Shifting a Molecular Generator Toward Developability with Iterative Importance Fine-Tuning

Anonymous Authors[1]

Abstract

Designing small molecules with strong target binding while maintaining favorable absorption, distribution, metabolism, excretion, and toxicity (ADMET) properties or synthetic accessibility remains a central challenge in drug discovery. Recent generative models such as GenMol enable efficient exploration of chemical space for hit identification and lead optimization. However, optimizing for a single objective (e.g., protein binding) often degrades other critical ADMET or synthesizability properties. We propose a framework that combines GenMol with Iterative Importance Fine-Tuning (IIFT), a reward-based post-training method that shifts the generative distribution toward molecules satisfying multiple objectives. IIFT requires only a scalar reward and does not assume differentiability, allowing incorporation of black-box oracle functions. We employ IIFT to develop two distribution-shifted variants of GenMol: **Viable GenMol**, which biases generation toward ADMET-favorable molecules, and **Tractable GenMol**, which biases generation toward synthetically accessible molecules. We evaluate these models in *de novo* generation, hit identification, and lead optimization, achieving replicable improvements across all three modalities while preserving drug-likeness, synthetic accessibility, and molecular diversity. We validate selected tractable lead-optimized candidates using binding free energy calculations of molecular affinity. Our results demonstrate that importance-based distribution shifting provides a practical approach to multi-objective molecular design under realistic drug discovery constraints.

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

1. Introduction

The discovery of novel small molecules with desired biological or physicochemical properties is a central challenge in computational drug discovery (Southey & Brunavs, 2023; Lin et al., 2020). The space of synthetically accessible organic molecules is vast–commonly estimated to exceed 10^{60} compounds–making exhaustive search impossible (Reymond, 2025; Kirkpatrick & Ellis, 2004). Consequently, recent years have seen increasing interest in machine learning methods capable of navigating chemical space and proposing candidate molecules with desirable functional properties (Paul et al., 2021; Patel et al., 2020).

Traditional machine learning approaches in chemoinformatics have largely focused on property prediction, where models estimate quantities such as binding affinity, solubility, toxicity, or drug-likeness from molecular representations (Mitchell, 2014; Lo et al., 2018; Niazi & Mariam, 2023). While these models enable efficient screening of large libraries, they do not directly address the inverse design problem: discovering entirely new molecules that satisfy specified objectives. To address this limitation, researchers have developed generative models capable of producing novel molecular structures directly (Meyers et al., 2021; Bian & Xie, 2021; Pang et al., 2024; Merz et al., 2020).

The **Gen**eralist **Mol**ecular (GenMol) generative model was introduced as a versatile framework for molecule generation across multiple drug discovery tasks (Lee et al., 2025). GenMol formulates molecular generation as a masked discrete diffusion process over tokenized molecular sequences, using a BERT-style transformer architecture trained to iteratively reconstruct masked tokens. Unlike autoregressive models that generate molecules token-by-token, GenMol employs non-autoregressive parallel decoding, allowing it to iteratively fill masked positions while leveraging bidirectional context across the entire molecule. This formulation improves sampling efficiency and enables flexible generation scenarios such as *de novo* molecule generation, target-guided hit identification, and lead optimization through fragment remasking. In the fragment remasking strategy, selected fragments of an input molecule are replaced with masked tokens and regenerated by the diffusion process, allowing efficient exploration of chemical space around ex-

isting scaffolds. The model further supports controllable generation through molecular context guidance, which adjusts predictions based on contextual molecular information during sampling. These capabilities make GenMol a promising foundation for generative drug discovery pipelines.

Despite substantial progress, generating molecules that satisfy specified functional objectives remains challenging (Bilodeau et al., 2022). Many molecular generative models are trained primarily as distribution-learning systems with likelihood-based objectives that encourage reproduction of training distributions without considering downstream properties such as desired biological or pharmacological profiles (Flam-Shepherd et al., 2022). Goal-directed molecular generation augments generative models with reward-based optimization, such as reinforcement-learning (RL) fine-tuning. In these approaches, generated molecules are scored by auxiliary property objectives such as bioactivity or affinity, and the generator is updated to favor higher-scoring regions of chemical space (Yang et al., 2023; Park et al., 2025). RL approaches can, however, be difficult to train and can over-optimize imperfect reward functions, leading to reduced diversity, distribution drift, and unrealistic molecules. In drug discovery, this challenge is compounded by the fact that reward signals are often noisy, biased, approximate and/or expensive to obtain, especially when they rely on learned predictors, docking, or other computationally intensive evaluations.

These challenges highlight a key tension in molecular generative modeling: balancing optimization of task-specific objectives with preservation of a realistic prior over chemical space. Methods that aggressively optimize rewards often distort the learned distribution, while methods that remain close to the training distribution fail to discover high-performing molecules.

IIFT is a recently-developed reward-based fine-tuning framework that reshapes the generative distribution through a sequence of importance-weighted updates (Denker et al., 2025). Rather than relying on RL or gradient-based guidance, IIFT iteratively samples trajectories from the current model, evaluates them using an external reward function, and accepts or rejects samples according to a reward-tilted distribution. Accepted samples are used to update the generative model through supervised regression, progressively shifting the model toward higher-reward regions of the sample space. IIFT has been demonstrated in low-dimensional sampling, class-conditional MNIST generation, and text-to-image models (Denker et al., 2025). A key advantage of IIFT is that it does not require differentiability of the reward function, enabling optimization with respect to black-box oracles. This makes IIFT a natural candidate for molecular generation, where the objectives are often black-box, non-differentiable, and/or expensive to obtain. Furthermore, by maintaining a replay buffer of accepted samples and performing incremental updates, IIFT also mitigates instability and mode collapse that can arise in RL–based fine-tuning.

In this work, we apply IIFT to GenMol to align molecular generation with external reward functions defined by drug-discovery-relevant oracles (Figure 1). We develop two distribution-shifted GenMol variants: **Viable GenMol**, which is aligned to favor molecules with improved ADMET profiles, and **Tractable GenMol**, which is aligned to favor molecules with improved retrosynthetic accessibility. We investigate these models in the settings of *de novo* generation, hit generation, and lead optimization using black-box reward signals for ADMET and retrosynthetic accessibility. We find that IIFT improves GenMol's ability to generate molecules satisfying the relevant downstream objectives, while maintaining diversity and preserving strong baseline performance in de novo drug-likeness, property-guided hit identification, and docking-based lead optimization.

2. Related Work

Molecular generative models. Early approaches to molecular generation focused on sequence-based and latent-variable models operating on SMILES representations. Variational autoencoders (VAEs) (Gómez-Bombarelli et al., 2018) and recurrent neural networks (RNNs) enabled generation and optimization of syntactically valid molecules. Graph-based models later improved chemical validity by directly modeling molecular structure, including methods such as JT-VAE (Jin et al., 2018), MolGAN (De Cao & Kipf, 2018), and NeVAE (Samanta et al., 2019).

Diffusion models for molecular design. More recent work has explored diffusion-based generative models for molecules, which provide a flexible framework for modeling complex distributions. Continuous and geometric diffusion models such as EDM (Hoogeboom et al., 2022) and GeoDiff (Xu et al., 2022) demonstrated strong performance in generating 3D molecular structures, while discrete diffusion approaches such as DiGress (Vignac et al., 2023) extended these ideas to molecular graphs. GenMol (Lee et al., 2025) introduces a masked discrete diffusion formulation over tokenized molecular sequences. By enabling selective modification of molecular substructures, it is well suited to lead optimization tasks that preserve a core scaffold.

Steering of molecular generative models. A common approach to molecular optimization is to treat generation as a sequential decision-making problem and RL optimization of a reward function. Methods such as REINVENT-Transformer (Xu et al., 2024) optimize generative models with respect to property predictors. While effective in some settings, these methods can be unstable and prone to reward

hacking, typically requiring careful reward design and tuning. Several recent works have explored alternatives to RL for steering generative models. GFlowNets (Bengio et al., 2026) aim to sample from reward-proportional distributions by learning stochastic policies over trajectories. Other approaches, including IIFT (Denker et al., 2025), shift model distributions using reweighted datasets rather than policy gradients, iteratively constructing an accepted-sample distribution via importance-based resampling and updating the model using a supervised diffusion objective.

3. Methods

3.1. GenMol

GenMol (Lee et al., 2025) is a generalist masked discrete-diffusion model over molecular sequences that supports *de novo* generation, hit identification, and lead optimization. Molecules are built from SAFE tokens, a fragment-based alternative to SMILES strings that are better-suited for scaffold generation or fragment linking (Noutahi et al., 2023). *De novo* generation starts without a conditioning scaffold or seed molecule, so improvements reflect whether fine-tuning can bias the distribution toward higher-reward molecules. Hit identification focuses on discovering molecules that satisfy a target objective without requiring preservation of a specific starting scaffold or seed compound. Lead optimization seeks improved analogs that preserve core structural features while improving potency and developability.

For *de novo* generation, GenMol starts from a masked SAFE sequence and iteratively fills masked positions with a masked discrete diffusion sampler (Lee et al., 2025). Because GenMol predicts masked tokens in parallel with a BERT-style denoiser (Devlin et al., 2019) rather than autoregressively, it can leverage bidirectional molecular context (Lee et al., 2025). We use this unconditional setting to test whether IIFT can shift the base GenMol distribution toward molecules with improved oracle values.

For hit identification, GenMol performs goal-directed search by constructing a fragment vocabulary from an initial molecular set and using fragment attachment and fragment remasking to explore chemical space (Lee et al., 2025). This setting naturally lends itself to practical molecular optimization (PMO), where the objective is to discover molecules with high oracle scores under a limited evaluation budget. PMO in GenMol is intended to find high-scoring candidates within the model's sampled distribution. By contrast, our use of IIFT is intended to reshape the underlying distribution itself toward a specified property objective rather than merely filtering favorable samples after generation. Like the original work, we test our post-trained GenMol's ability to perform hit identification on the 23 oracles provided by the Therapeutic Data Commons (Huang et al., 2021), but place a more detailed focus on the multi-property optimization (MPO) objectives and the DRD2, JNK3, and GSK3β oracles most relevant to drug discovery.

For lead optimization, GenMol applies a fragment-remasking strategy starting using fragments derived from a seed molecule to generate candidate analogs and iteratively expand the search space beyond the original scaffold (Lee et al., 2025). Here, we employ QuickVina2 (Alhossary et al., 2015) docking scores as the target oracle. The original GenMol work evaluated fifteen target-lead pairs spanning BRAF, PARP1, 5HT1B, Factor VII, and JAK2. We consider all of these as well as BAZ2A, BAZ2B, CREBBP, and thrombin (Drouin et al., 2015; Xu et al., 2016; Weitz, 2007).

3.2. Reward oracles

We consider two black-box reward signals for IIFT fine-tuning of GenMol: an ADMET oracle based on ADMET-AI (Swanson et al., 2024), which defines **Viable GenMol**, and a synthesizability oracle based on AiZynthFinder (Genheden et al., 2020; Saigiridharan et al., 2024), which defines **Tractable GenMol**. We use ADMET-AI as a broad-coverage predictor of ADMET properties achieving the highest average rank on the TDC ADMET leaderboard (Huang et al., 2021) across 22 benchmark datasets while remaining fast enough for large-scale evaluation. For synthesizability, we use AiZynthFinder, an open-source retrosynthetic planner that identifies plausible routes to purchasable precursors (Genheden et al., 2020), as a modern alternative to the SAScore evaluation (Ertl & Schuffenhauer, 2009). Our composite ADMET and synthesizability rewards are intended as practical developability filters rather than predictors of clinical success, so we therefore evaluate them alongside chemical motif-based analyses.

For **Viable GenMol**, we use ADMET-AI predictions as inputs to a composite ADMET score. Given a molecule x, ADMET-AI returns a panel of predicted endpoint values, including physicochemical properties, absorption and permeability proxies, metabolism-related liabilities, and toxicity-related endpoints. Our implementation groups these predictions into four pillars: *developability*, *exposure*, *metabolism*, and *toxicity*. Within each pillar, we aggregate endpoint-level desirabilities using a geometric mean, and then combine the four pillar scores using a weighted geometric mean. Numerical weights and parameters are reported in Appendix A.1.1. We additionally apply penalties when predicted liabilities for the Ames mutagenicity test (Ames), human Ether-à-go-go-Related Gene (hERG) activity, and drug-induced liver injury (DILI) (Vijay et al., 2018; David & Hamilton, 2010; Lamothe et al., 2016) exceed preset thresholds. This produces a final score $s_{\mathrm{ADMET}}(x) \in (0, 1]$, which we convert into a reward as, $r_{\mathrm{ADMET}}(x) = \min[0, \log(s_{\mathrm{ADMET}}(x)/s^*)]$. We adopt

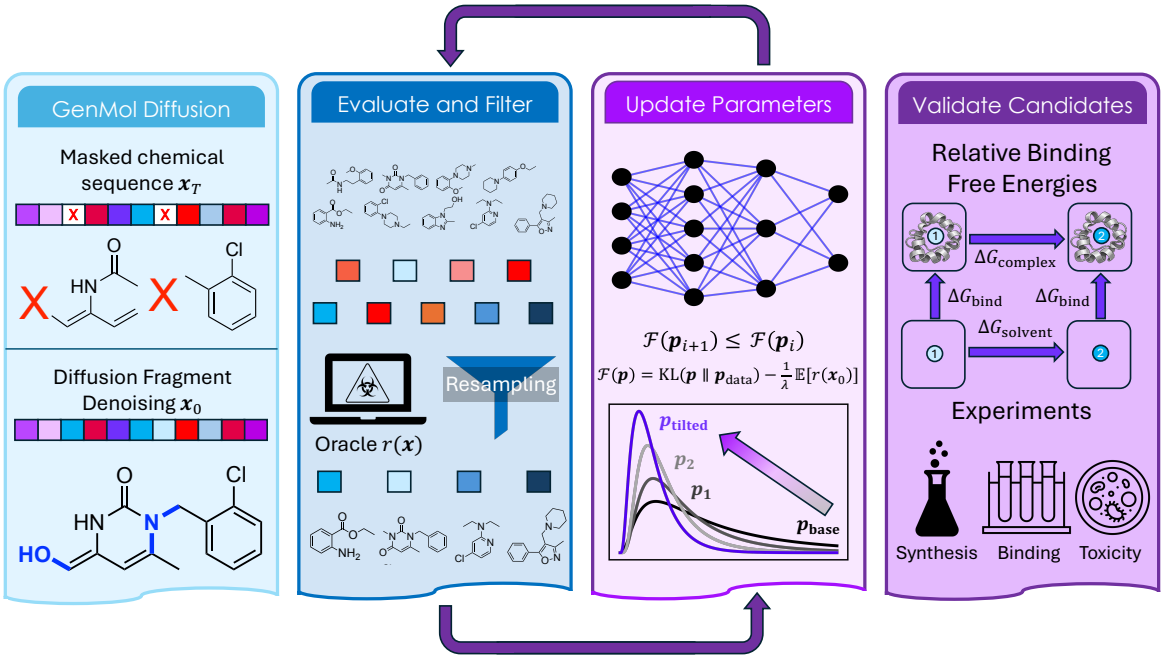

Figure 1. Overview of the GenMol-IIFT workflow. A masked molecular sequence is generated with GenMol's discrete diffusion model and refined through fragment denoising to produce candidate molecules. These candidates are evaluated and filtered using reward or oracle signals, after which selected samples are resampled and used to update the model parameters and shift the generative distribution toward a reward-tilted distribution. The resulting candidates can be validated using downstream high-fidelity computational or experimental assays.

$s^* = 0.2$ as a recentering that makes molecules near the desired ADMET regime have reward close to zero, while less desirable molecules receive more negative values.

For **Tractable GenMol**, we use AiZynthFinder to obtain a retrosynthesis-based score. For each molecule, AiZynthFinder performs retrosynthetic search under a specified configuration, stock, and policy, and returns summary statistics describing the search result. Our oracle converts these outputs into a $[0, 1]$ bounded score, $r_{\mathrm{synth}}(x) = w_{\mathrm{solved}}\,\mathbf{1}\{\mathrm{solved}\} + w_{\mathrm{stock}}\,f_{\mathrm{stock}} + w_{\mathrm{routes}}\,f_{\mathrm{routes}} + w_{\mathrm{score}}\,f_{\mathrm{top}} + w_{\mathrm{steps}}\,f_{\mathrm{steps}}$, where the f_i terms correspond to whether a route is solved, the fraction of leaf precursors found in stock, the number of solved routes, the top route score, and a preference for shorter routes. Exact weights w_i are reported in Appendix A.1.2.

3.3. Iterative importance fine-tuning (IIFT) of GenMol

We consider a pre-trained GenMol p_θ over molecules $x \in \mathcal{X}$ together with a scalar reward function r_{ADMET} or r_{synth}. Our goal is to bias generation toward a reward-tilted target distribution while avoiding RL-style policy gradients (Denker et al., 2025) which can suffer from high-variance updates, sensitivity to reward scaling, and poor sample efficiency when oracle evaluations are expensive or rewards are sparse (Sutton et al., 1999).

The IIFT framework comprises three iterative steps: sampling from the current model, constructing an accepted set of samples using reward-dependent acceptance probabilities, and then updating the model with a supervised diffusion fine-tuning loss defined on the accepted-sample distribution (Denker et al., 2025). We implement this procedure for GenMol as follows. At iteration k, we sample a batch of molecules from the current GenMol model and evaluate the scalar reward $r(x)$ on the valid molecules. We then assign each sample a resampling score, $\log s(x) = r(x)/T + \log \rho(x)$, where T is a temperature hyperparameter and $\log \rho(x)$ is the cumulative log-probability difference between the guided sampler and the unguided GenMol backbone along the sampled trajectory.

For every token that is newly unmasked and sampled, we accumulate the difference between its log-probability under the guided model and that under the base model, where both quantities are obtained from the corresponding denoiser logits before sampling the token. Summing these terms across all newly sampled tokens and denoising steps yields a pathwise log-likelihood ratio between the guided and base generative processes that can be viewed as a log Radon-Nikodym derivative of the guided sampling measure with respect to the base sampling measure (Oksendal, 2003).

The accepted objects are the final tokenized molecular sequences associated with those molecules. These sequences are appended to a replay buffer $\mathcal{D}$. When the

buffer exceeds a fixed capacity, the oldest entries are removed in first-in-first-out (FIFO) order. Minibatches are then drawn uniformly from $\mathcal{D}$ for supervised fine-tuning of GenMol (Denker et al., 2025). The replay buffer can be conceived as a finite approximation of the accepted-sample distribution. Iterative acceptance and fine-tuning progressively shifts the model toward a reward-tilted target distribution.

Model updates are performed using the standard GenMol masked discrete diffusion loss on samples drawn from the replay buffer, $\mathcal{L}_{\mathrm{IIFT}}(\theta) = \mathbb{E}_{x\sim\mathcal{D};\, t\sim\mathcal{U}[0,T]}\big[\mathcal{L}_{\mathrm{MDLM}}(x_t;\theta)\big]$. Importantly, the reward does not appear explicitly in this loss, and thus does not have to be backpropagated through the model weights or even be differentiable. Instead, the reward influences training through the accepted-sample distribution as only molecules that survive the resampling step reside within $\mathcal{D}$. For each replayed sequence, training follows the standard MDLM corruption-and-denoising procedure: a diffusion time is sampled, the sequence is corrupted, the frozen backbone produces base logits and hidden states, the guidance head modifies those outputs, and the final loss is the usual MDLM objective averaged over the minibatch. Full training details are provided in Section A.2.1.

IIFT fine-tuning can be understood as targeting a reward-tilted distribution, $p_{\mathrm{tilted}}(x) \propto p_{\mathrm{base}}(x)\exp\big(r(x)/\lambda\big)$, where p_{base} denotes the base generative distribution and $\lambda > 0$ controls the strength of reward preference. In practice, IIFT approaches this target iteratively: molecules are sampled from the current model, scored using the reward and sampler-dependent trajectories, preferentially retained through adaptive resampling, and then used to fine-tune the model with the standard diffusion objective. It can be shown that this induces a monotonic decrease of a free-energy objective, $F(p) = \mathrm{KL}\big(p \,\|\, p_{\mathrm{data}}\big) - \frac{1}{\lambda}\,\mathbb{E}_{x\sim p}[r(x)]$, whose minimizer is exactly p_{tilted} (Denker et al., 2025).

We use two related but distinct IIFT regimes. For both *de novo* generation and hit identification, we use the standard setting in which the model is iteratively fine-tuned on selected *de novo* samples drawn from the current generator. This setting shifts the model toward higher-reward regions of chemical space without imposing an additional task-specific search structure beyond sample generation, scoring, and replay-based fine-tuning. In this regime, training is tracked both by *de novo* generation every epoch and evaluating reward values, quantitative estimate of drug-likeness (QED), synthetic accessibility (SA), and diversity scores of each molecules (Figure A.1), as well as tracking the IIFT loss (Figure A.2). Hyperparameters and implementation details can be found in Section A.2.2.

For lead optimization, we instead use a "hybrid" IIFT procedure that combines local search around a seed molecule with reward-aware post-training. Candidate molecules are generated within a constrained lead-optimization loop that recombines fragments derived from the seed molecule and refines the resulting molecules with masked-token modification. The lead-search population is updated only with candidates that satisfy the predefined lead-optimization criteria (i.e., QED, SA, reward oracle, and docking $\Delta\Delta G$ filters). IIFT resampling is then applied to molecules produced by this lead-optimization loop, and only a subset of these candidates is retained for training and appended to $\mathcal{D}$. We adopt this on-the-fly formulation to keep the overall computational budget aligned with the underlying lead-optimization task. An alternative approach would perform a separate post-training phase on lead-optimized samples and then run lead-optimization using the updated model, but this would effectively allocate two stages of optimization compute: one for training and one for sampling. In contrast, we reuse the same stream of candidate molecules encountered during optimization for immediate replay-buffer updates rather than granting the method an additional offline adaptation phase. Hyperparameters and implementation details can be found in Section A.2.3.

4. Results

In practice, a useful generative model should not only produce molecules with better average reward, but should increase the number of candidates that satisfy task-relevant success criteria while remaining acceptable under auxiliary developability constraints. We summarize in Table 1 the performance of the two IIFT fine-tuned models **Viable GenMol** and **Tractable GenMol** across the three modalities of *de novo* generation, hit identification, and lead optimization.

Table 1. Summary of results across GenMol modalities.

Modality	Measurement	Improvement (↑)
	Viable GenMol	
De Novo	Hit Count	+146.5%
De Novo	Ames/hERG/DILI Filter	+92.0%
Hit ID	Hit Count	+23.8%
Hit ID	Ames/hERG/DILI Filter	+17.7%
Lead Opt	Hit Count	+11.7%
Lead Opt	Ames/hERG/DILI Filter	+4.5%
	Tractable GenMol	
De Novo	Hit Count	+62.9%
De Novo	Problematic Motif Filter	+20.2%
Hit ID	Hit Count	+13.7%
Hit ID	Problematic Motif Filter	+8.8%
Lead Opt	Hit Count	+5.7%
Lead Opt	Problematic Motif Filter	+3.2%

We report improvements in the number of practically useful hits, where the definition of hit depends on the modality. In *de novo* generation, where there is no separate task oracle,

a hit is defined directly by the reward objective itself: a molecule must exceed thresholds of ADMET-AI ≥ 0.6 or AiZynthFinder ≥ 2.0. In hit identification, a hit must satisfy both the task oracle and the reward oracle: specifically, it must score above the 60$^{\text{th}}$ percentile of the base model under the target oracle and also exceed the relevant ADMET-AI or AiZynthFinder threshold (Figures A.3 and A.4). In lead optimization, the target condition is adapted to the scaffold-constrained setting: a hit must improve upon the original lead under the docking oracle while also satisfying the corresponding reward threshold.

We also report in Table 1 the change in hit count after filtering out by simple chemistry-motivated exclusions: motifs well-known to be liabilities for the Ames mutagenicity test (Ames), human Ether-à-go-go-Related Gene (hERG) activity, and drug-induced liver injury (DILI) (Huang et al., 2021) for **Viable GenMol**, and problematic synthesis-related motifs for **Tractable GenMol**. Both sets of chemical motifs flagged by these filters are reported in Appendix A.3, and bar plots illustrating the rates of molecules passing these filters are reported in Figure A.5.

In addition to the summary statistics in Table 1, which shows hit count elevations across the board, we illustrate the distribution shifts induced by IIFT under all 23 TDC oracles for hit identification in Figures A.6 and A.7, and for lead optimization in Figures A.8 and A.9. We observe particularly high performance of **Viable GenMol** and **Tractable GenMol** over base GenMol in the low-budget regime of oracle evaluations. At larger budgets, base GenMol PMO routines tend to narrow the gap by discovering many local variations of already successful hits, but the IIFT-guided models continue to outperform the base model at all oracle evaluation budgets. The performance trends with budget are provided in Figures A.10–A.13, but our analyses focus on a budget in hit identification of 1000 generated molecules and in lead optimization, where each oracle call is more expensive, of 400 generated molecules.

4.1. *De Novo* Generation

We first evaluate IIFT in the *de novo* generation setting, where the model is asked to shift the unconditional molecular distribution toward improved oracle values without conditioning on an input scaffold. This provides a direct test of whether importance-based fine-tuning can improve global sample quality while preserving chemical realism and diversity.

Table 2 illustrates the primary *de novo* generation results for Viable and **Tractable GenMol**, while Figure 2A illustrates the distribution shifts in the ADMET-based and synthesizability-based rewards. When fine-tuned on the ADMET reward, **Viable GenMol** substantially improves the mean reward from 0.431 to 0.765, while also improving mean synthetic accessibility from 2.887 to 2.348. Although the fraction of molecules with QED ≥ 0.55 decreases slightly from 0.976 to 0.903, diversity increases from 0.814 to 0.834. Thus, ADMET-guided IIFT shifts the generated distribution toward molecules with substantially better oracle values without collapsing diversity, and, in fact, modestly improves it.

When fine-tuned on the synthesizability reward, **Tractable GenMol** increases the mean reward from 1.893 to 2.234 and substantially improves mean SA from 2.887 to 2.159, again with only a small reduction in the QED success rate from 0.976 to 0.949. Diversity remains nearly unchanged, decreasing only slightly from 0.814 to 0.809.

Taken together, these results show that IIFT can effectively steer GenMol in *de novo* generation toward different reward objectives while largely preserving the desirable properties of the base generator. The relatively small changes in QED and diversity suggest that these gains are achieved through a meaningful reward-aligned distribution shift rather than through mode collapse or a loss of molecular variety. This addresses a known failure mode of reward-driven molecular optimization wherein reinforcement-learning-based generators can become trapped in local optima and repeatedly generate structurally similar, high-scoring molecules (Gummesson Svensson et al., 2025).

Table 2. De novo generation results. Reward is the average $\exp(r(x))$ across all molecules. QED is the fraction of molecules for which QED ≥ 0.55, SA is the synthetic accessibility score (Ertl & Schuffenhauer, 2009), and diversity is reported using the TDC Diversity evaluator (Huang et al., 2021).

Method	Reward (↑)	QED (↑)	SA (↓)	Diversity (↑)
ADMET Reward				
Base	0.431	**0.976**	2.887	0.814
Viable	**0.765**	0.903	**2.348**	**0.834**
Synthesizability Reward				
Base	1.893	**0.976**	2.887	**0.814**
Tractable	**2.234**	0.949	**2.159**	0.809

4.2. Hit Identification

GenMol hit identification proceeds through goal-directed fragment-based search under a limited budget of oracle evaluations, with the objective of discovering molecules that score highly under a task-specific target oracle. IIFT acts upstream of that search by biasing the underlying generative distribution toward ADMET favorability or retrosynthetic accessibility, with fragments and fragment combinations encountered during search drawn from a distribution shifted toward more promising regions of chemical space. We then compare the number of such hits produced by the guided and base models across tasks. Since our goal is to assess distri-

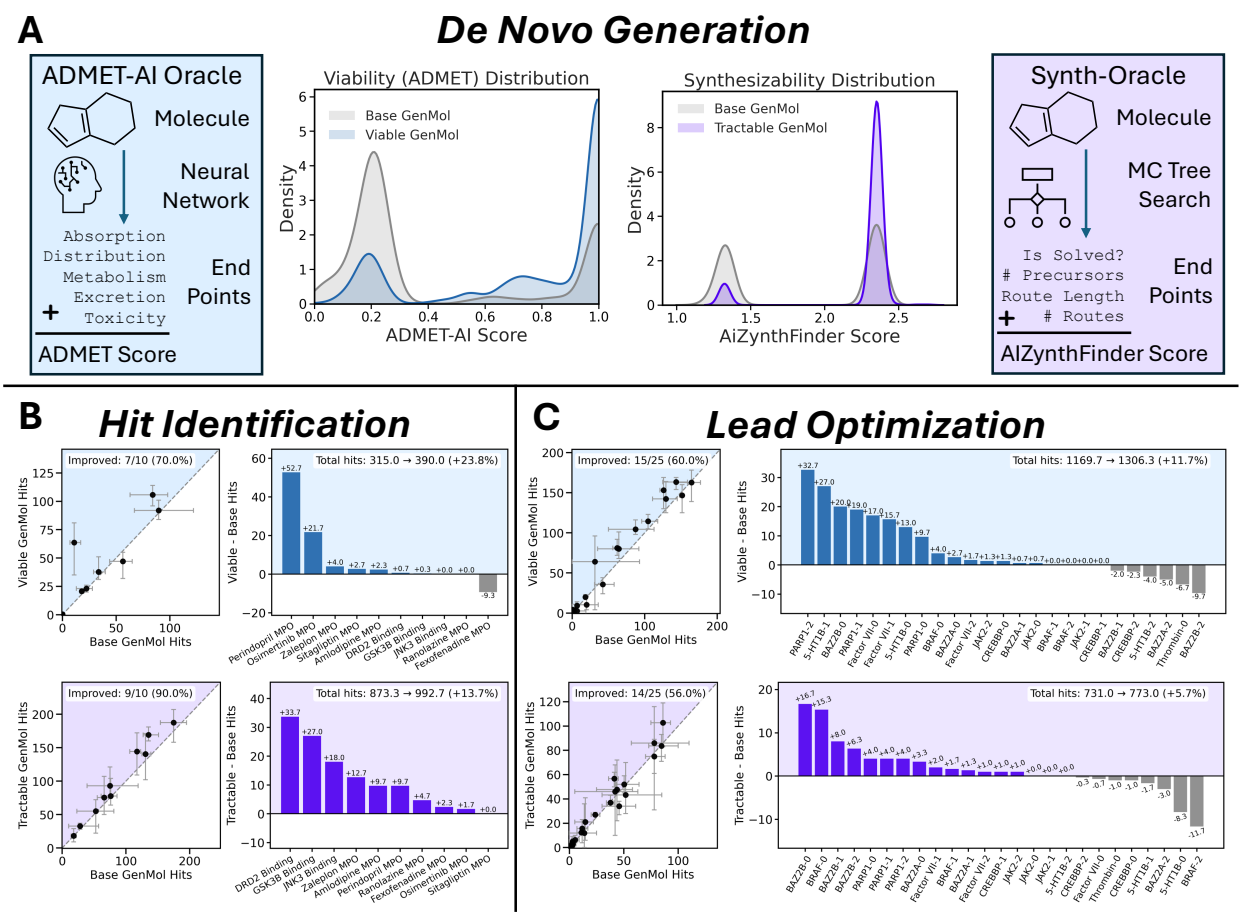

Figure 2. Multi-objective molecular optimization tasks comparing Viable and **Tractable GenMol** to the base model. (A) IIFT shifts the GenMol sampling distribution in *de novo* generation toward greater ADMET favorability or retrosynthetic tractability by using ADMET-AI and AiZynthFinder-based synthesizability oracles as reward functions. The ADMET- and synthesizability-based oracle workflows are illustrated in blue and purple, respectively. Additional training curves showing that high QED, favorable SA score, and molecular diversity are largely preserved are provided in Figure A.1. (B) In hit identification, both **Viable GenMol** and **Tractable GenMol** increase the number of hits relative to the base GenMol model across PMO targets, with gains summarized by parity plots and per-target hit differences. (C) In lead optimization, oracle-guided fine-tuning similarly improves hit recovery for the majority of compound/initial lead pairs while retaining the original GenMol lead-optimization workflow. In (B) and (C), points are reported as an average of three runs where the error bars depict the minimum and the maximum of these runs.

bution shifting in practically meaningful molecular-design settings, we focus here on the drug-discovery-relevant PMO tasks of multi-property optimization (MPO) objectives and DRD2, JNK3, and GSK3β oracles within the Therapeutic Data Commons (Huang et al., 2021). We report the remaining benchmarks in Figures A.6 and A.7 for completeness.

Figure 2B summarizes the hit identification comparisons. **Viable GenMol** improves hit counts in $7/10$ tasks and increases the total number of hits from an average of 315.0 to 390.0 ($+23.8\%$). **Tractable GenMol** improves hit counts in $9/10$ tasks and increases the total from an average of 873.3 to 992.7 ($+13.7\%$). The per-task delta plots show that the gains are heterogeneous: some tasks exhibit large improvements in hit rate, whereas a smaller number show little change or even slight degradation. We speculated that the slight degradation in hit rate relative to the base model may be due to a constriction of the viable molecular search space under the ADMET or synthesizability reward, but we were unable to find a metric that consistently explained the variation in change in hits across tasks. In practice, however, our empirical results establish that IIFT consistently improves performance across a majority of hit identification applications.

4.3. Lead Optimization

Lead optimization imposes a stricter and more practically consequential test than hit identification. Instead of searching broadly for strong molecules anywhere in chemical space, the model must improve a specific starting lead while

retaining a desired molecular scaffold. In our setting, as in the original GenMol work (Lee et al., 2025), the target oracle is the QuickVina2 score (Alhossary et al., 2015) for the target protein, and GenMol performs optimization by generating analogs from seed-derived fragments through iterative fragment remasking. Because the search is anchored to a particular starting molecule, different seeds for the same target can lead to very different optimization trajectories and accessible regions of chemical space, and we examined whether incorporating IIFT within each of these optimization trajectories can lead to elevated drug-like hits.

Figure 2C illustrates the lead optimization comparisons. **Viable GenMol** improves the hit count in $15/25$ lead instances, is unchanged in $4/25$, and increases the total number of hits from 1169.7 to 1306.3 ($+11.7\%$). **Tractable GenMol** improves hit counts in $14/25$ while remaining unchanged in $3/25$ instances, and the total number of hits improves slightly from 731.0 to 773.0 ($+5.7\%$). Again, a small number of applications exhibit a degradation in hit rate, but in the majority of tasks, IIFT significantly improves the hit rate for scaffold-aware optimization relevant to medicinal chemistry.

4.4. Fragment and binding analysis of synthetically-accessible lead candidates

We further examined lead-optimization candidates produced by **Tractable GenMol** to determine whether the synthesizability-oriented distribution shift yields chemically more realistic leads rather than merely changing oracle values. We focused on BAZ2A and BAZ2B as two targets anticipated to be amenable to relatively accessible experimental follow-up. We first analyzed the sets of lead-optimization hits for the presence of selected motifs that distinguish the starting leads and are associated with different apparent route complexity under retrosynthetic analysis. In particular, we compared the frequency of a motif that is difficult to find a retrosynthesis path for with one that is simple to synthesize among the top-50 docking hits returned by the two models. We selected the difficult motifs of the benzimidazoline aryl sulfonamide from an initial BAZ2A lead and the N-alkyl lactam from an initial BAZ2B lead and the simple motifs of the p-substituted phenyl aminomethyl group from the BAZ2A lead and the secondary benzamide from the BAZ2B lead. As shown in Figure 3A for BAZ2A and Figure 3B for BAZ2B, base GenMol hits will frequently contain the difficult-to-synthesize motif, whereas **Tractable GenMol** shifts toward the easy-to-synthesize motif. Specifically, **Tractable GenMol** reduces the incidence of the difficult benzimidazoline aryl sulfonamide among the top 50 hits from an initial BAZ2A lead from 37.3% to 25.3%, and the incidence of the difficult N-alkyl lactam from an initial BAZ2B lead from 46.0% to 35.3%. Correspondingly, the model elevates the incidence of the easy p-substituted phenyl aminomethyl from an initial BAZ2A lead from 36.7% to 55.3%, and the incidence of the easy secondary benzamide from an initial BAZ2B lead from 37.3% to 54.0%.

To move beyond motif-level analysis, we next isolated three lead-optimization candidates that were especially attractive from a synthetic standpoint since they were identified as being obtainable in a single synthetic step from readily available precursors (Figure 3C, Figure A.14). These candidates were then subjected to computationally expensive relative binding free energy (RBFE) calculations using OpenFE (Baumann et al., 2026) (Section A.4) as a more stringent physics-based evaluation of binding affinity than molecular docking (Shirts et al., 2007). As illustrated in Figure 3C, although all three candidates exhibited favorable docking improvements relative to the initial lead, RBFE revealed that only two of the three possessed negative $\Delta\Delta G_{\mathrm{RBFE}}$ values indicative of stronger binding affinity of the **Tractable GenMol** generated candidate relative to the initial lead molecule.

In follow-up experimental work, we are in the process of synthesizing the two candidates with favorable RBFE results (red star and blue star in Figure 3) from commercially available or readily prepared building blocks using the one-step synthetic routes and will be assayed against the corresponding BAZ2 target proteins for biological activity. More broadly, this prospective synthesis campaign will provide an initial experimental test of the success of IIFT reward-shifted molecular generation in reducing the gap between *in silico* optimization and practical medicinal chemistry by identifying candidates that satisfy both binding and synthesis constraints before committing experimental resources.

5. Conclusion

We applied Iterative Importance Fine-Tuning (IIFT) as a reward-based post-training method to shift the generative distribution of molecular candidates from GenMol to favor improved developability while preserving the strengths of the base discrete-diffusion generator. Specifically, we developed two fine-tuned variants of GenMol: **Viable GenMol**, which biases generation toward molecules with improved ADMET profiles, and **Tractable GenMol**, which promotes generation of molecules with improved retrosynthetic accessibility. Because IIFT only requires scalar reward evaluations, the same framework can incorporate black-box oracles such as ADMET-AI and AiZynthFinder without differentiating through the reward or performing reinforcement-learning-style policy optimization.

Across *de novo* generation, hit identification, and lead optimization, both guided models increased the number of practically useful hits relative to base GenMol. These gains were strongest in low-budget settings, where improving the sampling distribution is most valuable due to high time or labor

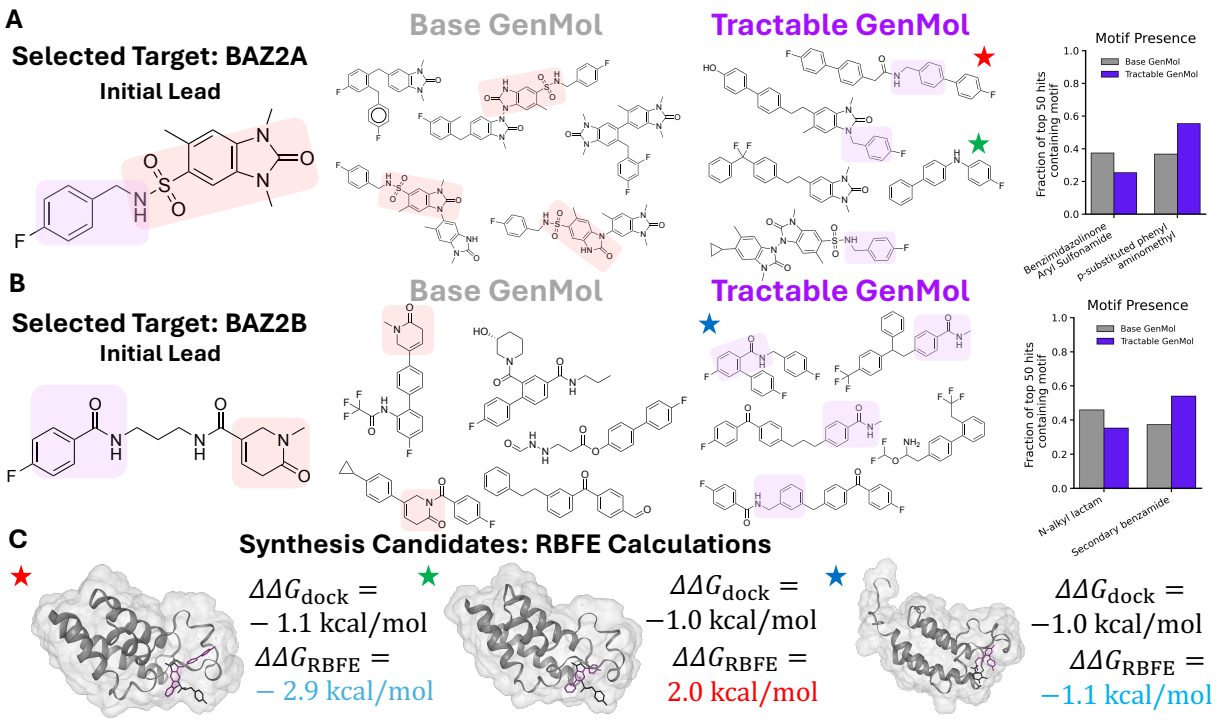

Figure 3. Motif-level analysis of **Tractable GenMol** in lead optimization. Representative top docking hits generated for BAZ2A (A) and BAZ2B (B), comparing base GenMol with **Tractable GenMol**. The initial lead structures are shown at left, with highlighted motifs used for tracking. Base GenMol-generated molecules are shown in the left-center, with shaded red regions indicating recurring difficult-to-synthesize motifs from the initial lead molecules (benzimidazoline aryl sulfonamide and N-alkyl lactams) reduced in quantity by **Tractable GenMol**. **Tractable GenMol**-generated molecules are shown in the right-center, with shaded purple regions indicating recurring tractable motifs from the initial lead molecules (p-substituted phenyl aminomethyl groups and secondary benzamides) enriched or preserved by **Tractable GenMol**. The bar plots on the right depict the frequency of the presence of these motifs within each set of generated molecules within the top-50 docking hits in each lead optimization run. (C) Three selected synthesis candidates from **Tractable GenMol** were further evaluated with RBFE calculations as a computational filter. For each candidate, the docking improvement relative to the initial lead, $\Delta\Delta G_{\mathrm{dock}}$, and the RBFE-estimated binding free energy change, $\Delta\Delta G_{\mathrm{RBFE}}$, are reported. Two candidates produce negative $\Delta\Delta G_{\mathrm{RBFE}}$ values indicating stronger affinity to the protein target for the **Tractable GenMol** candidate relative to the initial lead.

costs associated with oracle evaluations. Importantly, the improvements were not limited to the direct reward values: the guided models also increased the frequency of molecules passing chemistry-motivated liability filters, while largely maintaining drug-likeness, synthetic accessibility, and diversity. These results indicate that IIFT produces a meaningful distribution shift rather than simply exploiting a narrow oracle-specific failure mode.

We also showed that **Tractable GenMol** can alter the chemical character of lead-optimized candidates in a way that is consistent with practical synthesis considerations. In BAZ2A and BAZ2B case studies, the tractability-guided model reduced the frequency of challenging motifs, enriched more accessible motifs, and produced candidates prioritized for readily accessible synthesis and favorable relative binding free energy calculations. Together, these results suggest that importance-based post-training is a practical strategy for multi-objective molecular generation, enabling existing molecular diffusion models to be adapted toward downstream drug-discovery constraints without redesigning the generator or requiring differentiable objective functions.

Impact Statement

This work aims to advance machine learning methods for molecular design by improving the ability of generative models to account for developability constraints such as ADMET properties and synthetic accessibility. As with other molecular generation methods, potential misuse could involve the design of harmful compounds; however, our focus is on improving filtering, tractability, and candidate prioritization in drug-discovery workflows. We do not identify additional societal impacts beyond those broadly associated with machine learning for drug discovery.

References

Alehashem, M. S., Ariffin, A. B., Savage, P. B., Yehya Dabdawb, W. A., and Thomas, N. F. Treasures old and new: what we can learn regarding the macrocyclic problem from past and present efforts in natural product total synthesis. *RSC Adv.*, 10, 2020.

Alhossary, A., Handoko, S. D., Mu, Y., and Kwoh, C.-K. Fast, accurate, and reliable molecular docking with QuickVina 2. *Bioinformatics*, 31(13):2214–2216, 07 2015.

Baell, J. B. and Nissink, J. W. M. Seven year itch: Pan-assay interference compounds (PAINS) in 2017—utility and limitations. *ACS Chemical Biology*, 13(1):36–44, 2018.

Barnette, D. A., Schleiff, M. A., Datta, A., Flynn, N., Swamidass, S. J., and Miller, G. P. Meloxicam methyl group determines enzyme specificity for thiazole bioactivation compared to sudoxicam. *Toxicology Letters*, 338: 10–20, 2021.

Baumann, H. M., Horton, J. T., Henry, M. M., Travitz, A., Ries, B., Gowers, R. J., Swenson, D. W. H., Pulido, I., Rufa, D., Dotson, D. L., Bansal, N., Bluck, J. P., Broughton, H., Campbell, K., Cao, L., Frieg, B., Gapsys, V., Göddeke, H., Klähn, M., Lakkaraju, S. K., Linker, S. M., Löhr, T., Magarkar, A., Pérez-Conesa, S., Purkey, H. E., Saribekyan, H., Scheen, J., Schindler, C. E. M., Steinbrecher, T., Stern, C. D., Suriana, P., Swope, W. C., Tresadern, G., Tsidilkovski, L., Wei, B., Williams, A. H., Wu, Y., Zhang, I., Chodera, J. D., Eastwood, J. R. B., Mobley, D. L., and Alibay, I. Large-scale collaborative assessment of binding free energy calculations for drug discovery using openfe. *ChemRxiv*, 2026(0317), 2026.

Bengio, Y., Lahlou, S., Deleu, T., Hu, E. J., Tiwari, M., and Bengio, E. GFlowNet foundations, 2026.

Benigni, R. and Bossa, C. Mechanisms of chemical carcinogenicity and mutagenicity: A review with implications for predictive toxicology. *Chemical Reviews*, 111(4): 2507–2536, 2011.

Bian, Y. and Xie, X.-Q. Generative chemistry: drug discovery with deep learning generative models. *Journal of Molecular Modeling*, 27(3):71, Feb 2021.

Bickerton, G. R., Paolini, G. V., Besnard, J., Muresan, S., and Hopkins, A. L. Quantifying the chemical beauty of drugs. *Nature Chemistry*, 4(2):90–98, 2012.

Bilodeau, C., Jin, W., Jaakkola, T., Barzilay, R., and Jensen, K. F. Generative models for molecular discovery: Recent advances and challenges. *WIREs Computational Molecular Science*, 12(5):e1608, 2022.

Broccatelli, F., Carosati, E., Neri, A., Frosini, M., Goracci, L., Oprea, T. I., and Cruciani, G. A novel approach for predicting P-Glycoprotein (ABCB1) inhibition using molecular interaction fields. *Journal of Medicinal Chemistry*, 54(6):1740–1751, 2011.

Bruns, R. F. and Watson, I. A. Rules for identifying potentially reactive or promiscuous compounds. *Journal of Medicinal Chemistry*, 55(22):9763–9772, 11 2012.

Bussi, G. and Parrinello, M. Accurate sampling using langevin dynamics. *Phys. Rev. E*, 75:056707, May 2007.

Case, D. A., Aktulga, H. M., Belfon, K., Cerutti, D. S., Cisneros, G. A., Cruzeiro, V. W. D., Forouzesh, N., Giese, T. J., Götz, A. W., Gohlke, H., Izadi, S., Kasavajhala, K., Kaymak, M. C., King, E., Kurtzman, T., Lee, T.-S., Li, P., Liu, J., Luchko, T., Luo, R., Manathunga, M., Machado, M. R., Nguyen, H. M., O'Hearn, K. A., Onufriev, A. V., Pan, F., Pantano, S., Qi, R., Rahnamoun, A., Risheh, A., Schott-Verdugo, S., Shajan, A., Swails, J., Wang, J., Wei, H., Wu, X., Wu, Y., Zhang, S., Zhao, S., Zhu, Q., Cheatham, T. E. I., Roe, D. R., Roitberg, A., Simmerling, C., York, D. M., Nagan, M. C., and Merz, K. M. J. Ambertools. *Journal of Chemical Information and Modeling*, 63(20):6183–6191, 2023.

Cournia, Z., Allen, B., and Sherman, W. Relative binding free energy calculations in drug discovery: Recent advances and practical considerations. *Journal of Chemical Information and Modeling*, 57(12):2911–2937, 2017.

Darden, T., York, D., and Pedersen, L. Particle mesh Ewald: An N log(N) method for Ewald sums in large systems. *The Journal of Chemical Physics*, 98(12):10089–10092, 06 1993.

David, S. and Hamilton, J. P. Drug-induced liver injury. *US Gastroenterol. Hepatol. Rev.*, 6:73–80, January 2010.

De Cao, N. and Kipf, T. Molgan: An implicit generative model for small molecular graphs. *arXiv preprint arXiv:1805.11973*, 2018.

Denker, A., Padhy, S., Vargas, F., and Hertrich, J. Iterative importance fine-tuning of diffusion models. *arXiv preprint arXiv:2502.04468*, 2025.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.

Di, L., Keefer, C., Scott, D. O., Strelevitz, T. J., Chang, G., Bi, Y.-A., Lai, Y., Duckworth, J., Fenner, K., Troutman, M. D., and Obach, R. S. Mechanistic insights from comparing intrinsic clearance values between human liver microsomes and hepatocytes to guide drug design. *European Journal of Medicinal Chemistry*, 57:441–448, 2012.

Drouin, L., McGrath, S., Vidler, L. R., Chaikuad, A., Monteiro, O., Tallant, C., Philpott, M., Rogers, C., Fedorov, O., Liu, M., Akhtar, W., Hayes, A., Raynaud, F., Müller, S., Knapp, S., and Hoelder, S. Structure enabled design of BAZ2-ICR, a chemical probe targeting the bromodomains of BAZ2A and BAZ2B. *Journal of Medicinal Chemistry*, 58(5):2553–2559, 2015.

Eastman, P., Swails, J., Chodera, J. D., McGibbon, R. T., Zhao, Y., Beauchamp, K. A., Wang, L.-P., Simmonett, A. C., Harrigan, M. P., Stern, C. D., Wiewiora, R. P., Brooks, B. R., and Pande, V. S. OpenMM 7: Rapid development of high performance algorithms for molecular dynamics. *PLOS Computational Biology*, 13(7): e1005659, 2017.

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

Flam-Shepherd, D., Zhu, K., and Aspuru-Guzik, A. Language models can learn complex molecular distributions. *Nature Communications*, 13(1):3293, Jun 2022.

Fleck, M., Wieder, M., and Boresch, S. Dummy atoms in alchemical free energy calculations. *Journal of Chemical Theory and Computation*, 17(7):4403–4419, 2021.

Genheden, S., Thakkar, A., Chadimova, V., Reymond, J.-L., Engkvist, O., and Bjerrum, E. J. Aizynthfinder: A fast robust and flexible open-source software for retrosynthetic planning. *ChemRxiv*, 2020(0615), 2020.

Gomez, Y. K., Natale, A. M., Lincoff, J., Wolgemuth, C. W., Rosenberg, J. M., and Grabe, M. Taking the Monte-Carlo gamble: How not to buckle under the pressure! *Journal of Computational Chemistry*, 43(6):431–434, 2022.

Gómez-Bombarelli, R., Wei, J. N., Duvenaud, D., Hernández-Lobato, J. M., Sánchez-Lengeling, B., Sheberla, D., Aguilera-Iparraguirre, J., Hirzel, T. D., Adams, R. P., and Aspuru-Guzik, A. Automatic chemical design using a data-driven continuous representation of molecules. *ACS Central Science*, 4(2):268–276, 2018.

Gummesson Svensson, H., Tyrchan, C., Engkvist, O., and Haghir Chehreghani, M. Diversity-aware reinforcement learning for de novo drug design. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence*, IJCAI-2025, pp. 9194–9204. International Joint Conferences on Artificial Intelligence Organization, 2025.

Hoogeboom, E., Satorras, V. G., Vignac, C., and Welling, M. Equivariant diffusion for molecule generation in 3D, 2022.

Huang, K., Fu, T., Gao, W., Zhao, Y., Roohani, Y., Leskovec, J., Coley, C. W., Xiao, C., Sun, J., and Zitnik, M. Therapeutics data commons: Machine learning datasets and tasks for drug discovery and development, 2021.

Huang, R., Xia, M., Nguyen, D.-T., Zhao, T., Sakamuru, S., Zhao, J., Shahane, S. A., Rossoshek, A., and Simeonov, A. Tox21Challenge to build predictive models of nuclear receptor and stress response pathways as mediated by exposure to environmental chemicals and drugs. *Frontiers in Environmental Science*, Volume 3 - 2015, 2016.

Jakalian, A., Jack, D. B., and Bayly, C. I. Fast, efficient generation of high-quality atomic charges. AM1-BCC model: II. parameterization and validation. *Journal of Computational Chemistry*, 23(16):1623–1641, 2002.

Jin, W., Barzilay, R., and Jaakkola, T. Junction tree variational autoencoder for molecular graph generation. In *Proceedings of the 35th International Conference on Machine Learning*, pp. 2323–2332, 2018.

Kirkpatrick, P. and Ellis, C. Chemical space. *Nature*, 432 (7019):823–823, Dec 2004.

Lagunin, A. A., Dearden, J. C., Filimonov, D. A., and Poroikov, V. V. Computer-aided rodent carcinogenicity prediction. *Mutation Research/Genetic Toxicology and Environmental Mutagenesis*, 586(2):138–146, 2005.

Lamothe, S. M., Guo, J., Li, W., Yang, T., and Zhang, S. The human ether-a-go-go-related gene (hERG) potassium channel represents an unusual target for protease-mediated damage. *Journal of Biological Chemistry*, 291 (39):20387–20401, 2016.

Lee, S., Kreis, K., Veccham, S. P., Liu, M., Reidenbach, D., Peng, Y., Paliwal, S., Nie, W., and Vahdat, A. GenMol: A drug discovery generalist with discrete diffusion, 2025.

Li, J. and Wang, Z.-X. Nickel-catalyzed amination of aryl 2-pyridyl ethers via cleavage of the carbon–oxygen bond. *Organic Letters*, 19(14):3723–3726, 2017.

Lin, J. H. CYP induction-mediated drug interactions: in vitro assessment and clinical implications. *Pharmaceutical Research*, 23(6):1089–1116, Jun 2006.

Lin, X., Li, X., and Lin, X. A review on applications of computational methods in drug screening and design. *Molecules*, 25(6), 2020.

Lipinski, C. A. Lead- and drug-like compounds: the rule-of-five revolution. *Drug Discovery Today: Technologies*, 1(4):337–341, 2004.

Lo, Y.-C., Rensi, S. E., Torng, W., and Altman, R. B. Machine learning in chemoinformatics and drug discovery. *Drug Discovery Today*, 23(8):1538–1546, 2018.

Lombardo, F. and Jing, Y. In silico prediction of volume of distribution in humans. Extensive data set and the exploration of linear and nonlinear methods coupled with molecular interaction fields descriptors. *Journal of Chemical Information and Modeling*, 56(10):2042–2052, 2016.

Maier, J. A., Martinez, C., Kasavajhala, K., Wickstrom, L., Hauser, K. E., and Simmerling, C. ff14SB: Improving the accuracy of protein side chain and backbone parameters from ff99SB. *Journal of Chemical Theory and Computation*, 11(8):3696–3713, 2015.

Mark, P. and Nilsson, L. Structure and dynamics of the TIP3P, SPC, and SPC/E Water Models at 298 K. *The Journal of Physical Chemistry A*, 105(43):9954–9960, 2001.

Merz, K. M. J., De Fabritiis, G., and Wei, G.-W. Generative models for molecular design. *Journal of Chemical Information and Modeling*, 60(12):5635–5636, 2020.

Meyers, J., Fabian, B., and Brown, N. De novo molecular design and generative models. *Drug Discovery Today*, 26 (11):2707–2715, 2021.

Mitchell, J. B. O. Machine learning methods in chemoinformatics. *WIREs Computational Molecular Science*, 4(5): 468–481, 2014.

Niazi, S. K. and Mariam, Z. Recent advances in machine-learning-based chemoinformatics: A comprehensive review. *International Journal of Molecular Sciences*, 24 (14), 2023.

Noutahi, E., Gabellini, C., Craig, M., Lim, J. S. C., and Tossou, P. Gotta be SAFE: A new framework for molecular design, 2023.

Obach, R. S. Prediction of human clearance of twenty-nine drugs from hepatic microsomal intrinsic clearance data: An examination of in vitro half-life approach and nonspecific binding to microsomes. *Drug Metabolism and Disposition*, 27(11):1350–1359, 1999.

Oksendal, B. *Stochastic Differential Equations: An Introduction with Applications*. Springer, 6 edition, 2003.

Ononamadu, C. J. and Ibrahim, A. Molecular docking and prediction of ADME/drug-likeness properties of potentially active antidiabetic compounds isolated from aqueous-methanol extracts of Gymnema sylvestre and Combretum micranthum. *BioTechnologia*, 102(1):85–99, 2021.

Owais, W. and Kleinhofs, A. Metabolic activation of the mutagen azide in biological systems. *Mutation Research - Fundamental and Molecular Mechanisms of Mutagenesis*, 197(2):313–323, 1988.

Pang, C., Qiao, J., Zeng, X., Zou, Q., and Wei, L. Deep generative models in de novo drug molecule generation. *Journal of Chemical Information and Modeling*, 64(7): 2174–2194, 2024.

Papadatos, G., Davies, M., Dedman, N., Chambers, J., Gaulton, A., Siddle, J., Koks, R., Irvine, S. A., Pettersson, J., Goncharoff, N., Hersey, A., and Overington, J. P. SureChEMBL: a large-scale, chemically annotated patent document database. *Nucleic Acids Research*, 44(D1): D1220–D1228, 01 2016.

Park, J., Ahn, J., Choi, J., and Kim, J. Mol-air: Molecular reinforcement learning with adaptive intrinsic rewards for goal-directed molecular generation. *Journal of Chemical Information and Modeling*, 65(5):2283–2296, 2025.

Patel, L., Shukla, T., Huang, X., Ussery, D. W., and Wang, S. Machine learning methods in drug discovery. *Molecules*, 25(22), 2020.

Paul, D., Sanap, G., Shenoy, S., Kalyane, D., Kalia, K., and Tekade, R. K. Artificial intelligence in drug discovery and development. *Drug Discovery Today*, 26(1):80–93, 2021.

Rappe, A. K., Casewit, C. J., Colwell, K. S., Goddard III, W. A., and Skiff, W. M. Uff, a full periodic table force field for molecular mechanics and molecular dynamics simulations. *Journal of the American Chemical Society*, 114(25):10024–10035, Dec 1992.

RDKit. https://www.rdkit.org.

Reymond, J.-L. Chemical space as a unifying theme for chemistry. *Journal of Cheminformatics*, 17(1):6, Jan 2025.

Riniker, S. and Landrum, G. A. Better informed distance geometry: Using what we know to improve conformation generation. *Journal of Chemical Information and Modeling*, 55(12):2562–2574, 2015.

Saigiridharan, L., Hassen, A. K., Lai, H., Torren-Peraire, P., Engkvist, O., and Genheden, S. AiZynthFinder 4.0: developments based on learnings from 3 years of industrial application. *Journal of Cheminformatics*, 16(1):57, May 2024.

Samanta, B., De, A., Jana, G., Chattaraj, P. K., Ganguly, N., and Gomez-Rodriguez, M. NeVAE: A deep generative model for molecular graphs, 2019.

Schmidt, R., Klein, R., and Rarey, M. Maximum common substructure searching in combinatorial make-on-demand compound spaces. *Journal of Chemical Information and Modeling*, 62(9):2133–2150, 2022.

Sharma, B., Chenthamarakshan, V., Dhurandhar, A., Pereira, S., Hendler, J. A., Dordick, J. S., and Das, P. Accurate clinical toxicity prediction using multi-task deep neural nets and contrastive molecular explanations. *Scientific Reports*, 13(1):4908, Mar 2023.

Shirts, M. R. and Chodera, J. D. Statistically optimal analysis of samples from multiple equilibrium states. *The Journal of Chemical Physics*, 129(12):124105, 09 2008.

Shirts, M. R., Mobley, D. L., and Chodera, J. D. Chapter 4 alchemical free energy calculations: Ready for prime time? In Spellmeyer, D. and Wheeler, R. (eds.), *Annual Reports in Computational Chemistry*, volume 3, pp. 41–59. Elsevier, 2007.

Southey, M. W. Y. and Brunavs, M. Introduction to small molecule drug discovery and preclinical development. *Frontiers in Drug Discovery*, Volume 3 - 2023, 2023.

Springer, C. and Sokolnicki, K. L. A fingerprint pair analysis of hERG inhibition data. *Chemistry Central Journal*, 7 (1):167, 2013.

Sutton, R. S., McAllester, D., Singh, S., and Mansour, Y. Policy gradient methods for reinforcement learning with function approximation. In *Advances in Neural Information Processing Systems*, 1999.

Swanson, K., Walther, P., Leitz, J., Mukherjee, S., Wu, J. C., Shivnaraine, R. V., and Zou, J. ADMET-AI: a machine learning ADMET platform for evaluation of large-scale chemical libraries. *Bioinformatics*, 40(7):btae416, 06 2024.

Tian, M., Peng, Y., and Zheng, J. Metabolic activation and hepatotoxicity of furan-containing compounds. *Drug Metabolism and Disposition*, 50(5):655–670, 2022.

Vignac, C., Krawczuk, I., Siraudin, A., Wang, B., Cevher, V., and Frossard, P. DiGress: Discrete denoising diffusion for graph generation, 2023.

Vijay, U., Gupta, S., Mathur, P., Suravajhala, P., and Bhatnagar, P. Microbial mutagenicity assay: Ames test. *Bio-protocol*, 8(6):e2763, Mar 2018.

Walters, W. and Murcko, M. A. Prediction of 'drug-likeness'. *Advanced Drug Delivery Reviews*, 54(3):255–271, 2002.

Wang, L., Behara, P. K., Thompson, M. W., Gokey, T., Wang, Y., Wagner, J. R., Cole, D. J., Gilson, M. K., Shirts, M. R., and Mobley, D. L. The open force field initiative: Open software and open science for molecular modeling. *The Journal of Physical Chemistry B*, 128(29): 7043–7067, 2024.

Wang, N.-N., Dong, J., Deng, Y.-H., Zhu, M.-F., Wen, M., Yao, Z.-J., Lu, A.-P., Wang, J.-B., and Cao, D.-S. ADME properties evaluation in drug discovery: Prediction of Caco-2 cell permeability using a combination of NSGA-II and boosting. *Journal of Chemical Information and Modeling*, 56(4):763–773, 2016.

Weitz, J. Factor Xa or thrombin: Is thrombin a better target? *Journal of Thrombosis and Haemostasis*, 5:65–67, 2007.

Wenlock, M. and Tomkinson, N. Experimental in vitro dmpk and physicochemical data on a set of publicly disclosed compounds, 2015. ChEMBL document CHEMBL3301361.

Xu, M., Unzue, A., Dong, J., Spiliotopoulos, D., Nevado, C., and Caflisch, A. Discovery of CREBBP bromodomain inhibitors by high-throughput docking and hit optimization guided by molecular dynamics. *Journal of Medicinal Chemistry*, 59(4):1340–1349, 2016.

Xu, M., Yu, L., Song, Y., Shi, C., Ermon, S., and Tang, J. GeoDiff: a geometric diffusion model for molecular conformation generation, 2022.

Xu, P., Feng, T., Fu, T., Laghuvarapu, S., and Sun, J. Molecular de novo design through transformer-based reinforcement learning, 2024.

Yang, Y., Hsieh, C.-Y., Kang, Y., Hou, T., Liu, H., and Yao, X. Deep generation model guided by the docking score for active molecular design. *Journal of Chemical Information and Modeling*, 63(10):2983–2991, 2023.

Zhang, Z., Liu, X., Yan, K., Tuckerman, M. E., and Liu, J. Unified efficient thermostat scheme for the canonical ensemble with holonomic or isokinetic constraints via molecular dynamics. *The Journal of Physical Chemistry A*, 123(28):6056–6079, 2019.

A. Appendix

A.1. Reward Specifications

A.1.1. Viable GenMol: ADMET-AI Reward

Our ADMET reward is a composite oracle built from endpoint-level predictions returned by ADMET-AI (Swanson et al., 2024). The purpose of this reward is not to optimize any single ADMET endpoint in isolation, but to bias generation toward molecules that are simultaneously reasonable across several aspects of developability. To do this, we convert heterogeneous endpoint predictions into a common desirability scale and then aggregate them into a single scalar reward. Each endpoint is first mapped to a normalized desirability score in $[0, 1]$ (see Table A.1), where larger values indicate that the prediction lies closer to the preferred regime for that endpoint. The precise transformation depends on the endpoint type, but the principle is always the same: the raw prediction is converted into a score measuring how favorable that value is from the standpoint of developability. For endpoints already expressed on a bounded probability-like scale, this is done directly or through a simple monotonic transformation. For continuous physicochemical and pharmacokinetic descriptors, we use the percentile fields provided by ADMET-AI when available and map them to desirability scores by measuring how close they are to a reference region within the approved-drug distribution. This places all endpoints on a shared scale before aggregation.

We organize the resulting desirability terms into four pillars: *developability*, *exposure*, *metabolism*, and *safety*. The developability pillar combines the quantitative estimate of drug-likeness (QED), medicinal-chemistry alert indicators, Lipinski-style drug-likeness terms, and selected physicochemical descriptors such as molecular weight, logP, hydrogen-bond donors and acceptors, topological polar surface area (TPSA), and stereochemical complexity. The exposure pillar combines descriptors related to absorption, permeability, solubility, clearance, half-life, plasma protein binding, and volume of distribution. The metabolism pillar collects Cytochrome P450 (CYP)-related interaction and substrate endpoints. The safety pillar aggregates predicted toxicological and related liability signals, including explicit toxicity endpoints and broader assay-panel outputs when available.

Within each pillar, the endpoint-level desirabilities are combined using a geometric mean. If $\{d_i(x)\}$ denotes the set of desirability scores assigned to molecule x for the endpoints in a given pillar, the corresponding pillar score is

$$s(x) = \exp\left(\frac{1}{N_{\text{endpoints}}} \sum_{\text{endpoints } i} \log\big(\max(\varepsilon, d_i(x))\big)\right),$$

where $\varepsilon = 10^{-6}$ avoids numerical issues at zero. We use a geometric mean rather than an arithmetic mean because it is conservative: a severe deficiency on one endpoint cannot be fully compensated for by strong performance on others. This is appropriate for ADMET optimization, where a molecule that is excellent on several properties but clearly poor on one critical property is usually not desirable.

The four pillar scores are then aggregated into a single base ADMET score using a weighted geometric mean,

$$s_{\text{base}}(x) = \exp\left(\frac{w_{\text{dev}} \log s_{\text{dev}}(x) + w_{\text{exp}} \log s_{\text{exp}}(x) + w_{\text{met}} \log s_{\text{met}}(x) + w_{\text{safe}} \log s_{\text{safe}}(x)}{w_{\text{dev}} + w_{\text{exp}} + w_{\text{met}} + w_{\text{safe}}}\right), \tag{1}$$

with weights

$$w_{\text{dev}} = 0.30, \qquad w_{\text{exp}} = 0.30, \qquad w_{\text{met}} = 0.15, \qquad w_{\text{safe}} = 0.25.$$

These values place somewhat greater emphasis on overall developability, exposure, and safety than on metabolism, while still requiring all four pillars to remain acceptable.

After computing the base score, we apply an additional multiplicative penalty for a small set of endpoints that we treat as pathological, namely the Ames mutagenicity test (Ames), human Ether-à-go-go-Related Gene (hERG) activity, and drug-induced liver injury (DILI). If the predicted value for any of these exceeds a predefined threshold, the score is multiplied by a penalty factor. If multiple such endpoints exceed their thresholds, the penalties compound multiplicatively. Denoting the resulting score by $s_{\text{ADMET}}(x)$, we finally convert it into the reward used by IIFT,

$$r_{\text{ADMET}}(x) = \alpha \min\left(0,\, \log \frac{s_{\text{ADMET}}(x)}{s^*}\right), \tag{2}$$

where $s^* = 0.2$ and $\alpha = 1.0$. This transformation recenters the reward so that molecules near the desired ADMET regime receive reward close to zero, molecules below that regime receive increasingly negative reward, and molecules above it

are not rewarded further. In this sense, the oracle is designed to raise the floor of developability rather than to encourage unbounded optimization of the composite score.

A.1.2. TRACTABLE GENMOL: AIZYNTHFINDER REWARD

Our synthesizability reward is a composite oracle built from retrosynthetic planning outputs returned by AiZynthFinder (Genheden et al., 2020; Saigiridharan et al., 2024). The purpose of this reward is not merely to assign a heuristic notion of synthetic ease to a molecule, but to bias generation toward molecules for which explicit retrosynthetic routes can be found under a fixed planning setup. To do this, we run AiZynthFinder on each candidate molecule and convert the resulting search statistics into a bounded scalar score that reflects route existence, precursor availability, route multiplicity, planner confidence, and route length.

For each molecule x, AiZynthFinder performs retrosynthetic search under a fixed configuration, stock, and policy setup. In our implementation, the stock is `zinc`, the expansion policy is `uspto`, and the filter policy is `uspto`. The planner returns a collection of summary statistics describing the search outcome, including whether any solved route was found, how many leaf precursors appear in stock, how many solved routes were identified, the top route score, and the number of retrosynthetic steps in the returned solution. We combine these quantities into a single synthesizability score

$$\begin{aligned} s_{\text{synth}}(x) = w_{\text{solved}}\,\mathbf{1}\{\text{solved}\} + w_{\text{stock}}\,f_{\text{stock}}(x) \\ + w_{\text{routes}}\,f_{\text{routes}}(x) + w_{\text{score}}\,f_{\text{top}}(x) \\ + w_{\text{steps}}\,f_{\text{steps}}(x), \end{aligned} \tag{3}$$

where each term lies in $[0, 1]$, so that the final score is also bounded in $[0, 1]$.

The first term, $\mathbf{1}\{\text{solved}\}$, is an indicator for whether AiZynthFinder found at least one solved retrosynthetic route. This is the strongest signal in the score and captures the most basic notion of tractability: whether the planner can identify a route to purchasable precursors at all. The second term,

$$f_{\text{stock}}(x) = \frac{n_{\text{in\_stock}}(x)}{n_{\text{precursors}}(x)},$$

measures the fraction of leaf precursors that are found in stock. This rewards molecules whose proposed routes terminate in more readily available building blocks. The third term,

$$f_{\text{routes}}(x) = \frac{\min\big(n_{\text{solved\_routes}}(x), C_{\text{routes}}\big)}{C_{\text{routes}}},$$

is a normalized measure of route multiplicity, where $n_{\text{solved\_routes}}(x)$ is the number of solved routes returned by the planner and C_{routes} is a fixed cap. In our implementation, $C_{\text{routes}} = 10$. This term favors molecules for which the planner can identify multiple feasible retrosynthetic decompositions, but prevents this component from dominating once the number of solved routes becomes large.

The fourth term, $f_{\text{top}}(x)$, is the top route score reported by AiZynthFinder. We use this quantity directly as returned by the planner. Its precise interpretation is inherited from AiZynthFinder, but in our oracle it plays the role of a route-quality or planner-confidence term. The fifth term,

$$f_{\text{steps}}(x) = 1 - \frac{\min\big(n_{\text{steps}}(x), C_{\text{steps}}\big)}{C_{\text{steps}}},$$

encodes a preference for shorter routes, where $n_{\text{steps}}(x)$ is the number of retrosynthetic steps and C_{steps} is a fixed cap. In our implementation, $C_{\text{steps}} = 8$. This term assigns larger values to shorter routes and decays linearly until the cap is reached, after which additional steps are not penalized further.

The default weights are

$$w_{\text{solved}} = 0.45, \qquad w_{\text{stock}} = 0.20, \qquad w_{\text{routes}} = 0.15, \qquad w_{\text{score}} = 0.15, \qquad w_{\text{steps}} = 0.05.$$

These weights place the greatest emphasis on the existence of at least one solved route, followed by precursor availability, with somewhat smaller contributions from route multiplicity and the planner's top route score, and the smallest contribution from the route-length preference. This reflects the intended use of the oracle: the most important distinction is whether a plausible synthetic route exists at all, while secondary terms help rank molecules within the set of route-feasible candidates.

A.2. IIFT Specifications

A.2.1. Training

For both **Viable GenMol** and **Tractable GenMol**, IIFT is implemented as replay-buffer fine-tuning of a guidance head on top of a frozen GenMol backbone. At each outer iteration, the sampler generates a batch of molecules from a masked prompt, evaluates valid molecules with the selected reward oracle, computes reward-dependent importance scores, adaptively resamples the batch, and appends the accepted final token sequences to a fixed-size FIFO replay buffer. The model is then updated by drawing minibatches uniformly from this buffer and applying the standard masked discrete diffusion loss to the guidance-modified logits, while the backbone remains frozen.

The resampling score is computed as,

$$\log w(x) = \frac{r(x)}{T} + \log \rho(x),$$

where $r(x)$ is the oracle reward, T is the reward temperature, and $\log \rho(x)$ is the sampler-provided discrete trajectory log-ratio. Invalid molecules are assigned reward $-\infty$ and are excluded from the valid resampling pool.

The two IIFT variants differ primarily in the strength and duration of post-training. **Viable GenMol** uses a lower learning rate, more outer iterations, and substantially more inner updates per outer iteration, while **Tractable GenMol** uses a higher learning rate, fewer outer iterations, and a lower reward temperature. This reflects the higher computational cost of retrosynthesis-based scoring relative to ADMET scoring and makes the AiZynthFinder-aligned run more efficient. Additional training parameters are reported in Table A.2.

A.2.2. Inference: de novo generation and hit optimization

Inference in *de novo* generation and hit identification is done identically to the GenMol base model, with the model weights now including the IIFT-controlled guidance head. In *de novo* generation, the softmax temperature of 0.5 and randomness parameter of 0.5 are used. For practical molecular optimization (PMO) hit identification, each run comprises 100 molecules per PMO iteration until reaching 1,000 target-oracle calls. The fragment remasking parameter γ is set to zero to avoid "warmup" loops without any model inference calls. Additionally, in PMO, the fragment population is updated using a control-aware score,

$$s_{\text{pop}}(x) = s_{\text{target}}(x) \exp(0.25\, r(x)),$$

rather than the target oracle alone, so fragments from molecules that are both target-active and control-favorable are preferentially retained; we apply this change to both the reward-guided and base model we compare against.

A.2.3. "Hybrid" IIFT: lead optimization

For lead optimization, we use an on-the-fly hybrid IIFT procedure in which GenMol performs its standard fragment-based lead-optimization loop while simultaneously fine-tuning the guidance head on high-control-reward candidates. Each run starts from the same GenMol checkpoint and a target-specific active molecule selected by the starting molecule index. At each iteration, GenMol samples 100 candidate molecules by attaching fragments from the current population and applying fragment remasking. Candidates are scored by docking, QED, synthetic accessibility, similarity to the starting molecule, and the ADMET control oracle.

The lead-optimization population is updated only with candidates that pass the lead filters. Accepted lead candidates must improve over the starting docking score, satisfy QED ≥ 0.6, SA ≤ 4, Tanimoto similarity ≥ 0.4, and pass the control threshold described in Table A.3. In parallel, hybrid IIFT uses the ADMET control reward to resample generated molecules for training. The retained molecules are tokenized, appended to a fixed-size FIFO replay buffer, and used to update only the guidance head with the standard GenMol masked diffusion loss while keeping the backbone frozen.

The hybrid IIFT update uses reward-temperature scaling with $T_{\text{hybrid}} = 1.0$, adaptive resampling with effective sample size threshold 0.95, and a replay buffer of 256 tokenized molecules. Each lead-optimization iteration performs 64 inner guidance-head updates with batch size 64 and learning rate 10^{-5}.

Motif	Problem / Liability	Source
N-Nitroso	Ames mutagenicity; metabolic activation to DNA-reactive diazonium ions.	(Benigni & Bossa, 2011)
Azide	Potent mutagenicity risk and potential for chemical explosivity.	(Owais & Kleinhofs, 1988)
Lipophilic Amine	hERG K^+ channel inhibition	(Springer & Sokolnicki, 2013)
Furan Ring	DILI (liver injury); oxidized by P450s to reactive cis-enedials.	(Tian et al., 2022)
Thiazole	Risk of metabolic ring-opening to reactive thioacyl species.	(Barnette et al., 2021)

A.3. Chemical Motif Filters

A.3.1. Viable GenMol: Ames, hERG, DILI liabilties

A.3.2. Tractable GenMol: Problematic Motifs

Motif	Problem / Liability	Source
Acid Chloride	Unstable, will form intermediates	(Bruns & Watson, 2012)
Aldehyde	High reactivity with nucleophiles	(Walters & Murcko, 2002)
N–C–N pattern	Unstable, falls apart during storage or assay	(Bruns & Watson, 2012)
O–C–N pattern	Unstable, falls apart during storage or assay	(Bruns & Watson, 2012)
Anhydride	Very reactive	(Papadatos et al., 2016)
Cycle of ≥ 8 atoms	Entropic cost of macrocyclization	(Alehashem et al., 2020)
≥ 6 nitrogens	Anticipated challenges in route design	Experimentalist Feedback
Adjacent nitrogens	Anticipated challenges in route design	Experimentalist Feedback
Isocyanate	Reactive electrophile	(Papadatos et al., 2016)
Sulfonyl chloride	Reactive electrophile	(Papadatos et al., 2016)

A.4. Relative Binding Free Energy (RBFE) Calculations

We used relative binding free energy (RBFE) calculations to estimate changes in binding affinity between a reference ligand A and a generated candidate ligand B (Cournia et al., 2017). Directly computing an absolute binding free energy for each ligand would require estimating the reversible work of physically transferring each molecule from bulk solvent into the protein binding site, which is expensive and often inefficient. RBFE instead computes the binding free energy difference through an alchemical thermodynamic cycle in which a molecular-dynamics potential energy function is modified by a coupling parameter λ so that the simulated ligand is gradually transformed from ligand A into ligand B. At $\lambda = 0$, the Hamiltonian corresponds to the system containing ligand A; at $\lambda = 1$, it corresponds to the system containing ligand B; and at intermediate λ values, the force-field parameters, nonbonded interactions, and bonded terms are interpolated through a hybrid topology that contains atoms from both endpoint ligands. The same alchemical transformation is performed in two environments: once for the ligand in aqueous solution and once for the ligand bound to the protein. Because free energy is a state function, the difference between these two nonphysical alchemical transformation free energies is equal to the physical relative binding free energy, as depicted in the rightmost panel of Figure 1,

$$\begin{aligned}
\Delta\Delta G_{\text{bind}}(A \to B) &= \Delta G_{\text{complex}}(A \to B) - \Delta G_{\text{solvent}}(A \to B) && (4)\\
&= (G^B_{\text{complex}} - G^A_{\text{complex}}) - (G^B_{\text{solvent}} - G^A_{\text{solvent}}) && (5)\\
&= (G^B_{\text{complex}} - G^B_{\text{solvent}}) - (G^A_{\text{complex}} - G^A_{\text{solvent}}) && (6)\\
&= \Delta G^B - \Delta G^A, && (7)
\end{aligned}$$

where, $\Delta G_{\text{complex}}(A \to B) = G^B_{\text{complex}} - G^A_{\text{complex}}$ is the free energy change for transforming A into B in the protein-bound state, $\Delta G_{\text{solvent}}(A \to B) = G^B_{\text{solvent}} - G^A_{\text{solvent}}$ is the corresponding transformation free energy in solvent, $\Delta G^A = G^A_{\text{complex}} - G^A_{\text{solvent}}$ is the free energy of binding of A, and $\Delta G^B = G^B_{\text{complex}} - G^B_{\text{solvent}}$ is the free energy of binding of B. With this convention, negative values of $\Delta\Delta G_{\text{bind}}$ indicate that the generated ligand B is predicted to bind more favorably than the reference ligand A.

RBFE calculations were performed using the OpenFE software suite and the OpenFE relative hybrid topology protocol (Baumann et al., 2026; Fleck et al., 2021). Protein structures were obtained from experimentally resolved PDB entries and prepared by adding missing atoms and hydrogens using OpenMM's `PDBFixer` (Eastman et al., 2017). Candidate ligands were generated from SMILES strings using RDKit (RDKit). Hydrogens were added, three-dimensional conformers were

generated using ETKDGv3, and the resulting conformers were optimized with the UFF force field (Riniker & Landrum, 2015; Rappe et al., 1992). Each candidate ligand was then placed into the binding site relative to the corresponding reference ligand.

For each ligand pair, the maximum common substructure (MCS) between the reference and candidate ligands was identified using RDKit with element and bond-order matching, ring-only ring matching, and complete-ring matching (Schmidt et al., 2022). Candidate conformers were generated with ETKDGv3, aligned to the reference ligand using the rigid ring-containing portion of the MCS, and filtered for physically unreasonable geometries and protein–ligand clashes. We rejected candidate placements with heavy-atom nonbonded overlaps, minimum protein–ligand heavy-atom distance below 1.0 Å, or rigid common substructure RMSD greater than 0.5 Å.

After placement, partial charges were assigned once before transformation construction using AM1-BCC charges with the AmberTools backend through the OpenFF/OpenFE charge-generation interface (Jakalian et al., 2002; Case et al., 2023; Wang et al., 2024). Ligand atom mappings were constructed from the RDKit MCS correspondence (RDKit). Two chemical systems were then built for each transformation: a solvent system containing the ligand and explicit solvent, and a complex system containing the ligand, explicit solvent, and protein.

For each ligand pair $A \to B$, we used a hybrid topology representation in which atoms shared between the two ligands are mapped onto one another and non-shared atoms are alchemically introduced or removed. The alchemical path is controlled by a coupling parameter $\lambda \in [0, 1]$, where $\lambda = 0$ corresponds to the reference ligand A and $\lambda = 1$ corresponds to the generated ligand B. We simulated 16 λ windows for both the solvent and complex legs, with one replica initialized at each window. The windows were evenly spaced between 0 and 1, corresponding to $\lambda \in \left\{0, \frac{1}{15}, \frac{2}{15}, \ldots, 1\right\}$. The solvent and complex legs were constructed as separate OpenFE transformations and executed independently.

Molecular dynamics simulations were performed with OpenMM (Eastman et al., 2017) through the OpenFE relative hybrid topology protocol (Baumann et al., 2026). Unless otherwise specified, we used the default settings of the OpenFE protocol. The ligand was parameterized with OpenFF-2.1.1 (Wang et al., 2024), the protein with Amberff14SB (Maier et al., 2015), and water molecules with the TIP3P water model (Mark & Nilsson, 2001). Systems were solvated in explicit solvent under periodic boundary conditions with 1.5 nm box padding in a dodecahedral box, neutralizing counterions, and Na^+ and Cl^- added to an ionic concentration of 0.15 M. Long-range electrostatics were treated using particle mesh Ewald (PME) (Darden et al., 1993), and Lennard-Jones interactions were truncated at 1.0 nm. Hydrogen mass repartitioning was applied using a hydrogen mass of 3.0 amu. Temperature was maintained at 298.15 K using a Langevin thermostat/integrator with the LFMiddle discretization (Bussi & Parrinello, 2007; Zhang et al., 2019), and pressure was maintained at 1.0 bar using a Monte Carlo Barostat barostat (Gomez et al., 2022). Each λ state was energy minimized for 5,000 steps, equilibrated in the NPT ensemble for 1.0 ns, and sampled in the NPT ensemble for 5.0 ns with an integration timestep of 2.0 fs. Each solvent and complex leg was run with three independent OpenFE protocol repeats. Each repeat contained the full set of 16 λ windows, and the resulting repeat-level estimates were combined by the OpenFE `gather` procedure (Baumann et al., 2026) to obtain a single free energy estimate. Simulations were run on a single GPU running CUDA 12.2.

For each leg, samples from all λ windows were analyzed together using MBAR (Shirts & Chodera, 2008) to compute $\Delta G_{\text{solvent}}(A \to B)$ and $\Delta G_{\text{complex}}(A \to B)$.

A.5. Supplementary Tables

Table A.1. Endpoint-level transformations used in the composite ADMET reward. Each raw endpoint prediction is converted to a desirability score $d \in [0, 1]$. Here p denotes the clipped endpoint value in $[0, 1]$, and q denotes the percentile of the predicted endpoint value relative to the approved-drug reference distribution provided by ADMET-AI, rescaled from $[0, 100]$ to $[0, 1]$.

Endpoint(s)	**Pillar**	**Desirability score**	**Comments**
QED	Developability	$d = p$	Directly uses the predicted QED value. (Bickerton et al., 2012)
Lipinski	Developability	$d = p/5$	Normalized by the maximum Lipinski count of 5. (Lipinski, 2004)
PAINS alert, BRENK alert, NIH alert	Developability	$d = 1 - p$	Penalizes the presence of medicinal-chemistry alerts (Baell & Nissink, 2018; Ononamadu & Ibrahim, 2021; RDKit).
molecular weight, logP, hydrogen bond acceptors, hydrogen bond donors, tpsa, stereo centers	Developability	$d = 1 - \frac{\lvert q-0.5 \rvert}{0.5}$	Uses the approved-drug percentile field when available; values closer to the middle of the approved-drug distribution receive larger desirability (RDKit).
HIA Hou, Bioavailability Ma, PAMPA NCATS	Exposure	$d = p$	Directly uses the predicted endpoint value.
Clearance Hepatocyte AZ, Clearance Microsome AZ, Half Life Obach, PPBR AZ, VDss Lombardo, Caco2 Wang, Solubility AqSolDB	Exposure	$d = 1 - \frac{\lvert q-0.5 \rvert}{0.5}$	Uses the approved-drug percentile field when available; values closer to the middle of the approved-drug distribution receive larger desirability. (Lombardo & Jing, 2016; Wang et al., 2016; Wenlock & Tomkinson, 2015; Obach, 1999; Di et al., 2012)
CYP1A2 Veith, CYP2C19 Veith, CYP2C9 Veith, CYP2D6 Veith, CYP3A4 Veith, CYP2C9 Substrate CarbonMangels, CYP2D6 Substrate CarbonMangels, CYP3A4 Substrate CarbonMangels	Metabolism	$d = 1 - p$	Penalizes predicted CYP-related liabilities (Lin, 2006).
Ames, DILI, ClinTox, Carcinogens Lagunin, hERG, Skin Reaction, Pgp Broccatelli	Safety	$d = 1 - p$	Penalizes predicted safety-related liabilities (Vijay et al., 2018; David & Hamilton, 2010; Lamothe et al., 2016; Broccatelli et al., 2011; Lagunin et al., 2005; Sharma et al., 2023).
`NR-` or `SR-` endpoints	Safety	$d = 1 - p$	Tox21-style assay outputs for nuclear receptor (`NR-`) and stress-response (`SR-`) pathways (Huang et al., 2016).

Table A.2. IIFT training hyperparameters for **Viable GenMol** and **Tractable GenMol**. Both variants use the same frozen GenMol backbone and train only the guidance head using accepted samples stored in a FIFO replay buffer.

Hyperparameter	**Viable GenMol (ADMET-AI)**	**Tractable GenMol (AiZynthFinder)**
Guidance hidden dimension	512	512
Guidance time embedding dimension	64	64
Guidance dropout	0.1	0.1
Max num. epochs	200	50
Generated batch size per iteration	64	64
Minimum added mask length	40	40
Replay buffer size	256	256
Replay buffer replacement	FIFO	FIFO
Inner updates per epoch	512	60
Inner batch size	128	128
IIFT learning rate	1×10^{-5}	1×10^{-4}
Optimizer	AdamW	AdamW
Learning-rate schedule	Cosine decay	Cosine decay
Minimum scheduler LR	$\mathrm{lr}/100$	$\mathrm{lr}/100$
Gradient clipping	Enabled	Enabled
Gradient-clip value	0.1	0.1
Sampling softmax temperature	1.2	1.2
Sampling randomness	2.0	2.0
Reward temperature	0.5	0.25
Effective Sample Size threshold	0.95	0.95
Minimum keep rate	0.1	0.1
Maximum keep rate	1.0	1.0

Table A.3. Online IIFT parameters for lead optimization.

Parameter	**Value**
Maximum number of lead-optimization iterations	10
Generated molecules per iteration	100
Similarity threshold	0.4
Fragment remasking parameter γ	0.0
Hybrid learning rate	1×10^{-5}
Replay buffer size	256
Replay buffer replacement	FIFO
Inner updates per iteration	64
Inner batch size	64
Hybrid reward temperature	0.5 (Viable), 0.1 (Trabtable)
ESS threshold	0.95
Minimum keep rate	0.3 (Viable), 0.1 (Tractable)
Maximum keep rate	0.95
Gradient clipping	Enabled
Gradient-clip value	0.1
Learning-rate schedule	Cosine decay
Minimum scheduler LR	$\mathrm{lr}/100$

A.6. Supplementary Figures

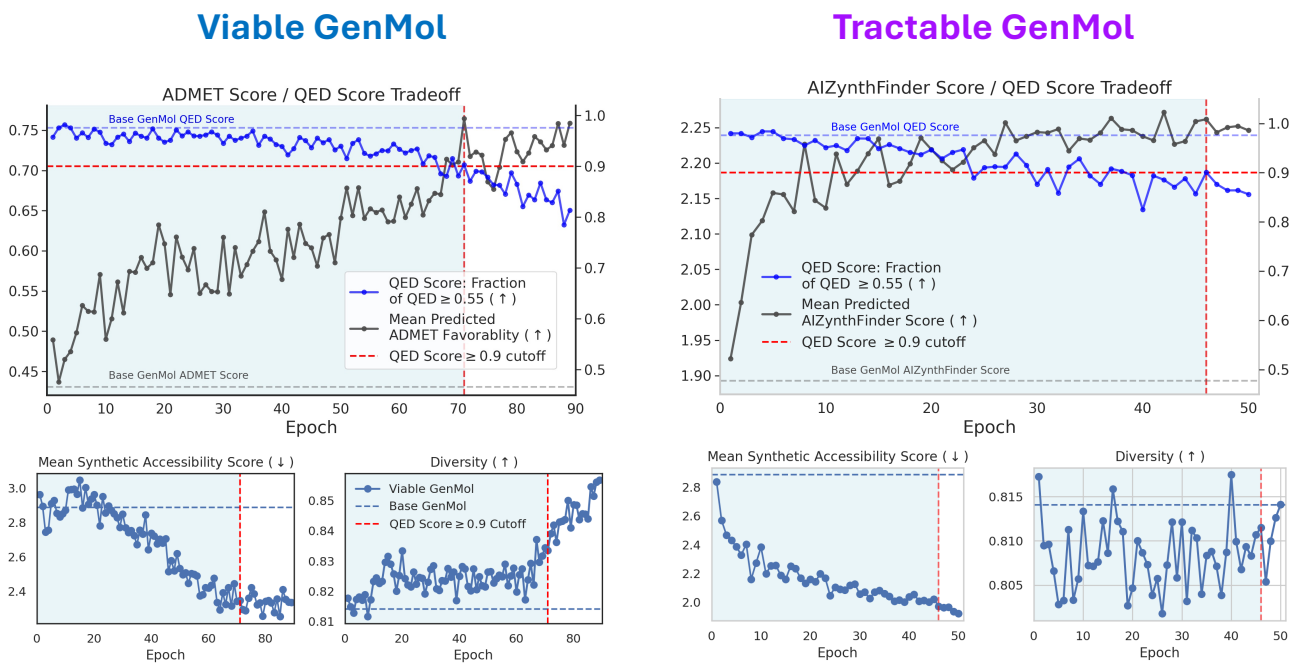

Figure A.1. IIFT Training curves of **Viable GenMol** and **Tractable GenMol**. 500 *de novo* candiates are generated using the fine-tuned model after every epoch in **Viable GenMol** (left) and **Tractable GenMol** (right). In the top panels, we show the tradeoff between the control oracle and "QED score". Here, the QED score is measured as the fraction of generated molecules with QED ≥ 0.55. We treat the portion of training before the final epoch at which this fraction remains above 0.9 as acceptable (highlighted in blue); training beyond that point is designated as the reward-hacking regime. The bottom panels show the evolution of the synthetic accessibility score and the diversity as evaluated by the TDC diversity evaluator (Huang et al., 2021), indicating that neither of these measures degrades with training.

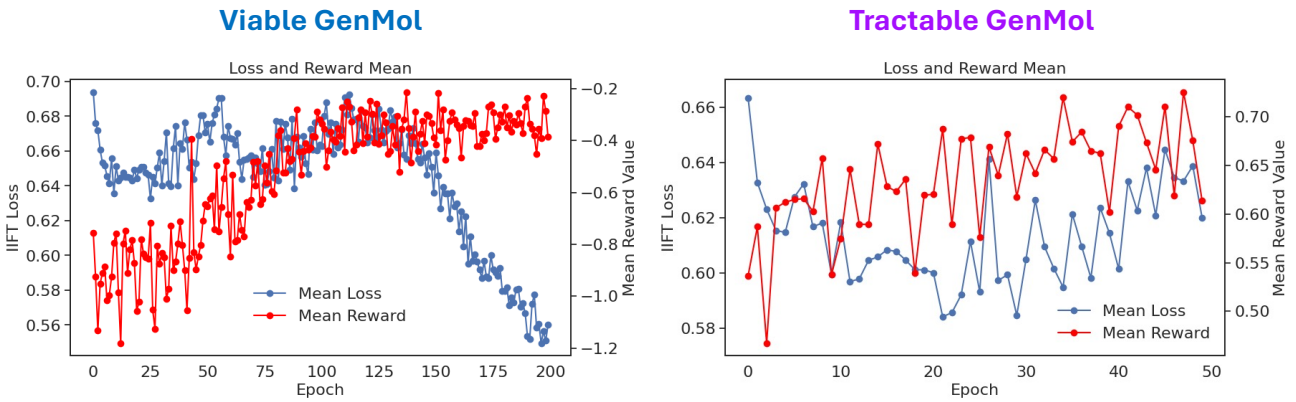

Figure A.2. Training IIFT Loss (blue) and mean reward (red) in every epoch. Loss and reward are computed over the candidates generated via the model in every epoch for **Viable GenMol** (left) and **Tractable GenMol** (right).

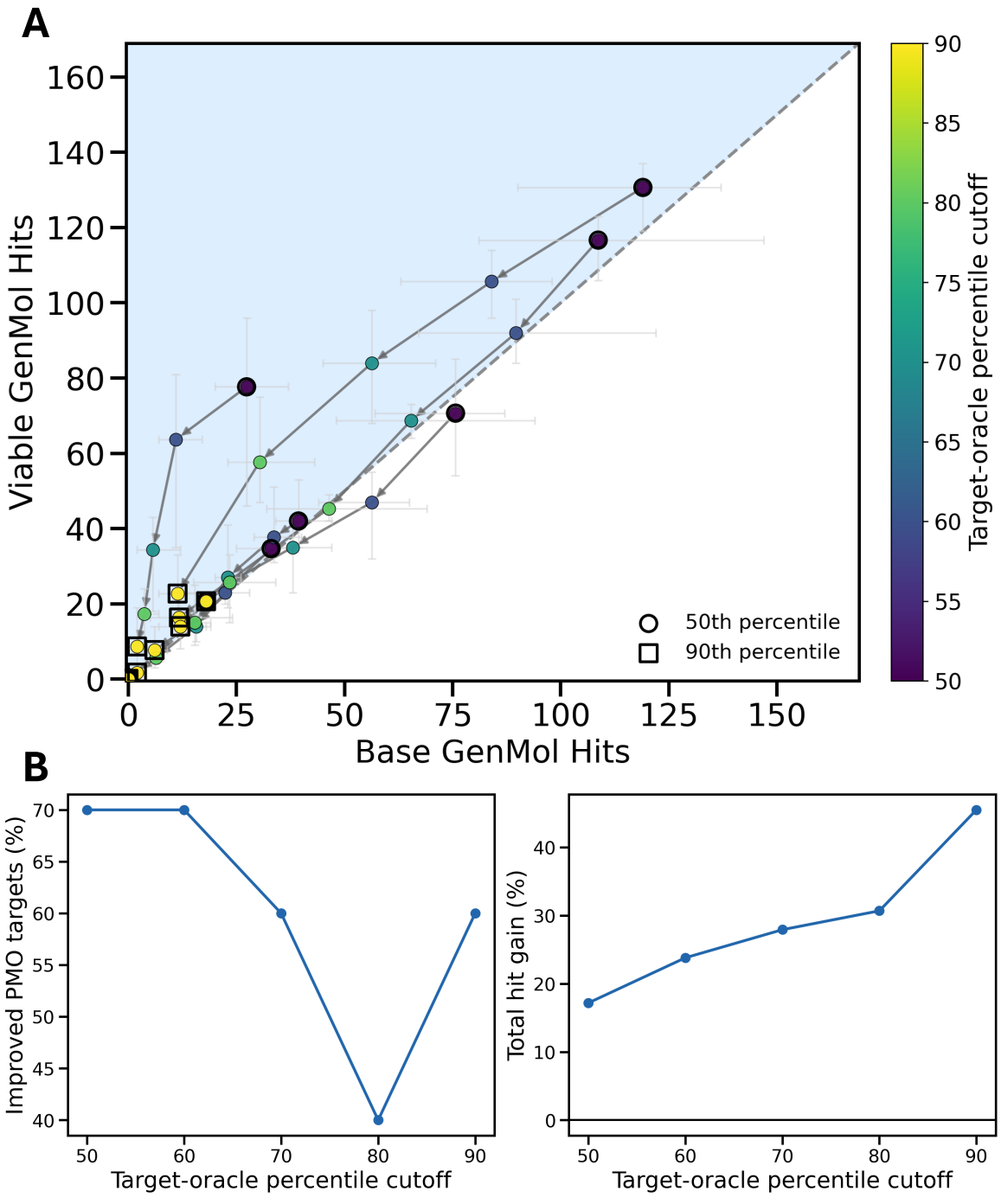

Figure A.3. Effect of PMO hit threshold on hit count for **Viable GenMol**. (A) Parity plot comparing the number of PMO hits found by **Viable GenMol** against base GenMol as the target oracle threshold increases. Each point corresponds to a PMO task at a target oracle threshold, points above the diagonal indicate more hits for the guided model. Error bars show variability across runs, and highlighted markers denote the largest generation cutoffs. (B) Threshold dependence of the improvement, showing the fraction of PMO tasks improved (left) and the percent gain in the number of hits relative to base GenMol (right) as a function of the number of generated molecules.

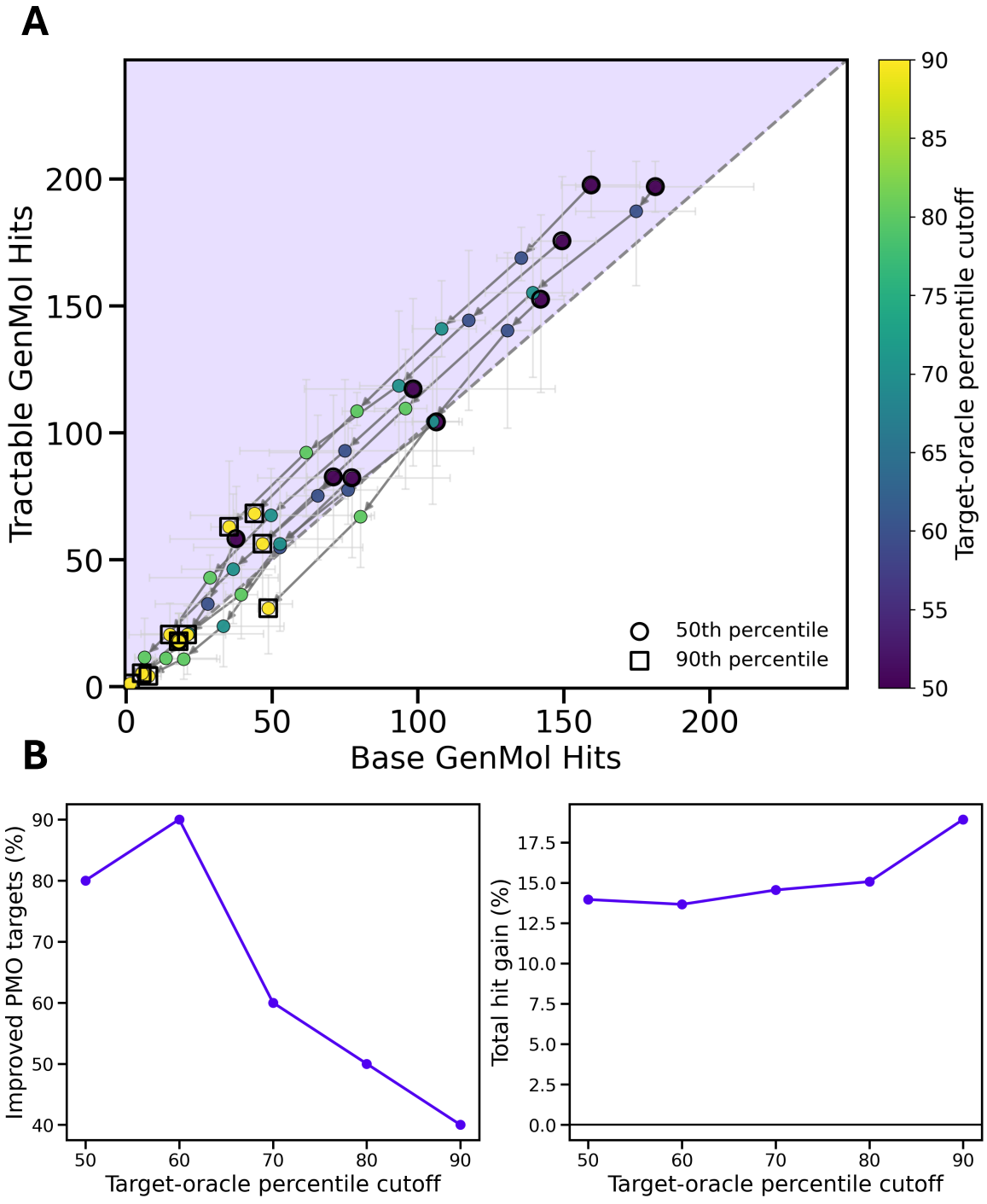

Figure A.4. Effect of PMO hit threshold on hit count for **Tractable GenMol**. (A) Parity plot comparing the number of PMO hits found by **Tractable GenMol** against base GenMol as the target oracle threshold increases. Each point corresponds to a PMO task at a target oracle threshold, points above the diagonal indicate more hits for the guided model. Error bars show variability across runs, and highlighted markers denote the largest generation cutoffs. (B) Threshold dependence of the improvement, showing the fraction of PMO tasks improved (left) and the percent gain in the number of hits relative to base GenMol (right) as a function of the number of generated molecules.

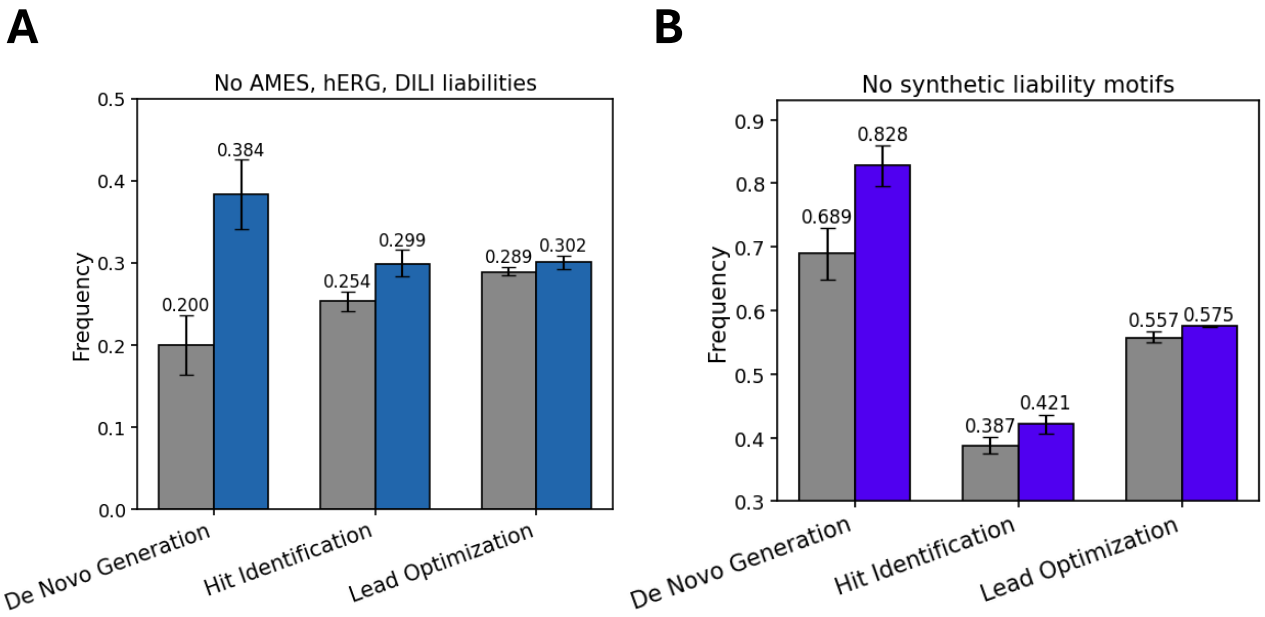

Figure A.5. Filter-based evaluation of generated molecules across *de novo* generation, hit identification, and lead optimization. (A) Frequency of generated molecules without predicted Ames, hERG, or DILI liabilities. (B) Frequency of generated molecules passing synthesizability-motivated structural filters, excluding molecules with problematic synthetic liability motifs. Gray bars denote Base GenMol, while colored bars denote the corresponding controlled model, **Viable GenMol** in (A) and **Tractable GenMol** in (B). Bars show mean frequencies, with error bars indicating the minimum and maximum across 3 runs.

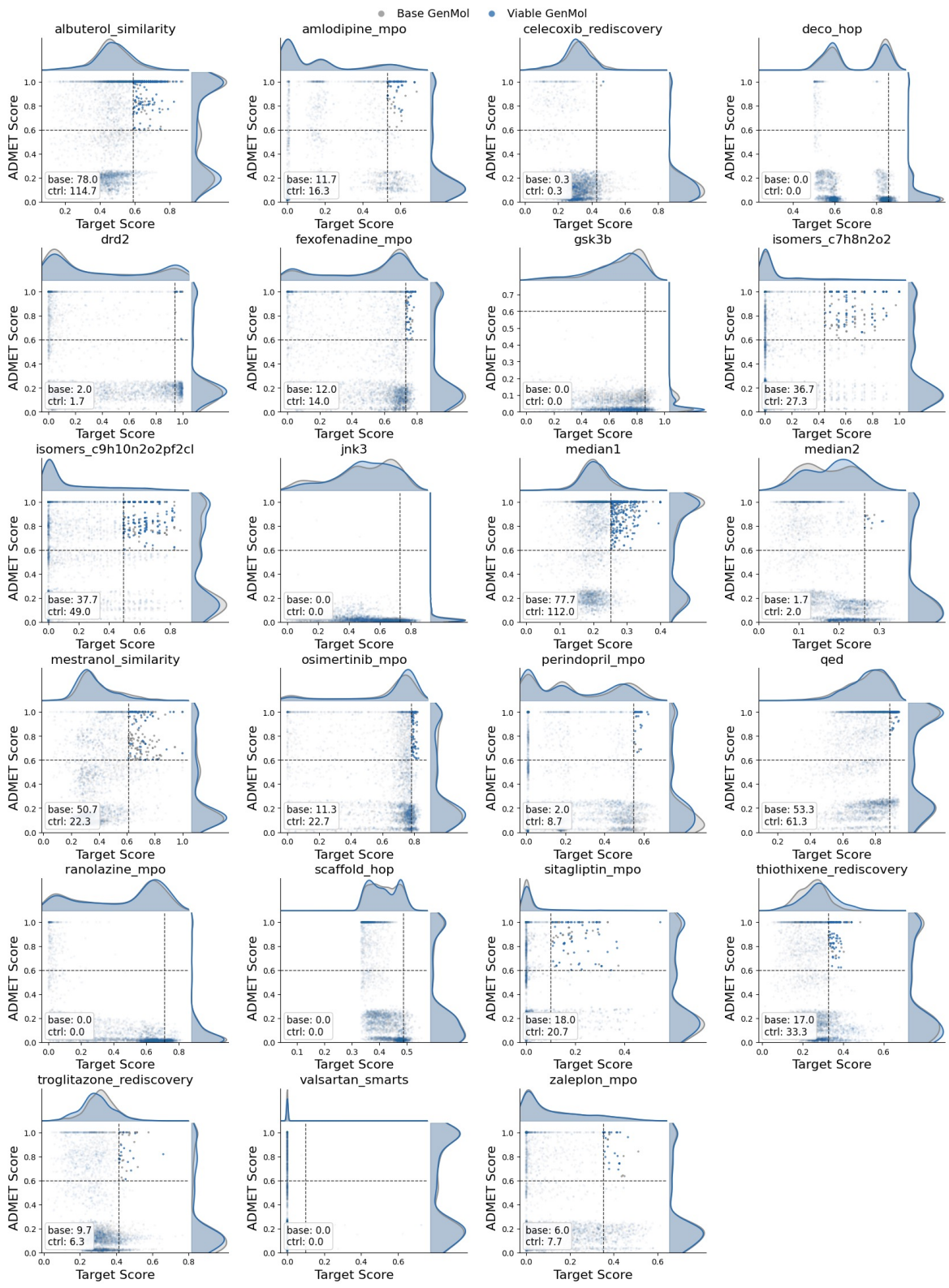

Figure A.6. Distributions of the ADMET scores and the target oracle scores for all 23 TDC oracles (Huang et al., 2021). Average hits per PMO campaign of 1,000 oracle calls are reported in each plot. A "hit" is defined as a molecule that achieves a target-oracle score above the 60$^{\text{th}}$ percentile of the base model's distribution for the corresponding task, and also an ADMET oracle score of above 0.6.

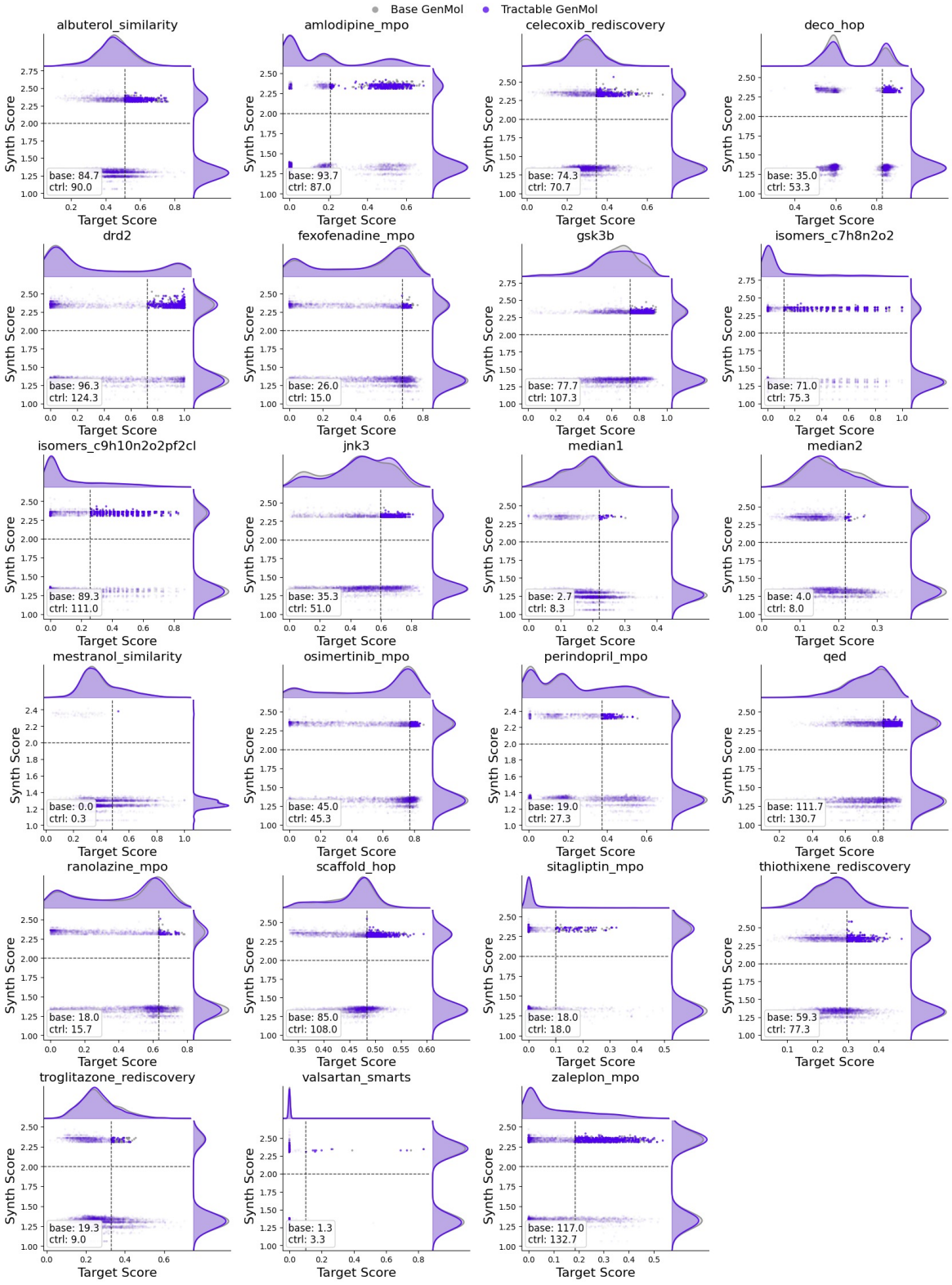

Figure A.7. Distributions of the AiZynthFinder scores and the target oracle scores for all 23 TDC oracles (Huang et al., 2021). Average hits per PMO campaign of 1,000 oracle calls are reported in each plot. A "hit" is defined as a molecule that achieves a target-oracle score above the 60$^{\text{th}}$ percentile of the base model's distribution for the corresponding task, and also an AiZynthFinder oracle score of above 2.0.

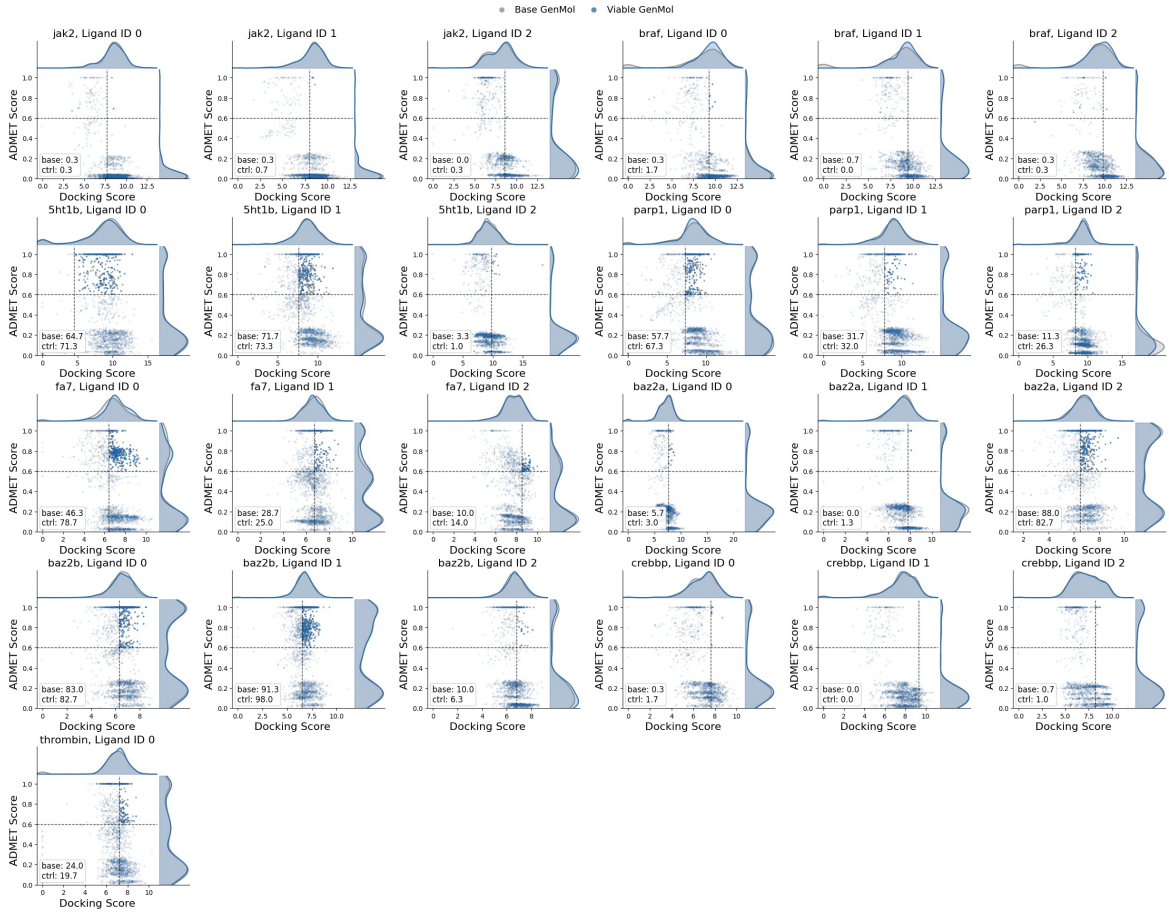

Figure A.8. Distributions of the ADMET scores and the target oracle scores for all 25 protein/lead pairs. Average hits per lead optimization campaign of 4 rounds of 100 molecule generations are reported in each graph. A "hit" is defined as a molecule that achieves a docking free energy below that of the initial lead compound, and also an ADMET oracle score of above 0.6.

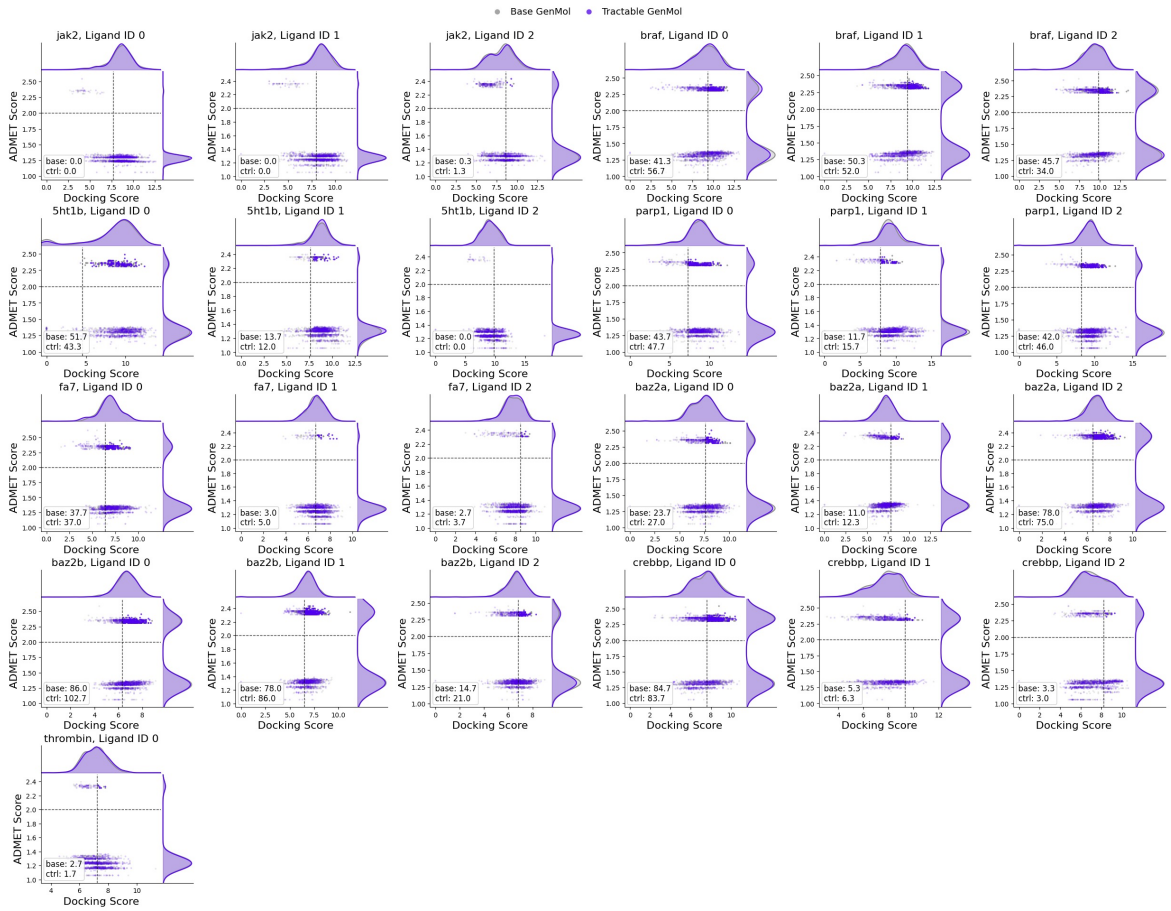

Figure A.9. Distributions of the AiZynthFinder scores and the target oracle scores for all 25 protein/lead pairs. Average hits per lead optimization campaign of 4 rounds of 100 molecule generations are reported in each graph. A "hit" is defined as a molecule that achieves a docking free energy below that of the initial lead compound, and also an AiZynthFinder oracle score of above 2.0.

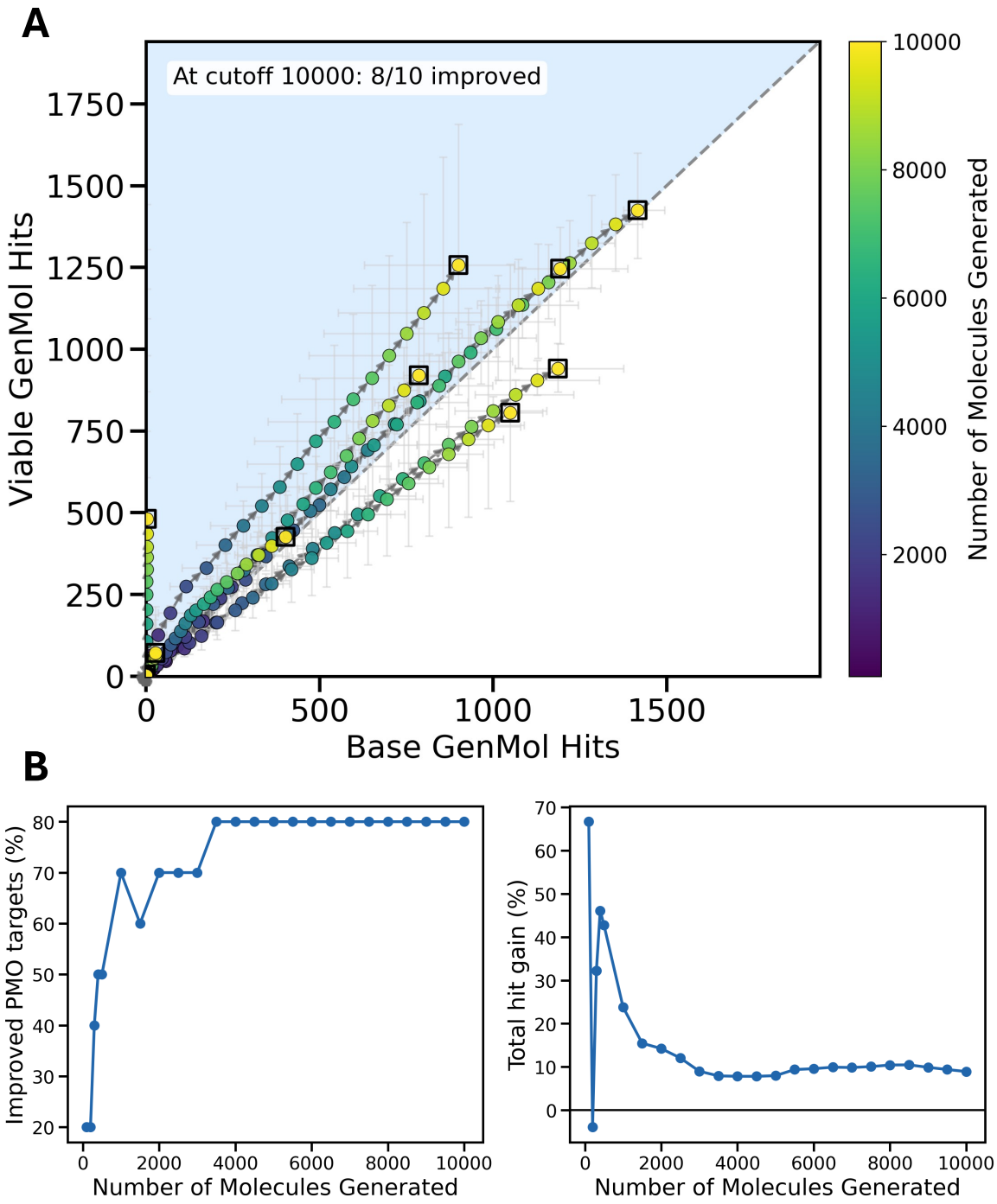

Figure A.10. Effect of oracle-evaluation budget on PMO hit improvement for **Viable GenMol**. (A) Parity plot comparing the number of PMO hits found by **Viable GenMol** against base GenMol as the number of generated molecules increases. Each point corresponds to a PMO task at a given generation cutoff, colored by the number of molecules generated; points above the diagonal indicate more hits for the guided model. Error bars show variability across runs, and highlighted markers denote the largest generation cutoffs. By the final cutoff of 10,000 generated molecules, **Viable GenMol** improves hit counts in $8/10$ PMO tasks. (B) Budget dependence of the improvement, showing the fraction of PMO tasks improved (left) and the percent gain in the number of hits relative to base GenMol (right) as a function of the number of generated molecules. The largest gains appear in the low-budget regime, while at larger budgets the base PMO procedure partially catches up by accumulating many local variations of successful candidates, although the guided model continues to improve more tasks overall.

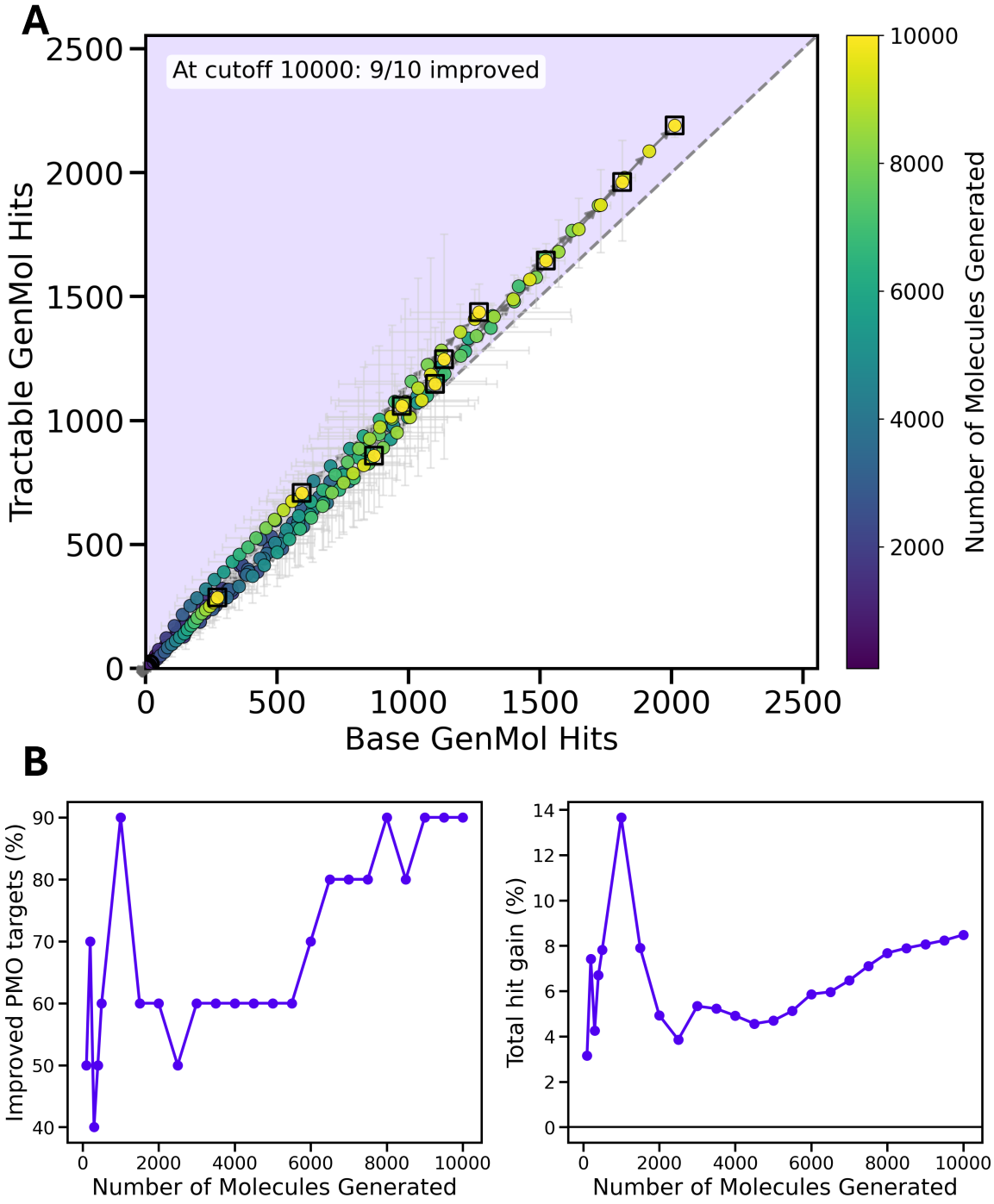

Figure A.11. Effect of oracle-evaluation budget on PMO hit improvement for **Tractable GenMol**. (A) Parity plot comparing the number of PMO hits found by **Tractable GenMol** against base GenMol as the number of generated molecules increases. Each point corresponds to a PMO task at a given generation cutoff, colored by the number of molecules generated; points above the diagonal indicate more hits for the guided model. Error bars show variability across runs, and highlighted markers denote the largest generation cutoffs. By the final cutoff of 10,000 generated molecules, **Tractable GenMol** improves hit counts in $9/10$ PMO tasks. (B) Budget dependence of the improvement, showing the fraction of PMO tasks improved (left) and the percent gain in the number of hits relative to base GenMol (right) as a function of the number of generated molecules. The largest gains appear in the low-budget regime, while at larger budgets the base PMO procedure partially catches up by accumulating many local variations of successful candidates, although the guided model continues to improve more tasks overall.

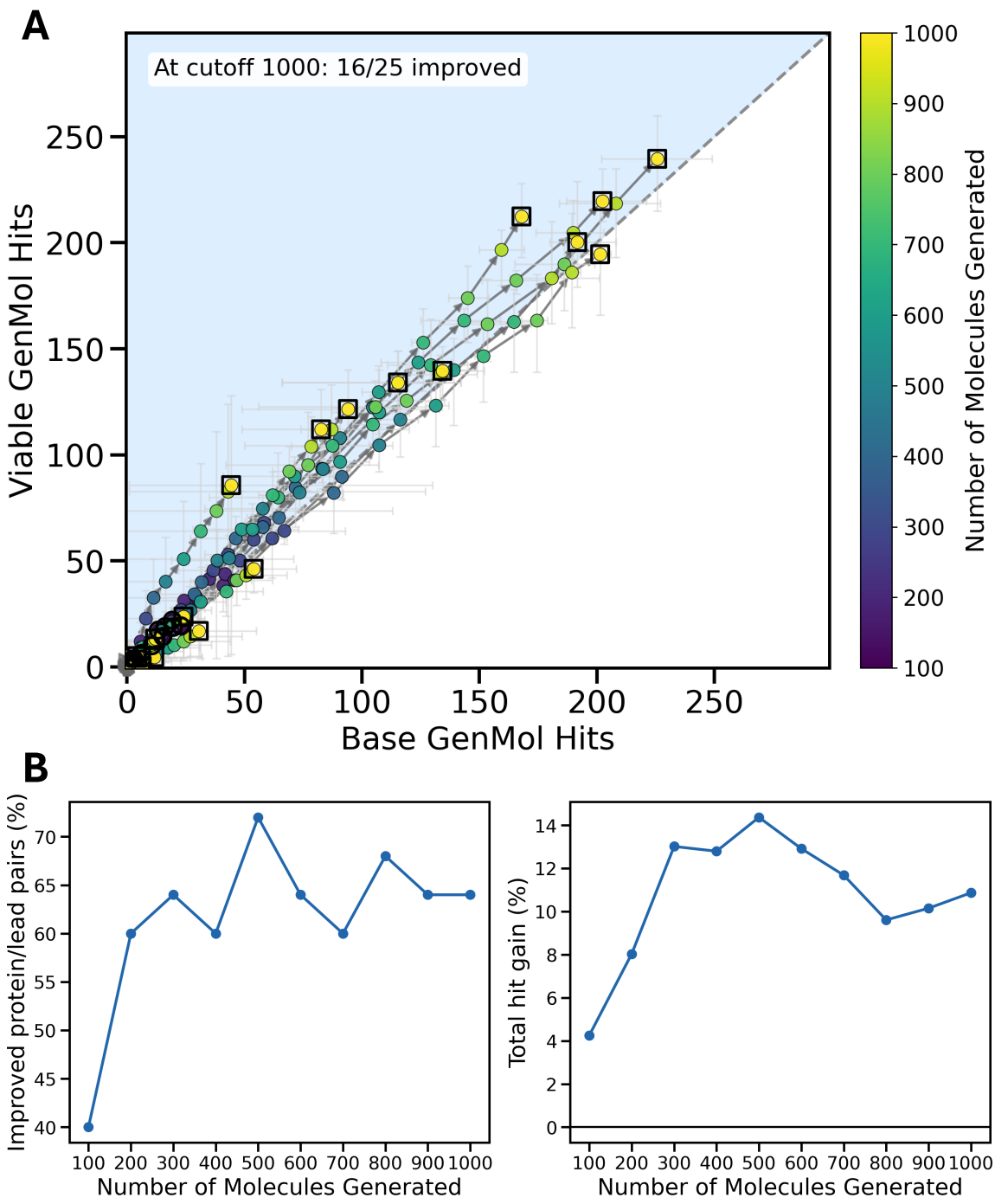

Figure A.12. Effect of oracle-evaluation budget on lead optimization hit improvement for **Viable GenMol**. (A) Parity plot comparing the number of PMO hits found by **Viable GenMol** against base GenMol as the number of generated molecules increases. Each point corresponds to a lead optimization task at a given generation cutoff, colored by the number of molecules generated; points above the diagonal indicate more hits for the guided model. Error bars show variability across runs, and highlighted markers denote the largest generation cutoffs. By the final cutoff of 1,000 generated molecules, **Viable GenMol** improves hit counts in $16/25$ PMO tasks. (B) Budget dependence of the improvement, showing the fraction of lead optimization tasks improved (left) and the percent gain in the number of hits relative to base GenMol (right) as a function of the number of generated molecules. The largest gains appear in the low-budget regime, while at larger budgets the base lead optimization procedure partially catches up by accumulating many local variations of successful candidates, although the guided model continues to improve more tasks overall.

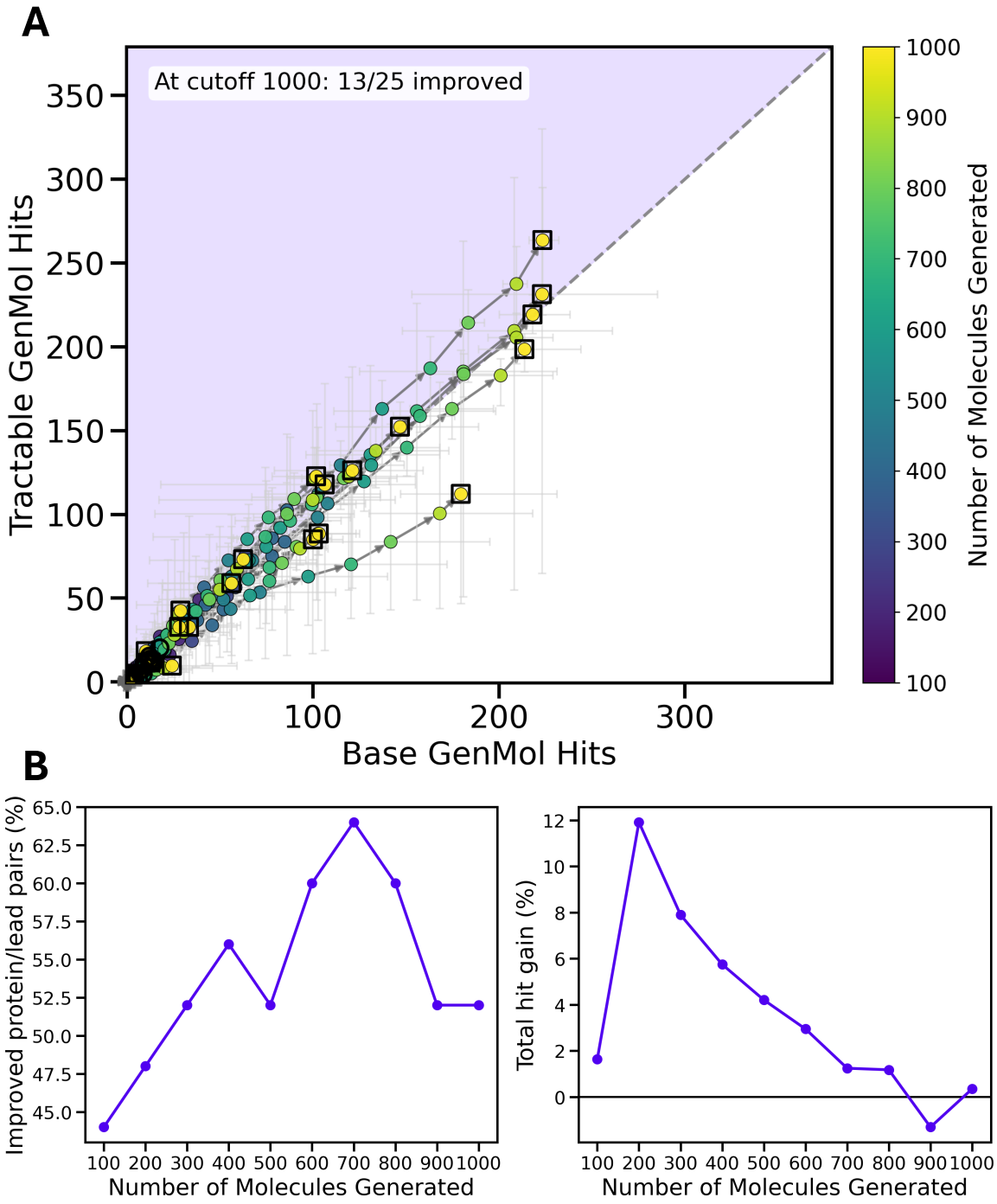

Figure A.13. Effect of oracle-evaluation budget on lead optimization hit improvement for **Tractable GenMol**. (A) Parity plot comparing the number of lead optimization hits found by **Tractable GenMol** against base GenMol as the number of generated molecules increases. Each point corresponds to a lead optimization task at a given generation cutoff, colored by the number of molecules generated; points above the diagonal indicate more hits for the guided model. Error bars show variability across runs, and highlighted markers denote the largest generation cutoffs. By the final cutoff of 1,000 generated molecules, **Tractable GenMol** improves hit counts in $13/25$ PMO tasks. (B) Budget dependence of the improvement, showing the fraction of lead optimization tasks improved (left) and the percent gain in the number of hits relative to base GenMol (right) as a function of the number of generated molecules. The largest gains appear in the low-budget regime, while at larger budgets the base lead optimization procedure partially catches up by accumulating many local variations of successful candidates, although the guided model continues to improve more tasks overall, except for the single data point at a cutoff of 900 molecule generations, primarily due to a single outlier visualized in (A).

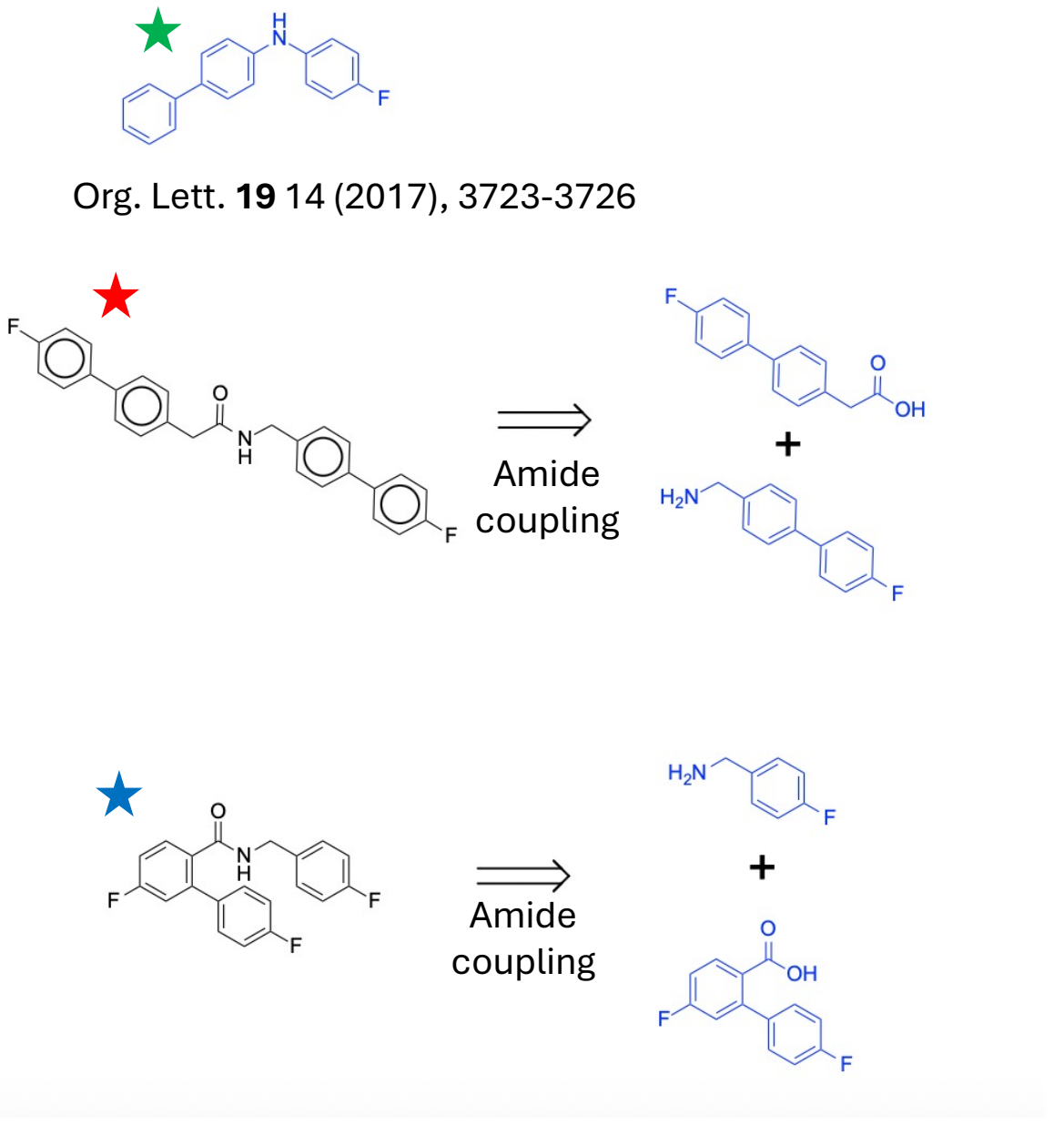

Figure A.14. Retrosynthesis routes of the three starred molecules in Figure 3. The green-starred compound has previously been synthesized by Li & Wang (2017), while the red-starred and blue-starred compound can be synthesized from commercially available reagents (blue compounds) via a single-step amide coupling.