CONFORMERDIFF-ADAPTIVE: A SAMPLER-AGNOSTIC WRAPPER FOR PER-MOLECULE NFE ALLOCATION IN CONFORMER GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

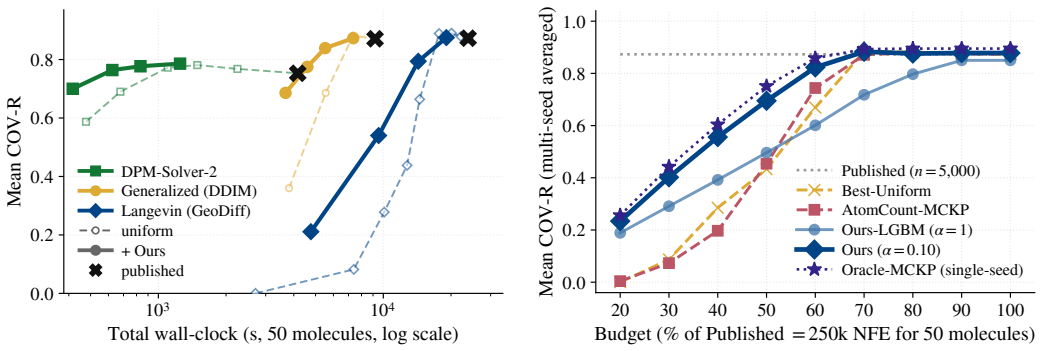
Sampling from diffusion-based molecular conformer generators is expensive: models such as GEODIFF use thousands of denoising steps for *every* molecule, irrespective of conformational complexity. We propose ConformerDiff-Adaptive, a backbone-frozen wrapper that allocates a global number-of-function-evaluations (NFE) budget heterogeneously across a batch of molecules to maximize aggregate coverage. The wrapper has three complementary components: an exact multi-choice knapsack allocator (which beats a greedy heuristic by 19.7 pp on average); a calibrated monotone heavy-atom-count prior that carries the dominant in-distribution signal; and a low-weight LIGHTGBM residual on cheap RDKIT features for molecule-specific structure beyond size. On GEODIFF's published test_data_1k benchmark with 3-seed averaged ground truth, our wrapper outperforms the strongest baseline by $+23.0$ **to** $+35.9$ **pp** absolute COV-R across the compute-constrained regime (20–50% of the published NFE budget, all $p < 0.001$ paired bootstrap). At 80% budget and above the advantage disappears: allocation degenerates to a uniform schedule, and we show this is intrinsic rather than a limitation of our predictor. The wrapper is sampler-agnostic across three substantially different samplers – Langevin dynamics, DDIM-style generalized, and DPM-Solver-2 (Heun) – and backbone-agnostic across GEODIFF and AGDIFF. The pipeline modifies no backbone parameters and requires ~ 3 GPU-hours of one-time per-deployment calibration. Code, trained predictors and calibration labels are available at `https://anonymous.4open.science/r/ConformerDiff-B5F0`.

1 INTRODUCTION

Sampling from diffusion-based molecular conformer generators is expensive: GEODIFF (Xu et al., 2022), the canonical SE(3)-equivariant model for this task, uses 5,000 denoising steps for *every* molecule. Subsequent work has improved the backbone (Wyzykowski et al., 2025; Liu et al., 2025b; Frank et al., 2025), the parameterization (Jing et al., 2022; Xu et al., 2023; Song et al., 2024; Wang et al., 2024; Hassan et al., 2024; Cao et al., 2025), and the distillation regime (Fan et al., 2024), but left one lever untouched: every molecule in a batch gets the same step count, regardless of conformational complexity. On the published GEOM-Drugs test split (Axelrod & Gómez-Bombarelli, 2022) we measure that the easiest decile of molecules saturates by NFE$= 3{,}000$ while the hardest reaches only COV-R≈ 0.45 at NFE$= 5{,}000$ – a two-fold spread in attainable coverage at identical compute, which a uniform schedule cannot exploit. Nor is this an artifact of slow samplers: few-step flow-matching models (Hassan et al., 2024; Cao et al., 2025; Liu et al., 2025b) lower the absolute NFE budget but not the cross-molecule difficulty variance.

Adaptive per-input computation is a recurring efficiency theme in machine learning – per-input depth (Graves, 2016; Banino et al., 2021), early exits (Teerapittayanon et al., 2016), speculative decoding (Leviathan et al., 2023; Chen et al., 2023) – and, in the diffusion domain, adaptive timesteps within a single image (Schusterbauer et al., 2026; Zhao et al., 2025), per-prompt step prediction (Li et al., 2024; Liu et al., 2025a; Anagnostidis et al., 2025), and training-free model swapping across timesteps (Pan et al., 2025). To our knowledge no prior work reallocates a shared budget *across* a batch of inputs in molecular conformer generation, where chemical-graph features make per-input difficulty directly observable.

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.



(a) Wall-clock vs. coverage. (b) Budget vs. coverage.

Figure 1: **Allocation buys coverage where compute is scarce and stops paying once it is ample.** a Measured uniform frontier (dashed) vs. ConformerDiff-Adaptive (solid) for three samplers; changing sampler alone (Langevin → generalized) buys $3.2\times$ less wall-clock at $+1.1$ pp. b On 50 held-out molecules, ConformerDiff-Adaptive leads below 70% budget; above it the curves converge and differences are not significant. Oracle-MCKP is single-seed and noise-inflated – a diagnostic, not an attainable bound.

We propose ConformerDiff-Adaptive: a backbone-frozen inference-time wrapper that, given a global NFE budget B across a batch of N molecules, assigns each molecule a per-instance $n_i \in \mathcal{G}$ with $\sum_i n_i \leq B$ so as to maximize aggregate coverage. It has three components: an exact multi-choice knapsack dynamic program (Kellerer et al., 2004) for the joint allocation; a calibrated isotonic atom-count prior, fit on $\sim$150 in-domain molecules, carrying the dominant in-distribution signal; and a LIGHTGBM residual on cheap cheminformatics features for out-of-distribution molecules. We train no backbone parameters; the wrapper's predictors are lightweight CPU models with sub-second overhead.

Contributions.

- **Heterogeneous NFE allocation.** The first backbone-frozen wrapper that reallocates a global NFE budget *across* a batch of molecules, as an exact multi-choice knapsack DP – beating a greedy heuristic by 19.7 pp on average.
- **A calibrated size prior does most of the work.** A monotone isotonic atom-count prior fit on $\sim$150 molecules reaches 99.7% of the full method and, re-calibrated per deployment, carries the gain across domain shift; a low-weight LIGHTGBM residual matters only when target molecules fall outside the prior's support.
- **Large gains where compute is scarce.** $+23.0$ to $+35.9$ pp COV-R over the strongest baseline at 20–50% budget (all $p < 0.001$), across three samplers and two backbones.
- **When allocation stops paying.** A saturation criterion for the regime boundary, plus evidence that single-seed evaluation manufactures apparent gains above it.

The result splits into two regimes. Below roughly 70% of the published budget, allocation buys a large amount of coverage; above it the coverage curves have flattened, and every allocation the knapsack can express is within noise of a uniform schedule. We give the saturation criterion that separates the two, and show that the per-molecule headroom that single-seed evaluation appears to find in the saturated regime is a measurement artefact. This null result is a property of the problem, not of our predictor: once most molecules sit at or above their own saturation rung there is nothing left to trade, and the knapsack correctly returns something close to uniform. Code, trained predictors and calibration data are released anonymously (Appendix P).

2 RELATED WORK

Diffusion-based molecular conformer generation. GEODIFF (Xu et al., 2022) established the SE(3)-equivariant Markov-chain diffusion paradigm for the 3D conformer task. Subsequent work has refined the geometric prior (Torsional Diffusion (Jing et al., 2022) diffuses only over torsion

angles; Shifting Scores (Zhou et al., 2023) re-targets the prior toward Maxwell-Boltzmann), the backbone architecture (AGDIFF (Wyzykowski et al., 2025) adds attention; NExT-Mol (Liu et al., 2025b) couples a relational diffusion transformer with a 1D language model; DiTMC (Frank et al., 2025) applies a diffusion transformer directly), the generative paradigm (GeoLDM (Xu et al., 2023) uses latent diffusion; GeoBFN (Song et al., 2024) uses Bayesian flow networks; MCF (Wang et al., 2024) parameterizes conformers as continuous functions; Equivariant Blurring Diffusion (Park & Shen, 2024) adopts a hierarchical denoising prior), the flow-matching family (ET-Flow (Hassan et al., 2024), SO(3)-Averaged Flow Matching (Cao et al., 2025), ConfFlow (Shah & Koltun, 2024)), and the distillation regime (EC-Conf (Fan et al., 2024), via equivariant consistency). ConformerDiff-Adaptive is *orthogonal* to all of the above: we do not propose a new generator; we wrap any compatible frozen backbone with a per-molecule-chosen NFE. We empirically verify orthogonality with GEODIFF and AGDIFF. The single closest chemistry-side budget-reallocation work is the Flow-Matching Refiner (Xu & Gao, 2025), which reallocates budget between a generator and a refiner stage (across pipeline stages) rather than across molecules.

Accelerated diffusion sampling. A second line reduces *per-sample* sampling cost. DDIM (Song et al., 2021) introduced non-Markovian deterministic samplers; the DPM-Solver family (Lu et al., 2022; 2025) provides higher-order ODE solvers (DPM-Solver-1/2/3, DPM-Solver++); EDM (Karras et al., 2022) unifies the design space; consistency models (Song et al., 2023) enable one-step generation via distillation; progressive distillation (Salimans & Ho, 2022) halves NFE iteratively. Recent extensions include instance-aware timestep scheduling (Yuan et al., 2026), optimal stepsize search (Pei et al., 2025), hierarchical schedule optimization (Zhu et al., 2025), rollover budget forcing (Kim et al., 2025), adaptive flow-path samplers (Jin et al., 2026), and adaptive stochastic coefficients (Wang et al., 2025). These methods cut *per-sample* cost – some even adapt the schedule per individual sample – but none reallocates a *shared* budget *across* a batch of molecules under a joint constraint. Our wrapper *stacks* with these accelerators: we empirically combine ConformerDiff-Adaptive with DDIM-style generalized sampling and with DPM-Solver-2 (Section 5.3).

Adaptive computation and per-input compute allocation. General-ML adaptive computation has a long history. ACT (Graves, 2016) and PonderNet (Banino et al., 2021) let recurrent models choose their own depth per input; BranchyNet (Teerapittayanon et al., 2016) adds early-exit branches; speculative decoding (Leviathan et al., 2023; Chen et al., 2023) short-circuits autoregressive LLMs. Scardapane et al. (2024) survey the broader conditional-computation literature. In the diffusion domain, recent vision work explores adaptive compute *within* a single sample: BudgetFusion (Li et al., 2024) predicts an optimal NFE *per text prompt* in Stable Diffusion; SpeCa (Liu et al., 2025a) caches features speculatively in image/video DiTs; FlexiDiT (Anagnostidis et al., 2025) makes one DiT support multiple per-sample budgets; Patch Forcing (Schusterbauer et al., 2026) adapts compute per spatial patch within one image; DyDiT (Zhao et al., 2025) routes compute timestep-wise and spatially; T-Stitch (Pan et al., 2025) swaps model size across timesteps in a training-free wrapper. ConformerDiff-Adaptive differs along three axes: (i) we operate in the molecular conformer domain rather than vision; (ii) we allocate compute *across* a batch of inputs under a shared budget, not within a single sample; and (iii) we use a single joint optimization (multi-choice knapsack DP) rather than independent per-input prediction. Joint MCKP optimization extracts shared-budget gains that independent per-input prediction cannot: only a joint allocation can move NFE from an easy molecule to a hard one (Kellerer et al., 2004).

3 PRELIMINARIES

3.1 CONFORMER GENERATION AND THE COVERAGE METRIC

Let $\mathcal{M}$ be the space of molecular graphs and $\mathbb{R}^{|m|\times 3}$ the space of 3D conformations of a molecule $m \in \mathcal{M}$ with $|m|$ heavy atoms. A diffusion-based conformer generator g maps a molecular graph m and a sampling-step count $n \in \mathbb{N}$ to k generated conformations, each in $\mathbb{R}^{|m|\times 3}$. We use the GEOM-Drugs convention (Axelrod & Gómez-Bombarelli, 2022): $k = 2 \cdot |\text{refs}(m)|$ generated conformations per molecule, where $|\text{refs}(m)|$ is the number of ground-truth reference conformations available for m.

The standard quality metric is **coverage of references** (COV-R). For a molecule m with $K = |\text{refs}(m)|$ references $\{C_j^{\text{ref}}\}_{j=1}^{K}$ and $L = 2K$ generated conformers $\{C_l^{\text{gen}}\}_{l=1}^{L}$:

$$\text{COV-R}(m, n) = \frac{1}{K}\Big|\{\, j \in \{1, \ldots, K\} : \min_{l} \text{RMSD}(C_j^{\text{ref}}, C_l^{\text{gen}}) \le \delta \,\}\Big| \tag{1}$$

where RMSD is best-RMS-alignment after hydrogen removal, and $\delta = 1.25$ Å on GEOM-Drugs (the published convention; $\delta = 0.5$ Å on GEOM-QM9). COV-R$(m, n) \in [0, 1]$, higher is better; we optimize for COV-R, the published primary metric.

3.2 Number of Function Evaluations

For a fixed backbone g and sampler, we define **NFE** as the total number of model forward passes used to produce one molecule's conformer ensemble. For Langevin dynamics (LD, GeoDiff's default), NFE equals the number of denoising iterations n. For DPM-Solver-2 (Heun's method), each macro-step uses 2 forward passes and so NFE $= 2 \cdot n_{\text{outer}}$. NFE is the operational quantity that determines wall-clock cost, which is proportional to NFE to leading order (per-pass cost varies only mildly with molecule size).

3.3 Three Samplers We Wrap

We test the wrapper on three substantively different sampling regimes spanning the contemporary diffusion-sampler design space: Langevin dynamics (LD, the published GeoDiff default; stochastic, slow per step), the DDIM-style generalized sampler (Song et al., 2021) (deterministic skeleton with optional noise; faster per step), and DPM-Solver-2 (Lu et al., 2022) (a deterministic second-order ODE solver; far fewer steps required). Full update equations and schedule definitions for each are in Appendix B. Their per-step cost and saturation behaviour differ substantially, which is why a sampler-invariant wrapper is a non-trivial property rather than a given.

3.4 Problem: Heterogeneous NFE Allocation

Given a batch of N molecules $\{m_i\}_{i=1}^{N}$, a finite NFE grid $\mathcal{G} = \{n^{(1)} < n^{(2)} < \cdots < n^{(G)}\}$ with $G = |\mathcal{G}|$ grid points, and a global NFE budget $B \le N \cdot n^{(G)}$, the *heterogeneous NFE allocation problem* is:

$$\max_{\{n_i\} \in \mathcal{G}^N} \frac{1}{N} \sum_{i=1}^{N} \text{COV-R}(m_i, n_i) \qquad \text{s.t.} \sum_{i=1}^{N} n_i \le B. \tag{2}$$

Eq. (2) is a multi-choice knapsack problem (Kellerer et al., 2004). We need a surrogate for the unknown COV-R(m_i, n_i), which Section 4 addresses.

4 Method: ConformerDiff-Adaptive

The heterogeneous-allocation problem of Eq. (2) is a multi-choice knapsack instance in which the value of choosing $n_i \in \mathcal{G}$ for molecule m_i is the unknown coverage COV-R(m_i, n_i). The wrapper estimates that value from cheap cheminformatics features and solves the resulting knapsack exactly.

Figure 2 shows the headroom this exploits: measured coverage curves differ sharply across molecules, both in the step count at which they saturate and in the ceiling they reach, so any single uniform step count is simultaneously wasteful for some molecules and insufficient for others. We organize the wrapper into three deployment phases: a one-time predictor trained on a public distribution, a per-deployment calibration that aligns the predictor to the deployer's molecule distribution, and an inference-time allocator that runs sub-second on CPU.

4.1 The Coverage Predictor: Why Two Estimators

The wrapper needs a numerical proxy for COV-R(m, n) given only chemical-graph information about m. We decompose this proxy into two estimators with complementary roles, because two

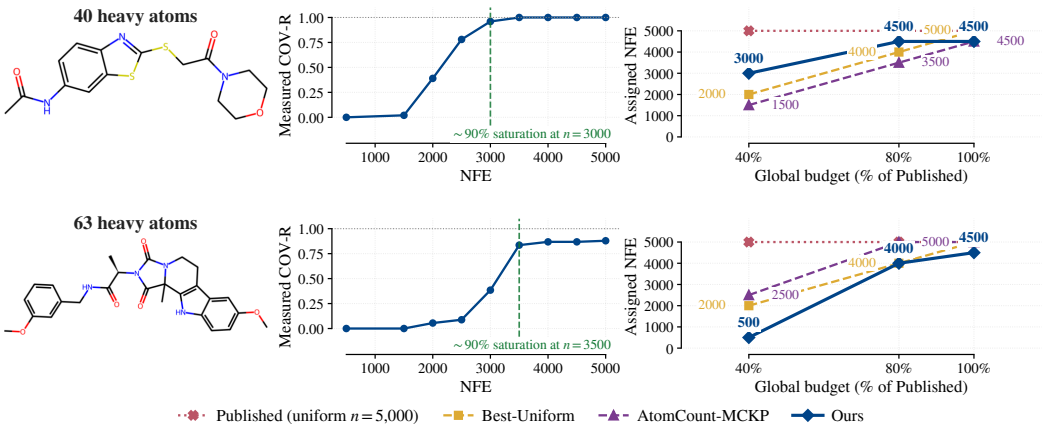


Figure 2: **Per-molecule case studies: two examples.** *Left:* 2D structure and heavy-atom count. *Middle:* measured coverage vs. NFE, ∼90% saturation marked. *Right:* NFE assigned by each method as the budget varies. The 40-atom molecule saturates by $n \approx 3{,}000$ and is handed well under n=5,000; the 63-atom one reaches only 0.74 at n=5,000 and is dropped to the floor rung n=500 when the budget is tightest – neither move is available to a uniform schedule. All six: Appendix L.

distinct signals live in the data and combining them in a single regressor mixes very different sources of variance.

The dominant signal is low-resolution and population-level: at any fixed n, mean coverage is a near-monotone function of molecule size (approximated here by heavy-atom count). On a small per-deployment calibration set this signal is recoverable at no flexibility cost, and our ablations (Section 5.4) show it carries the bulk of the in-distribution performance. For each $n \in \mathcal{G}$ we fit a one-dimensional isotonic regression

$$\mathrm{iso}_n : \ -\mathrm{heavy_atoms}(m) \longmapsto \bar{c}(m, n), \tag{3}$$

where $\bar{c}(m, n)$ is the multi-seed averaged measured coverage on the calibration set. The negation makes iso_n monotone-decreasing in molecule size. The collection $\{\mathrm{iso}_n : n \in \mathcal{G}\}$ – a few kilobytes total – captures the size-vs-coverage curve at every NFE operating point.

The second signal is high-resolution and molecule-specific: distinct substructures (flexible rings, long aliphatic chains, multiple chiral centers) drive coverage variance even between two molecules of identical heavy-atom count. A LightGBM (Ke et al., 2017) regressor φ trained on a large public distribution captures this residual signal, which becomes essential when molecules fall outside the size range the prior was fit on (Section 5.6). We train φ on GeoDiff's 1000 validation molecules at the 9-point NFE grid $\mathcal{G} = \{500, 1500, 2000, 2500, 3000, 3500, 4000, 4500, 5000\}$ from cheminformatics features computed in ≤ 2 ms per molecule on CPU; the full feature list, training hyperparameters, and runtime budget are deferred to Appendix F. Its labels are quantized – GeoDiff val has only 5 reference conformers per molecule, so COV-R$(m, n) \in \{0, 0.2, \ldots, 1.0\}$ – which is precisely why it under-exploits the smooth size-vs-coverage trend and is best used as a residual on top of the calibrated prior rather than as the primary estimator.

The two estimators are combined by convex weighting:

$$\hat{c}(m, n) \ = \ \alpha\, \varphi(m, n) \ + \ (1 - \alpha)\, \mathrm{iso}_n\big(-\mathrm{heavy_atoms}(m)\big), \tag{4}$$

clipped to $[0, 1]$ and enforced monotone non-decreasing in n per molecule by isotonic projection along the NFE axis. The single hyperparameter α is selected by cross-validation on the calibration set. We use $\alpha = 0.10$ throughout; performance is flat between $\alpha = 0.05$ and $\alpha = 0.20$ (Appendix K). Section 5.4 confirms that on in-distribution evaluation the isotonic prior alone recovers 99.7% of the full wrapper's performance, while Section 5.6 shows that under domain shift re-calibrating this prior restores the gain, and that the LightGBM residual matters only when target molecules fall outside the prior's support.

4.2 Exact Allocation via Multi-Choice Knapsack DP

Substituting $\hat{c}$ for the unknown COV-R in Eq. (2) yields a multi-choice knapsack problem (Kellerer et al., 2004): each molecule must pick exactly one element of $\mathcal{G}$, and the summed costs must respect B. Because $\mathcal{G}$ is a finite integer set with small $\gcd$ ($s = 500$ for our grid), the standard knapsack DP runs in tabular form. Let $g_k = n^{(k)}/s$ and $B' = \lfloor B/s \rfloor$, and define

$$f_i(b) = \max_{k\,:\,g_k \le b} \Big[\, f_{i-1}(b - g_k) + \hat{c}(m_i, n^{(k)}) \,\Big], \qquad f_0(0) = 0, \quad f_0(b) = -\infty \ (b > 0). \tag{5}$$

The optimum is $\max_{b \le B'} f_N(b)$, recovered with $O(N)$ choice backtracking. The time complexity is $O(N\,|\mathcal{G}|\,B')$; for $N \le 200$, $|\mathcal{G}| = 9$, and $B' \le 2{,}000$ this is at most 3.6×10^6 operations – well under one second on CPU.

The DP's output is easiest to read through the Lagrangian relaxation of Eq. (2): pricing NFE at $\lambda \ge 0$ decouples it across molecules,

$$n_i(\lambda) \;=\; \arg\max_{n \in \mathcal{G}} \,\big[\, \hat{c}(m_i, n) - \lambda\, n \,\big], \tag{6}$$

with λ set by the budget. A tight budget means a high price, so a molecule whose predicted gain over the floor rung is below λ per NFE at every rung is sent to the floor (the triage of Section 5.2). As B grows, $\lambda \to 0$ and each molecule settles at the smallest rung attaining its predicted plateau, which is coverage-equivalent to a uniform schedule once most have saturated (Section 5.5).

The exact DP is not interchangeable with cheaper alternatives in this setting. A marginal-gain greedy heuristic, which ranks molecules by predicted coverage gain per unit NFE, fails on flat segments of the predicted curve: when the predicted curve is flat from $n^{(k)}$ to $n^{(k+1)}$ but rises again at $n^{(k+2)}$ (a delayed jump), the greedy marginal gain at step $k{+}1$ is zero, so the molecule is never advanced to the rung where its coverage actually increases. The exact DP is robust to flat segments. A different alternative – adopted by BudgetFusion (Li et al., 2024) in text-to-image – predicts an optimal NFE *independently* for each input, without coupling under a shared budget. Independent prediction cannot move NFE from an easy molecule to a hard one, so it is dominated by Eq. (5) – strictly so whenever the budget forces a trade-off. Section 5.4 measures the empirical cost of both substitutions.

4.3 Sampler- and Backbone-Agnostic Interface

The wrapper observes only $(m, n, \text{COV-R}(m, n))$ triples and outputs a per-molecule NFE assignment $\{n_i\}$. The diffusion backbone – weights, noise schedule, sampler choice – is treated as a black box. To deploy on a new sampler, the only change needed is to re-run Phase B calibration with that sampler and refit the isotonic prior $\{\text{iso}_n\}$; the LightGBM predictor φ from Phase A can be reused unchanged because the prior dominates the convex blend at $\alpha = 0.10$. The same procedure applies to a new backbone (e.g., AGDIFF instead of GeoDiff). The empirical verification is in Section 5.3 and Appendix D; the per-sampler update equations are in Appendix B.

5 Experiments

5.1 Setup

We evaluate on the GEOM-Drugs and GEOM-QM9 benchmarks (Axelrod & Gómez-Bombarelli, 2022). The main backbone is the released GeoDiff (Xu et al., 2022) checkpoint at its published Langevin-dynamics sampler configuration; we additionally evaluate AGDIFF (Wyzykowski et al., 2025) for backbone orthogonality. Coverage is reported as mean COV-R at $\delta = 1.25$ Å on Drugs (0.5 Å on QM9), the published convention. Each (m, n) sampling cell averages 3 independent seeds (standard for stochastic generators (Xu et al., 2022; Karras et al., 2022)); single-seed numbers appear in Appendix J for transparency.

We compare against two budget-respecting baselines: **Best-Uniform**, the single best uniform NFE that fits the budget, and **AtomCount-MCKP**, a per-NFE Gaussian-process regression on heavy-atom count fed into the same knapsack allocator. **Published** (uniform $n = 5{,}000$) is a full-budget reference, feasible only at 100% and drawn as a constant line elsewhere. **Oracle-MCKP** runs the same allocator on single-seed measured labels; it is a noise-inflated diagnostic, not an attainable bound. Method

Table 1: **Main result on 50 held-out molecules.** Mean COV-R on the 9-point NFE grid, 3-seed ground truth; prior fit on the 150 calibration molecules. Δ is Ours minus the column-labelled baseline (2000 paired-bootstrap resamples, unrounded, so it may differ 0.1 pp from the rounded cells). ✓✓ $= p < 0.001$; ✓ $= p < 0.05$; NS = not significant. The rule after 60% is the saturation boundary: below it allocation pays, above it all methods are within noise of the best uniform schedule. Oracle-MCKP uses single-seed labels and is noise-inflated – a diagnostic, not an attainable bound.

Budget	Published	Best-Uniform	AtomCount-MCKP	Ours-LGBM	**Ours**	Δ vs AC	Δ vs Pub	Oracle (1-seed)
20% (50k)	0.873	0.000	0.004	0.189	**0.234**	**+23.0 ✓✓**	–	0.256
30% (75k)	0.873	0.087	0.073	0.291	**0.402**	**+32.9 ✓✓**	–	0.442
40% (100k)	0.873	0.285	0.197	0.392	**0.556**	**+35.9 ✓✓**	–	0.604
50% (125k)	0.873	0.433	0.454	0.496	**0.695**	**+24.2 ✓✓**	–	0.751
60% (150k)	0.873	0.669	0.744	0.601	**0.823**	**+7.9 ✓**	–	0.858
70% (175k)	0.873	0.884	0.871	0.718	0.883	+1.2	+1.0	0.893
80% (200k)	0.873	0.882	0.879	0.797	0.876	−0.3 (NS)	+0.3 (NS)	0.895
90% (225k)	0.873	0.878	0.874	0.850	0.878	+0.4 (NS)	+0.6	0.895
100% (250k)	0.873	0.873	0.874	0.850	0.878	+0.4 (NS)	+0.5	0.895

variants are **Ours-LGBM** (α=1), **Ours-Prior** (α=0) and **Ours** (α=0.10). The 200-molecule test set is split into 150 calibration molecules (to fit the prior) and 50 held-out molecules (evaluation only); bootstrap 95% CIs use 2000 resamples. Our pipeline reproduces the published GEODIFF coverage (0.892 vs. 0.8913; Appendix C). Full definitions and hyperparameters are in Appendix A–F.

5.2 MAIN RESULT: CONFORMERDIFF-ADAPTIVE ON GEODIFF

Table 1 gives the result on the 50 held-out molecules with 3-seed ground truth, plotted in Figure 1b. It splits into two regimes. *Below* 70% *budget, allocation wins by a wide margin*: uniform baselines cannot give any single molecule enough NFE to cross its coverage threshold while respecting the global cap. At 40% budget, where AtomCount-MCKP barely passes 0.20 and Best-Uniform stalls at 0.29, Ours already reaches 0.56 – $+35.9$ **pp** over AtomCount-MCKP, the strongest adaptive baseline ($\Delta = +0.357$, 95% CI $[+0.255, +0.457]$, $p < 0.001$). The mechanism is triage rather than concentration. Because the allocator maximizes total coverage under a global cap, it buys the cheapest coverage first: it funds each molecule whose curve turns up within reach of the budget to at or just below its own saturation rung, and drops the molecules whose coverage stays flat until $n \geq 3{,}500$ to the grid's floor rung n=500. At 40% budget this floors 71 **of the** 200 benchmark molecules; measured COV-R at n=500 is exactly zero, so those molecules consume 8.9% of the budget and return nothing. The decile with the lowest attainable coverage receives 2.5% of the budget, against 9–16% for the highest-attainable decile. No uniform schedule can express this trade; Figure 2 shows it molecule by molecule (all six cases in Appendix L).

Above 70% *budget, the advantage disappears.* At 80% Ours reaches 0.876 against 0.882 for Best-Uniform and 0.879 for AtomCount-MCKP; none of the three pairwise differences is significant ($p = 0.269, 0.951, 0.766$). This is not a failure of the predictor. Once the budget is large enough that most molecules can be placed at or above their own saturation rung, every allocation the knapsack can express is within measurement noise of a uniform schedule, and the knapsack correctly returns something close to one. The apparent per-molecule headroom that a single-seed evaluation shows in this regime is an artefact of selecting on noise, which we quantify in Section 5.5 and Appendix E.

5.3 SAMPLER- AND BACKBONE-AGNOSTIC STACKABILITY

Table 2 applies ConformerDiff-Adaptive to three substantively different samplers, each shown below and above its own coverage plateau (per-sampler bars in Figure 3). The pattern is identical: large gains below the plateau and none above it, with the plateau at a different absolute budget for each sampler – so the gain is governed by where a sampler sits on its own coverage curve, not by which sampler it is. The samplers' wall-clock tiers (Figure 1a) come from per-step acceleration, not allocation (the sampler swap alone buys $3.2\times$ less wall-clock); the wrapper adds its gain on top of whichever tier is chosen, at $\sim$0.5 s of CPU overhead per 50-molecule batch against $\sim 2 \times 10^4$ s of backbone sampling (Appendix N). The gain also transfers across backbones: deployed on AGDIFF with a

Table 2: **Sampler-agnostic stackability.** Each sampler is shown below its own coverage plateau, where allocation pays, and above it, where it does not; the plateau sits at a different absolute budget for each. All rows use a size prior only (φ unused, so the samplers are comparable), 5-fold cross-validated (prior fit on 40 molecules, applied to 10); DPM-Solver-2 also uses disjoint allocation and scoring seeds. Cal-Uniform is the best uniform schedule chosen on the same folds. Protocol differs from Table 1; cells are not comparable across tables. Δ uses unrounded values.

Sampler family	Budget	Published	Cal-Uniform	**Ours**	Δ
Langevin dynamics (GEODIFF)	40%	0.873	0.278	**0.540**	**+26.2**
	80%	0.873	0.885	0.875	−1.0
Generalized (DDIM-style)	50%	0.871	0.360	**0.775**	**+41.4**
	80%	0.871	0.882	0.873	−0.9
DPM-Solver-2 (Heun)	15%	0.750	0.691	**0.764**	**+7.2**
	30%	0.750	0.782	0.786	+0.4

Table 3: **Component ablation.** Mean COV-R averaged over the 9 budget levels on the 50 held-out molecules. Each block varies one component – the coverage predictor, the allocator, or the blend weight α – with the other two held fixed. Δ is against the full method. The full α sweep is in Appendix K.

Varying	Variant	# feat.	Avg COV-R	Δ
Predictor	**Ours** (blend, α=0.10)	1077	**0.692**	–
	Ours-Prior only (α=0)	–	0.690	−0.2
	Ours-LIGHTGBM only (α=1)	1077	0.576	**−11.6**
	Ours, 2-feature blend	2	0.692	−0.06
Allocator	Exact multi-choice knapsack DP	–	**0.692**	–
	Marginal-gain greedy	–	0.494	**−19.7**
Blend α	$\alpha = 0.00$ (prior only)	–	0.690	−0.2
	$\alpha = 0.05$–0.20 (flat region)	1077	**0.692**	–
	$\alpha = 0.50$	1077	0.685	−0.7
	$\alpha = 1.00$ (LIGHTGBM only)	1077	0.576	−11.6

cross-validated size prior (φ unused), the allocation gain agrees with GEODIFF to within 2.3 pp at every budget tested (Appendix D; that comparison uses the 4-point grid on 200 molecules, so its absolute gains are not comparable to Table 1).

5.4 COMPONENT ABLATIONS

Three component questions matter for the story. *Which predictor carries the signal?* The isotonic prior alone (Ours-Prior, α=0) reaches 99.7% of the full method's performance, while the LightGBM predictor alone (Ours-LGBM, α=1) loses 11.6 pp on average (Table 3). The calibrated per-NFE prior on heavy-atom count is the load-bearing component; the LightGBM adds a small residual when blended at $\alpha = 0.10$. *How much feature engineering is needed?* A 2-feature LightGBM using only heavy-atom count and rotatable-bond count loses 0.06 pp from the 1077-feature variant – essentially free, and evidence that the predictor's signal is dominated by molecule size and flexibility, not fingerprint detail. *Is the joint allocator essential?* Yes, and by a wide margin: a marginal-gain greedy heuristic costs 19.7 **pp** on average and 20–34 pp at moderate budgets, because greedy stalls on flat segments of the predicted coverage curve where the marginal gain per unit NFE is zero. The deficit itself traces the two regimes: it peaks at 34.3 pp at 60% budget and closes to 0.5 pp at full budget, where nothing is left to allocate.

5.5 WHEN DOES PER-MOLECULE ALLOCATION PAY?

The two regimes are separated by a property of the sampler, not of the allocator: allocation has something to trade only while a non-trivial fraction of molecules sit on the steep part of their coverage curve. Once the budget is large enough to place most molecules at or above their own saturation rung, the optimal allocation is close to uniform and the knapsack correctly returns it – which is why

Table 4: **Transfer to GEOM-QM9.** Budgets (of 9) at which the wrapper beats Best-Uniform by >1 / >3 pp, with the prior transferred from GEOM-Drugs or re-calibrated on the target set (5-fold, out-of-fold). Single-seed labels on the 4-point grid; Best-Uniform is chosen on the evaluation set itself, so values are not comparable to Table 1. No molecule is in φ's training pool.

QM9 set	Prior	α=0	α=0.10
99 molecules	from Drugs	0/9 / 0/9	2/9 / 2/9
	re-calibrated	4/9 / 4/9	4/9 / 4/9
398 molecules	from Drugs	0/9 / 0/9	5/9 / 2/9
	re-calibrated	7/9 / 6/9	7/9 / 6/9

Ours and Best-Uniform agree to within noise above 70% budget. This yields a practical criterion. Measure the calibration set's mean coverage curve; allocate per molecule while the budget places the median molecule below its own saturation rung, and fall back to the best uniform schedule above it. On our benchmark the crossover sits between 60% and 70% of the published budget, and the same criterion locates it for each of the three samplers in Table 2 at a different absolute budget. The same boundary explains why single-seed evaluation misleads here. Write a single-seed measurement as $\tilde{c}(m,n) = c(m,n) + \varepsilon_{m,n}$, with c the seed-averaged coverage and $\mathbb{E}[\varepsilon] = 0$. Because a maximum is convex, any allocator that selects on $\tilde{c}$ and is scored on the same $\tilde{c}$ is biased upward:

$$\mathbb{E}_{\varepsilon}\Big[\max_{\{n_i\}\in\mathcal{F}} \frac{1}{N}\sum_i \tilde{c}(m_i,n_i)\Big] \;\geq\; \max_{\{n_i\}\in\mathcal{F}} \frac{1}{N}\sum_i c(m_i,n_i), \tag{7}$$

where $\mathcal{F}$ is the feasible set of Eq. (2); the gap grows with the number of near-tied allocations, i.e. once real headroom is gone. Oracle-MCKP is this estimator – 1.0–1.9 pp above Ours at 70–100% budget (Table 1) – and the same effect produced the saturated-regime gain that an earlier single-seed version of this work reported (Appendix E).

5.6 Cross-Domain Generalization

Table 4 tests a true cross-domain shift to GEOM-QM9, whose molecules ($\leq$9 heavy atoms) fall below the size range the Drugs prior was fit on (10–69). Transferred unchanged, the prior collapses to a single curve shared by every molecule, so at α=0 the knapsack has no molecule-level signal and beats Best-Uniform by more than 1 pp at no budget; the LightGBM residual is then the only signal left (significant at 14 of 18 budgets, up to $+52.7$ pp over α=0). Re-calibrating the prior on the target set restores the gain at either blend weight, after which the residual changes coverage by at most $\pm$3 pp (significant at 1 of 18 budgets). Per-deployment re-calibration is therefore what carries the gain across domains; the residual is a safeguard for when that step is skipped.

6 Conclusion

We presented ConformerDiff-Adaptive, a backbone-frozen wrapper that allocates a global NFE budget heterogeneously across a batch of molecules for diffusion-based conformer generation, pairing an exact multi-choice knapsack DP allocator with a calibrated heavy-atom-count isotonic prior and a low-weight LightGBM residual. On the published GeoDiff benchmark with 3-seed averaged ground truth it outperforms the strongest baseline by $+23.0$ **to** $+35.9$ **pp** across the compute-constrained regime (20–50% of the published budget), and the same pattern holds across three samplers and two backbones. Above roughly 70% budget the advantage disappears, because the coverage curves have flattened and the optimal allocation is close to uniform; we give the saturation criterion that separates the two regimes, so a deployer can tell in advance which one they are in. Eq. (7) generalizes beyond this setting: any per-input allocator scored on the labels it selects on is biased upward, so such work should use disjoint seeds for allocation and scoring, or report multi-seed ground truth; with a single seed, a null result can be made to look positive.

Limitations. Without in-domain labels only the LightGBM residual applies, which is substantially weaker than the calibrated prior; we validated on the GeoDiff family only, leaving flow-matching backbones to future work; and the grid has no zero rung, so molecules the allocator gives up on still consume the floor rung's NFE; adding n=0 is a natural extension.

AI USE STATEMENT

Of the subtasks for which ICLR requires disclosure, we used generative AI tools only to help *interpret results*: language-model assistants were given read access to our experiment logs and asked to check that the numbers quoted in the manuscript matched the tables and logs they referred to, and to flag internal inconsistencies. Two such checks changed the paper: an earlier draft described the allocator's behaviour under a tight budget in the direction opposite to what the logs showed, and an earlier figure plotted an oracle allocation under our method's label. Both were verified against the logs and corrected by the authors.

We did *not* use generative AI tools to generate synthetic data, develop theoretical or conceptual frameworks, formulate mathematical claims, prove or assist in proving claims, propose or refine hypotheses, design or give feedback on research methodology or experiments, implement methods, clean or reformat datasets, or perform qualitative or thematic data analysis. All experiments, code and analysis in this paper were designed, implemented and run by the authors; no experimental result, table entry or figure was produced by a language model.

Additionally, among the subtasks for which disclosure is recommended, we used AI assistants to draft and edit prose, tighten captions, suggest paper structure, and help identify and format references. Every reference was verified by the authors against its arXiv or venue record, and every number in the manuscript was checked against our own pipeline output. We have reviewed all AI-assisted work and take responsibility for the final content of this work, including all text, claims and artifacts.

REPRODUCIBILITY STATEMENT

The frozen backbone, checkpoint, environment versions, hardware and sampler hyperparameters are specified in Appendix F; the NFE grid and its construction in Appendix G; the bootstrap and significance protocol in Appendix H; and the multi-seed averaging protocol in Appendix I. Baselines and the evaluation metric are defined in Appendix A. Because the saturated-regime results are sensitive to the seed protocol, we state for each table whether it uses 3-seed or single-seed ground truth, and report the single-seed comparison itself in Appendix J together with an analysis of why single-seed evaluation inflates it (Appendix E). Code, trained predictors, calibration labels and the bootstrap outputs needed to regenerate every table are released anonymously (Appendix P).

REFERENCES

Sotiris Anagnostidis, Gregor Bachmann, Yeongmin Kim, Jonas Kohler, Markos Georgopoulos, Artsiom Sanakoyeu, Yuming Du, Albert Pumarola, Ali Thabet, and Edgar Schönfeld. FlexiDiT: Your diffusion transformer can easily generate high-quality samples with less compute. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 28316–28326, 2025.

Simon Axelrod and Rafael Gómez-Bombarelli. GEOM, energy-annotated molecular conformations for property prediction and molecular generation. *Scientific Data*, 9(1):185, 2022. doi: 10.1038/s41597-022-01288-4.

Andrea Banino, Jan Balaguer, and Charles Blundell. PonderNet: Learning to ponder. In *8th ICML Workshop on Automated Machine Learning (AutoML)*, 2021.

Zhonglin Cao, Mario Geiger, Allan dos Santos Costa, Danny Reidenbach, Karsten Kreis, Tomas Geffner, Franco Pellegrini, Guoqing Zhou, and Emine Küçükbenli. Efficient molecular conformer generation with SO(3)-averaged flow matching and reflow. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 6522–6542, 2025.

Charlie Chen, Sebastian Borgeaud, Geoffrey Irving, Jean-Baptiste Lespiau, Laurent Sifre, and John Jumper. Accelerating large language model decoding with speculative sampling, 2023.

Zhiguang Fan, Yuedong Yang, Mingyuan Xu, and Hongming Chen. EC-Conf: A ultra-fast diffusion model for molecular conformation generation with equivariant consistency. *Journal of Cheminformatics*, 16(1):1–15, 2024. doi: 10.1186/s13321-024-00893-2.

J. Thorben Frank, Winfried Ripken, Gregor Lied, Klaus-Robert Müller, Oliver T. Unke, and Stefan Chmiela. Sampling 3D molecular conformers with diffusion transformers. In *Advances in Neural Information Processing Systems*, volume 38, 2025.

Alex Graves. Adaptive computation time for recurrent neural networks, 2016.

Majdi Hassan, Nikhil Shenoy, Jungyoon Lee, Hannes Stärk, Stephan Thaler, and Dominique Beaini. ET-flow: Equivariant flow-matching for molecular conformer generation. In *Advances in Neural Information Processing Systems*, volume 37, 2024.

Cheng Jin, Zhenyu Xiao, and Yuantao Gu. A-FloPS: Accelerating diffusion models via adaptive flow path sampler. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2026.

Bowen Jing, Gabriele Corso, Jeffrey Chang, Regina Barzilay, and Tommi S. Jaakkola. Torsional diffusion for molecular conformer generation. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. In *Advances in Neural Information Processing Systems*, volume 35, pp. 26565–26577, 2022.

Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. LightGBM: A highly efficient gradient boosting decision tree. In *Advances in Neural Information Processing Systems*, 2017.

Hans Kellerer, Ulrich Pferschy, and David Pisinger. *Knapsack Problems*. Springer, Berlin, Heidelberg, 2004. Chapter 11: The Multiple-Choice Knapsack Problem, pp. 317–347.

Jaihoon Kim, Taehoon Yoon, Jisung Hwang, and Minhyuk Sung. Inference-time scaling for flow models via stochastic generation and rollover budget forcing. In *Advances in Neural Information Processing Systems*, 2025.

Yaniv Leviathan, Matan Kalman, and Yossi Matias. Fast inference from transformers via speculative decoding. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 19274–19286. PMLR, 2023.

Qinchan Li, Kenneth Chen, Changyue Su, and Qi Sun. BudgetFusion: Perceptually-guided adaptive diffusion models, 2024.

Jiacheng Liu, Chang Zou, Yuanhuiyi Lyu, Fei Ren, Shaobo Wang, Kaixin Li, and Linfeng Zhang. SpeCa: Accelerating diffusion transformers with speculative feature caching. In *Proceedings of the ACM International Conference on Multimedia*, 2025a.

Zhiyuan Liu, Yanchen Luo, Han Huang, Enzhi Zhang, Sihang Li, Junfeng Fang, Yaorui Shi, Xiang Wang, Kenji Kawaguchi, and Tat-Seng Chua. NExT-Mol: 3D diffusion meets 1D language modeling for 3D molecule generation. In *International Conference on Learning Representations*, 2025b.

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

Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. DPM-Solver++: Fast solver for guided sampling of diffusion probabilistic models. *Machine Intelligence Research*, 22 (4):730–751, 2025. doi: 10.1007/s11633-025-1562-4.

Zizheng Pan, Bohan Zhuang, De-An Huang, Weili Nie, Zhiding Yu, Chaowei Xiao, Jianfei Cai, and Anima Anandkumar. T-Stitch: Accelerating sampling in pre-trained diffusion models with trajectory stitching. In *International Conference on Learning Representations*, 2025.

Jiwoong Park and Yang Shen. Equivariant blurring diffusion for hierarchical molecular conformer generation. In *Advances in Neural Information Processing Systems*, volume 37, 2024.

Jianning Pei, Han Hu, and Shuyang Gu. Optimal stepsize for diffusion sampling, 2025.

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

Simone Scardapane, Alessandro Baiocchi, Alessio Devoto, Valerio Marsocci, Pasquale Minervini, and Jary Pomponi. Conditional computation in neural networks: Principles and research trends. *Intelligenza Artificiale*, 18(1):175–190, 2024. doi: 10.3233/IA-240035.

Johannes Schusterbauer, Ming Gui, Yusong Li, Pingchuan Ma, Felix Krause, and Björn Ommer. Denoising, fast and slow: Difficulty-aware adaptive sampling for image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026.

Sohil Atul Shah and Vladlen Koltun. Conformation generation using transformer flows, 2024.

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

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

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

Surat Teerapittayanon, Bradley McDanel, and H. T. Kung. BranchyNet: Fast inference via early exiting from deep neural networks. In *23rd International Conference on Pattern Recognition (ICPR)*, pp. 2464–2469. IEEE, 2016.

Ruoyu Wang, Beier Zhu, Junzhi Li, Liangyu Yuan, and Chi Zhang. Adaptive stochastic coefficients for accelerating diffusion sampling. In *Advances in Neural Information Processing Systems*, 2025.

Yuyang Wang, Ahmed A. A. Elhag, Navdeep Jaitly, Joshua M. Susskind, and Miguel Ángel Bautista. Swallowing the bitter pill: Simplified scalable conformer generation. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 50400–50418. PMLR, 2024.

André Brasil Vieira Wyzykowski, Fatemeh Fathi Niazi, and Alex Dickson. AGDIFF: Attention-enhanced diffusion for molecular geometry prediction. *Journal of Chemical Information and Modeling*, 65, 2025. doi: 10.1021/acs.jcim.4c01896.

Minkai Xu, Lantao Yu, Yang Song, Chence Shi, Stefano Ermon, and Jian Tang. GeoDiff: A geometric diffusion model for molecular conformation generation. In *International Conference on Learning Representations*, 2022.

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

Xiangyang Xu and Hongyang Gao. Flow-matching based refiner for molecular conformer generation, 2025.

Liangyu Yuan, Ruoyu Wang, Tong Zhao, Dingwen Fu, Mingkun Lei, Beier Zhu, and Chi Zhang. Few-step diffusion sampling through instance-aware discretizations, 2026.

Wangbo Zhao, Yizeng Han, Jiasheng Tang, Kai Wang, Yibing Song, Gao Huang, Fan Wang, and Yang You. Dynamic diffusion transformer. In *International Conference on Learning Representations*, 2025.

Zihan Zhou, Ruiying Liu, Chaolong Ying, Ruimao Zhang, and Tianshu Yu. Molecular conformation generation via shifting scores, 2023.

Aihua Zhu, Rui Su, Qinglin Zhao, Li Feng, Meng Shen, and Shibo He. Hierarchical schedule optimization for fast and robust diffusion model sampling, 2025.

A BASELINES, DATASETS, AND METRICS

Baselines. We use two budget-respecting baselines, a full-budget reference, and a diagnostic upper bound.

Published. Uniform NFE at the published grid maximum $n = 5{,}000$ on every molecule. This is the configuration reported in Xu et al. (2022) as the operating point. Its NFE total is $N \cdot 5{,}000$, so for budgets $B < N \cdot 5{,}000$ Published is technically over-budget; we still report it as a horizontal reference line.

Best-Uniform. For each global budget B, choose the single uniform NFE $n^\star \in \mathcal{G}$ maximizing mean COV-R on the 150-molecule calibration set when applied to every molecule, subject to $N \cdot n^\star \le B$. This isolates the gain from *adaptive* (per-molecule) allocation relative to the best *uniform* schedule.

AtomCount-MCKP. At each $n \in \mathcal{G}$, fit a Gaussian-process regression $\mathrm{GP}_n(\text{heavy_atoms}(m)) \to \text{COV-R}(m, n)$ on the calibration set. Plug $\hat{c} = \mathrm{GP}_n$ into the same MCKP allocator the main method uses. This is the strongest single-feature baseline: it captures the published intuition that molecule size predicts difficulty, while still benefiting from MCKP's joint allocation.

Note that AtomCount-MCKP performs a per-NFE Gaussian process regression on heavy-atom count and does *not* enforce monotone-decreasing coverage in molecule size or monotone non-decreasing coverage in n per molecule. Our Ours-Prior ablation (α=0 in Table 3) is the apples-to-apples comparison for the design choice that matters – it is the monotone-isotonic version of the same single-feature regression. The ~ 33 pp gap between AtomCount-MCKP and Ours at 40% budget is therefore explained by GP regression producing non-monotone coverage predictions that mislead the MCKP allocator on the steep part of the coverage-vs-size curve; once monotonicity is enforced (Ours-Prior), 99.7% of Ours's performance is recovered.

Oracle-MCKP. The MCKP allocator given measured coverage labels of the held-out set at allocation time. Because those labels are single-seed, the allocator selects, per molecule, whichever rung happens to be scored high by noise, so the resulting value is inflated and is *not* an attainable bound. We report it as a diagnostic: the amount by which it exceeds the 3-seed methods measures how much noise a selection step can harvest.

Datasets. We use GEOM-Drugs (Axelrod & Gómez-Bombarelli, 2022) via the splits packaged with the GEODIFF release. The training pool for φ is the 1000 molecules in GEODIFF's validation split. The benchmark test set is test_data_1k, the same 200 drug-like molecules that Xu et al. (2022) report final coverage on. Phase B calibration uses 150 molecules from this 200-molecule set; the remaining 50 are held out for evaluation.

Metrics. For a molecule m with K reference conformers $\{C_j^{\text{ref}}\}_{j=1}^{K}$ and L generated conformers $\{C_l^{\text{gen}}\}_{l=1}^{L}$, with heavy-atom RMSD threshold δ:

$$\text{COV-R}(m, n) = \frac{1}{K}\Big|\{\, j : \min_l \mathrm{RMSD}(C_l^{\text{gen}}, C_j^{\text{ref}}) \le \delta \,\}\Big|. \tag{8}$$

On Drugs $\delta = 1.25\,\text{Å}$, on QM9 $\delta = 0.5\,\text{Å}$ (published conventions). Mean COV-R averages over molecules. We optimize for COV-R, the published primary metric.

B THREE SAMPLERS WE WRAP: EQUATIONS

Let $\{x_t\}_{t=0}^{T}$ be the noised conformer trajectory with variance schedule $\bar{\alpha}_t = \prod_{s \le t}(1 - \beta_s)$ and noise predictor $\epsilon_\theta(x_t, m, t)$ provided by the frozen backbone.

Langevin dynamics (GEODIFF default). GEODIFF's sampler uses fixed-stepsize Langevin updates:

$$x_{t-1} = x_t + \tfrac{1}{2}\, s_t\, \nabla_{x_t} \log p_t(x_t) \;+\; \sqrt{s_t}\,\xi, \quad \xi \sim \mathcal{N}(0, I), \tag{9}$$

with score $\nabla_{x_t} \log p_t(x_t) = -\epsilon_\theta(x_t, m, t)/\sqrt{1 - \bar{\alpha}_t}$ and step-size s_t set by `step_lr` $= 10^{-6}$. Each LD iteration is one model forward pass, so NFE $= n$ where n is the number of iterations.

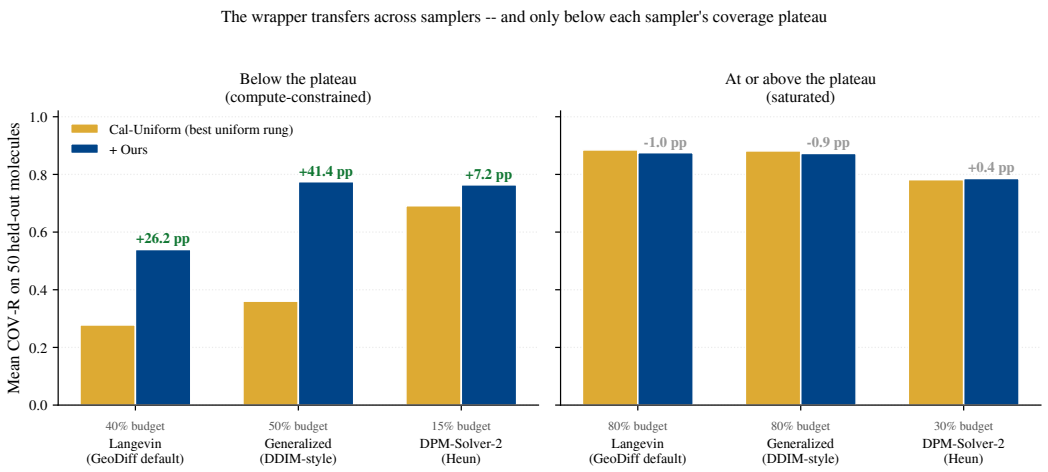


Figure 3: **Sampler-agnostic stackability.** Mean COV-R on 50 held-out molecules, wrapper against the Cal-Uniform baseline. *Left:* below the plateau, where allocation pays (Langevin @40%, generalized @50%, DPM-Solver-2 @15%). *Right:* at or above it, where it does not (@80%, @80%, @30%). Budgets are annotated beneath each bar; cells match Table 2 one-to-one.

Generalized (DDIM-style) sampler. This is GEODIFF's built-in DDIM-style stepping with optional noise (Song et al., 2021). Given consecutive timesteps $t > t'$ on a sub-sampled schedule, the update is

$$\hat{x}_0 = \tfrac{1}{\sqrt{\bar{\alpha}_t}}\big(x_t - \sqrt{1-\bar{\alpha}_t}\,\epsilon_\theta(x_t, m, t)\big), \tag{10}$$

$$\sigma_{t'} = \eta\sqrt{(1-\bar{\alpha}_{t'})/(1-\bar{\alpha}_t)}\sqrt{1-\bar{\alpha}_t/\bar{\alpha}_{t'}}, \tag{11}$$

$$x_{t'} = \sqrt{\bar{\alpha}_{t'}}\,\hat{x}_0 + \sqrt{1-\bar{\alpha}_{t'}-\sigma_{t'}^2}\,\epsilon_\theta(x_t, m, t) + \sigma_{t'}\,\xi. \tag{12}$$

We use $\eta = 1.0$ matching GEODIFF's default. Each step is one model forward pass; the schedule is sub-sampled so $\mathrm{NFE} \ll T$.

DPM-Solver-2 (Heun's method). A deterministic second-order ODE solver (Lu et al., 2022). For each macro-step from t_i to t_{i+1}, with midpoint t_{mid}:

1. Compute $\epsilon_1 = \epsilon_\theta(x_{t_i}, m, t_i)$.
2. First-order predict $\hat{x}_{t_{\text{mid}}}$ at t_{mid} using ϵ_1.
3. Compute $\epsilon_2 = \epsilon_\theta(\hat{x}_{t_{\text{mid}}}, m, t_{\text{mid}})$.
4. Heun average $\bar{\epsilon} = (\epsilon_1 + \epsilon_2)/2$ and step from t_i to t_{i+1} using $\bar{\epsilon}$.

Each macro-step costs two model evaluations, so $\mathrm{NFE} = 2n_{\text{outer}}$. We use a DDIM-style sub-sampled schedule spanning $[0, T)$ rather than the truncated last-n schedule GEODIFF uses for LD: ODE-style solvers integrate the reverse process directly and require schedule support that reaches $t = 0$.

Wrapper interface. For every sampler above, the wrapper observes only $(m, n, \text{COV-R}(m, n))$ triples and returns a per-molecule NFE assignment $\{n_i\}$. The sampler-side change consists of re-running Phase B calibration with the new sampler and re-fitting the isotonic prior $\{\text{iso}_n\}$; the LightGBM predictor φ trained in Phase A is reused.

C SANITY: REPRODUCING PUBLISHED GEODIFF

We report COV-R under two settings, which use distinct subsets and conventions; clarifying them up front avoids ambiguity. *(i)* The **reproduction value**: running the released checkpoint at uniform n=5,000 on the full 200-molecule test_data_1k gives 0.892, matching the COV-R reported by Xu et al. (2022) (0.8913) within 0.001 absolute COV-R ($\sim$ 0.1 pp), with the median likewise close (0.969 vs 0.9655) – a faithful reproduction. *(ii)* The **held-50 3-seed averaged Published value** is 0.873 on

Table 5: Cross-backbone transfer, rebuilt with a cross-validated size prior (φ unused) so that no molecule is scored by a predictor trained on it. Same allocator, deployed on GEODIFF versus AGDIFF on the same 200 dev molecules at the 4-point grid. Gains agree within 2.3 pp.

Budget	Gain on GEODIFF	Gain on AGDIFF	$\lvert\Delta\rvert$
30%	+21.0 pp	+20.9 pp	0.1
50%	+34.5 pp	+33.0 pp	1.5
70%	+14.4 pp	+12.1 pp	2.3
90%	+20.0 pp	+19.4 pp	0.6

the 50 held-out molecules; this is the baseline against which all main-text deltas are computed. The held-50 vs full-200 absolute difference (0.873 vs 0.892) reflects standard small-sample variance on a random 50-molecule subset; the held-50 evaluation is the methodologically clean choice because its calibration and evaluation sets are disjoint.

D BACKBONE STACKABILITY: AGDIFF

If allocation gain were tied to one backbone's failure modes, we would expect it to vanish on a stronger backbone. It does not. Taking the predictor φ trained on GEODIFF val and deploying it directly on AGDIFF (Wyzykowski et al., 2025) – without any retraining or relabelling – gives the cross-backbone transfer summary in Table 5: the adaptive-allocation gain agrees within 2.3 pp at every budget. Because the 200 dev molecules lie inside φ's training pool, we rebuild this comparison with a cross-validated size prior and φ unused, so the numbers below measure what transfers without any predictor leakage. What transfers is the size prior, not φ. Architectural improvements and per-input compute allocation are genuinely orthogonal efficiency axes, and the gain transfers cleanly across them.

E WHY SINGLE-SEED EVALUATION MANUFACTURES HEADROOM

Under single-seed ground truth the same comparison at 80% budget appears to favour our method, and an earlier version of this work reported a gain there. With three seeds the gain vanishes. The reason is selection on noise: the allocator picks, for each molecule, the rung that maximizes its *estimated* coverage, so any single-seed estimate that happens to be high is preferentially selected, and the resulting allocation inherits that upward bias. The Oracle-MCKP column of Table 1 makes the effect visible – computed on single-seed labels it exceeds every 3-seed method by more than a point at high budget, which is not an attainable bound but a measure of how much noise the selection step can harvest. We therefore recommend that evaluations of per-input compute allocation use disjoint seeds for allocation and for scoring, or report multi-seed ground truth; with a single seed a null result can be made to look positive.

F REPRODUCIBILITY SPECIFICATIONS

Backbone and environment. GEODIFF: the released pre-trained GEOM-Drugs checkpoint (2022 release) from the official repository. Environment: PyTorch 1.12.1+cu113, PyTorch Geometric 2.0.4, RDKit 2022.9.5, NumPy 1.26.4. Hardware: 8× NVIDIA RTX 3090 (24 GB). We pin one exact repository commit and checkpoint, both recorded in the released code, because the reproduction value (Appendix C) is sensitive to the released weights: a later checkpoint would shift the absolute COV-R and break comparability with the published 0.8913.

Sampling configuration. GEODIFF LD sampling config: `step_lr` $= 10^{-6}$, `w_global` $= 1.0$, `global_start_sigma` $= 0.5$, `eta` $= 1.0$, `num_confs` $= 2 \cdot |\text{refs}|$. These are GEODIFF's published defaults; we change none of them, so that the only difference between the baseline and our wrapper is the per-molecule choice of step count n, not any sampler hyperparameter. The `num_confs` $= 2 \cdot |\text{refs}|$ rule follows the GEOM-Drugs evaluation convention (Appendix A) and fixes the number of generated conformers per molecule independently of n, so that n controls only sampling depth and not ensemble size.

Predictor. LightGBM: 800 trees, 63 leaves, learning rate 0.05, early stopping on a 20% held-out fold, with an isotonic post-hoc calibrator fit on the train fold. The 63-leaf depth and early stopping guard against overfitting the 1,077-dimensional RDKIT feature vector on only 1,000 training molecules; the post-hoc isotonic calibrator maps raw LightGBM scores onto the $[0, 1]$ coverage scale monotonically, which is what allows φ to be blended directly with the isotonic prior in Eq. (4) without a scale mismatch. Later RDKit releases changed the definition of `NumHAcceptors`, one of φ's input features; recomputing features under RDKit 2022.9.5 reproduces every cell of the α sweep exactly, so we pin that version in the released code.

Per-phase cost. Phase A (one-time, distribution-level): $\sim$7 GPU-h to generate COV-R labels across the NFE grid on the 1,000 training molecules, then < 5 min CPU to fit LightGBM. Phase B (per-deployment): $\sim$3 GPU-h for calibration sampling on the in-domain set, < 30 s CPU to fit the isotonics. Phase C (per-batch inference): sub-second on CPU (Appendix N). Only Phase B recurs when the deployer changes molecule distribution or sampler; Phase A is amortized across all deployments.

G NFE-GRID CONSTRUCTION

The original GEODIFF test convention reports coverage at a single NFE point ($\mathcal{G}_0 = \{5{,}000\}$), which is sufficient to quote a headline number but gives the allocator no intermediate operating points to choose from. We therefore evaluate each molecule on a grid of candidate step counts.

From a coarse to a dense grid. We begin with the 4-point grid $\mathcal{G}_4 = \{500, 1500, 3000, 5000\}$ and then densify to the 9-point grid $\mathcal{G}_9 = \{500, 1500, 2000, 2500, 3000, 3500, 4000, 4500, 5000\}$ by sampling 5 additional points on the same 200 molecules. The added points are concentrated in the 2,000–4,500 range rather than spread uniformly, because that is where the per-molecule coverage curve has its steepest rise: at low n almost all molecules are far from saturation (coverage near zero), and beyond $n \approx 4{,}500$ most have already plateaued, so extra resolution there would not change any allocation decision. The dense grid matters most at moderate budgets, where a one-rung change in a molecule's assignment can move it across the steep part of its curve.

Grid spacing and the budget axis. The dense grid has a greatest common divisor of $s = 500$, so every attainable allocation total is a multiple of 500; the DP allocator (Eq. (5)) quantizes the budget axis at this same granularity, which keeps the discretized budget $B' = \lfloor B/s \rfloor$ small enough for the exact knapsack solve while still resolving the steep region. All budget levels reported in the main table (20%–100% of $N \cdot 5{,}000$) are exact multiples of this grid.

H BOOTSTRAP CONFIDENCE INTERVALS AND SIGNIFICANCE TESTING

Procedure. For each (method, budget) cell we resample the 50 held-out molecules with replacement 1,000 times and report the mean COV-R together with a 95% percentile bootstrap confidence interval. For a paired comparison (e.g. Ours vs. AtomCount-MCKP) we recompute the per-molecule difference on each resample, using the *same* resampled molecule indices for both methods, and report the mean Δ and the 95% CI of the resampled Δ distribution. The one-sided p-value is the empirical fraction of resamples with $\Delta \leq 0$.

Why this design. The comparison is paired because both methods are evaluated on the same 50 molecules, and a molecule that is intrinsically hard inflates or deflates both methods together; pairing removes that shared molecule-level variance and isolates the difference attributable to the allocation strategy. We use a percentile bootstrap rather than a parametric t-test because per-molecule coverage is bounded in $[0, 1]$ and heavily skewed (many molecules at 0 or near 1 at a given budget), so a normal-error assumption is unwarranted. The one-sided test reflects the directional hypothesis that the adaptive allocator does not *hurt* relative to the baseline.

Relation to multi-seed averaging. The two variance-reduction steps address different noise sources and compose. Multi-seed averaging (Appendix I) removes *within-cell* stochastic-sampling noise so that each COV-R(m, n) label is a stable estimate; the paired bootstrap then quantifies the remaining

Table 6: Single-seed coverage on the held-out 50 molecules.

Budget (% Pub.)	Published	AtomCount-MCKP	Ours	Δ vs AC
20%	0.873	0.000	0.203	+20.3
40%	0.873	0.201	0.536	+33.5
60%	0.873	0.744	0.810	+6.6
80%	0.873	0.875	0.878	+0.3 (NS)
100%	0.873	0.870	0.879	+0.9 (NS)

Table 7: Sensitivity to blend weight α; mean COV-R averaged over the 9 budget levels on the 50 held-out molecules.

α	0.00	0.05	0.10	0.20	0.30	0.50	0.70	1.00
Mean COV-R	0.690	0.692	**0.692**	0.692	0.690	0.685	0.659	0.576

across-molecule signal, i.e. whether the measured Δ would persist under resampling of the molecule set. Without seed averaging, sampling noise enters both the labels the allocator selects on and the labels it is scored against, which biases high-budget comparisons upward rather than merely widening them; the single-seed comparison (Appendix J) shows the size of the effect.

I MULTI-SEED AVERAGING

For each (m, n) sampling cell we average COV-R over 3 independent sampling seeds, where a seed controls the random noise injected during sampling. This matters specifically because the published GEODIFF sampler is Langevin dynamics, a *stochastic* sampler (Appendix B): two runs at the same n on the same molecule draw different noise and yield different conformer ensembles, hence different coverage. A single seed therefore reports a noisy estimate of the cell's true coverage, and at high n – where the true gains over the baseline are only a few points – that per-cell noise is comparable in size to the signal. Averaging over seeds reduces this variance so that the isotonic prior is fit to stable targets and the reported deltas reflect allocation, not seed luck.

Multi-seed averaging is standard practice for evaluating stochastic generative models (Xu et al., 2022; Karras et al., 2022). We apply it identically on both the 150-molecule calibration set (so the prior and predictor are fit to denoised labels) and the 50-molecule evaluation set (so the reported metrics are denoised), keeping the two sets on the same measurement footing. The deterministic solver DPM-Solver-2 ($\eta = 0$) is far less seed-sensitive, but we average it the same way for consistency.

J SINGLE-SEED COMPARISON

For completeness, Table 6 reports single-seed COV-R on the same held-out 50 molecules at the dense 9-point grid (no seed averaging).

Low-budget gains are nearly identical with and without seed averaging because coverage curves at $n \leq 2{,}000$ are close to zero for all methods. The high-budget rows are the protocol-sensitive ones: under a single seed Ours appears to lead at 80% and 100% budget, whereas with 3-seed ground truth (Table 1) none of those differences is significant. Appendix E explains why: an allocator selects the rung that maximizes each molecule's estimated coverage, so single-seed noise is preferentially harvested. We report 3-seed results in the main text and treat the single-seed table as a cautionary comparison rather than a result.

K SENSITIVITY TO THE BLEND WEIGHT α

Table 7 sweeps the convex-combination weight α in $\hat{c} = \alpha\varphi + (1 - \alpha)\text{iso}$. Between $\alpha = 0.05$ and $\alpha = 0.20$ mean COV-R varies by 0.0027; outside this window, performance degrades smoothly toward the LightGBM-only endpoint. We use $\alpha = 0.10$ throughout the main paper.

The sweep confirms that the blend is dominated by the prior. The two endpoints are the ablation variants of Table 3: $\alpha = 0$ is the prior-only Ours-Prior (0.690) and $\alpha = 1$ is the LightGBM-only Ours-LGBM (0.576), an 11.6 pp gap that establishes the prior as the load-bearing component. Adding a small LightGBM weight lifts the mean marginally, from 0.690 to 0.692. Beyond $\alpha = 0.20$ performance falls monotonically toward the LightGBM-only endpoint (0.690 at 0.30, 0.685 at 0.50, 0.659 at 0.70), because the noisier learned predictor increasingly overrides the calibrated prior. The flat top between $\alpha = 0.05$ and $\alpha = 0.20$ (spread 0.0027) means the exact weight is not delicate; we fix $\alpha = 0.10$ throughout, in the middle of the flat region, so that the molecule-specific residual stays in the estimate: Section 5.6 shows it is the only molecule-level signal left when target molecules fall outside the prior's support.

L PER-MOLECULE CASE STUDIES

Figure 4 illustrates the allocator's per-molecule behaviour on six benchmark molecules spanning the heavy-atom range, showing each molecule's measured coverage curve and the NFE assigned to it by each method as the global budget varies.

M FAILURE-CASE ANALYSIS

Across all (method, budget) pairs in the main table, one failure mode occurs. It is small in magnitude but worth naming.

Saturation reversal at full budget. At 100% budget, the wrapper sometimes assigns slightly less than the full $n = 5{,}000$ to molecules whose predicted coverage curve is already flat at $n = 4{,}500$, and reallocates the saved NFE to a harder molecule that does not benefit. The net effect is small (< 1 pp absolute) and, at full budget, within the noise band established in Appendix E. This is a consequence of the predictor's discretized NFE grid; a finer grid would reduce it.

In the 70–100% budget range the single-seed Oracle-MCKP column of Table 1 sits 1.0–1.9 pp above Ours. As Appendix E explains, that gap is not a measure of predictor error: the oracle allocates on the same single-seed labels it is scored against, so most of the margin is harvested measurement noise rather than attainable headroom.

N COMPONENT WALL-CLOCK BREAKDOWN

Table 8 gives the wall-clock cost of the wrapper components, on a batch of $N = 50$ molecules at an average per-molecule NFE of 4,000. The wrapper's inference overhead is negligible relative to the diffusion backbone.

Table 8: Per-batch wall-clock breakdown (50 molecules, average $n = 4{,}000$, single RTX 3090).

Component	Wall-clock
RDKIT feature extraction (CPU)	∼0.10 s
LightGBM inference at 9 NFE points (CPU)	∼0.05 s
Isotonic regression lookup (CPU)	<0.01 s
Multi-choice knapsack DP (CPU)	∼0.30 s
Total wrapper overhead	∼0.5 s
LD sampling at average $n = 4{,}000$ (GPU)	∼19,782 s

The total wrapper overhead of ∼0.5 s is about 2.5×10^{-5} of the ∼19,782 s spent in backbone sampling – more than four orders of magnitude smaller – so the allocator's cost does not appear in any end-to-end speedup figure. Within the overhead, the exact multi-choice knapsack DP is the largest single component at ∼0.30 s, yet it remains sub-second because the problem is small (Section 4.2): $N \leq 200$ molecules, $|\mathcal{G}| = 9$ rungs, and a budget axis quantized at $s = 500$ (Appendix G). All wrapper components run on CPU and in parallel with no GPU contention, so they overlap trivially with the GPU sampling that dominates the runtime. This is the quantitative basis for treating the allocator as "free" relative to the backbone.

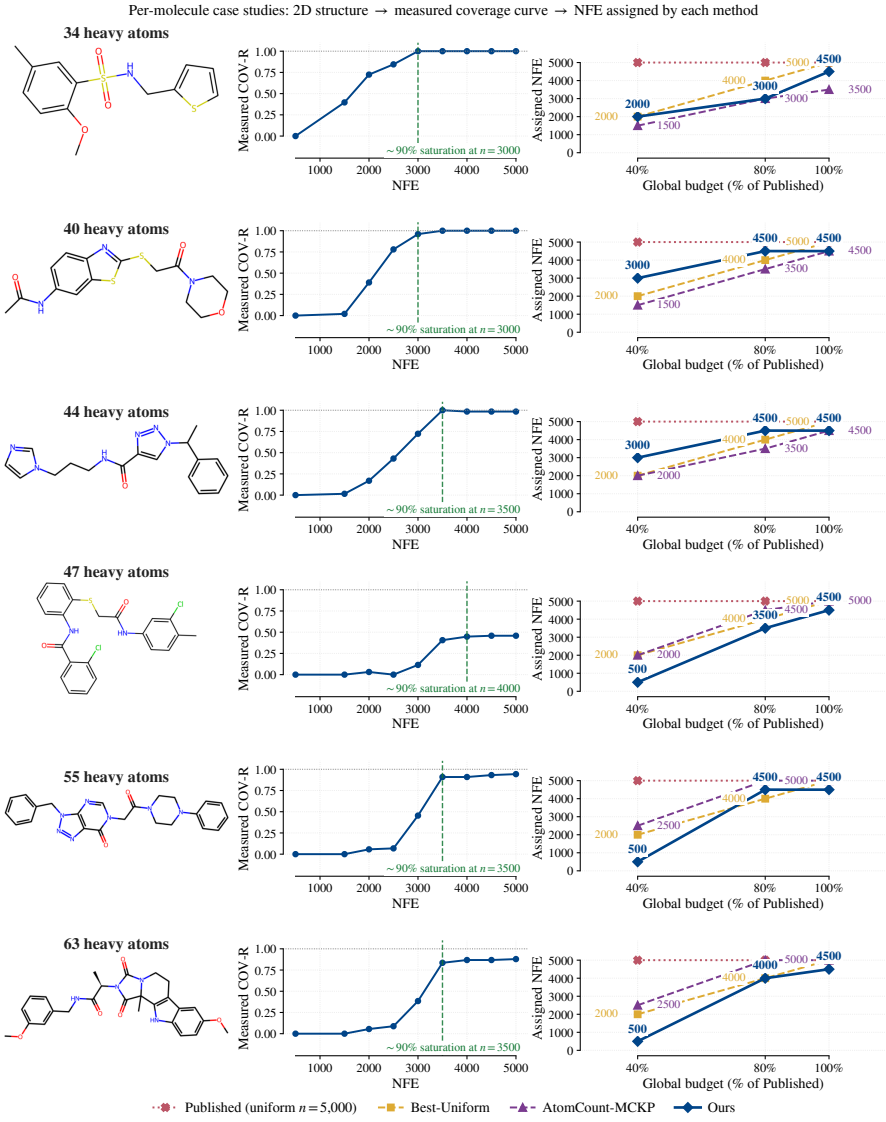


Figure 4: **Per-molecule case studies: all six molecules.** Panels as in Figure 2, for six benchmark molecules spanning the heavy-atom range. ConformerDiff-Adaptive hands easy molecules at or just below their own saturation rung, well under the published n=5,000; at 40% budget the hardest case is dropped to the floor rung n=500, where measured coverage is zero, so the saved NFE lifts more tractable cases.

O DPM-SOLVER-2 IMPLEMENTATION DETAILS

We added a DPM-Solver-2 sampling routine to GEODIFF's backbone, performing Heun-style second-order stepping with two model evaluations per macro-step. The schedule is DDIM-style sub-sampled across $[0, T)$ rather than GEODIFF's truncated last-n schedule, because the truncated schedule is appropriate for iterative LD refinement but not for ODE-style accelerated solvers that integrate the reverse process directly. The step-size formula matches GEODIFF's generalized mode with $\eta = 0$ (deterministic). The full per-macro-step procedure is given in Appendix B.

NFE accounting. Because each macro-step costs two model evaluations, the reported NFE is $2\, n_{\text{outer}}$, where n_{outer} is the number of macro-steps. An allocation of 300 NFE therefore corresponds to 150 macro-steps, which is why DPM-Solver-2 reaches a usable coverage at a fraction of LD's step

count. We hold the wrapper's NFE grid fixed in these units, so an allocation of "300 NFE" means the same compute budget regardless of sampler.

Why the deterministic ceiling is lower. At $\eta = 0$ the solver is fully deterministic and performs no stochastic exploration of the conformer landscape. For molecules with many near-degenerate low-energy conformers, that exploration is what lets stochastic LD eventually cover diverse references, so the deterministic solver saturates at a lower coverage ceiling (below 0.8 in Table 2, against about 0.88 for LD) than LD even when given more steps. This is a property of the sampler, not the wrapper: the allocator still extracts a significant gain within the DPM-Solver-2 tier ($+7.2$ pp below its plateau, Table 2), it simply operates under a lower absolute ceiling. A stochastic higher-order solver would raise the ceiling while preserving the wrapper's relative gain.

P CODE AVAILABILITY

All experimental code, trained predictors φ, isotonic priors iso_n, calibration labels, and bootstrap CSVs are released anonymously at `https://anonymous.4open.science/r/ConformerDiff-B5F0` (deanonymized upon acceptance).