and computes finite-difference estimates along these directions. The resulting estimates are averaged to obtain the guidance signal. **ES** (Salimans et al., 2017) also provides guidance through a zeroth-order gradient estimate. It samples multiple Gaussian perturbations around the current clean estimate, evaluates the perturbed candidates with the evaluator, and aggregates the perturbations weighted by their evaluator values. **REINFORCE** (Williams, 1992) provides guidance through a score-function estimator. It treats perturbation sampling as a stochastic policy and uses evaluator values as rewards to estimate the guidance signal. **RS-ZO** (Yu et al., 2025) performs zeroth-order gradient estimation in a low-dimensional random subspace, reducing the variance of traditional full-space zeroth-order estimators. For surrogate-based guidance, following ChemGuide (Shen et al., 2025), we compare with surrogate-model-based guidance that replaces the non-differentiable xTB evaluator with differentiable neural predictors and uses their gradients for guidance. Specifically, we follow the settings in Shen et al. (2025) and adopt the surrogate model **AIMNet2** (Anstine et al., 2025). We further evaluate another surrogate model, **eSEN-sm-cons.** (Levine et al., 2025). We also include a non-differentiable Best-of-N selection baseline: **SCG** (Huang et al., 2024b). We adapt **SCG** (Huang et al., 2024b) to our setting by sampling multiple candidate next states from the diffusion model at each guided denoising step and selecting the one with the best evaluator value. We also include **Tree Search** (Kocsis & Szepesvári, 2006) as a non-differentiable baseline. **Tree Search** maintains multiple candidate branches across search steps, iteratively extends them, and selects the final candidate according to evaluator feedback. We further adapt **Sequential Monte Carlo (SMC)** (Naesseth et al., 2019) to our setting by maintaining multiple denoising particles and sequentially reweighting and resampling them according to their evaluator values. These methods either rely on gradients (e.g., SPSA, RGF, or surrogate-based guidance), even when evaluator gradients are not directly available, or use costly evaluator queries primarily to select or retain promising candidates (e.g., SCG), which can be query-inefficient. These limitations can make such approaches unsuitable in 3D molecular guidance with expensive, non-differentiable evaluators. Different from these methods, CGMD adopts a design tailored to this setting, making fuller use of each set of evaluator queries while avoiding the need for evaluator gradients, by aggregating multiple sampled perturbations with exponential weights. Crucially, CGMD couples this control construction with evaluator-informed sampling to reuse past evaluator feedback for query-efficient guidance, and

hierarchy-preserving control to coordinate heterogeneous constraints in mixed-guidance settings. As shown in Tables 23 and 24, CGMD achieves the best overall performance among the compared methods.

Table 23: Additional analysis on our design for non-differentiable guidance with GeoLDM on QM9 under GFN2-xTB force guidance. Note that AIMNet2 and eSEN-sm-cons. (Levine et al., 2025) are surrogate models whose gradients serve as differentiable approximations of the non-differentiable evaluator

Method	Force RMS ↓	Energy above Ground State ↓	Validity ↑	Uniqueness ↑	Atom Stability ↑	Molecule Stability ↑
Unconditional	0.0111	0.0050	89.80%	**100.00%**	98.93%	88.80%
ChemGuide (w/ SPSA) (Shen et al., 2025; Spall, 1992)	0.0104	0.0042	91.40%	**100.00%**	99.02%	90.60%
ChemGuide (w/ RGF) (Shen et al., 2025; Nesterov & Spokoiny, 2017)	0.0103	0.0049	90.80%	99.78%	99.01%	90.40%
ChemGuide (w/ ES) (Shen et al., 2025; Salimans et al., 2017)	0.0105	0.0053	88.20%	**100.00%**	99.05%	90.60%
AIMNet2 (Anstine et al., 2025)	0.0110	0.0055	89.00%	**100.00%**	98.75%	88.00%
eSEN-sm-cons. (Levine et al., 2025)	0.0107	0.0055	88.80%	**100.00%**	98.97%	89.20%
ChemGuide (w/ REINFORCE) (Shen et al., 2025; Williams, 1992)	0.0109	0.0045	91.40%	**100.00%**	98.55%	86.60%
ChemGuide (w/ RS-ZO) (Shen et al., 2025; Yu et al., 2025)	0.0102	0.0043	91.40%	**100.00%**	99.11%	90.60%
SCG (Huang et al., 2024b)	0.0102	0.0043	91.20%	**100.00%**	98.85%	89.60%
Tree search (Kocsis & Szepesvári, 2006)	0.0107	0.0049	87.20%	**100.00%**	98.75%	88.40%
SMC (Naesseth et al., 2019)	0.0105	0.0053	87.00%	99.54%	98.74%	88.20%
Ours	**0.0068**	**0.0021**	**93.20%**	100.00%	**99.19%**	**92.80%**

Table 24: Additional analysis on our design for non-differentiable guidance with GeoLDM on GEOM under GFN2-xTB force guidance. Note that AIMNet2 and eSEN-sm-cons. (Levine et al., 2025) are surrogate models whose gradients serve as differentiable approximations of the non-differentiable evaluator

Method	Force RMS ↓	Energy above Ground State ↓	Validity ↑	Uniqueness ↑	Atom Stability ↑
Unconditional	0.0478	0.2248	49.00%	**100.00%**	84.53%
ChemGuide (w/ SPSA) (Shen et al., 2025; Spall, 1992)	0.0411	0.1935	50.40%	**100.00%**	84.36%
ChemGuide (w/ RGF) (Shen et al., 2025; Nesterov & Spokoiny, 2017)	0.0427	0.2041	49.60%	**100.00%**	84.32%
ChemGuide (w/ ES) (Shen et al., 2025; Salimans et al., 2017)	0.0420	0.2085	48.20%	**100.00%**	84.45%
AIMNet2 (Anstine et al., 2025)	0.0414	0.2283	49.80%	**100.00%**	85.13%
eSEN-sm-cons. (Levine et al., 2025)	0.0398	0.2213	49.80%	**100.00%**	84.79%
ChemGuide (w/ REINFORCE) (Shen et al., 2025; Williams, 1992)	0.0420	0.2067	48.20%	99.59%	83.27%
ChemGuide (w/ RS-ZO) (Shen et al., 2025; Yu et al., 2025)	0.0413	0.1960	50.40%	**100.00%**	84.45%
SCG (Huang et al., 2024b)	0.0398	0.2012	49.20%	**100.00%**	84.34%
Tree search (Kocsis & Szepesvári, 2006)	0.0447	0.2168	47.80%	99.58%	84.26%
SMC (Naesseth et al., 2019)	0.0431	0.2154	47.20%	**100.00%**	83.48%
Ours	**0.0211**	**0.1003**	**59.60%**	100.00%	**85.96%**

Evaluation Using a Different Evaluator Software. In the main experiments, following the main experimental setting of ChemGuide (Shen et al., 2025), we use the same evaluator software for guidance and evaluation. We further consider the additional experimental setting in ChemGuide (Shen et al., 2025), where the generated molecules are evaluated using software different from that used for guidance. Following this setting, we use GFN2-xTB for guidance and Gaussian16 (Frisch et al., 2016) for evaluation. Following ChemGuide, we report validity, which measures the percentage of generated molecules that are valid and is calculated using DFT here. We also report strict validity, under which a molecule is counted as valid only if its generated geometry is no more than 50 kcal/mol (0.07968 Eh) above its optimized geometry, together with energy above the ground state in both Eh and kcal/mol. The results are reported in Table 25.

Table 25: Results with GFN2-xTB for guidance and Gaussian16 for evaluation on QM9 with GeoLDM over 500 generated molecules.

Method	Validity ↑	Strict Validity ↑	Energy above Ground State (Eh) ↓	Energy above GS (kcal/mol) ↓
Unconditional	93.60%	93.20%	0.004480	2.811
ChemGuide (w/ SPSA) (Shen et al., 2025; Spall, 1992)	96.20%	95.60%	0.004051	2.542
ChemGuide (w/ RGF) (Shen et al., 2025; Nesterov & Spokoiny, 2017)	96.00%	95.20%	0.004173	2.619
ChemGuide (w/ ES) (Shen et al., 2025; Salimans et al., 2017)	96.00%	95.40%	0.004136	2.596
Ours	**97.60%**	**96.40%**	**0.002030**	**1.274**

Comparison with Post-hoc xTB Optimization. To further evaluate the effectiveness of applying xTB-based force guidance during generation, we compare CGMD with post-hoc xTB geometry optimization. Starting from molecules generated by the unconditional model, the post-hoc baseline optimizes their atomic coordinates using xTB after generation. We compare this baseline with unconditional generation and our inference-time xTB-guided generation. As xTB optimization can alter or even corrupt molecular structures (Nikitin et al., 2025), for a fair comparison, we limit the number of optimization iterations and the maximum displacement to keep validity and molecule stability close to their pre-optimization levels. The results are reported in Table 26. As shown, CGMD achieves superior performance compared with the baselines.

Table 26: Comparison with post-hoc GFN2-xTB geometry optimization using GeoLDM on QM9 over 500 generated molecules.

Method	Force RMS ↓	Energy above Ground State ↓	Validity ↑	Uniqueness ↑	Atom Stability ↑	Molecule Stability ↑
Unconditional	0.0111	0.0050	89.80%	**100.00%**	98.93%	88.80%
Unconditional + Post-hoc xTB	0.0094	0.0047	89.80%	99.11%	99.09%	88.80%
Ours	**0.0068**	**0.0021**	**93.20%**	**100.00%**	**99.19%**	**92.80%**

Comparison with Per-Step Post-hoc xTB Optimization. To further compare guidance during generation with direct geometry optimization, we construct a per-step post-hoc baseline using EDM. At each guided denoising step, this baseline applies GFN2-xTB geometry optimization to the predicted clean molecular structure. As noted above, xTB optimization can alter or even corrupt molecular structures (Nikitin et al., 2025). Therefore, we limit the number of optimization iterations and the maximum displacement so that validity and molecule stability remain close to their pre-optimization levels. The results are reported in Table 27. As shown, CGMD achieves the best overall performance among the compared baselines.

Table 27: Comparison with per-step post-hoc GFN2-xTB geometry optimization using EDM on QM9 over 500 generated molecules.

Method	Force RMS ↓	Energy above Ground State ↓	Validity ↑	Uniqueness ↑	Atom Stability ↑	Molecule Stability ↑
Unconditional (EDM)	0.0114	0.0072	86.60%	**100.00%**	98.53%	83.60%
Per-Step Post-hoc GFN2-xTB Optimization	0.0104	0.0062	86.60%	99.31%	98.57%	83.60%
Ours (EDM)	**0.0096**	**0.0053**	**87.20%**	**100.00%**	**98.69%**	**86.80%**

Comparison under a Fixed Number of Valid Candidates. In the main experiments, following prior work (Shen et al., 2025), we compare all methods by generating the same number of molecules (500), thereby controlling the generation budget for a fair comparison. In practical molecular screening, however, a fixed candidate budget may instead be used, e.g., testing approximately 100 compounds (Ackloo et al., 2022; Huang et al., 2021). We therefore additionally consider a complementary setting where the objective is to obtain 100 valid molecules. For each method, we continuously generate and evaluate molecules until 100 valid molecules are obtained. We further include a generate-and-select baseline. Within the same wall-clock time required by CGMD to generate and evaluate 100 valid molecules, this baseline generates molecules unconditionally, evaluates them, and selects the 100 valid candidates with the lowest evaluator loss. Results are reported in Table 28. As shown, CGMD achieves superior performance compared with the baselines.

Table 28: Comparison under a fixed number of 100 valid candidates using GFN2-xTB force guidance with GeoLDM on QM9.

Method	Force RMS ↓	Energy above Ground State ↓	Validity ↑	Uniqueness ↑	Atom Stability ↑	Molecule Stability ↑
Unconditional	0.0110	0.0049	**100.00%**	**100.00%**	98.98%	89.00%
ChemGuide (w/ SPSA) (Shen et al., 2025; Spall, 1992)	0.0103	0.0041	**100.00%**	**100.00%**	99.32%	94.00%
ChemGuide (w/ RGF) (Shen et al., 2025; Nesterov & Spokoiny, 2017)	0.0101	0.0047	**100.00%**	**100.00%**	99.37%	94.00%
ChemGuide (w/ ES) (Shen et al., 2025; Salimans et al., 2017)	0.0104	0.0052	**100.00%**	**100.00%**	99.34%	93.00%
AIMNet2 (Anstine et al., 2025)	0.0107	0.0053	**100.00%**	**100.00%**	99.23%	94.00%
eSEN-sm-cons. (Levine et al., 2025)	0.0105	0.0054	**100.00%**	**100.00%**	99.37%	93.00%
Generate-and-Select	0.0084	0.0035	**100.00%**	**100.00%**	99.57%	95.00%
Ours	**0.0053**	**0.0019**	**100.00%**	**100.00%**	**99.91%**	**99.00%**

Further Analysis on the Number of Evaluator Queries. Figure 11 compares CGMD against w/o evaluator-informed sampling, which removes the evaluator-informed sampling distribution and samples from the original Gaussian distribution. As shown, the variant without evaluator-informed sampling remains worse than CGMD even with substantially more evaluator queries, highlighting the effectiveness of our evaluator-informed sampling strategy.

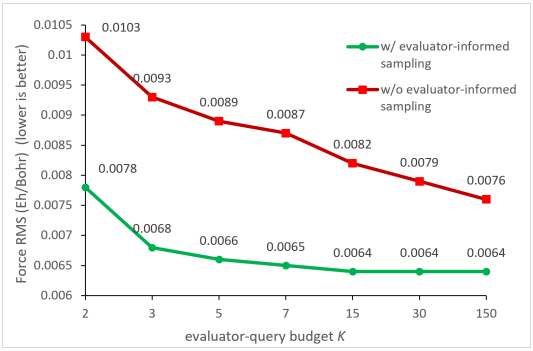


Figure 11: Comparison between the naive Monte Carlo realization of Eq. 7 and our evaluator-informed sampling under different numbers of evaluator queries per step.

Further Analysis on Evaluator-Informed Sampling. To further understand the effect of evaluator-informed sampling, we compare CGMD with the *w/o evaluator-informed sampling* variant, which removes the evaluator-informed sampling strategy and adopts naive Monte Carlo sampling.

First, we examine whether the evaluator loss decreases consistently along the guided denoising trajectory. Specifically, we measure the proportion of inference steps where the evaluator loss decreases after applying the sampled perturbation. As shown in Table 29, CGMD achieves a higher ratio of loss-decreasing steps than the *w/o evaluator-informed sampling* variant. This indicates that evaluator-informed sampling leads to more stable guidance during inference.

Table 29: Comparison of the proportion of inference steps where the evaluator loss decreases on QM9 with GeoLDM under GFN2-xTB force guidance. Higher values indicate that the sampling strategy more frequently produces loss-reducing perturbations.

Method	Loss-decreasing Steps (%) ↑
w/o evaluator-informed sampling	68.20%
Ours	**90.15%**

Second, we compare the validity of perturbations produced by different sampling strategies along the CGMD inference trajectory. We measure the proportion of sampled perturbations whose corresponding terminal clean-state estimates are valid under our xTB-based validity criterion (Appendix E.3).

As shown in Table 30, evaluator-informed sampling yields a higher proportion of valid perturbations under the same per-step evaluator-query budget, indicating that it more effectively samples perturbations from valid molecular regions.

Table 30: Comparison of the proportion of valid perturbations under different sampling strategies. Experiments are conducted on QM9 with GeoLDM under GFN2-xTB force guidance. Both methods use the same per-step evaluator-query budget.

Method	Valid Perturbations (%) ↑
w/o evaluator-informed sampling	48.11%
Ours	**63.44%**

Impact of Monte Carlo Estimation. We use the standard and well-established plain Monte Carlo method (Owen, 2013) to approximate the expectation in Eq. 7 and build our evaluator-informed sampling and hierarchy-preserving control upon it. Here, we also adapt two classical Monte Carlo variants, the difference estimator (Owen, 2013) and antithetic sampling (Owen, 2013), to our setting and incorporate them into CGMD. We compare these three Monte Carlo implementations of CGMD. For each variant, we report results using its best-performing configuration under the same per-step evaluator-query budget. As shown in Table 31, CGMD consistently achieves strong performance with different Monte Carlo methods.

Table 31: Comparison of Monte Carlo estimators within CGMD on QM9 with GeoLDM under GFN2-xTB force guidance.

Method	Force RMS ↓	Energy above Ground State ↓	Validity ↑	Uniqueness ↑	Atom Stability ↑	Molecule Stability ↑
CGMD with Plain Monte Carlo	0.0068	0.0021	93.20%	100.00%	99.19%	92.80%
CGMD with Difference-Estimator-Based Monte Carlo	0.0072	0.0023	92.80%	100.00%	99.15%	92.40%
CGMD with Antithetic-Sampling-Based Monte Carlo	0.0070	0.0025	93.00%	100.00%	99.13%	92.20%

Further Analysis on Hierarchy-Preserving Control. To further understand the effect of hierarchy-preserving control, we compare the loss recorded in denoising trajectories with and without hierarchy-preserving control under the mixed-guidance setting. Specifically, we track the force loss and the property loss across guided denoising steps, where the property objective is static molecular polarizability α. The detailed definitions of these losses are provided in Appendix E.2. As shown in Fig. 12, without hierarchy-preserving control, the two losses fluctuate more strongly along the denoising trajectory. In contrast, hierarchy-preserving control leads to smoother decreases in both losses, indicating more effective coordination between force guidance and property guidance.

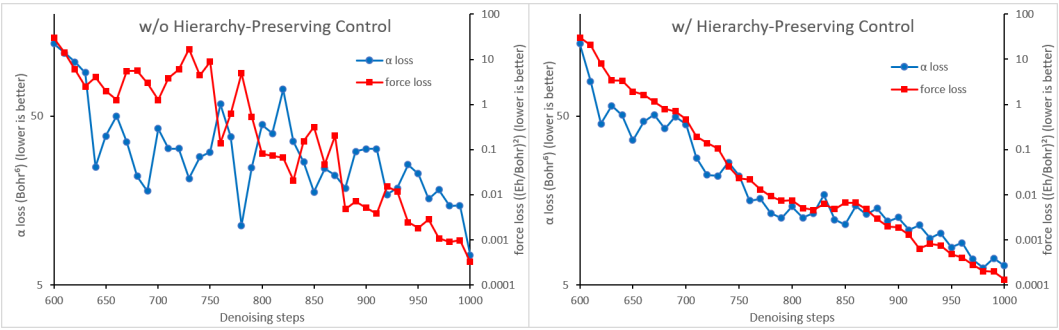


Figure 12: Visualization of loss recorded across diffusion guidance steps under mixed guidance. We show force loss and α loss across guidance denoising steps. Left: w/o Hierarchy-Preserving Control. Right: w/ Hierarchy-Preserving Control.

Runtime Analysis. In Table 32, we report the inference-time comparison between our method and the compared methods. Here, guidance computation refers to constructing the guidance signal from the evaluator outputs. As shown, on the one hand, compared with methods that use software evaluators, while our method substantially outperforms them, its inference time remains comparable to theirs. This is because the computation required for our guidance (measured from receiving the evaluator outputs to obtaining the guidance), similar to that of existing gradient-estimation methods, accounts for only a very small fraction of the overall conditional generation time (see Table 33). On the other hand, compared with surrogate-based methods, our method not only achieves substantially better performance but also has a shorter inference time. This is because accurate force surrogates in molecular generation often require complex neural models, making surrogate-based methods slower.

Table 32: Inference-time performance and runtime comparison for generating 500 molecules under GFN2-xTB force guidance using GeoLDM on QM9.

Method	Time ↓	Force RMS ↓	Energy above Ground State ↓
ChemGuide (w/ SPSA) (Shen et al., 2025; Spall, 1992)	2h 02m	0.0104	0.0042
ChemGuide (w/ RGF) (Shen et al., 2025; Nesterov & Spokoiny, 2017)	**2h 01m**	0.0103	0.0049
ChemGuide (w/ ES) (Shen et al., 2025; Salimans et al., 2017)	2h 03m	0.0105	0.0053
AIMNet2 (Anstine et al., 2025)	2h 34m	0.0110	0.0055
eSEN-sm-cons. (Levine et al., 2025)	2h 16m	0.0107	0.0055
Ours	2h 06m	**0.0068**	**0.0021**

Table 33: Guidance computation overhead for generating 500 molecules under GFN2-xTB force guidance using GeoLDM on QM9.

Method	Guidance Computation (%)
ChemGuide (w/ SPSA)	0.92%
ChemGuide (w/ RGF)	0.90%
ChemGuide (w/ ES)	0.97%
Ours	1.18%

Further Analysis on Tweedie's Formula. In this work, we propose CGMD as a control-based framework for non-differentiable diffusion guidance. Within this framework, following common practice in diffusion guidance (Ye et al., 2024; Han et al., 2024; Shen et al., 2025), we use Tweedie's formula to readily obtain terminal clean-state estimates at intermediate guidance steps while notably avoiding gradient recovery altogether. To examine whether this design materially changes the final generated distribution, we compare the complete CGMD pipeline with the *w/o Tweedie* variant. At every guidance step, this variant replaces the Tweedie estimate with a full rollout of the remaining unconditional diffusion process to the terminal state before querying the evaluator, while keeping all other settings identical. Thus, we compare the final distributions produced by the two complete pipelines rather than their outputs at a single denoising step. We report molecule-level NSPDK-MMD (Costa & De Grave, 2010) and 3D geometry MMDs (Luo & Ji, 2022; Huang et al., 2024a). As shown in Table 34, the differences are consistently small. Accordingly, we retain Tweedie's formula rather than performing full terminal rollouts.

Table 34: Distributional discrepancy between the complete CGMD pipelines with and without Tweedie's formula on QM9 with GeoLDM under GFN2-xTB force guidance over 500 generated molecules.

Comparison	NSPDK-MMD ↓	Bond-length MMD ↓	Bond-angle MMD ↓	Dihedral-angle MMD ↓
w/ Tweedie vs. w/o Tweedie	0.000264	0.000745	0.000232	0.000037

D More Details about CGMD

D.1 Naive Monte Carlo Instantiation

In this section, we present a naive Monte Carlo instantiation of Eq. 7. Recall that Eq. 7 expresses the non-differentiable control increment as an analytically intractable Brownian-increment expectation weighted by terminal evaluator values. The naive instantiation samples Brownian increments directly from their original distribution. Let p_0 denote the distribution of $\mathrm{d}\mathbf{B}_t$, i.e., $p_0 = \mathcal{N}(\mathbf{0}, \mathrm{d}t\,\mathbf{I})$. For brevity, we write $\hat{\mathbf{z}}_T = \mathcal{T}_t(\mathbf{z}_t)$ for the terminal clean-state estimate of $\mathbf{z}_t$ obtained by the Tweedie formula (Efron, 2011), where $\mathcal{T}_t$ denotes the corresponding Tweedie operator at step t, so that $\hat{\mathbf{z}}_T^{(k)} = \mathcal{T}_t(\mathbf{z}_t^{(k)})$ refers to the same Tweedie estimate as in the main text (Eqs. 9 and 11).

Concretely, for $k = 1, \ldots, K$, we draw $\mathrm{d}\mathbf{B}_t^{(k)} \sim p_0$, perturb the current state as $\mathbf{z}_t^{(k)} = \mathbf{z}_t + g(t)\mathrm{d}\mathbf{B}_t^{(k)}$, and obtain the corresponding terminal clean-state estimate $\hat{\mathbf{z}}_T^{(k)}$ from $\mathbf{z}_t^{(k)}$ using the Tweedie formula. The unnormalized evaluation weight is $r_t^{(k)} = \exp[-L(\hat{\mathbf{z}}_T^{(k)})]$, and the normalized weight is $\bar{r}_t^{(k)} = r_t^{(k)} / \sum_{j=1}^{K} r_t^{(j)}$. The naive Monte Carlo estimator is then

$$\widehat{\mathbf{c}}_t \mathrm{d}t = \sum_{k=1}^{K} \bar{r}_t^{(k)} \mathrm{d}\mathbf{B}_t^{(k)}. \tag{13}$$

Although this estimator provides a practical Tweedie-based construction for Eq. 7, it samples Brownian increments blindly from p_0. In molecular generation, most such perturbations can receive negligible terminal weights, leading to a small effective sample size and making the estimate query-inefficient. The proposal construction in the following sections keeps the same target expectation but changes where samples are drawn.

D.2 More Details about Constructing Evaluator-Informed Anchors

As introduced in Section 4.1, CGMD constructs an evaluator-informed sampling distribution by mapping cached evaluator queries from earlier denoising steps to the current step. The mapping requirement is summarized by Eq. 8 as a fixed-point equation: given a cached terminal clean-state estimate $\hat{\mathbf{z}}_T^{(j)}$, find a current-step state $\bar{\mathbf{z}}_t^{(j)}$ whose Tweedie clean-state estimate equals $\hat{\mathbf{z}}_T^{(j)}$.

In this section, the explicit form of f in Eq. 8 is derived, followed by a practical solution that avoids fixed-point iteration.

Recall that under the DDPM forward noising process, a clean state $\mathbf{z}_T$ and a noisy state $\mathbf{z}_t$ at step t are related by

$$\mathbf{z}_t = \sqrt{\bar{\alpha}_{T-t}}\,\mathbf{z}_T + \sigma_t \boldsymbol{\epsilon}, \qquad \sigma_t = \sqrt{1-\bar{\alpha}_{T-t}}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(0, I), \tag{14}$$

where $\bar{\alpha}_{T-t} = \exp(-\int_0^{T-t} \beta(s)\,\mathrm{d}s)$. The Tweedie clean-state estimator at step t is $\mathcal{T}_t(\mathbf{z}_t) = \frac{\mathbf{z}_t - \sigma_t \hat{\boldsymbol{\epsilon}}_\theta(\mathbf{z}_t, T-t)}{\sqrt{\bar{\alpha}_{T-t}}}$, where $\hat{\boldsymbol{\epsilon}}_\theta$ is the noise predictor of the diffusion backbone. Imposing the requirement $\mathcal{T}_t(\bar{\mathbf{z}}_t^{(j)}) = \hat{\mathbf{z}}_T^{(j)}$ and rearranging gives

$$\bar{\mathbf{z}}_t^{(j)} \;=\; \sqrt{\bar{\alpha}_{T-t}}\,\hat{\mathbf{z}}_T^{(j)} + \sigma_t\,\hat{\boldsymbol{\epsilon}}_\theta\big(\bar{\mathbf{z}}_t^{(j)},\, T-t\big). \tag{15}$$

This leads to Eq. 8, with

$$f\big(\bar{\mathbf{z}}_t^{(j)};\, \hat{\mathbf{z}}_T^{(j)}\big) \;=\; \sqrt{\bar{\alpha}_{T-t}}\,\hat{\mathbf{z}}_T^{(j)} + \sigma_t\,\hat{\boldsymbol{\epsilon}}_\theta\big(\bar{\mathbf{z}}_t^{(j)},\, T-t\big). \tag{16}$$

The dependence of the right-hand side on $\bar{\mathbf{z}}_t^{(j)}$ through $\hat{\boldsymbol{\epsilon}}_\theta$ makes Eq. 15 a genuine fixed-point equation, which in general requires fixed-point iteration to solve.

Instead of directly solving Eq. 15 by iteration, we exploit the fact that the cached evaluator query already comes with a noise coordinate associated with the cached endpoint $\hat{\mathbf{z}}_T^{(j)}$. Specifically, for each cached evaluator query, we have stored a noisy state $\mathbf{z}_{s,j}$ from the earlier noise level s at which the query was made, the corresponding terminal clean-state estimate $\hat{\mathbf{z}}_T^{(j)}$, and its evaluator value $L(\hat{\mathbf{z}}_T^{(j)})$. From the forward noising relation at level s,

$$\mathbf{z}_{s,j} \;=\; \sqrt{\bar{\alpha}_{T-s}}\,\hat{\mathbf{z}}_T^{(j)} + \sigma_s\,\boldsymbol{\xi}^{(j)}, \qquad \sigma_s = \sqrt{1-\bar{\alpha}_{T-s}}, \tag{17}$$

we recover the noise coordinate of the cached endpoint:

$$\boldsymbol{\xi}^{(j)} \;=\; \frac{\mathbf{z}_{s,j} - \sqrt{\bar{\alpha}_{T-s}}\,\hat{\mathbf{z}}_T^{(j)}}{\sigma_s}. \tag{18}$$

Substituting $\boldsymbol{\xi}^{(j)}$ in place of $\hat{\boldsymbol{\epsilon}}_\theta(\bar{\mathbf{z}}_t^{(j)}, T-t)$ in f yields a closed-form approximation:

$$\bar{\mathbf{z}}_t^{(j)} \;=\; \sqrt{\bar{\alpha}_{T-t}}\,\hat{\mathbf{z}}_T^{(j)} + \sigma_t\,\boldsymbol{\xi}^{(j)}. \tag{19}$$

This bypasses fixed-point iteration and requires no additional forward passes through the noise predictor.

D.3 More Details about Fitting the Student-t Mixture Proposal

In this section, we describe how the proposal distribution $q_{\phi_t}(\mathrm{d}\mathbf{B}_t)$ used in Eq. 9 is fit from the evaluator-informed anchors constructed in Appendix D.2. Specifically, at step t, we have a set of Brownian-increment centers $\{(\boldsymbol{\mu}_t^{(j)}, L^{(j)})\}_{j=1}^{N_t}$, where N_t is the number of cached anchors available at this step and $L^{(j)} = L(\hat{\mathbf{z}}_T^{(j)})$ is the evaluator value stored from the original cached query (as defined in Appendix D.2).

Conversion to Brownian-increment coordinates. The anchor $\bar{\mathbf{z}}_t^{(j)}$ derived above lives in the molecular state space, while the proposal distribution q_{ϕ_t} in Eq. 9 is parameterized in Brownian-increment coordinates. We therefore convert each state-space anchor $\bar{\mathbf{z}}_t^{(j)}$ into the Brownian increment

that would move $\mathbf{z}_t$ toward it. Since a sampled Brownian increment $\mathrm{d}\mathbf{B}_t$ perturbs the current state as $\mathbf{z}_t^{\mathrm{pert}} = \mathbf{z}_t + g(t)\mathrm{d}\mathbf{B}_t$, the corresponding Brownian-increment center is

$$\boldsymbol{\mu}_t^{(j)} = \frac{\bar{\mathbf{z}}_t^{(j)} - \mathbf{z}_t}{g(t)}. \tag{20}$$

We follow ChemGuide (Shen et al., 2025) and use the binary mask $\mathbf{m}_t$ for guidance applied only to atomic coordinates, with $\odot$ denoting element-wise multiplication:

$$\boldsymbol{\mu}_t^{(j)} = \mathbf{m}_t \odot \frac{\bar{\mathbf{z}}_t^{(j)} - \mathbf{z}_t}{g(t)}. \tag{21}$$

This yields a set of current-step Brownian-increment centers $\{(\boldsymbol{\mu}_t^{(j)}, L^{(j)})\}_j$, where each center $\boldsymbol{\mu}_t^{(j)}$ is paired with the evaluator value $L^{(j)} = L(\hat{\mathbf{z}}_T^{(j)})$ stored from the original cached query. The construction reuses historical evaluator information solely to determine where fresh perturbations should be sampled at the current step, without invoking any additional evaluator query.

Inherited evaluator weights. We convert each stored evaluator value into an inherited evaluator weight $r^{(j)} = \exp(-L^{(j)})$ and normalize these inherited weights as

$$\omega_t^{(j)} = \frac{r^{(j)}}{\sum_{k=1}^{N_t} r^{(k)}}. \tag{22}$$

These weighted centers summarize where previously evaluated low-cost endpoints would appear in the current Brownian-increment space. They are used only for fitting the proposal distribution and do not require additional evaluator queries.

Student-t mixture proposal. To capture multiple promising perturbation regions while remaining robust to imperfect anchors, we fit a Student-t mixture to the weighted centers. The proposal distribution is defined as

$$q_{\phi_t}(\mathrm{d}\mathbf{B}_t) = \sum_{a=1}^{M} \lambda_t^{(a)}\, t_\nu\Big(\mathrm{d}\mathbf{B}_t;\, \boldsymbol{\mu}_t^{(a)},\, \boldsymbol{\Sigma}_t\Big), \tag{23}$$

where M is the number of mixture components, $\lambda_t^{(a)}$ is the mixture weight of component a, and $t_\nu(\cdot; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ denotes the density of a multivariate Student-t distribution with degree of freedom ν, location $\boldsymbol{\mu}$, and scale matrix $\boldsymbol{\Sigma}$. The proposal parameters are $\phi_t = \{\lambda_t^{(a)}, \boldsymbol{\mu}_t^{(a)}, \boldsymbol{\Sigma}_t\}_{a=1}^M$ (Zhou et al., 2014), with ν fixed.

Fitting via weighted EM. The proposal parameters are fitted by weighted EM through the objective

$$\max_{\phi_t} \sum_{j=1}^{N_t} \omega_t^{(j)} \log q_{\phi_t}\Big(\boldsymbol{\mu}_t^{(j)}\Big). \tag{24}$$

Each EM iteration admits closed-form weighted updates for the Student-t mixture parameters. This fitting step is performed only over cached centers with inherited evaluator weights, so its cost is negligible compared with fresh evaluator queries.

D.4 More Details about the Score Function

In this section, we describe how the score $\nabla_{\mathbf{z}_t} \log p_t(\mathbf{z}_t)$ in the reverse-time SDE (Eq. 1) is obtained from the pretrained diffusion backbone. Recall that under the forward-time convention $\mathbf{z}_t = \mathcal{G}_{T-t}$ from Section 4.1, $\mathbf{z}_T$ denotes the clean state, so the marginal density indexed by the forward-time step t used here corresponds to noise level $T - t$. In our experiments, the molecular diffusion backbones use the DDPM-style noise-prediction parameterization, so the score used by the sampler is obtained from the predicted noise as follows.

Under the DDPM forward noising process,

$$\mathbf{z}_t = \sqrt{\bar{\alpha}_{T-t}}\mathbf{z}_T + \sqrt{1 - \bar{\alpha}_{T-t}}\boldsymbol{\epsilon}, \qquad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \tag{25}$$

where $\bar{\alpha}_{T-t} = \exp(-\int_0^{T-t} \beta(s)\,\mathrm{d}s)$. The corresponding conditional Gaussian score is $\nabla_{\mathbf{z}_t} \log q(\mathbf{z}_t \mid \mathbf{z}_T) = -\boldsymbol{\epsilon}/\sqrt{1-\bar{\alpha}_{T-t}}$. Therefore, when the backbone predicts the noise $\hat{\boldsymbol{\epsilon}}_\theta(\mathbf{z}_t, T-t)$, we instantiate the score used by the sampler as

$$\widehat{\mathbf{s}}_\theta(\mathbf{z}_t, t) := -\frac{\hat{\boldsymbol{\epsilon}}_\theta(\mathbf{z}_t, T-t)}{\sqrt{1-\bar{\alpha}_{T-t}}}. \tag{26}$$

The implemented reverse drift is then obtained by using $\widehat{\mathbf{s}}_\theta(\mathbf{z}_t, t)$ in place of the score term in Eq. 1. This subsection only clarifies the score parameterization used in implementation. The theoretical derivations remain written in terms of $\nabla_{\mathbf{z}_t} \log p_{T-t}(\mathbf{z}_t)$.

The forward-time drift used in the rest of this paper enters only through $\mathbf{b}(\mathbf{z}_t, t) = \frac{1}{2}\beta(T-t)\mathbf{z}_t + \beta(T-t)\nabla_{\mathbf{z}_t} \log p_{T-t}(\mathbf{z}_t)$ and is therefore agnostic to which of the two parameterizations the backbone uses.

E More Details about Experiments

E.1 Algorithm

We summarize the complete inference-time procedure of CGMD in Algorithm 1. At each guided denoising step, CGMD computes the terminal clean-state estimate $\hat{\mathbf{z}}_T = \mathcal{T}_t(\mathbf{z}_t)$. For force guidance, CGMD constructs the proposal $q_{\phi_t}(\mathrm{d}\mathbf{B}_t)$ from the endpoint memory as described in Appendix D.3, draws fresh Brownian increments, evaluates their terminal clean-state estimates, and forms the stability control $\mathbf{c}_t^f$ via Eq. 9. In the force-guidance setting, this stability control is directly used as the per-step control term, i.e., $\mathbf{c}_t = \mathbf{c}_t^f$. In the mixed-guidance setting, CGMD additionally forms the property control $\mathbf{c}_t^r$ via Eq. 11. Following the natural hierarchy in molecular design, CGMD treats $\mathbf{c}_t^f$ as the foundation control and $\mathbf{c}_t^r$ as the refinement control, and combines them through the hierarchy-preserving control in Eq. 12. The resulting control term $\mathbf{c}_t$ is then injected into the reverse update. To match the experimental protocol in Section E.2, the per-step control term $\mathbf{c}_t$ is applied only during the last 400 of the $T = 1000$ denoising steps.

Algorithm 1 Control-Guided Molecular Diffusion (CGMD)

Input: pretrained 3D molecular diffusion model p_θ, design target $\mathbf{y}$, stability evaluator L, property likelihood $p(\mathbf{y} \mid \cdot)$, number of perturbations K, guidance start step T_{guide}.
Output: generated molecule $\mathbf{z}_T$.
$\mathbf{z}_0 \sim p(\mathbf{z}_0)$
for all t **from** 0 **to** $T-1$ **do**
  $\mathbf{z}_{t+1} \sim p_\theta(\mathbf{z}_{t+1} \mid \mathbf{z}_t)$
  if $t \geq T_{\mathrm{guide}}$ **then**
    $\hat{\mathbf{z}}_T \leftarrow \mathcal{T}_t(\mathbf{z}_t)$
    construct $q_{\phi_t}(\mathrm{d}\mathbf{B}_t)$ from cached evaluator-informed anchors
    $\mathrm{d}\mathbf{B}_t^{(k)} \sim q_{\phi_t}(\mathrm{d}\mathbf{B}_t), \qquad k = 1, \ldots, K$
    $\mathbf{z}_t^{(k)} \leftarrow \mathbf{z}_t + g(t)\mathrm{d}\mathbf{B}_t^{(k)}$
    $\hat{\mathbf{z}}_T^{(k)} \leftarrow \mathcal{T}_t(\mathbf{z}_t^{(k)})$
    $r_t^{(k)} \leftarrow \exp\Big(-L(\hat{\mathbf{z}}_T^{(k)})\Big)\, \frac{p_0(\mathrm{d}\mathbf{B}_t^{(k)})}{q_{\phi_t}(\mathrm{d}\mathbf{B}_t^{(k)})}$
    $\widehat{\mathbf{c}}_t \mathrm{d}t \leftarrow \frac{\sum_{k=1}^K r_t^{(k)} \mathrm{d}\mathbf{B}_t^{(k)}}{\sum_{k=1}^K r_t^{(k)}}$
    cache $\{(t, \mathbf{z}_t^{(k)}, \hat{\mathbf{z}}_T^{(k)}, L(\hat{\mathbf{z}}_T^{(k)}))\}_{k=1}^K$ as evaluator-informed anchors for later steps
    if property guidance is used **then**
      obtain the final hierarchy-preserving control $\widehat{\mathbf{c}}_t$
    end if
    apply $\widehat{\mathbf{c}}_t$ to update $\mathbf{z}_{t+1}$
  end if
end for
return $\mathbf{z}_T$

E.2 More Details about Implementation

For experimental settings following ChemGuide (Shen et al., 2025), we adopt the same evaluator definitions, guidance settings, and objective configurations.

Datasets. We conduct our experiments on the QM9 (Ramakrishnan et al., 2014) and GEOM (Axelrod & Gomez-Bombarelli, 2022) benchmarks. QM9 contains 134K drug-like organic molecules, each carrying no more than nine heavy (non-hydrogen) atoms. GEOM, formally the Geometric Ensemble Of Molecules, is the larger benchmark of the two: it spans 450K molecules whose heavy-atom counts reach as high as 91 (24.9 on average), and additionally releases 37M conformations, each annotated with its geometry, energy, and statistical weight.

3D molecular diffusion models We use GeoLDM (Xu et al., 2023) as the diffusion backbone, and additionally test our framework with the popular EDM (Hoogeboom et al., 2022). For both backbones, we use the official released implementations and pretrained checkpoints.

Constraint evaluators. We use GFN2-xTB (Bannwarth et al., 2019) and g-xTB (Froitzheim et al., 2025) as the non-differentiable evaluators for force evaluation. GFN2-xTB is an accurate and broadly parametrized self-consistent tight-binding quantum chemical method with multipole electrostatics and density-dependent dispersion contributions, whose total energy decomposes as $E_{\text{GFN2-xTB}} = E_{\text{rep}} + E_{\text{disp}} + E_{\text{EHT}} + E_{\text{IES+IXC}} + E_{\text{AES}} + E_{\text{AXC}} + G_{\text{Fermi}}$, covering repulsion, dispersion, extended Hückel, isotropic and anisotropic electrostatic and exchange-correlation terms, and the Fermi-smearing free-energy contribution. In practice, we use xTB version 6.6.1, and call it through its standard command-line interface to obtain per-atom forces and total energies for each candidate structure. We further adopt g-xTB (Froitzheim et al., 2025), a next-generation general-purpose extended tight-binding electronic-structure method designed to achieve DFT-like accuracy at tight-binding speed, which substantially improves over GFN2-xTB across broad thermochemical and molecular benchmarks (Froitzheim et al., 2025). To ensure a fair comparison, we use the differentiable property regressors released with the official ChemGuide implementation (Shen et al., 2025). We conduct mixed-guidance evaluation only on QM9 because GEOM lacks the property labels needed to train the corresponding property regressors. Following the same setup, we use the same loss convention and define both the non-differentiable force loss and the differentiable property loss as mean squared errors.

Sampling configuration. We use DDPM-style sampling with $T = 1000$ denoising steps for all methods and, following Shen et al. (2025) and Han et al. (2024), inject inference-time guidance only during the last 400 denoising steps. Following ChemGuide (Shen et al., 2025), we use 500 generated molecules for force-guidance experiments and 200 generated molecules for mixed-guidance experiments, while the additional analysis in Table 28 follows its separately specified setting.

Hyperparameters of CGMD. We follow ChemGuide (Shen et al., 2025), setting `acc`=1.0 for GFN2-xTB force calculations, and use the same accuracy setting for g-xTB. For perturbation sampling, we draw $K = 3$ samples per guidance step. For force guidance, L is computed as the mean squared magnitude of the atomic forces computed by the evaluator, normalized by κ ($L = \text{MSE}/\kappa$). Here κ is absorbed into L, i.e., L in Eq. 2 and Theorem 1 denotes the normalized evaluator loss. Since the evaluator values of the K candidates within each guidance step typically differ by roughly 3×10^{-6} on QM9 and 3×10^{-3} on GEOM, we set $\kappa = 3 \times 10^{-6}$ on QM9 and $\kappa = 3 \times 10^{-3}$ on GEOM. For the Student-t mixture, we use $M = 2$ components with $\nu = 5$ and a 5-step anchor window. For numerical stability in the hierarchy-preserving projection, we set $\delta_{\text{proj}} = 10^{-12}$. We initialize component means and assignments using k-means, with the scale and mixture weights initialized from the resulting clusters. The weighted EM fitting uses a maximum of 40 iterations and a convergence tolerance of 10^{-3}.

Hardware, software, and runtime. Specifically, we use xTB 6.6.1 for GFN2-xTB and g-xTB 2.0.1. All experiments are conducted on a server with an Intel Xeon Platinum 8468 CPU and Intel Data Center GPU Max 1550. Under this setup, generating 500 molecules takes approximately 2 hours on QM9 and 11 hours on GEOM.

E.3 More Details about Evaluation Metrics

Metrics for non-differentiable guidance evaluation. We describe each metric used in the non-differentiable guidance experiments, all of which follow the evaluation protocol of ChemGuide (Shen et al., 2025).

Force RMS (Eh/Bohr ↓): The root mean square of the force vectors acting on each atom in the generated molecule, measuring the extent to which the structure deviates from mechanical equilibrium.

Energy above Ground State (Eh ↓): The energy difference between the generated molecular geometry before and after xTB relaxation, quantifying the energy excess of the generated structure relative to the ground-state conformation.

Validity (% ↑): The proportion of generated molecules that are valid and successfully evaluated.

Uniqueness (% ↑): The proportion of non-duplicate molecules.

Atom Stability (% ↑): The proportion of atoms in all generated molecules that satisfy the correct valence rules (Xu et al., 2023).

Molecule Stability (% ↑): The proportion of all generated molecules in which every atom satisfies the correct valence rules (Xu et al., 2023).

Metrics for mixed-guidance evaluation. Following ChemGuide (Shen et al., 2025), for the six differentiable molecular property objectives in the mixed-guidance experiments, we evaluate each property separately, each combined with xTB-based force guidance. For each property, we adopt the same empirical target distribution as theirs. At inference time, the guidance target value used for each generated molecule is sampled from this empirical distribution. Following the GeoLDM evaluation protocol (Xu et al., 2023), we report MAE using the evaluator reproduced by ChemGuide under the same setup as GeoLDM. Following ChemGuide (Shen et al., 2025), we report force and energy metrics.

α (Bohr^3 ↓): Norm of the static molecular polarizability.

μ (Debye ↓): Norm of the molecular dipole moment.

C_v (cal/(mol·K) ↓): Molecular heat capacity at room temperature.

$\varepsilon_{\mathrm{HOMO}}$ (eV ↓): Energy of the highest occupied molecular orbital.

$\varepsilon_{\mathrm{LUMO}}$ (eV ↓): Energy of the lowest unoccupied molecular orbital.

$\Delta\varepsilon$ (eV ↓): HOMO–LUMO energy gap, defined as $\varepsilon_{\mathrm{LUMO}} - \varepsilon_{\mathrm{HOMO}}$.

F Proofs and Derivations

F.1 Connection Between Eq. 1 and Eq. 4

In Eq. 1, the time variable runs backward from T to 0. We denote this backward time by τ to avoid confusion. To express the dynamics in the standard controlled SDE form (Eq. 4), we introduce a forward time $t = T - \tau$ and define $\mathbf{z}_t = \mathcal{G}_\tau = \mathcal{G}_{T-t}$.

Under this change of variable, $\mathrm{d}\tau = -\mathrm{d}t$, which flips the sign of the drift in Eq. 1. The reversed Brownian motion $\mathrm{d}\bar{\mathbf{B}}_\tau$ becomes a forward Brownian motion $\mathrm{d}\mathbf{B}_t$ under the time reversal. Substituting these gives

$$\mathrm{d}\mathbf{z}_t = \left[\tfrac{1}{2}\beta(T-t)\,\mathbf{z}_t + \beta(T-t)\,\nabla_{\mathbf{z}_t}\log p_{T-t}(\mathbf{z}_t)\right]\mathrm{d}t + \beta(T-t)\,\mathbf{s}_{T-t}\,\mathrm{d}t + \sqrt{\beta(T-t)}\,\mathrm{d}\mathbf{B}_t. \quad (27)$$

Matching term by term against Eq. 4 yields $\mathbf{b}(\mathbf{z}_t, t) = \frac{1}{2}\beta(T-t)\,\mathbf{z}_t + \beta(T-t)\,\nabla_{\mathbf{z}_t}\log p_{T-t}(\mathbf{z}_t)$, $g(t) = \sqrt{\beta(T-t)}$, and $\mathbf{c}_t = \sqrt{\beta(T-t)}\,\mathbf{s}_{T-t}$.

F.2 Derivation of Eq. 6

In this section, Eq. 6 is derived by writing the reverse denoising dynamics in the forward-time convention used by the controlled process in Eq. 4. Specifically, we view the process as $\mathbf{z}_t = \mathcal{G}_{T-t}$, so that t increases from the initial noisy state to the terminal clean state. Under this convention, starting from any intermediate state $\mathbf{z}_t = \mathbf{z}$, the remaining controlled trajectory evolves over $s \in [t, T]$ according to

$$\mathrm{d}\mathbf{z}_s = \mathbf{b}(\mathbf{z}_s, s)\,\mathrm{d}s + g(s)\mathbf{c}_s\,\mathrm{d}s + g(s)\,\mathrm{d}\mathbf{B}_s, \qquad s \in [t, T], \tag{28}$$

where the control $\mathbf{c}_s$ is penalized through a quadratic running cost. Since the objective is evaluated at the terminal state of this forward-time process, the terminal cost is $L(\mathbf{z}_T)$, and the corresponding cost-to-go is

$$\mathcal{C}_{\mathbf{c}}(\mathbf{z}, t) = \mathbb{E}\left[L(\mathbf{z}_T) + \int_t^T \frac{1}{2}\|\mathbf{c}_s\|_2^2\,\mathrm{d}s \,\middle|\, \mathbf{z}_t = \mathbf{z}\right], \tag{29}$$

with value function $V(\mathbf{z}, t) = \inf_{\mathbf{c}} \mathcal{C}_{\mathbf{c}}(\mathbf{z}, t)$. This future-cost formulation leads to the HJB equation

$$-\partial_t V = \min_{\mathbf{c}} \left\{(\mathbf{b} + g\mathbf{c})^\top \nabla V + \frac{1}{2}g^2\Delta V + \frac{1}{2}\|\mathbf{c}\|_2^2\right\}, \qquad V(\mathbf{z}, T) = L(\mathbf{z}). \tag{30}$$

The terminal condition appears at time T because no future cost remains once the terminal state is reached. To identify the optimal control, we only need to minimize the terms depending on $\mathbf{c}$, namely $g\mathbf{c}^\top \nabla V + \frac{1}{2}\|\mathbf{c}\|_2^2$. This gives

$$\mathbf{c}_t^* = -g(t)\nabla V(\mathbf{z}_t, t). \tag{31}$$

Substituting this minimizer back into the HJB equation removes the minimization and yields the nonlinear PDE

$$\partial_t V + \mathbf{b}^\top \nabla V + \frac{1}{2}g^2\Delta V - \frac{1}{2}g^2\|\nabla V\|_2^2 = 0. \tag{32}$$

The nonlinearity comes from the quadratic gradient term. To linearize the equation, we introduce the desirability function (Theodorou et al., 2010; Kappen, 2005)

$$h(\mathbf{z}, t) = \exp[-V(\mathbf{z}, t)]. \tag{33}$$

Using

$$\partial_t V = -\frac{\partial_t h}{h}, \qquad \nabla V = -\frac{\nabla h}{h}, \qquad \Delta V = -\frac{\Delta h}{h} + \frac{\|\nabla h\|_2^2}{h^2}, \tag{34}$$

we substitute these identities into the PDE for V. The two quadratic terms in ∇h cancel, leaving the backward Kolmogorov equation

$$\partial_t h + \mathbf{b}^\top \nabla h + \frac{1}{2}g^2\Delta h = 0, \qquad h(\mathbf{z}, T) = \exp[-L(\mathbf{z})]. \tag{35}$$

This linear equation is exactly the backward equation associated with the uncontrolled diffusion. Therefore, by the Feynman–Kac formula under the uncontrolled path measure $\mathcal{P}_0$, where

$$\mathrm{d}\mathbf{z}_s = \mathbf{b}(\mathbf{z}_s, s)\,\mathrm{d}s + g(s)\,\mathrm{d}\mathbf{B}_s, \tag{36}$$

we have

$$h(\mathbf{z}_t, t) = \mathbb{E}_{\mathcal{P}_0}\left[\exp[-L(\mathbf{z}_T)] \,|\, \mathbf{z}_t\right]. \tag{37}$$

Finally, translating back from h to V converts the value-gradient control into a log-expectation form:

$$\mathbf{c}_t^* = -g(t)\nabla V(\mathbf{z}_t, t) = g(t)\nabla_{\mathbf{z}_t} \log h(\mathbf{z}_t, t) = g(t)\nabla_{\mathbf{z}_t} \log \mathbb{E}_{\mathcal{P}_0}\left[\exp[-L(\mathbf{z}_T)] \,|\, \mathbf{z}_t\right], \tag{38}$$

which gives Eq. 6.

F.3 Derivation of Eq. 7

In this section, Eq. 7, the gradient-free stochastic-increment representation of the optimal control, is derived starting from Eq. 6. Recall that Eq. 6 expresses the optimal control as a gradient of the desirability function,

$$\mathbf{c}_t^* = g(t)\nabla_{\mathbf{z}_t} \log h(\mathbf{z}_t, t) = g(t)\frac{\nabla_{\mathbf{z}_t} h(\mathbf{z}_t, t)}{h(\mathbf{z}_t, t)}. \tag{39}$$

This form is still not directly suitable for non-differentiable guidance, because it requires differentiating a conditional expectation with respect to the noisy state $\mathbf{z}_t$. To remove this gradient, we expose the one-step Gaussian transition of the uncontrolled process. Under an Euler discretization,

$$\mathbf{z}_{t+\mathrm{d}t} = \mathbf{z}_t + \mathbf{b}(\mathbf{z}_t, t)\,\mathrm{d}t + g(t)\sqrt{\mathrm{d}t}\,\boldsymbol{\epsilon}, \qquad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \tag{40}$$

and the Markov property together with the tower rule gives

$$h(\mathbf{z}_t, t) = \mathbb{E}_{\mathcal{P}_0}\left[h(\mathbf{z}_{t+\mathrm{d}t}, t+\mathrm{d}t) \,|\, \mathbf{z}_t\right]. \tag{41}$$

Thus the gradient in Eq. 6 can be shifted onto this local Gaussian expectation. Retaining leading-order terms over the infinitesimal Euler step and exchanging differentiation with expectation under the usual regularity conditions, we obtain

$$\nabla_{\mathbf{z}_t} h(\mathbf{z}_t, t) = \mathbb{E}_{\boldsymbol{\epsilon}}\left[\nabla_{\mathbf{z}_{t+\mathrm{d}t}} h(\mathbf{z}_{t+\mathrm{d}t}, t+\mathrm{d}t)\right]. \tag{42}$$

At this point the gradient acts on a function of the Gaussian noise, which is exactly the setting where Stein's lemma can be applied. Since

$$\nabla_{\boldsymbol{\epsilon}} h(\mathbf{z}_{t+\mathrm{d}t}, t+\mathrm{d}t) = g(t)\sqrt{\mathrm{d}t}\,\nabla_{\mathbf{z}_{t+\mathrm{d}t}} h(\mathbf{z}_{t+\mathrm{d}t}, t+\mathrm{d}t), \tag{43}$$

Stein's identity gives

$$\nabla_{\mathbf{z}_t} h(\mathbf{z}_t, t) = \frac{1}{g(t)\sqrt{\mathrm{d}t}} \mathbb{E}_{\mathcal{P}_0}\left[\boldsymbol{\epsilon}\, h(\mathbf{z}_{t+\mathrm{d}t}, t+\mathrm{d}t) \,|\, \mathbf{z}_t\right]. \tag{44}$$

Substituting this expression into the control formula above yields

$$\mathbf{c}_t^* = \frac{1}{\sqrt{\mathrm{d}t}} \frac{\mathbb{E}_{\mathcal{P}_0}\left[\boldsymbol{\epsilon}\, h(\mathbf{z}_{t+\mathrm{d}t}, t+\mathrm{d}t) \,|\, \mathbf{z}_t\right]}{h(\mathbf{z}_t, t)}. \tag{45}$$

The remaining term $h(\mathbf{z}_{t+\mathrm{d}t}, t+\mathrm{d}t)$ is still a one-step-ahead desirability. To express it directly through the terminal non-differentiable evaluator cost, we again use the definition of h and the tower rule:

$$h(\mathbf{z}_{t+\mathrm{d}t}, t+\mathrm{d}t) = \mathbb{E}_{\mathcal{P}_0}\left[\exp[-L(\mathbf{z}_T)] \,|\, \mathbf{z}_{t+\mathrm{d}t}\right]. \tag{46}$$

Moreover, the Brownian increment satisfies $\mathrm{d}\mathbf{B}_t = \sqrt{\mathrm{d}t}\boldsymbol{\epsilon}$ and is measurable with respect to the one-step transition from $\mathbf{z}_t$ to $\mathbf{z}_{t+\mathrm{d}t}$. Therefore,

$$\mathbb{E}_{\mathcal{P}_0}\left[\mathrm{d}\mathbf{B}_t\, h(\mathbf{z}_{t+\mathrm{d}t}, t+\mathrm{d}t) \,|\, \mathbf{z}_t\right] = \mathbb{E}_{\mathcal{P}_0}\left[\mathrm{d}\mathbf{B}_t\, \exp[-L(\mathbf{z}_T)] \,|\, \mathbf{z}_t\right], \tag{47}$$

while the denominator is

$$h(\mathbf{z}_t, t) = \mathbb{E}_{\mathcal{P}_0}\left[\exp[-L(\mathbf{z}_T)] \,|\, \mathbf{z}_t\right]. \tag{48}$$

Combining these two identities gives the gradient-free stochastic-increment form

$$\mathbf{c}_t^*\,\mathrm{d}t = \frac{\mathbb{E}_{\mathcal{P}_0}\left[\exp[-L(\mathbf{z}_T)]\,\mathrm{d}\mathbf{B}_t \,|\, \mathbf{z}_t\right]}{\mathbb{E}_{\mathcal{P}_0}\left[\exp[-L(\mathbf{z}_T)] \,|\, \mathbf{z}_t\right]}, \tag{49}$$

which gives Eq. 7. This representation no longer requires differentiating the non-differentiable evaluator cost or estimating a non-differentiable gradient. The terminal cost enters only through evaluations of $\exp[-L(\mathbf{z}_T)]$ along uncontrolled continuations from $\mathbf{z}_t$.

For the Euler step, let $H_t(\mathbf{z}_t) := \mathbb{E}_{\boldsymbol{\epsilon}}[h(\mathbf{z}_{t+\mathrm{d}t}, t+\mathrm{d}t)]$ and $J_t := \partial \mathbf{z}_{t+\mathrm{d}t}/\partial \mathbf{z}_t = \mathbf{I} + (\nabla_{\mathbf{z}_t}\mathbf{b})\,\mathrm{d}t$. The chain rule and Stein's lemma give

$$g(t)\,\mathrm{d}t\,\nabla_{\mathbf{z}_t} \log H_t(\mathbf{z}_t) = J_t^\top \frac{\mathbb{E}_{\boldsymbol{\epsilon}}[h(\mathbf{z}_{t+\mathrm{d}t}, t+\mathrm{d}t)\,\mathrm{d}\mathbf{B}_t]}{H_t(\mathbf{z}_t)}.$$

Under the same regularity conditions, the ratio is $O(\mathrm{d}t)$ by Stein's identity, so omitting $J_t^\top - \mathbf{I}$ changes the increment only by $O((\mathrm{d}t)^2)$. As $\mathrm{d}t \to 0$, $\nabla \log H_t \to \nabla \log h$. This establishes Eq. 7. Moreover, for the finite-step Euler implementation, the error in the control increment from omitting the Jacobian correction is only $O((\mathrm{d}t)^2)$ per step.

F.4 Elaboration on Theorem 1

In this section, we provide the detailed proof of Theorem 1. The result concerns the ideal optimal control in Eq. 6. Eq. 7 is its equivalent stochastic-increment representation derived in Appendix F.3.

Lemma 1 (Desirability-ratio change of measure)**.** *Let $\mathcal{P}_0$ denote the path measure induced by the uncontrolled diffusion process, i.e., Eq. 4 with $\mathbf{c}_t = \mathbf{0}$. Let $\mathcal{P}^{c^*}$ denote the path measure induced by the optimal control law $\mathbf{c}_t^*$ in Eq. 6. Define the desirability function*

$$h(\mathbf{z}_t, t) = \mathbb{E}_{\mathcal{P}_0}\left[\exp(-L(\mathbf{z}_T)) \,|\, \mathbf{z}_t\right]. \tag{50}$$

Then, under the standard linearly-solvable stochastic optimal control formulation (Theodorou et al., 2010; Kappen, 2005), for any $0 \le t < s \le T$,

$$\mathcal{P}^{c^*}_{t,s}(\mathbf{z}_s \mid \mathbf{z}_t) = \mathcal{P}_{0,t,s}(\mathbf{z}_s \mid \mathbf{z}_t)\frac{h(\mathbf{z}_s, s)}{h(\mathbf{z}_t, t)}. \tag{51}$$

Proof. The value function satisfies $h(\mathbf{z}_t, t) = \exp[-V(\mathbf{z}_t, t)]$. By the Feynman–Kac representation,

$$h(\mathbf{z}_t, t) = \mathbb{E}_{\mathcal{P}_0}\left[h(\mathbf{z}_T, T) \,|\, \mathbf{z}_t\right]. \tag{52}$$

With terminal boundary $h(\mathbf{z}_T, T) = \exp[-L(\mathbf{z}_T)]$, this gives Eq. 50. Substituting this desirability function into the standard path-integral change-of-measure formula yields Eq. 51. □

Theorem 1. *Recall that, for a design target $\mathbf{y}$, the terminal cost is defined as*

$$L(\mathbf{z}_T) = -\log p(\mathbf{y} \mid \mathbf{z}_T) + C, \tag{53}$$

where C is independent of $\mathbf{z}_T$.

Consider the controlled diffusion process in Eq. 4, and let $\mathbf{c}_t^*$ be the ideal optimal control law in Eq. 6, equivalently written in stochastic-increment form as Eq. 7. For any fixed initial noise $\mathbf{z}_0 = \mathbf{z}$, the controlled terminal law is the state-conditioned tilt in Eq. 58. For the population reverse process associated with a completely noised DDPM, $\bar{\alpha}_T = 0$ implies the endpoint factorization $\mathcal{P}_0(\mathbf{z}_0, \mathbf{z}_T) = p_{\text{init}}(\mathbf{z}_0)p_\theta(\mathbf{z}_T)$ (Ho et al., 2020; Song et al., 2021; Domingo i Enrich et al., 2025). Under this factorization, the terminal clean-state marginal induced by the optimally controlled process satisfies

$$\mathcal{P}^{c^*}(\mathbf{z}_T) = \frac{p_\theta(\mathbf{z}_T)\exp[-L(\mathbf{z}_T)]}{\int p_\theta(\tilde{\mathbf{z}}_T)\exp[-L(\tilde{\mathbf{z}}_T)]\,\mathrm{d}\tilde{\mathbf{z}}_T}. \tag{54}$$

Equivalently, the optimally controlled process samples from the target distribution specified in Eq. 2, and

$$\mathcal{P}^{c^*}(\mathbf{z}_T) = p_\theta(\mathbf{z}_T \mid \mathbf{y}). \tag{55}$$

Proof. By Lemma 1, taking $t = 0$ and $s = T$ gives

$$\mathcal{P}^{c^*}(\mathbf{z}_T \mid \mathbf{z}_0) = \mathcal{P}_0(\mathbf{z}_T \mid \mathbf{z}_0)\frac{\exp[-L(\mathbf{z}_T)]}{h(\mathbf{z}_0, 0)}. \tag{56}$$

Using Eq. 50,

$$h(\mathbf{z}_0, 0) = \int \mathcal{P}_0(\tilde{\mathbf{z}}_T \mid \mathbf{z}_0)\exp[-L(\tilde{\mathbf{z}}_T)]\,\mathrm{d}\tilde{\mathbf{z}}_T. \tag{57}$$

Thus,

$$\mathcal{P}^{c^*}(\mathbf{z}_T \mid \mathbf{z}_0) = \frac{\mathcal{P}_0(\mathbf{z}_T \mid \mathbf{z}_0)\exp[-L(\mathbf{z}_T)]}{\int \mathcal{P}_0(\tilde{\mathbf{z}}_T \mid \mathbf{z}_0)\exp[-L(\tilde{\mathbf{z}}_T)]\,\mathrm{d}\tilde{\mathbf{z}}_T}. \tag{58}$$

For the population reverse process specified in the theorem, the endpoint factorization gives $\mathcal{P}_0(\mathbf{z}_T \mid \mathbf{z}_0) = p_\theta(\mathbf{z}_T)$. Hence, Eq. 57 is independent of $\mathbf{z}_0$ and equals

$$Z = \int p_\theta(\tilde{\mathbf{z}}_T)\exp[-L(\tilde{\mathbf{z}}_T)]\,\mathrm{d}\tilde{\mathbf{z}}_T. \tag{59}$$

The uncontrolled process is the original pretrained diffusion sampler, so its terminal marginal is

$$p_\theta(\mathbf{z}_T) = \int \mathcal{P}_0(\mathbf{z}_T \mid \mathbf{z}_0)p_{\text{init}}(\mathbf{z}_0)\,\mathrm{d}\mathbf{z}_0. \tag{60}$$

Under the tilted path measure, the terminal marginal is obtained by integrating the path tilt over the initial noise distribution:

$$\mathcal{P}^{c^*}(\mathbf{z}_T) = \frac{\exp[-L(\mathbf{z}_T)] \int \mathcal{P}_0(\mathbf{z}_T \mid \mathbf{z}_0) p_{\text{init}}(\mathbf{z}_0)\, \mathrm{d}\mathbf{z}_0}{\int \exp[-L(\tilde{\mathbf{z}}_T)] \int \mathcal{P}_0(\tilde{\mathbf{z}}_T \mid \mathbf{z}_0) p_{\text{init}}(\mathbf{z}_0)\, \mathrm{d}\mathbf{z}_0\, \mathrm{d}\tilde{\mathbf{z}}_T} \tag{61}$$

$$= \frac{p_\theta(\mathbf{z}_T) \exp[-L(\mathbf{z}_T)]}{\int p_\theta(\tilde{\mathbf{z}}_T) \exp[-L(\tilde{\mathbf{z}}_T)]\, \mathrm{d}\tilde{\mathbf{z}}_T}. \tag{62}$$

This proves Eq. 54.

By definition, $L(\mathbf{z}_T) = -\log p(\mathbf{y} \mid \mathbf{z}_T) + C$, so $\exp[-L(\mathbf{z}_T)] \propto p(\mathbf{y} \mid \mathbf{z}_T)$. Substituting this into Eq. 54 gives

$$\mathcal{P}^{c^*}(\mathbf{z}_T) = \frac{p_\theta(\mathbf{z}_T) p(\mathbf{y} \mid \mathbf{z}_T)}{\int p_\theta(\tilde{\mathbf{z}}_T) p(\mathbf{y} \mid \tilde{\mathbf{z}}_T)\, \mathrm{d}\tilde{\mathbf{z}}_T} = p_\theta(\mathbf{z}_T \mid \mathbf{y}), \tag{63}$$

where the last equality follows from Bayes' rule. This completes the proof of Theorem 1. □

F.5 Additional Analysis on the Sampling Distribution

The proposal distribution q_{ϕ_t} is introduced only to change where Brownian increments are sampled. It should not change the target expectation defined under the original Brownian-increment distribution p_0 in the naive realization. This subsection clarifies this point in two steps. First, we show that the density-ratio correction p_0/q_{ϕ_t} is exactly a change-of-measure correction (Owen, 2013), so, with $\hat{\mathbf{z}}_T$ obtained via Tweedie's formula, the numerator and denominator in the naive realization remain the same even when samples are drawn from q_{ϕ_t}. Second, we explain why choosing q_{ϕ_t} to cover high-weight perturbation regions improves the finite-sample stability of the resulting estimator.

Effect of Proposal Sampling on the Target Expectation The estimator in Eq. 9 draws samples from q_{ϕ_t} rather than p_0, raising the question of whether it still targets the same expectation as the naive realization. This invariance in fact holds: the density-ratio factor p_0/q_{ϕ_t} in $r_t^{(k)}$ exactly compensates for the change of sampling distribution, as we show below.

The target expectation of the naive realization corresponds to a tilted distribution over $\mathrm{d}\mathbf{B}_t$. Both its numerator and denominator are integrals against the evaluator-weighted density $\exp(-L(\hat{\mathbf{z}}_T))\, p_0(\mathrm{d}\mathbf{B}_t)$, whose normalization gives

$$\pi_t(\mathrm{d}\mathbf{B}_t) = \frac{\exp(-L(\hat{\mathbf{z}}_T))\, p_0(\mathrm{d}\mathbf{B}_t)}{\int \exp(-L(\hat{\mathbf{z}}'_T))\, p_0(\mathrm{d}\mathbf{B}'_t)\, \mathrm{d}\mathbf{B}'_t}. \tag{64}$$

Under π_t, the target control increment $\mathbf{c}_t^{\text{non-diff}}\, \mathrm{d}t$ is precisely the expectation of $\mathrm{d}\mathbf{B}_t$. It thus suffices to show that the weighted estimator in Eq. 9 targets the expectation under π_t.

For samples $\mathrm{d}\mathbf{B}_t \sim q_{\phi_t}$, recall that the unnormalized weight $r_t^{(k)}$ in Eq. 9, viewed as a function of $\mathrm{d}\mathbf{B}_t$, reads

$$r_t(\mathrm{d}\mathbf{B}_t) = \exp(-L(\hat{\mathbf{z}}_T))\, \frac{p_0(\mathrm{d}\mathbf{B}_t)}{q_{\phi_t}(\mathrm{d}\mathbf{B}_t)}, \tag{65}$$

and the corresponding effective density is the normalization of $r_t(\mathrm{d}\mathbf{B}_t)\, q_{\phi_t}(\mathrm{d}\mathbf{B}_t)$:

$$\tilde{\pi}_t(\mathrm{d}\mathbf{B}_t) = \frac{r_t(\mathrm{d}\mathbf{B}_t)\, q_{\phi_t}(\mathrm{d}\mathbf{B}_t)}{\int r_t(\mathrm{d}\mathbf{B}'_t)\, q_{\phi_t}(\mathrm{d}\mathbf{B}'_t)\, \mathrm{d}\mathbf{B}'_t}. \tag{66}$$

Substituting $r_t(\mathrm{d}\mathbf{B}_t)$ into $\tilde{\pi}_t(\mathrm{d}\mathbf{B}_t)$, the proposal density q_{ϕ_t} cancels in both the numerator and the denominator, yielding

$$\tilde{\pi}_t(\mathrm{d}\mathbf{B}_t) = \frac{\exp(-L(\hat{\mathbf{z}}_T))\, \frac{p_0(\mathrm{d}\mathbf{B}_t)}{q_{\phi_t}(\mathrm{d}\mathbf{B}_t)}\, q_{\phi_t}(\mathrm{d}\mathbf{B}_t)}{\int \exp(-L(\hat{\mathbf{z}}'_T))\, \frac{p_0(\mathrm{d}\mathbf{B}'_t)}{q_{\phi_t}(\mathrm{d}\mathbf{B}'_t)}\, q_{\phi_t}(\mathrm{d}\mathbf{B}'_t)\, \mathrm{d}\mathbf{B}'_t} = \frac{\exp(-L(\hat{\mathbf{z}}_T))\, p_0(\mathrm{d}\mathbf{B}_t)}{\int \exp(-L(\hat{\mathbf{z}}'_T))\, p_0(\mathrm{d}\mathbf{B}'_t)\, \mathrm{d}\mathbf{B}'_t} = \pi_t(\mathrm{d}\mathbf{B}_t). \tag{67}$$

As shown above, the effective sampling density $\tilde{\pi}_t$ therefore coincides exactly with the target tilted distribution π_t, confirming that the self-normalized estimator $\widehat{\mathbf{c}}_t^{\text{non-diff}}\, \mathrm{d}t$ in Eq. 9 converges to the same

expectation as the naive realization. The role of q_{ϕ_t} is therefore not to alter the target expectation, but to determine where samples are drawn, which in turn governs the finite-sample efficiency of the estimator, as discussed next.

So far, we have naturally arrived at Eq. 9 by following the practical construction order of CGMD. At a more fundamental level, the proposal reweighting itself corresponds to an exact change of measure already at the level of the original Eq. 7. For a fixed current state and fitted proposal, let $\mathcal{Q}_t$ sample $\mathrm{d}\mathbf{B}_t$ from q_{ϕ_t} while retaining the conditional distribution of the remaining trajectory under $\mathcal{P}_0$. Then, provided $q_{\phi_t} > 0$ wherever $p_0 > 0$, Eq. 7 can be written exactly as

$$\mathbf{c}_t^* \,\mathrm{d}t = \frac{\mathbb{E}_{\mathcal{Q}_t}\left[e^{-L(\mathbf{z}_T)}\,\frac{p_0(\mathrm{d}\mathbf{B}_t)}{q_{\phi_t}(\mathrm{d}\mathbf{B}_t)}\,\mathrm{d}\mathbf{B}_t \mid \mathbf{z}_t\right]}{\mathbb{E}_{\mathcal{Q}_t}\left[e^{-L(\mathbf{z}_T)}\,\frac{p_0(\mathrm{d}\mathbf{B}_t)}{q_{\phi_t}(\mathrm{d}\mathbf{B}_t)} \mid \mathbf{z}_t\right]}.$$

Applying the Tweedie clean-state instantiation and the K-sample self-normalized Monte Carlo estimator to this exact reformulation recovers Eq. 9.

Analysis of the proposal distribution. Although the target expectation is invariant to the choice of q_{ϕ_t}, the estimation error depends on it. Let $\boldsymbol{\mu}_t := \mathbb{E}_{\pi_t}[\mathrm{d}\mathbf{B}_t]$ denote the target increment, and let $\widehat{\boldsymbol{\mu}}_{t,K}$ denote its self-normalized estimator in Eq. 9. Conditioning on the current state and the history used to construct q_{ϕ_t}, a standard first-order analysis (Owen, 2013) gives the trace of the asymptotic covariance of $\sqrt{K}(\widehat{\boldsymbol{\mu}}_{t,K} - \boldsymbol{\mu}_t)$ as

$$A_t(q_{\phi_t}) := \mathbb{E}_{\mathrm{d}\mathbf{B}_t \sim q_{\phi_t}}\left[\left(\frac{\pi_t(\mathrm{d}\mathbf{B}_t)}{q_{\phi_t}(\mathrm{d}\mathbf{B}_t)}\right)^2 \|\mathrm{d}\mathbf{B}_t - \boldsymbol{\mu}_t\|_2^2\right]. \tag{68}$$

When $q_{\phi_t} = \pi_t$, all samples have equal importance weights. However, when q_{ϕ_t} assigns too little probability to a region of π_t, samples from that region receive large weights and can dominate the estimate. Eq. 68 captures this effect through the squared density ratio, together with the distance of each increment from $\boldsymbol{\mu}_t$.

This dependence on the density ratio can be characterized using the Rényi divergence. Let $\rho_t := \pi_t/q_{\phi_t}$ and $D_\alpha(\pi_t\|q_{\phi_t}) := \frac{1}{\alpha-1}\log\mathbb{E}_{q_{\phi_t}}[\rho_t^\alpha]$ (Van Erven & Harremos, 2014). We write the arbitrary Hölder exponent $p > 1$ as $p = \alpha - 1$, so that the resulting density-ratio moment corresponds to D_α. For any $\alpha > 2$, a change of measure followed by Hölder's inequality with conjugate exponents $\alpha - 1$ and $(\alpha-1)/(\alpha-2)$ gives

$$\begin{aligned} A_t(q_{\phi_t}) &= \mathbb{E}_{\pi_t}\left[\rho_t\,\|\mathrm{d}\mathbf{B}_t - \boldsymbol{\mu}_t\|_2^2\right] \\ &\le \left(\mathbb{E}_{\pi_t}[\rho_t^{\alpha-1}]\right)^{\frac{1}{\alpha-1}}\left(\mathbb{E}_{\pi_t}\left[\|\mathrm{d}\mathbf{B}_t - \boldsymbol{\mu}_t\|_2^{\frac{2(\alpha-1)}{\alpha-2}}\right]\right)^{\frac{\alpha-2}{\alpha-1}} \\ &= e^{D_\alpha(\pi_t\|q_{\phi_t})}\left(\mathbb{E}_{\pi_t}\left[\|\mathrm{d}\mathbf{B}_t - \boldsymbol{\mu}_t\|_2^{\frac{2(\alpha-1)}{\alpha-2}}\right]\right)^{\frac{\alpha-2}{\alpha-1}}. \end{aligned} \tag{69}$$

The last equality follows from $\mathbb{E}_{\pi_t}[\rho_t^{\alpha-1}] = \mathbb{E}_{q_{\phi_t}}[\rho_t^\alpha]$.

For $L \ge 0$ and $Z_t := \mathbb{E}_{p_0}[e^{-L(\hat{\mathbf{z}}_T)}] > 0$, the target satisfies $\pi_t \le p_0/Z_t$ and inherits all finite-order moments from the Gaussian p_0. With $\boldsymbol{\Sigma}_t \succ 0$, the Student-$t$ proposal has full support and heavier tails than p_0, so ρ_t is bounded and $D_\alpha(\pi_t\|q_{\phi_t})$ is finite. For $\nu > 4$, the Student-t proposal also has a finite fourth moment. These properties give the following MSE expansion and bound as $K \to \infty$:

$$\begin{aligned} \mathbb{E}\left[\|\widehat{\boldsymbol{\mu}}_{t,K} - \boldsymbol{\mu}_t\|_2^2\right] &= \frac{A_t(q_{\phi_t})}{K} + o(K^{-1}) \\ &\le \frac{e^{D_\alpha(\pi_t\|q_{\phi_t})}}{K}\left(\mathbb{E}_{\pi_t}\left[\|\mathrm{d}\mathbf{B}_t - \boldsymbol{\mu}_t\|_2^{\frac{2(\alpha-1)}{\alpha-2}}\right]\right)^{\frac{\alpha-2}{\alpha-1}} + o(K^{-1}), \qquad \alpha > 2. \end{aligned} \tag{70}$$

Bounded ρ_t makes the sample-average weight concentrate exponentially around one. Where this average is at least $1/2$, a ratio expansion and Cauchy–Schwarz give an $O(K^{-3/2})$ MSE remainder.

Elsewhere, the convex-combination form of the estimator and finite fourth moments give an $o(K^{-1})$ contribution.

Since $e^{D_2(\pi_t\|q_{\phi_t})} = 1 + \chi^2(\pi_t\|q_{\phi_t})$, the Hölder endpoint at $\alpha = 2$ recovers $A_t(q_{\phi_t}) \le B^2[1 + \chi^2(\pi_t\|q_{\phi_t})]$ for a centered increment bounded by B (Agapiou et al., 2017; Chatterjee & Diaconis, 2018). Eq. 69 extends this relation to unbounded increments through finite higher-order moments.

For fixed π_t and α, the moment factor in Eq. 70 is independent of q_{ϕ_t}, so a smaller $D_\alpha(\pi_t\|q_{\phi_t})$ tightens the leading term of the MSE bound. This motivates fitting q_{ϕ_t} to evaluator-informed anchors in Appendix D.3, so that previously observed high-weight perturbations guide where new samples are drawn.

F.6 Derivation of Differentiable Guidance

In this section, Eq. 10 is derived under the same optimal-control formulation, and Eq. 11 is obtained as its differentiable instantiation. Suppose the terminal objective is induced by a design target $\mathbf{y}$ through

$$L_{\mathbf{y}}(\mathbf{z}_T) = -\log p(\mathbf{y} \mid \mathbf{z}_T) + C, \tag{71}$$

where C is independent of $\mathbf{z}_T$. Substituting this cost into the desirability term in Eq. 6 gives

$$h_{\mathbf{y}}(\mathbf{z}_t, t) = \mathbb{E}_{\mathcal{P}_0}\left[\exp[-L_{\mathbf{y}}(\mathbf{z}_T)] \,|\, \mathbf{z}_t\right] = e^{-C} \int p(\mathbf{y} \mid \mathbf{z}_T) p(\mathbf{z}_T \mid \mathbf{z}_t)\, \mathrm{d}\mathbf{z}_T. \tag{72}$$

The integral is exactly the noisy-state marginal likelihood $p_t(\mathbf{y} \mid \mathbf{z}_t)$ up to the constant e^{-C}. Since this constant vanishes under $\nabla_{\mathbf{z}_t} \log$, Eq. 6 reduces to

$$\mathbf{c}_t^* = g(t) \nabla_{\mathbf{z}_t} \log p_t(\mathbf{y} \mid \mathbf{z}_t), \tag{73}$$

which gives Eq. 10.

The marginal likelihood $p_t(\mathbf{y} \mid \mathbf{z}_t)$ is generally not available in closed form. For a practical differentiable realization, we use the Tweedie clean estimator

$$\hat{\mathbf{z}}_T = \mathcal{T}_t(\mathbf{z}_t) \tag{74}$$

and approximate $p_t(\mathbf{y} \mid \mathbf{z}_t)$ by $p(\mathbf{y} \mid \hat{\mathbf{z}}_T)$. This plug-in approximation gives

$$\mathbf{c}_t = g(t) \nabla_{\mathbf{z}_t} \log p(\mathbf{y} \mid \hat{\mathbf{z}}_T), \tag{75}$$

which is Eq. 11. Equivalently, if a differentiable evaluator is written as $v(\hat{\mathbf{z}}_T) = -\log p(\mathbf{y} \mid \hat{\mathbf{z}}_T) + \mathrm{const}$, the same control can be written as $-g(t)\nabla_{\mathbf{z}_t} v(\hat{\mathbf{z}}_T)$.

F.7 Derivation of Eq. 12

In this section, we provide a formal derivation of the hierarchy-preserving control allocation used in Eq. 12. Recall that, in the mixed-guidance setting, both the foundation control and the refinement control have been formulated as control inputs acting on the same diffusion dynamics. At each denoising step t, we denote the foundation control by $c_t^f \in \mathbb{R}^n$ and the refinement control by $c_t^r \in \mathbb{R}^n$. The goal is to construct a final control $c_t \in \mathbb{R}^n$ by allocating these two control inputs in a way that respects their hierarchy:

$$c_t = \mathcal{A}(c_t^f, c_t^r), \tag{76}$$

where $\mathcal{A}(\cdot, \cdot)$ denotes the control allocation rule.

A naive allocation is to directly sum the two controls, i.e., $c_t = c_t^f + c_t^r$. However, this flat composition ignores the hierarchy structure between the two controls. In our setting, the foundation control c_t^f enforces the foundation stability-related objective, while the refinement control c_t^r further optimizes the target property. If c_t^r contains a component that conflicts with the direction of c_t^f, direct summation may alter or weaken the foundation control. We therefore formulate the allocation as a task-hierarchy control problem: the final control needs to first preserve c_t^f exactly, and only then incorporate c_t^r within the remaining degrees of freedom.

To formalize the foundation stability constraint, we require the component of the final control c_t along the foundation direction c_t^f to be identical to that of c_t^f itself. This requirement can be written as a linear constraint:

$$A_t c_t = b_t, \qquad A_t = (c_t^f)^\top, \qquad b_t = \|c_t^f\|_2^2. \tag{77}$$

Indeed, $A_t c_t = (c_t^f)^\top c_t$ measures the strength of c_t along the foundation direction, while $b_t = (c_t^f)^\top c_t^f$ is the corresponding strength of the foundation control itself. Thus, any feasible c_t satisfying this constraint preserves the foundation component exactly.

A particular solution to this foundation constraint is

$$c_t^{(0)} = A_t^\dagger b_t = c_t^f, \tag{78}$$

where $A_t^\dagger$ denotes the Moore–Penrose pseudoinverse of A_t. This follows from

$$A_t c_t^{(0)} = (c_t^f)^\top c_t^f = \|c_t^f\|_2^2 = b_t. \tag{79}$$

Therefore, $c_t^{(0)}$ already satisfies the foundation control requirement. If only the foundation task were considered, the final control would simply be c_t^f. The remaining question is how to incorporate the refinement control without violating this constraint.

The general solution to the linear constraint $A_t c_t = b_t$ can be written as

$$c_t = c_t^{(0)} + N_t z_t, \qquad A_t N_t = 0, \tag{80}$$

where N_t is a basis matrix for the null space of A_t, and z_t is a free coefficient vector. Since

$$\mathcal{N}(A_t) = \left\{ x \in \mathbb{R}^n : (c_t^f)^\top x = 0 \right\}, \tag{81}$$

the null space consists exactly of directions orthogonal to c_t^f. Consequently, adding any null-space component $N_t z_t$ does not affect the foundation constraint:

$$A_t(c_t^{(0)} + N_t z_t) = A_t c_t^{(0)} + A_t N_t z_t = b_t. \tag{82}$$

Thus, the refinement control can only be incorporated through the degrees of freedom lying in the null space of the foundation stability constraint.

We then use these remaining degrees of freedom to approximate the refinement control as closely as possible. Specifically, among all controls that satisfy the foundation constraint, we choose the one closest to c_t^r in the least-squares sense:

$$z_t^* = \arg\min_{z_t} \left\| c_t^{(0)} + N_t z_t - c_t^r \right\|_2^2. \tag{83}$$

This is a standard linear least-squares problem with solution

$$z_t^* = N_t^\dagger \left(c_t^r - c_t^{(0)} \right). \tag{84}$$

Substituting this solution back gives the allocated control

$$c_t^* = c_t^{(0)} + N_t N_t^\dagger \left(c_t^r - c_t^{(0)} \right). \tag{85}$$

This expression shows that the foundation control is preserved by $c_t^{(0)} = c_t^f$, while the refinement task is only pursued through its projection onto the null space of the foundation constraint.

In our case, the foundation constraint is one-dimensional because $A_t = (c_t^f)^\top$ has rank one when $c_t^f \neq 0$. Therefore, $N_t N_t^\dagger$ is the orthogonal projector onto the null space of A_t, equivalently the orthogonal complement of c_t^f:

$$N_t N_t^\dagger = P_{c_t^f}^\perp = I - \frac{c_t^f (c_t^f)^\top}{\|c_t^f\|_2^2}. \tag{86}$$

The orthogonal projection above requires dividing by $\|\mathbf{c}_t^f\|_2^2$, which can become numerically unstable when the foundational control is near zero. In our implementation, we replace the denominator with $\|\mathbf{c}_t^f\|_2^2 + \delta_{\text{proj}}$ for a small constant $\delta_{\text{proj}} > 0$.

G External Assets and Licenses

Datasets. We use the GEOM dataset (Axelrod & Gomez-Bombarelli, 2022) under the CC0 1.0 Universal license.[1] We use the QM9 dataset (Ramakrishnan et al., 2014) under the CC0 1.0 Universal license.[2]

Software and Evaluators. We use GFN2-xTB (Bannwarth et al., 2019) under the GNU Lesser General Public License v3.0 or later (LGPL-3.0-or-later).[3] We use g-xTB (Froitzheim et al., 2025) under the GNU General Public License v3.0 (GPL-3.0).[4]

Pretrained Models and Checkpoints. We use the pretrained GeoLDM model (Xu et al., 2023) under the MIT License.[5] We use the pretrained EDM model (Hoogeboom et al., 2022) under the MIT License.[6] We use the pretrained AIMNet2 model (Anstine et al., 2025) under the MIT License.[7] We use the pretrained eSEN-sm-conserving model (Levine et al., 2025) under the FAIR Chemistry License.[8] We use the pretrained property regressors publicly released by ChemGuide (Shen et al., 2025).[9]

Baseline Implementations. We use the ChemGuide implementation (Shen et al., 2025) under the MIT License.[10]

[1] https://doi.org/10.7910/DVN/JNGTDF
[2] https://doi.org/10.6084/m9.figshare.978904
[3] https://github.com/grimme-lab/xtb
[4] https://github.com/grimme-lab/g-xtb
[5] https://github.com/MinkaiXu/GeoLDM
[6] https://github.com/ehoogeboom/e3_diffusion_for_molecules
[7] https://huggingface.co/isayevlab/aimnet2-2025
[8] https://huggingface.co/facebook/OMol25
[9] https://drive.google.com/file/d/1A-CtbcSLLBou6EVajuqI7R_3U86oKvkd/view
[10] https://github.com/A-Chicharito-S/ChemGuide