
From Tokens to Blocks: A Block-Diffusion Perspective on Molecular Generation

Qianwei Yang^{1*} Dong Xu^{1*} Zhangfan Yang² Sisi Yuan¹ Zexuan Zhu¹ Jianqiang Li¹ Junkai Ji¹

Abstract

Drug discovery can be viewed as a combinatorial search over an immense chemical space, motivating the development of deep generative models for *de novo* molecular design. Among these, GPT-based molecular language models (MLM) have shown strong molecular design performance by learning chemical syntax and semantics from large-scale data. However, existing MLMs face two fundamental limitations: they inadequately capture the graph-structured nature of molecules when formulated as next-token prediction problems, and they typically lack explicit mechanisms for target-aware generation. Here, we propose SoftMol, a unified framework that co-designs molecular representation, model architecture, and search strategy for target-aware molecular generation. SoftMol introduces soft fragments, a rule-free block representation of SMILES that enables diffusion-native modeling, and develops SoftBD, the first block-diffusion molecular language model that combines local bidirectional diffusion with autoregressive generation under molecular structural constraints. To favor generated molecules with high drug-likeness and synthetic accessibility, SoftBD is trained on a carefully curated dataset named ZINC-Curated. SoftMol further integrates a gated Monte Carlo tree search to assemble fragments in a target-aware manner. Experimental results show that, compared with current state-of-the-art models, SoftMol achieves 100% chemical validity, improves binding affinity by 9.7%, yields a 2–3× increase in molecular diversity, and delivers a 6.6× speedup in inference efficiency. Our code is available at <https://github.com/SZU-ADDG/SoftMol>.

*Equal contribution ¹School of Artificial Intelligence, Shenzhen University, Shenzhen 518060, China ²School of Computer Science, University of Nottingham Ningbo, Ningbo 315100, China. Correspondence to: Junkai Ji <jijunkai@szu.edu.cn>.

1. Introduction

Drug discovery can be formulated as a search problem over a chemical space containing up to 10^{60} possible molecules (Polishchuk et al., 2013; Reymond, 2015). To address the scale of this problem, prior work has increasingly shifted from high-throughput screening toward *de novo* molecular design based on deep generative models (Chen et al., 2018; Vamathevan et al., 2019; Bilodeau et al., 2022). A variety of generative architectures have been developed for molecular generation, encompassing autoregressive Transformers (Bagal et al., 2021; Noutahi et al., 2024) alongside diffusion and flow-based models (Ho et al., 2020; Austin et al., 2021; Lee et al., 2025).

GPT-based molecular language models (MLM) have attracted considerable attention among these models, due to their strong empirical performance (Bagal et al., 2021; Flam-Shepherd et al., 2022). They represent small molecules as textual sequences, such as SMILES and SAFE strings (Weininger, 1988; Noutahi et al., 2024). By training on large-scale molecular datasets, these models learn the intrinsic syntactic and semantic properties of chemical structures, which enables the generation of chemically plausible molecules with high diversity and novelty (Krenn et al., 2020; Nigam et al., 2022).

However, these MLM-based approaches suffer from two fundamental limitations. First, small molecules are intrinsically not one-dimensional and unidirectional sequences but structured chemical graphs (Jin et al., 2018; De Cao & Kipf, 2018). Consequently, framing molecular generation as a next-token prediction problem is suboptimal, as it fails to adequately capture the local chemical context, including the interactions among neighboring atoms within molecular substructures (Atz et al., 2021; Vignac et al., 2023). Second, such generative models typically operate independently of the target protein, as the generation process cannot explicitly incorporate target-specific information (Bilodeau et al., 2022; Peng et al., 2022). As a result, these limitations render MLMs ill-suited for true target-aware drug design and substantially constrain their practical utility in realistic drug discovery workflows.

To address the aforementioned limitations, several molecular language models have explored discrete diffusion architectures to explicitly model interactions among all to-

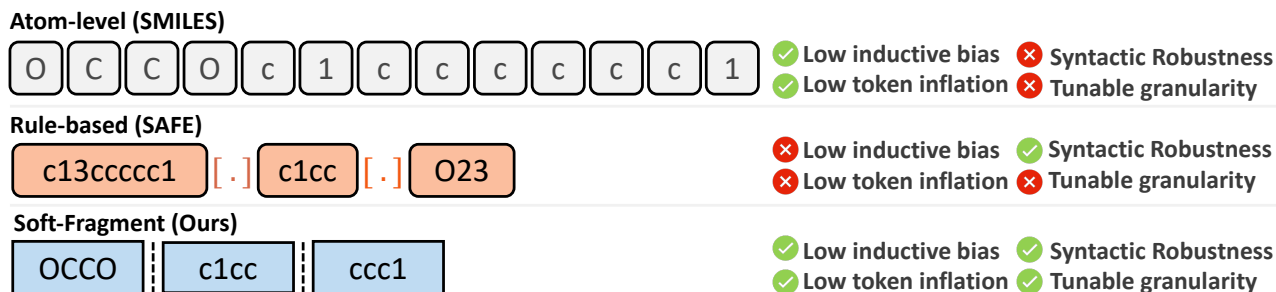


Figure 1. Comparison of molecular representation paradigms. **Top:** Atom-level SMILES provides minimal bias but lacks syntactic robustness. **Middle:** Rule-based fragments impose rigid vocabulary constraints and heuristic priors. **Bottom:** Our soft-fragment approach segments SMILES into fixed-length blocks, enabling tunable granularity and high robustness without heuristic rules or auxiliary tokens.

kens (Lee et al., 2025; Austin et al., 2021). However, diffusion-based models are designed to generate output sequences of arbitrary length (Han et al., 2022; Sahoo et al., 2024). In contrast, small molecules typically contain a bounded and highly structured number of tokens, ranging from only a few to several hundred (Sterling & Irwin, 2015). This mismatch conflicts with the intrinsic structural constraints of molecular representations (Kusner et al., 2017; Vignac et al., 2023). On the other hand, protein information has been used as prompt or conditional input to guide molecular language models toward target-specific molecule generation (Grechishnikova, 2021). However, cross-modal alignment is challenging, and experimental protein–ligand data remain scarce (Huang et al., 2021b). Together, these issues still limit the scalability and practical applicability of current MLMs for drug design.

This study proposes **SoftMol**, a unified framework that co-designs molecular representations, model architecture, and search strategy to systematically address these limitations. Specifically, SoftMol partitions SMILES sequences into contiguous, fixed-length blocks without chemistry-specific heuristics or auxiliary tokens, which we term **soft-fragments**, distinguishing them from traditional rule-based fragments. A block-diffusion-based masked language model, termed **SoftBD**, is then employed for molecular generation, enabling local bidirectional diffusion within each soft fragment while autoregressively conditioning on previously generated fragments. SoftBD is trained on a carefully curated, high-quality molecular dataset, termed ZINC-Curated, to promote the generation of molecules with strong drug-likeness and synthetic accessibility. Finally, SoftMol integrates a novel gated Monte Carlo tree search (MCTS) procedure to assemble the generated soft fragments in a target-aware manner, facilitating molecular design toward specific protein targets. Extensive experiments demonstrate that SoftMol significantly outperforms all existing molecular generative models on both *de novo* molecule generation and target-specific design tasks, establishing a new state of the art and revealing the decisive advantages of block-

diffusion-based modeling for molecular generation.

Our main contributions can be summarized as follows:

- We introduce **soft-fragments**, a rule-free block representation of SMILES that enables diffusion-native molecular modeling and tunable granularity across tasks.
- We propose **SoftBD**, the first block-diffusion molecular language model that reconciles bidirectional diffusion with autoregressive generation under molecular structural constraints.
- We develop **SoftMol**, a unified and target-aware framework that integrates SoftBD and gated MCTS, achieving state-of-the-art performance in both *de novo* and target-specific molecular design.

2. Preliminaries

2.1. Notation and Problem Setup

Molecular sequence representation. A molecule is represented as a discrete sequence $\mathbf{x} = (x_1, \dots, x_L)$ of length L , where each token $x_i \in \mathcal{V}$ belongs to a vocabulary \mathcal{V} comprising chemical primitives (atoms, bonds, ring markers) and control symbols ([BOS], [EOS], [PAD], [MASK]).

Task formulation. We address two fundamental tasks:

- De novo generation:* Learn a generative model $p_\theta(\mathbf{x})$ that approximates the distribution of valid, drug-like, and synthetically accessible molecules.
- Target-specific molecular design:* Given a target protein, identify molecules \mathbf{x}^* that maximize binding affinity while satisfying pharmacological constraints:

$$\begin{aligned} \mathbf{x}^* &= \underset{\mathbf{x}}{\operatorname{argmin}} \operatorname{DS}(\mathbf{x}) \\ \text{s.t. } \operatorname{QED}(\mathbf{x}) &> \tau_{\operatorname{QED}}, \operatorname{SA}(\mathbf{x}) < \tau_{\operatorname{SA}}, \end{aligned} \quad (1)$$

where DS represents the docking score (Alhossary et al., 2015) (lower indicates higher affinity), QED quantifies drug-likeness (Bickerton et al., 2012), and SA measures synthetic accessibility (Ertl & Schuffenhauer, 2009).

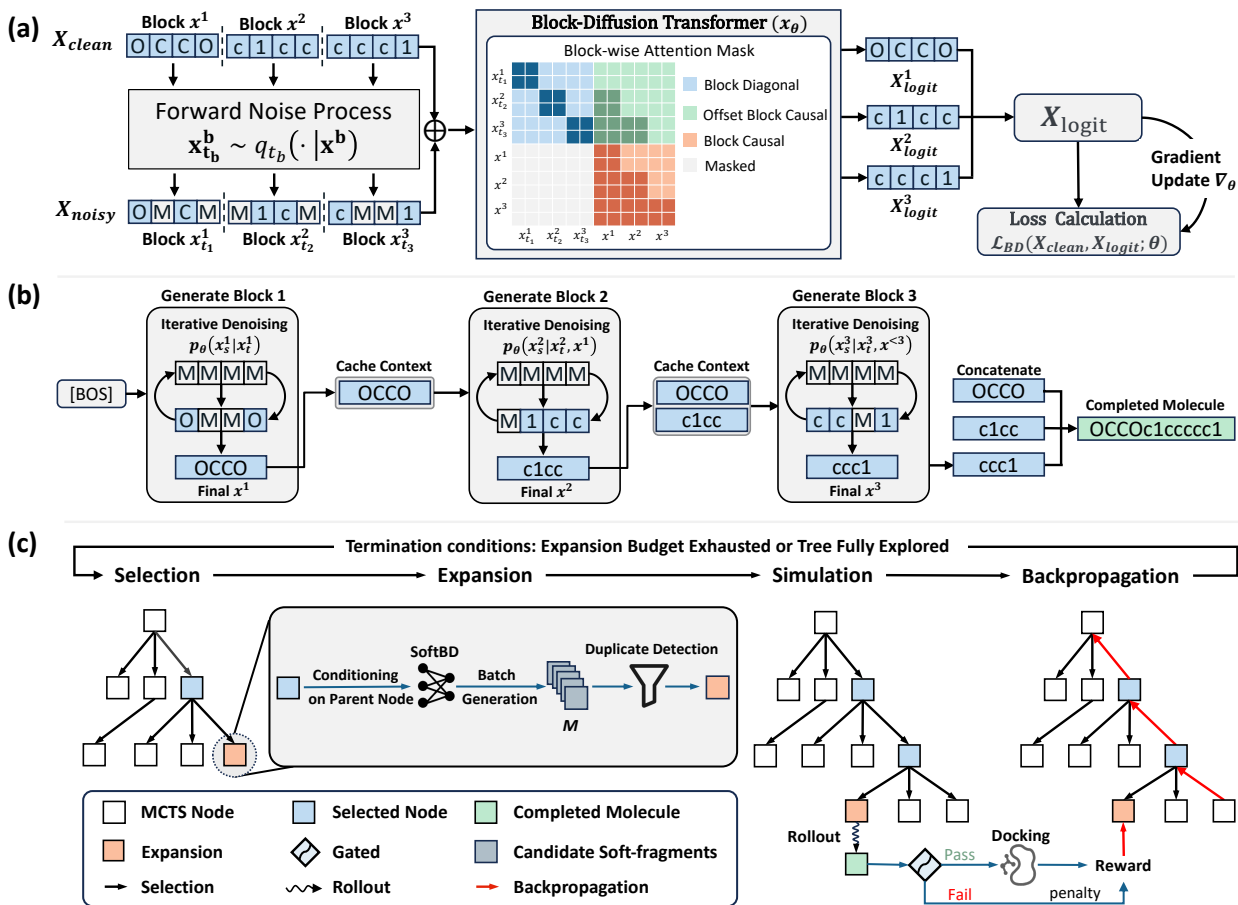


Figure 2. Overview of the SoftMol framework. (a) **Training:** The concatenated clean and noised sequences are processed by the Block-Diffusion Transformer with a block-wise attention mask enforcing intra-block bidirectional and inter-block causal dependencies. (b) **Sampling:** Starting from [BOS], molecules are generated semi-autoregressively via iterative denoising, with previously decoded blocks cached as context. (c) **Gated MCTS:** Selection traverses the tree via UCT; Expansion uses batched SoftBD generation with duplicate filtering; Simulation applies a tunable feasibility gate—candidates satisfying pharmacological constraints proceed to docking, while failures receive a penalty; Backpropagation updates node statistics.

2.2. Background: Generative Models for Sequences

Autoregressive models factorize the joint distribution as:

$$p_\theta(\mathbf{x}) = \prod_{i=1}^L p_\theta(x_i | \mathbf{x}_{<i}). \quad (2)$$

While enabling efficient parallel training, this strictly causal factorization precludes bidirectional context. This restricts the modeling of local substructures, where chemical validity typically relies on the mutual dependencies of all constituent atoms rather than a linear sequence history (Chithrananda et al., 2020).

Discrete diffusion models (Austin et al., 2021) define a forward process $q(\mathbf{x}_t | \mathbf{x}_0)$ that progressively corrupts tokens, and learn a reverse denoising distribution $p_\theta(\mathbf{x}_0 | \mathbf{x}_t)$. Unlike autoregressive models, diffusion facilitates bidirectional context modeling, allowing the model to capture interactions among all tokens simultaneously. However, standard

formulations typically operate on fixed-dimensional tensors (Austin et al., 2021), necessitating inefficient padding for molecular sequences. While variable-length extensions have been proposed (Han et al., 2022; Sahoo et al., 2024), they often require complex length-modeling priors that complicate the simple termination mechanism inherent to autoregressive approaches.

Block diffusion models (Arriola et al., 2025) unify these paradigms through a semi-autoregressive framework. This approach decomposes the sequence into tunable-granularity blocks modeled autoregressively, thereby retaining flexible termination while employing discrete diffusion to capture local bidirectional dependencies within each block:

$$p_\theta(\mathbf{x}) = \prod_{b=1}^B p_\theta(\mathbf{x}^b | \mathbf{x}^{<b}). \quad (3)$$

In this formulation, each conditional $p_\theta(\mathbf{x}^b | \mathbf{x}^{<b})$ is parameterized as a denoising network. This design decouples

global sequence generation from local structure modeling: the autoregressive prior manages sequence length and global coherence, while the block-wise diffusion employs bidirectional attention to capture the dense constraints of local chemical substructures.

3. Method

3.1. Soft-Fragment Representation

We define *soft-fragments* via a deterministic partitioning of a fixed-length molecular sequence. Given a SMILES string s , we first construct a dense sequence $\mathbf{x} = (x_1, \dots, x_L)$ of uniform length L by padding s with [PAD] tokens. This sequence is then partitioned into $B = L/K$ contiguous blocks of fixed length K . This approach constitutes a rule-free computational segmentation protocol (Figure 1), where the b -th block is simply defined as:

$$\mathbf{x}^b := (x_{(b-1)K+1}, \dots, x_{bK}), \quad b \in \{1, \dots, B\}. \quad (4)$$

By strictly enforcing the global length L to be a multiple of K , we ensure that every block possesses a uniform dimension without requiring conditional padding. This formulation structurally decouples representation from generation: a fixed granularity K_{train} is employed to capture intrinsic chemical syntax, while the sampling granularity K_{sample} serves as a flexible hyperparameter to control search resolution during inference (detailed analysis in Appendix 5).

3.2. SoftBD Architecture and Training

We implement SoftBD by instantiating Block Diffusion (Arriola et al., 2025) over soft-fragments, following the factorization in Equation (3).

Block-wise attention masking. We employ a hybrid attention mechanism (Arriola et al., 2025) (Figure 2(a)) on the concatenated noised (\mathbf{x}_t) and clean (\mathbf{x}_0) sequences. The applied mask $\mathcal{M}_{\text{full}} \in \{0, 1\}^{2L \times 2L}$ enforces inter-block causal dependencies while permitting intra-block bidirectional context. This structure is critical for soft-fragments: it enables the model to implicitly reconstruct chemical syntax and local bonds disrupted by arbitrary chunking, thereby eliminating the need for rule-based boundary preservation (details in Appendix B; empirical validation in Appendix F.1).

Training Objective. We minimize the Block Diffusion Negative Evidence Lower Bound (NELBO) (Arriola et al., 2025). Leveraging the block-wise attention mask, we employ a vectorized training strategy to compute gradients for the entire molecule in a single forward pass. Specifically, for each block b , we sample a time $t \sim \mathcal{U}[0, 1]$ and mask tokens with probability $1 - \alpha_t$. The objective function aggregates the weighted cross-entropy losses over masked positions

across all blocks (derivation in Appendix C):

$$\mathcal{L}_{\text{BD}}(\mathbf{x}; \theta) := \sum_{b=1}^B \mathbb{E}_{t, \mathbf{x}_t^b} \left[-\frac{\alpha'_t}{1 - \alpha_t} \mathcal{L}_{\text{CE}}(\mathbf{x}^b, p_\theta(\cdot | \mathbf{x}_t^b, \mathbf{x}^{<b})) \right], \quad (5)$$

where α_t denotes the noise schedule and α'_t its time derivative. Figure 2(a) illustrates this pipeline.

3.3. Adaptive Confidence Decoding for SoftBD

SoftBD generates molecules semi-autoregressively (Figure 2(b)): for each block $b = 1, \dots, B$, the model performs reverse diffusion conditioned on the cached history $\mathbf{x}^{<b}$. We implement *Adaptive Confidence Decoding* (Algorithm 2, Appendix D.4), integrating three optimization mechanisms.

First-Hitting Sampling. Unlike standard masked diffusion which relies on a fixed, often redundant schedule T , we employ First-Hitting sampling (Zheng et al., 2024) to analytically determine optimal transition times. For a sequence i with $m_t^{(i)}$ masked tokens, the diffusion time updates via:

$$t_{\text{next}}^{(i)} = t_{\text{curr}}^{(i)} \cdot (u^{(i)})^{1/m_t^{(i)}}, \quad u^{(i)} \sim \mathcal{U}[0, 1]. \quad (6)$$

This mechanism dynamically aligns the step size with generative entropy: large m_t triggers granular refinement steps, while small m_t allows rapid progression, effectively eliminating computationally wasteful “no-op” iterations.

Greedy Confidence Decoding. To ensure high structural fidelity, we employ a deterministic strategy that prioritizes the resolution of high-confidence substructures. At each adaptive step, we identify the position-token pair (j^*, v^*) that maximizes the local conditional probability across all masked indices $j \in \mathcal{M}_t$:

$$j^* = \operatorname{argmax}_{j \in \mathcal{M}_t} \left(\max_{v \in \mathcal{V}} p_\theta(x_j = v | \mathbf{x}_t^b, \mathbf{x}^{<b}) \right). \quad (7)$$

We deterministically unmask x_{j^*} to v^* . This confidence-ordered revealing process **allows** premature commitment to ambiguous tokens, ensuring that complex dependencies are resolved only when sufficient context is available. As shown in Table 4, this strategy not only guarantees syntactic correctness but also substantially improves pharmacological quality.

Batched Block-Wise Inference with Caching. To maximize throughput, we parallelize generation by decoding mini-batches of molecules as independent Markov chains. Crucially, we employ a caching mechanism that freezes completed blocks as context, restricting active denoising computation solely to the current block. Coupled with a sliding attention window and an absorbing [EOS] termination token, this strategy amortizes the iterative refinement cost, achieving near-constant per-token latency.

3.4. Gated MCTS for Target-Specific Search

We formulate target-specific molecular design as a Markov Decision Process (MDP) defined over the generative block space induced by SoftBD. A state s_b represents a partial molecule comprising b soft-fragments, and an action a corresponds to the generation of the subsequent block $\mathbf{x}^{b+1} \sim p_\theta(\cdot | s_b)$. Unlike retrieval-based methods, this *generative action space* eliminates reliance on rigid fragment libraries, enabling open-ended exploration with adaptive granularity controlled by K_{sample} . To effectively navigate this space under strict pharmacological constraints, we introduce a **Tunable Feasibility Gate** that decouples validity enforcement from binding affinity optimization. The search procedure follows four phases (Figure 2(c); Algorithm 3 in Appendix E).

Selection. We guide tree traversal using a modified Upper Confidence Bound (UCT) that balances expectation and maximization. Each node s tracks visit count $N(s)$, mean return $\bar{R}(s)$, and maximum return $R^{\max}(s)$. A child s_j is selected by maximizing:

$$\text{UCT}(s_j) = \lambda \bar{R}(s_j) + (1 - \lambda) R^{\max}(s_j) + C \sqrt{\frac{\ln N(s)}{N(s_j)}}, \quad (8)$$

where $\lambda \in [0, 1]$ modulates the trade-off between robust average performance and peak affinity discovery. To efficiently allocate computational budget, we employ a Children-Adaptive widening strategy (Tian et al., 2024) that dynamically adjusts the branching factor based on value uncertainty. The node-specific expansion cap $C_{\text{cap}}(s)$ is defined as:

$$C_{\text{cap}}(s) = \min\left(C_{\text{max}}, \max\left(C_{\text{min}}, \lfloor \beta I(s) \rfloor\right)\right), \quad (9)$$

$$I(s) := \max_k |\bar{R}(s_k) - \bar{R}(s)|,$$

where $I(s)$ measures the dispersion of child values. This mechanism triggers wider expansion in regions with high variance (potential high rewards) while conserving resources in low-value or converged subspaces.

Expansion. Upon reaching a leaf node s_b , we perform batched parallel exploration by sampling M next-block candidates from $p_\theta(\cdot | s_b)$. After filtering duplicates, a unique candidate is instantiated as a new child node. This approach leverages SoftBD’s batch inference capability to replace sequential rejection sampling with a single, high-throughput forward pass, efficiently discovering diverse transitions.

Simulation with a Tunable Feasibility Gate. From the expanded child, we perform a rollout using the SoftBD prior to complete the molecule \mathbf{x} . To strictly decouple pharmacological compliance from affinity optimization, we implement a *Tunable Feasibility Gate* parameterized by

$(\tau_{\text{QED}}, \tau_{\text{SA}})$. The reward function is defined as:

$$R(\mathbf{x}) = \begin{cases} -\text{DS}(\mathbf{x}) & \text{if } \text{QED}(\mathbf{x}) \geq \tau_{\text{QED}} \wedge \text{SA}(\mathbf{x}) \leq \tau_{\text{SA}}, \\ R_{\text{pen}} & \text{otherwise,} \end{cases} \quad (10)$$

where $\text{DS}(\mathbf{x})$ denotes the Vina docking score and R_{pen} is a penalty for violation. This mechanism functions as a hierarchical computational sieve: by preemptively filtering candidates based on 2D properties, it ensures that computationally expensive 3D docking is reserved exclusively for pharmacologically viable structures, thereby maximizing the return on computational investment. We instantiate two configurations: **SoftMol** with default thresholds ($\tau_{\text{QED}} = 0.5, \tau_{\text{SA}} = 5.0$), and **SoftMol (Unconstrained)** with ($\tau_{\text{QED}} = 0, \tau_{\text{SA}} = \infty$) to probe the affinity ceiling without pharmacological constraints.

Backpropagation. Finally, the reward signal $R(\mathbf{x})$ is propagated recursively from the leaf to the root. We update the nodal statistics $\{N, \bar{R}, R^{\max}\}$ along the trajectory, progressively refining the tree policy to focus exploration on the optimal structural subspace defined by the feasibility gate and affinity landscape.

4. Experiments

We systematically evaluate SoftMol across three key dimensions. First, we assess unsupervised generative performance and distribution matching against leading fragment-based baselines (Section 4.2). Second, we benchmark the full framework on target-specific molecular design tasks across diverse protein targets (Section 4.3). Finally, we investigate the method’s mechanistic underpinnings, including representation robustness and component-level ablations, in Appendix 5 and Appendix 6.

4.1. Datasets

We employ three large-scale datasets to evaluate general generative modeling and drug-likeness.

SMILES (324M). Following Noutahi et al. (2024), we aggregate a broad collection of molecules from ZINC (Irwin et al., 2012) and UniChem (Chambers et al., 2013). We filter sequences exceeding 72 tokens (0.77% rejection rate), yielding a final corpus of 324 million SMILES strings representing general chemical space.

SAFE (322M). To benchmark against rule-based fragmentation, we utilize the official SAFE dataset (Noutahi et al., 2024). From an initial pool of 326 million samples, we exclude sequences longer than 88 tokens (0.85% removal) to ensure compatibility with fixed-window training, resulting in a corpus of 322 million SAFE strings.

ZINC-Curated (427M). To prioritize pharmaceutical rele-

Table 1. *De novo* molecule generation results. The results are the means and standard deviations of 3 runs. p and τ denote the nucleus sampling probability and softmax temperature, respectively. All SoftBD results use $K_{\text{sample}} = 2$. The best results are highlighted in **bold**.

Method	Validity (%)	Uniqueness (%)	Quality (%)	Docking-Filter (%)	Diversity
SAFE-GPT (Noutahi et al., 2024)	93.2 ± 0.1	100.0 ± 0.0	54.4 ± 0.6	78.3 ± 0.5	0.879 ± 0.000
GenMol (Lee et al., 2025)	99.9 ± 0.1	96.0 ± 0.3	85.2 ± 0.4	97.8 ± 0.1	0.817 ± 0.000
SoftBD ($p=1.0, \tau=0.9$)	99.8 ± 0.0	100.0 ± 0.0	87.1 ± 0.2	98.5 ± 0.1	0.871 ± 0.000
SoftBD ($p=1.0, \tau=1.0$)	99.6 ± 0.0	100.0 ± 0.0	84.7 ± 0.2	97.8 ± 0.1	0.878 ± 0.000
SoftBD ($p=1.0, \tau=1.1$)	99.1 ± 0.0	100.0 ± 0.0	81.7 ± 0.3	96.5 ± 0.1	0.883 ± 0.000
SoftBD ($p=1.0, \tau=1.2$)	98.3 ± 0.0	100.0 ± 0.0	77.7 ± 0.3	94.2 ± 0.2	0.888 ± 0.000
SoftBD ($p=1.0, \tau=1.3$)	96.7 ± 0.1	100.0 ± 0.0	72.9 ± 0.3	91.1 ± 0.2	0.893 ± 0.000
SoftBD ($p=0.95, \tau=0.9$)	100.0 ± 0.0	98.4 ± 0.1	93.5 ± 0.2	99.8 ± 0.0	0.844 ± 0.000
SoftBD ($p=0.95, \tau=1.0$)	100.0 ± 0.0	99.4 ± 0.1	92.8 ± 0.0	99.7 ± 0.0	0.851 ± 0.000
SoftBD ($p=0.95, \tau=1.1$)	100.0 ± 0.0	99.6 ± 0.1	91.9 ± 0.1	99.6 ± 0.0	0.858 ± 0.000
SoftBD ($p=0.95, \tau=1.2$)	99.9 ± 0.0	99.8 ± 0.0	90.8 ± 0.1	99.3 ± 0.1	0.867 ± 0.000
SoftBD ($p=0.95, \tau=1.3$)	99.9 ± 0.0	99.8 ± 0.1	88.9 ± 0.2	98.9 ± 0.1	0.871 ± 0.000
SoftBD ($p=0.9, \tau=0.9$)	100.0 ± 0.0	90.0 ± 0.2	94.9 ± 0.2	99.9 ± 0.0	0.829 ± 0.000
SoftBD ($p=0.9, \tau=1.0$)	100.0 ± 0.0	96.0 ± 0.1	94.0 ± 0.2	99.8 ± 0.0	0.839 ± 0.000
SoftBD ($p=0.9, \tau=1.1$)	100.0 ± 0.0	98.0 ± 0.1	93.3 ± 0.3	99.8 ± 0.0	0.846 ± 0.000
SoftBD ($p=0.9, \tau=1.2$)	100.0 ± 0.0	99.1 ± 0.1	92.4 ± 0.2	99.7 ± 0.0	0.852 ± 0.000
SoftBD ($p=0.9, \tau=1.3$)	100.0 ± 0.0	99.3 ± 0.1	91.7 ± 0.2	99.6 ± 0.0	0.858 ± 0.000

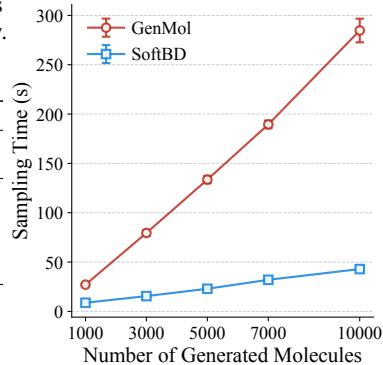


Figure 3. Sampling time comparison between GenMol and SoftBD; SoftBD achieves $\approx 6.6\times$ shorter sampling time for 10k molecules.

vance, we curate a high-quality subset of ZINC-22 (Sterling & Irwin, 2015) using a multi-stage filtration pipeline adapted from Chitsaz et al. (2025). The pipeline enforces physico-chemical constraints, structural validity, and diversity-aware stratification (Appendix D.1). We impose a maximum SMILES length of $L = 72$, which removes negligible data ($< 0.001\%$) while retaining approximately 427 million drug-like molecules in SMILES format.

4.2. De Novo Generation

Setup. We train SoftBD (89M parameters) on ZINC-Curated, aligning model capacity with leading baselines to ensure fair comparison. We report performance across three dimensions: (1) **Standard Metrics:** Validity, Uniqueness, and Diversity, following MOSES (Polykovskiy et al., 2020); (2) **Quality:** the proportion of unique, valid molecules meeting strict drug-likeness criteria ($\text{QED} \geq 0.6$, $\text{SA} \leq 4$) (Lee et al., 2025); and (3) **Docking-Filter:** a pre-screening proxy for downstream docking viability defined by $\text{QED} > 0.5$ and $\text{SA} < 5$ (Lee et al., 2024a). For rigorous benchmarking, we evaluate all models on identical hardware, generating 10,000 samples per experiment across three independent seeds. Further details are provided in Appendix D.7.

Baselines. We benchmark against two representative state-of-the-art fragment-based models. **SAFE-GPT** (Noutahi et al., 2024) (87M parameters) employs an autoregressive model trained on 1.1 billion SAFE strings from ZINC and UniChem. **GenMol** (Lee et al., 2025) (89M parameters) is a discrete diffusion model trained on the same extensive corpus, distinguished by its high sampling efficiency. Crucially, SoftBD consumes less than 40% of this training data (using only ZINC-Curated), thereby rigorously testing its data efficiency relative to these large-scale pre-trained baselines.

Results. Table 1 highlights SoftBD’s robust generative performance. Despite training on a significantly smaller corpus,

SoftBD matches or exceeds the large-scale baselines SAFE-GPT and GenMol on strict drug-likeness metrics (Quality and Docking-Filter) while preserving superior structural diversity. Notably, the model achieves 100% validity across most configuration settings, effectively refuting the assumption that explicit rule encoding is requisite for syntactic robustness in molecule generation. This robustness is further corroborated in Appendix F.1, where we demonstrate that SoftBD successfully repairs syntactically broken prefixes (e.g., unclosed rings) with near-perfect accuracy, confirming that the model learns intrinsic chemical syntax rather than relying on rigid tokenization rules. We also observe a controllable trade-off between quality and exploration: lower temperatures (τ) and nucleus probabilities (p) shift the distribution towards high-fidelity drug-like candidates, while relaxed constraints facilitate broad exploration. Computationally, SoftBD delivers a $\approx 6.6\times$ acceleration over the discrete diffusion baseline GenMol (Figure 3). Collectively, these gains underscore the synergy between the Block-Diffusion architecture, Adaptive Confidence Decoding, and the high-quality ZINC-Curated training corpus.

4.3. Target-specific molecular design

Setup. Following the protocol of Lee et al. (2024b), we benchmark SoftMol on protein-targeted generation tasks across five targets: parp1, fa7, 5ht1b, braf, and jak2. For each target, we generate 3,000 molecules across three independent runs. The primary evaluation metric is the **Novel Top-hit 5% Score**, defined as the mean docking score (DS) of the top 5% unique and novel generated hits. In contrast to methods such as f -RAG (Lee et al., 2024b) and GEAM (Lee et al., 2024a), which are trained or fine-tuned on relatively small-scale datasets like ZINC250K, SoftBD is pretrained on a large-scale molecular corpus without target-specific adaptation. Accordingly, we define a **Novel Hit** with three strict criteria to emphasize both affinity and drug-likeness:

Table 2. **Novel top-hit 5% docking score (kcal/mol) results.** The results are the means and the standard deviations of 3 runs. Lower is better; the best results are highlighted in **bold** and the second best are underlined.

Method	Target protein				
	parp1	fa7	5ht1b	braf	jak2
REINVENT (Olivecrona et al., 2017)	-8.702 ± 0.523	-7.205 ± 0.264	-8.770 ± 0.316	-8.392 ± 0.400	-8.165 ± 0.277
JTVAE (Jin et al., 2018)	-9.482 ± 0.132	-7.683 ± 0.048	-9.382 ± 0.332	-9.079 ± 0.069	-8.885 ± 0.026
GA+D (Nigam et al., 2019)	-8.365 ± 0.201	-6.539 ± 0.297	-8.567 ± 0.177	-9.371 ± 0.728	-8.610 ± 0.104
Graph-GA (Jensen, 2019)	-10.949 ± 0.532	-7.365 ± 0.326	-10.422 ± 0.670	-10.789 ± 0.341	-10.167 ± 0.576
MORLD (Jeon & Kim, 2020)	-7.532 ± 0.260	-6.263 ± 0.165	-7.869 ± 0.650	-8.040 ± 0.337	-7.816 ± 0.133
HierVAE (Jin et al., 2020a)	-9.487 ± 0.278	-6.812 ± 0.274	-8.081 ± 0.252	-8.978 ± 0.525	-8.285 ± 0.370
G EGL (Ahn et al., 2020)	-9.329 ± 0.170	-7.470 ± 0.013	-9.086 ± 0.067	-9.073 ± 0.047	-8.601 ± 0.038
RationaleRL (Jin et al., 2020b)	-10.663 ± 0.086	-8.129 ± 0.048	-9.005 ± 0.155	<i>No hit found</i>	-9.398 ± 0.076
FREED (Yang et al., 2021)	-10.579 ± 0.104	-8.378 ± 0.044	-10.714 ± 0.183	-10.561 ± 0.080	-9.735 ± 0.022
MARS (Xie et al., 2021a)	-9.716 ± 0.082	-7.839 ± 0.018	-9.804 ± 0.073	-9.569 ± 0.078	-9.150 ± 0.114
PS-VAE (Kong et al., 2022)	-9.978 ± 0.091	-8.028 ± 0.050	-9.887 ± 0.115	-9.637 ± 0.049	-9.464 ± 0.129
MOOD (Lee et al., 2023)	-10.865 ± 0.113	-8.160 ± 0.071	-11.145 ± 0.042	-11.063 ± 0.034	-10.147 ± 0.060
RetMol (Wang et al., 2022)	-8.590 ± 0.475	-5.448 ± 0.688	-6.980 ± 0.740	-8.811 ± 0.574	-7.133 ± 0.242
Genetic GFN (Ko et al., 2023)	-9.227 ± 0.644	-7.288 ± 0.433	-8.973 ± 0.804	-8.719 ± 0.190	-8.539 ± 0.592
GEAM (Lee et al., 2024a)	-12.891 ± 0.158	-9.890 ± 0.116	-12.374 ± 0.036	-12.342 ± 0.095	-11.816 ± 0.067
<i>f</i> -RAG (Lee et al., 2024b)	-12.945 ± 0.053	-9.899 ± 0.205	-12.670 ± 0.144	-12.390 ± 0.046	-11.842 ± 0.316
GenMol (Lee et al., 2025)	-11.773 ± 0.332	-8.967 ± 0.289	-11.914 ± 0.183	-11.394 ± 0.203	-10.417 ± 0.229
SoftMol	<u>-13.977 ± 0.044</u>	<u>-11.031 ± 0.045</u>	<u>-13.733 ± 0.031</u>	<u>-13.252 ± 0.018</u>	<u>-12.580 ± 0.007</u>
SoftMol (Unconstrained)	-14.168 ± 0.072	-11.235 ± 0.047	-13.903 ± 0.030	-13.430 ± 0.032	-12.807 ± 0.039

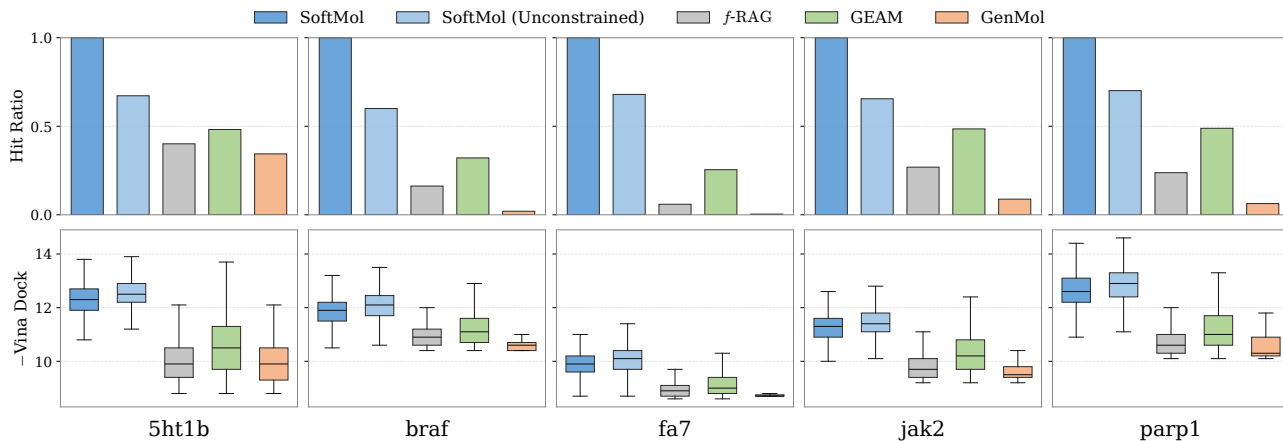


Figure 4. **Hit Ratio and DS Distributions.** **Top:** Hit Ratio, defined as the proportion of unique generated molecules simultaneously satisfying drug-likeness ($QED > 0.5$, $SA < 5.0$) and binding affinity ($DS < \text{median active}$) criteria. **Bottom:** Distribution of negative docking scores ($-DS$) for the identified hits satisfying all three criteria. Higher values indicate stronger affinity.

(1) $DS < \text{median DS of known actives}$; (2) $QED > 0.5$; and (3) $SA < 5.0$. We additionally report:

- **Uniqueness:** The number of unique valid molecules generated out of 3,000 attempts, indicating the absence of mode collapse.
- **Hit Ratio:** The proportion of unique generated molecules that qualify as novel hits, measuring generative success rate.
- **#Circles:** The number of distinct structural clusters among novel hits (Xie et al., 2021b), measuring diversity and coverage of the explored chemical space.

We report results for both **SoftMol** and **SoftMol (Unconstrained)**, as defined in Section 3.4. Implementation details are provided in Appendix D.8.

Baselines. We benchmark against 17 state-of-the-art methods spanning four categories. (1) *Fragment-Based Methods:* **JT-VAE** (Jin et al., 2018), **HierVAE** (Jin et al., 2020a), **MARS** (Xie et al., 2021a), **RationaleRL** (Jin et al., 2020b), **FREED** (Yang et al., 2021), **PS-VAE** (Kong et al., 2022), **GEAM** (Lee et al., 2024a), *f*-**RAG** (Lee et al., 2024b), and **RetMol** (Wang et al., 2022). (2) *Evolutionary Algorithms:* **Graph-GA** (Jensen, 2019), **G EGL** (Ahn et al., 2020), **Genetic-GFN** (Ko et al., 2023), and **GA+D** (Nigam et al., 2019). (3) *Reinforcement Learning:* **REINVENT** (Olivecrona et al., 2017) and **MORLD** (Jeon & Kim, 2020). (4) *Diffusion Models:* **MOOD** (Lee et al., 2023) and **GenMol** (Lee et al., 2025). For the binding affinity results in Table 2, we report the values from Lee et al. (2024b), with the exception of GenMol, which we reproduce using the

Table 3. #Circles of generated hit molecules. The #Circles threshold is set to 0.75. The results are the means and standard deviations of 3 runs. The best results are highlighted in **bold** and the second best are underlined.

Method	Target protein				
	parp1	fa7	5ht1b	braf	jak2
REINVENT (Olivecrona et al., 2017)	44.2 ± 15.5	23.2 ± 6.6	138.8 ± 19.4	18.0 ± 2.1	59.6 ± 8.1
MORLD (Jeon & Kim, 2020)	1.4 ± 1.5	0.2 ± 0.4	22.2 ± 16.1	1.4 ± 1.2	6.6 ± 3.7
HierVAE (Jin et al., 2020a)	4.8 ± 1.6	0.8 ± 0.7	5.8 ± 1.0	3.6 ± 1.4	4.8 ± 0.7
RationaleRL (Jin et al., 2020b)	61.3 ± 1.2	2.0 ± 0.0	312.7 ± 6.3	1.0 ± 0.0	199.3 ± 7.1
FREED (Yang et al., 2021)	34.8 ± 4.9	21.2 ± 4.0	88.2 ± 13.4	34.4 ± 8.2	59.6 ± 8.2
PS-VAE (Kong et al., 2022)	38.0 ± 6.4	18.0 ± 5.9	180.7 ± 11.6	16.0 ± 0.8	83.7 ± 11.9
MOOD (Lee et al., 2023)	86.4 ± 11.2	19.2 ± 4.0	144.4 ± 15.1	50.8 ± 3.8	81.8 ± 5.7
GEAM (Lee et al., 2024a)	123.0 ± 7.8	79.0 ± 9.2	144.3 ± 8.6	84.7 ± 8.6	118.3 ± 0.9
<i>f</i> -RAG (Lee et al., 2024b)	54.7 ± 8.1	31.7 ± 3.1	64.3 ± 18.2	36.0 ± 10.6	54.7 ± 7.4
GenMol (Lee et al., 2025)	3.0 ± 0.0	2.3 ± 0.6	20.3 ± 2.1	6.3 ± 1.5	2.3 ± 0.6
SoftMol	215.0 ± 6.2	254.3 ± 10.2	246.3 ± 21.1	291.7 ± 17.0	330.0 ± 18.5
SoftMol (Unconstrained)	160.3 ± 3.5	213.0 ± 4.6	186.7 ± 10.3	217.7 ± 13.6	264.0 ± 3.0

official implementation ($\delta = 0.4$). For the distributional and diversity metrics in Figure 4, Table 3, and Figure 5, we reproduce all compared baselines using their official implementations to ensure a consistent evaluation protocol.

Results. Binding Affinity. As evidenced in Table 2, a new state-of-the-art is established by SoftMol. Superior binding affinity is consistently achieved across all five protein targets compared to 17 baselines, despite the enforcement of strict pharmacological constraints. Notably, a substantial performance margin is observed over leading fragment-based methods, such as *f*-RAG and GEAM. This effectiveness is attributed to the generative soft-fragment action space, which facilitates the exploration of novel chemotypes beyond the rigid, predefined vocabularies inherent to prior approaches. Furthermore, the theoretical affinity ceiling is quantified by the unconstrained variant, which yields the lowest docking scores across all evaluated models.

Hit Ratio and DS Distributions. Figure 4 evaluates generation efficiency and hit quality. SoftMol consistently achieves a **100% Hit Ratio** across all targets, confirming that the feasibility gate effectively constrains the search to the drug-like manifold without impeding the discovery of high-affinity binders. Conversely, the lower hit ratios observed in baselines (e.g., GenMol, *f*-RAG) indicate a failure to balance affinity optimization with physicochemical validity. Notably, SoftMol (Unconstrained) outperforms most baselines in hit ratio even without explicit gating, suggesting that the pharmaceutically relevant chemical space is intrinsically captured by the pretrained prior. Furthermore, the docking score distributions (bottom) demonstrate that SoftMol closely approximates the affinity ceiling established by the unconstrained variant, implying that strict pharmacological compliance incurs negligible performance cost.

Diversity and Exploration. Finally, we investigate whether the improvements in binding affinity come at the cost of generative diversity, specifically regarding mode collapse. As shown in Figure 5, both SoftMol configurations consistently generate nearly 3,000 unique candidates per 3,000

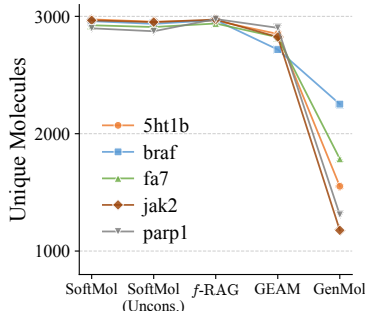


Figure 5. Number of unique molecules per 3,000 attempts. SoftMol maintains high uniqueness.

attempts. This confirms that the MCTS policy maintains robust exploration, avoiding the local optima traps that lead to severe redundancy in baselines like GenMol. Beyond uniqueness, structural diversity is quantified using the #Circles metric (Table 3). SoftMol achieves 2–3 \times higher diversity than the leading baseline (GEAM), validating that the generative soft-fragment space enables the construction of novel chemotypes that transcend the rigid vocabularies of retrieval-based methods. Interestingly, SoftMol (Unconstrained) exhibits slightly reduced diversity compared to the constrained variant. We attribute this to aggressive exploitation: without feasibility barriers, the search converges more intensely toward deep high-affinity basins. Nevertheless, even in this unconstrained regime, SoftMol significantly outperforms all baselines, demonstrating that its exploratory power is intrinsic to the representation and resilient to strong optimization pressures.

5. Analysis of Soft-Fragment Length

We investigate how the soft-fragment length governs the trade-off between global autoregressive modeling and local bidirectional diffusion. Unlike prior methods that couple training and generation granularities (e.g., token-level AR or fixed-fragment diffusion), SoftMol decouples them: K_{train} acts as the representation granularity for learning chemical syntax, while K_{sample} serves as a flexible inference knob controlling search step size. To characterize this interaction, we systematically train models with varying representation granularities $K_{\text{train}} \in \{1, 2, 3, 4, 6, 8, 12, 24, 36, 72\}$ and evaluate each across the same set of sampling granularities K_{sample} . The resulting performance landscape, visualized in Figure 6, reveals how SoftBD balances representation learning with decoding dynamics. Experimental details and further efficiency analysis are provided in Appendix D.9.

Robustness of Pharmacological Quality. Figure 6 (top row) identifies a broad “High-Performance Plateau” for $K_{\text{train}} \in [4, 12]$, where the model consistently achieves near-perfect Validity ($> 99.9\%$) and high Quality ($> 80\%$)

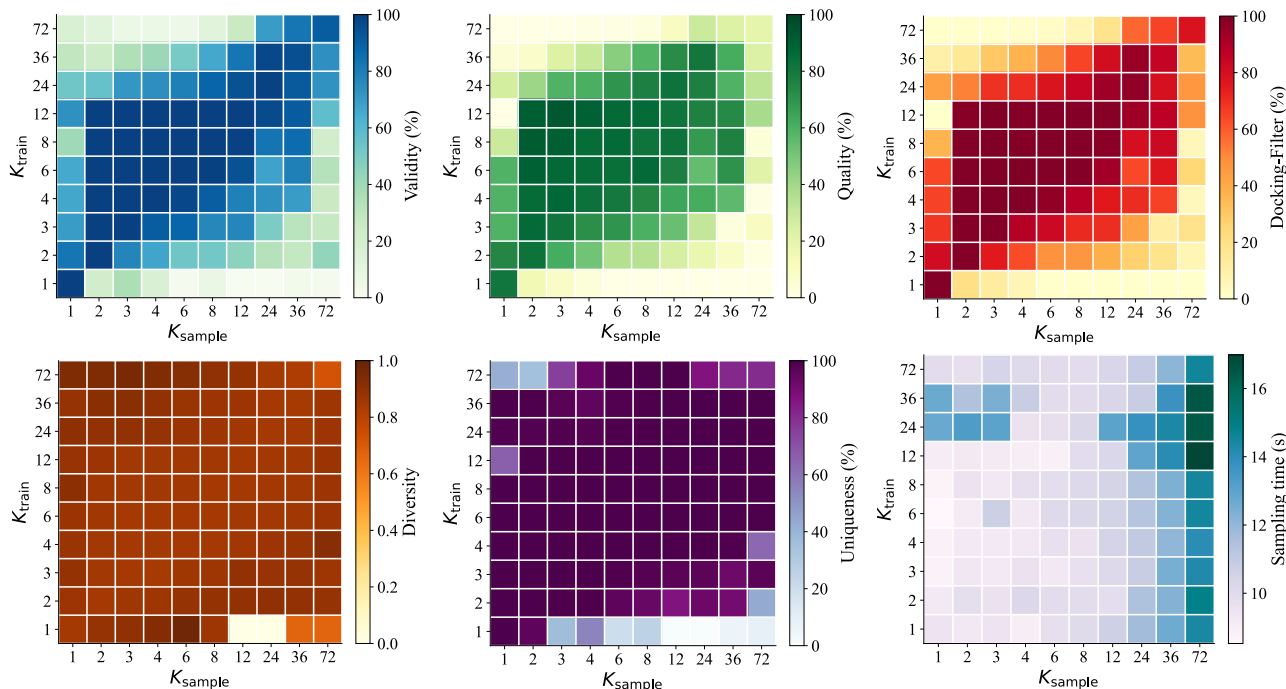


Figure 6. **Effect of Soft-Fragment Length.** Heatmaps display Validity, Quality, Docking-Filter, Diversity, Uniqueness, and Sampling Time across the full $K_{\text{train}} \times K_{\text{sample}}$ grid.

across a wide range of sampling granularities ($K_{\text{sample}} \in [2, 12]$). Similarly, the Docking-Filter rate follows this same high-performance plateau, confirming that drug-likeness capabilities are preserved whenever representation and sampling steps are aligned. This plateau highlights the stability of the soft-fragment representation. Performance degrades only at the pathological extremes: at $K_{\text{train}} = 1$, the lack of local bidirectional context leads to brittle syntax handling (Validity drops); at $K_{\text{train}} \geq 24$, the diffusion decoder becomes under-conditioned, failing to reconstruct complex substructures (Quality drops). Crucially, within the stable region, the model’s grasp of chemical syntax is not tied to a specific tokenization boundary.

Diversity and Computational Efficiency. Regarding exploration capability, the Diversity metric (bottom-left) remains consistently high across the entire grid, indicating that the soft-fragment representation robustly covers chemical space without collapsing into repetitive modes. Complementing this, Uniqueness (bottom-middle) generally mirrors Quality, remaining above 99% across the stability region and dropping only due to mismatched granularities. However, Sampling Time (bottom-right) reveals a critical practical constraint: while latency remains stable at ≈ 9 –10 s for blocks $K_{\text{sample}} \leq 12$, it climbs to 13–17 s for extremely large blocks ($K_{\text{sample}} \geq 24$). This occurs because the diffusion attention mechanism scales quadratically with block length, outweighing the benefits of fewer autoregressive steps. Although $K_{\text{sample}} = 1$ offers low latency, it severely degrades pharmacological Quality. Taken together, the

range $K_{\text{sample}} \in [2, 12]$ offers the optimal trade-off between quality and efficiency.

Task-Adaptive Granularity Selection. The decoupled nature of SoftMol enables inference-time adaptation without retraining. Leveraging the robustness observed in the plateau, we select $K_{\text{train}} = 8$ as our canonical configuration and vary K_{sample} to maximize performance for specific downstream tasks. For *de novo* generation, we use a finer granularity ($K_{\text{sample}} = 2$) to maximize local precision and Quality. Conversely, for target-specific optimization via MCTS, we employ a coarser granularity ($K_{\text{sample}} = 8$) to expand larger valid substructures per step, effectively reducing the search depth and computational cost. This flexibility—dictated by the task rather than the training scheme—is a unique advantage of the computation-centric soft-fragment paradigm.

6. Ablation Studies

To rigorously validate the design of SoftMol, we dissect the framework into two core dimensions: *Generative Foundations* and *Search & Inference Dynamics*. First, we quantify the efficiency gains from our decoding strategies (Table 4) and isolate the impact of data quality and representation efficiency (Table 5). Second, we analyze the scaling behavior of the model and the interplay between granularity and search budget in Gated MCTS (Figure 7). Detailed experimental protocols are provided in Appendix D.10.

Table 4. Ablation of Adaptive Confidence Decoding. The results represent the means and standard deviations across 3 independent runs, with 1,000 molecules generated per run. All results use $K_{\text{sample}} = 8$. The best results are highlighted in **bold**.

Components			Metrics			
FH	GCD	Batch	Validity (%)	Quality (%)	Diversity	Time (s)
OFF	OFF	OFF	88.6 ± 1.0	55.4 ± 1.5	0.855 ± 0.001	1308.4 ± 7.4
ON	OFF	OFF	90.1 ± 0.5	55.2 ± 1.8	0.853 ± 0.000	512.5 ± 14.7
ON	ON	OFF	100.0 ± 0.0	80.4 ± 0.9	0.845 ± 0.001	471.6 ± 32.7
ON	OFF	ON	90.6 ± 1.0	55.7 ± 0.7	0.854 ± 0.000	9.6 ± 0.5
ON	ON	ON	100.0 ± 0.0	81.9 ± 2.0	0.845 ± 0.001	10.3 ± 0.9

Ablation of Adaptive Confidence Decoding. Table 4 quantifies the hierarchical contribution of each inference component. First, *First-Hitting Sampling* (FH) yields a $2.6\times$ speedup by dynamically eliminating redundant denoising steps. Second, *Greedy Confidence Decoding* (GCD) functions as a critical quality enhancement, bridging the validity gap (88.6% \rightarrow 100.0%) and substantially improving pharmacological quality by 26% (55.4% \rightarrow 81.9%). Third, *Batched Inference* (Batch) unlocks massive parallelism, amplifying throughput by $46\times$ compared to non-batched execution. Collectively, this integrated strategy achieves a $130\times$ acceleration while guaranteeing perfect structural validity, facilitating efficient large-scale screening.

Ablation of Pretraining Corpus. Modern NLP has established that high-quality generation requires both pre-training and post-training alignment (Achiem et al., 2023). We demonstrate that a carefully curated pretraining corpus alone can substantially worsen pharmacological quality without requiring additional post-training techniques such as DPO or reinforcement learning. To disentangle architectural inductive biases from data distribution effects, we compare SoftBD models trained on raw SMILES versus our curated ZINC-Curated (Table 5). We observe that in our experiments, validity remains high across both corpora, suggesting that the soft-fragment Block-Diffusion architecture effectively maintains structural correctness. In contrast, the pharmacological profile faithfully mirrors the training distribution: transitioning to the curated dataset yields a substantial gain in Quality with negligible degradation in Diversity. This demonstrates that SoftBD acts as a high-fidelity generative model that effectively decouples syntax learning from property optimization, while simultaneously validating our dataset’s contribution in shifting the generative Pareto frontier by filtering non-drug-like noise without sacrificing chemical space diversity.

Ablation of Representation Efficiency. We additionally evaluate the SAFE representation (Noutahi et al., 2024) under our Block-Diffusion architecture. Despite SAFE’s design goal of enforcing chemical validity through rule-based fragmentation, SAFE dataset yields lower Validity (91.9%) and Quality (59.0%) compared to both raw SMILES and ZINC-Curated (Table 5). We attribute this to two factors: (1)

Table 5. Ablation of pretraining corpus and model scale. The results represent the means and standard deviations across 3 independent runs, with 1,000 molecules generated per run. All results use $K_{\text{sample}} = 8$. The best results are highlighted in **bold**.

Dataset	Params	Validity (%)	Quality (%)	Diversity	Time (s)
SMILES	55M	100.0 ± 0.0	61.4 ± 1.7	0.854 ± 0.001	5.357 ± 0.660
	72M	99.9 ± 0.0	61.4 ± 2.1	0.852 ± 0.000	8.167 ± 0.339
SAFE	74M	91.9 ± 0.0	59.0 ± 0.0	0.870 ± 0.002	9.487 ± 0.434
	55M	99.9 ± 0.1	81.0 ± 0.7	0.845 ± 0.001	5.975 ± 0.565
ZINC-Curated	72M	99.9 ± 0.1	80.6 ± 0.3	0.845 ± 0.001	7.865 ± 0.586
	89M	100.0 ± 0.0	81.9 ± 2.0	0.845 ± 0.001	8.896 ± 0.377
	116M	100.0 ± 0.0	80.7 ± 0.4	0.844 ± 0.000	10.976 ± 0.667
	624M	100.0 ± 0.0	81.0 ± 0.4	0.846 ± 0.001	33.965 ± 0.928

SAFE’s auxiliary connectivity tokens and variable-length fragments inflate sequence complexity, leading to higher token consumption per molecule; and (2) the rigid fragmentation heuristics conflict with our architecture’s inductive bias, which learns chemical syntax directly from fixed-length soft-fragments. These results suggest that explicit rule-based representations are not only unnecessary for validity but may actively hinder generation quality when combined with architectures designed for computation-centric representations.

Ablation of Model Scaling. Given that the curated ZINC-Curated dataset yields superior pharmacological quality, we adopt it as the standard corpus to examine the scaling properties of SoftMol. We train models ranging from 55M to 624M parameters and observe a rapid saturation of generation quality: a compact 55M model already attains strong performance in Quality and Docking-Filter, with only marginal fluctuations when scaling to 624M (Table 5). Sampling latency nevertheless grows roughly linearly with parameter count. These findings demonstrate that the superior performance of SoftMol over fragment-based baselines is driven by the efficiency of the representation and architecture rather than brute-force model scaling. Consequently, the 89M model utilized in our main experiments represents a Pareto-optimal choice, delivering saturated quality with a highly favorable computational footprint.

Ablation of Gated MCTS. We conduct a comprehensive ablation study to isolate the impact of Gated MCTS configuration and SoftBD inference parameters, performing 360 experiments (9 variables \times 4 settings \times 5 targets \times 2 models) by varying individual parameters from the default configuration (Table 9). Detailed analysis is provided in Appendix D.10.

Granularity and budget interaction. Figure 7 investigates the interplay between soft-fragment granularity K_{sample} and search budget N_{max} . Optimal granularity proves to be dynamic: under a strict budget ($N_{\text{max}} = 1000$), coarser fragments ($K_{\text{sample}} = 8$) maximize efficiency by reaching valid chemical space with fewer steps; with increased compute ($N_{\text{max}} \in \{5000, 10000\}$), finer granularity ($K_{\text{sample}} = 4$) yields superior docking scores by enabling more precise

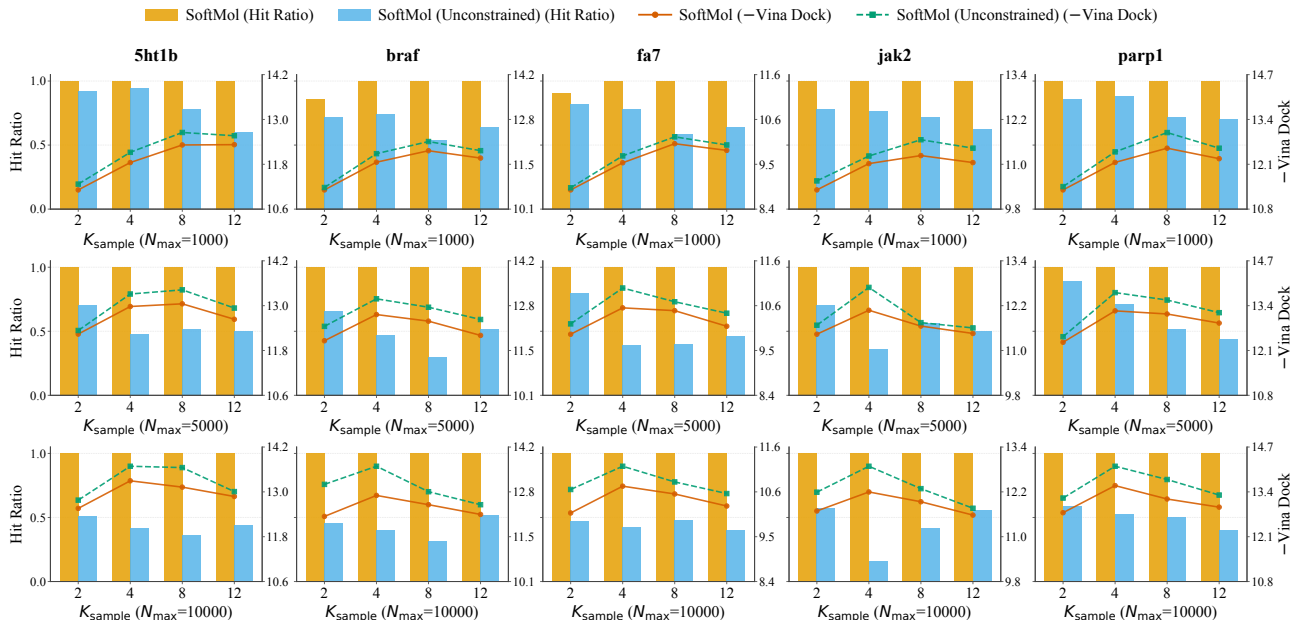


Figure 7. Ablation on Granularity and Search Budget. Hit Ratio and docking scores for varying K_{sample} across different search budgets $N_{\max} \in \{1000, 5000, 10000\}$. Results are averaged across 5 targets from 50 independent runs.

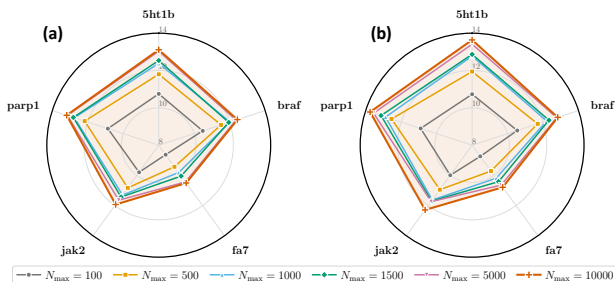


Figure 8. Search Budget Scaling. (a) SoftMol. (b) SoftMol (Unconstrained). DS across five targets under varying search budgets at $K_{\text{sample}} = 8$. Results are averaged over 50 independent runs.

local optimization. Crucially, fixed extrema fail: fragment-level steps ($K_{\text{sample}} = 12$) limit tree depth to 4–6 levels, restricting combinatorial complexity, whereas atom-level steps ($K_{\text{sample}} = 2$) yield incomplete molecules even at $N_{\max} = 10000$. From the Hit Ratio perspective, SoftMol maintains near-perfect success rates ($> 99\%$) across all settings, validating the effectiveness of the feasibility gate in confining the search to the drug-like manifold. In contrast, the unconstrained baseline exhibits significant degradation at extreme granularities ($K_{\text{sample}} = 2$ and $K_{\text{sample}} = 12$), confirming that the gate is essential for navigating the trade-off between affinity optimization and chemical validity.

Search budget scaling. Figure 8 characterizes the compute-performance trade-off across search budgets spanning two orders of magnitude. Three trends emerge: (1) *Monotonic improvement*: binding affinity increases with budget, with steepest gains at $N_{\max} \leq 1000$; (2) *Gradual saturation*: performance plateaus beyond $N_{\max} = 5000$ at our default

granularity ($K_{\text{sample}} = 8$), indicating sufficient exploration of the current search space; notably, as suggested by Figure 7, employing a finer granularity (e.g., $K_{\text{sample}} = 4$) could potentially unlock further gains by extending the optimization horizon given a larger budget; (3) *Constraint cost*: the gap between configurations quantifies the affinity penalty of strict pharmacological compliance, which remains constant across budgets. Notably, even $N_{\max} = 100$ exceeds many baselines in Table 2, confirming the efficiency of the soft-fragment action space.

7. Conclusion

This work presents SoftMol, a unified framework for target-aware molecular generation that systematically co-designs representation, model architecture, and search strategy. Central to this approach is the soft-fragment representation, a rule-free block formulation that enables diffusion-native modeling with tunable granularity. Building on this foundation, the SoftBD architecture implements the first molecular block-diffusion language model, synergizing intra-block bidirectional denoising with inter-block autoregressive conditioning. To ensure high-throughput sampling without compromising structural validity, *Adaptive Confidence Decoding* is integrated, while a gated MCTS mechanism explicitly decouples binding affinity optimization from drug-likeness constraints. Empirically, SoftMol resolves the trade-off between generation quality and efficiency: it achieves 100% chemical validity and a $6.6\times$ speedup, while delivering a 9.7% improvement in binding affinity and $2\text{--}3\times$ higher diversity compared to state-of-the-art methods.

Impact Statement

SoftMol reduces computational barriers in target-specific molecular design, potentially accelerating therapeutic development. However, this efficiency could be misused to design harmful compounds, as the model does not distinguish between therapeutic and toxic targets. Responsible deployment requires toxicity filters, ethical oversight, and controlled access to sensitive optimization objectives.

References

- Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altschmidt, J., Altman, S., Anadkat, S., et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.
- Ahn, S., Kim, J., Lee, H., Shin, J., and Yoon, S. J. Guiding deep molecular optimization with genetic exploration. In *Advances in Neural Information Processing Systems*, volume 33, pp. 12008–12019, 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, 2015.
- Arriola, M., Gokaslan, A. K., Chiu, J. T., Yang, Z., Qi, Z., Han, J., Sahoo, S. S., and Kuleshov, V. Block diffusion: Interpolating between autoregressive and diffusion language models. In *International Conference on Learning Representations*, 2025.
- Atz, K., Grisoni, F., and Schneider, G. Geometric deep learning on molecular representations. *Nature Machine Intelligence*, 3(12):1023–1032, 2021.
- Austin, J., Johnson, D. D., Ho, J., Tarlow, D., and van den Berg, R. Structured denoising diffusion models in discrete state-spaces. In *Advances in Neural Information Processing Systems*, volume 34, pp. 17981–17993, 2021.
- Bagal, V., Aggarwal, R., Vinod, P. K., and Priyakumar, U. D. Molgpt: Molecular generation using a transformer-decoder model. *Journal of Chemical Information and Modeling*, 62(9):2064–2076, 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. *Wiley Interdisciplinary Reviews: Computational Molecular Science*, 12(5):e1608, 2022.
- Chambers, J., Davies, M., Gaulton, A., Hersey, A., Velankar, S., Petryszak, R., Hastings, J., Bellis, L., and Overington, J. P. Unichem: a unified chemical structure cross-referencing and identifier tracking system. *Journal of cheminformatics*, 5:1–13, 2013.
- Chaslot, G., Bakkes, S., Szita, I., and Spronck, P. Monte-carlo tree search: A new framework for game ai. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, volume 4, pp. 216–217, 2008.
- Chen, H., Engkvist, O., Wang, Y., Olivecrona, M., and Blaschke, T. The rise of deep learning in drug discovery. *Drug Discovery Today*, 23(6):1241–1250, 2018.
- Chithrananda, S., Grand, G., and Ramsundar, B. Chemberta: Large-scale self-supervised pretraining for molecular property prediction. *arXiv preprint arXiv:2010.09885*, 2020.
- Chitsaz, K., Balaji, R., Fournier, Q., Bhatt, N. P., and Chandar, S. Novomolgen: Rethinking molecular language model pretraining. *arXiv preprint arXiv:2508.13408*, 2025.
- De Cao, N. and Kipf, T. Molgan: An implicit generative model for small molecular graphs. *arXiv preprint arXiv:1805.11973*, 2018.
- Degen, J., Wegscheid-Gerlach, C., Zaliani, A., and Rarey, M. On the art of compiling and using “drug-like” chemical fragment spaces. *ChemMedChem*, 3(10):1503–1507, 2008.
- 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):1–11, 2009.
- Flam-Shepherd, D., Zhu, K., and Aspuru-Guzik, A. Language models can learn complex molecular distributions. *Nature Communications*, 13(1):3293, 2022.
- Goel, M., Raghunathan, S., Laghuvarapu, S., and Priyakumar, U. D. MoleGuLAR: Molecule generation using reinforcement learning with alternating rewards. *Journal of Chemical Information and Modeling*, 61(12):5815–5826, 2021.
- Grechishnikova, D. Transformer neural network for protein-specific de novo drug generation as a machine translation problem. *Scientific reports*, 11(1):321, 2021.
- Han, X., Kumar, S., and Tsvetkov, Y. Ssd-lm: Semi-autoregressive simplex-based diffusion language model for text generation and modular control. In *Association for Computational Linguistics*, 2022.
- Ho, J., Jain, A., and Abbeel, P. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, pp. 6840–6851, 2020.

- 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. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1, 2021a.
- 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. *arXiv preprint arXiv:2102.09548*, 2021b.
- Irwin, J. J., Sterling, T., Mysinger, M. M., Bolstad, E. S., and Coleman, R. G. Zinc: a free tool to discover chemistry for biology. *Journal of chemical information and modeling*, 52(7):1757–1768, 2012.
- Jensen, J. H. A graph-based genetic algorithm and generative model/monte carlo tree search for the exploration of chemical space. *Chemical Science*, 10(12):3567–3572, 2019.
- Jeon, W. J. and Kim, D. Autonomous molecule generation using reinforcement learning and docking. *Scientific Reports*, 10(1):22104, 2020.
- Ji, J., Yang, Z., Xu, D., Bai, R., Li, J., Hou, T., and Zhu, Z. Toward closed-loop molecular discovery via language model, property alignment and strategic search, 2025. URL <https://arxiv.org/abs/2512.09566>.
- Jin, W., Barzilay, R., and Jaakkola, T. Junction tree variational autoencoder for molecular graph generation. In *International conference on machine learning*, pp. 2323–2332. PMLR, 2018.
- Jin, W., Barzilay, R., and Jaakkola, T. Hierarchical generation of molecular graphs using structural motifs. *arXiv preprint*, 2020a.
- Jin, W., Barzilay, R., and Jaakkola, T. Multi-objective molecule generation using interpretable substructures. In *International conference on machine learning*, pp. 4849–4859. PMLR, 2020b.
- Ko, S. et al. Genetic-guided gflownets for sample efficient molecular optimization. In *Advances in Neural Information Processing Systems*, 2023.
- Kong, X., Huang, W., Tan, Z., and Liu, Y. Molecule generation by principal subgraph mining and assembling. *Advances in Neural Information Processing Systems*, 35: 2550–2563, 2022.
- Krenn, M., Häse, F., Nigam, A., Friederich, P., and Aspuru-Guzik, A. Self-referencing embedded strings (SELFIES): A 100% robust molecular string representation. *Machine Learning: Science and Technology*, 1(4):045024, 2020.
- Kusner, M. J., Paige, B., and Hernández-Lobato, J. M. Grammar variational autoencoder. In *Proceedings of the International Conference on Machine Learning*, 2017.
- Landrum, G. et al. Rdkit: Open-source cheminformatics, 2016. <http://www.rdkit.org>.
- Lee, S., Jo, J., and Hwang, S. J. Exploring chemical space with score-based out-of-distribution generation. In *International Conference on Machine Learning*, pp. 18872–18892. PMLR, 2023.
- Lee, S., Lee, S., Kawaguchi, K., and Hwang, S. J. Drug discovery with dynamic goal-aware fragments. *Proceedings of the 41th International Conference on Machine Learning*, 2024a.
- Lee, S., Kreis, K., Veccham, S. P., Liu, M., Reidenbach, D., Peng, Y., Paliwal, S. G., Nie, W., and Vahdat, A. Genmol: A drug discovery generalist with discrete diffusion. In *Forty-second International Conference on Machine Learning*, 2025. URL <https://openreview.net/forum?id=KM7pXWG1xj>.
- Lee, S. et al. Molecule generation with fragment retrieval augmentation (*f*-rag). *Advances in Neural Information Processing Systems*, 37, 2024b.
- Lewell, X. Q., Judd, D. B., Watson, S. P., and Hann, M. M. Recap retrosynthetic combinatorial analysis procedure: a powerful new technique for identifying privileged molecular fragments with useful applications in combinatorial chemistry. *Journal of chemical information and computer sciences*, 38(3):511–522, 1998.
- Li, X. et al. Smiles pair encoding: A data-driven substructure tokenization algorithm for deep learning. *Journal of Chemical Information and Modeling*, 61(4):1560–1569, 2021.
- Nigam, A., Pollice, R., and Aspuru-Guzik, A. Augmenting genetic algorithms with deep neural networks for exploring chemical space. In *International Conference on Learning Representations*, 2019.
- Nigam, A. et al. Group selfies: A robust fragment-based molecular string representation. *arXiv preprint arXiv:2211.13322*, 2022.
- Noutahi, E. et al. Gotta be safe: a new framework for molecular design. *Digital Discovery*, 3:796–804, 2024.
- Olivecrona, M., Blaschke, T., Engkvist, O., and Chen, H. Molecular de-novo design through deep reinforcement learning. *Journal of Cheminformatics*, 9(1):1–14, 2017.
- Peng, X. et al. Pocket2mol: Efficient molecular sampling based on 3d protein pockets. In *International Conference on Machine Learning*, 2022.

- Polishchuk, P. G., Madzhidov, T. I., and Varnek, A. Estimation of the size of drug-like chemical space based on gdb-17 data. *Journal of Computer-Aided Molecular Design*, 27(8):675–679, 2013.
- Polykovskiy, D., Zhebrak, A., Sanchez-Lengeling, B., Golovanov, S., Tatanov, O., Belyaev, S., Kurbanov, R., Artamonov, A., Aladinskiy, V., Veselov, M., et al. Molecular sets (moses): a benchmarking platform for molecular generation models. *Frontiers in pharmacology*, 11:565644, 2020.
- Reymond, J.-L. The chemical space project. *Accounts of Chemical Research*, 48(3):722–730, 2015.
- Sahoo, S. S., Arriola, M., Schiff, Y., Gokaslan, A., Marroquin, E., Chiu, J. T., Rush, A., and Kuleshov, V. Simple and effective masked diffusion language models. *Advances in Neural Information Processing Systems*, 37, 2024.
- Sterling, T. and Irwin, J. J. Zinc 15–ligand discovery for everyone. *Journal of Chemical Information and Modeling*, 55(11):2324–2337, 2015.
- Tano, K., Hachino, T., Kaji, D., and Harada, K. Avoiding reward hacking in multi-objective molecular design. *Journal of Chemical Information and Modeling*, 64(18):5555–5568, 2024.
- Tian, Y. et al. Toward self-improvement of llms via imagination, searching, and criticizing. *Advances in Neural Information Processing Systems*, 37, 2024.
- Tom, G., Yu, E., Yoshikawa, N., Jorner, K., and Aspuru-Guzik, A. Stereochemistry-aware string-based molecular generation. *PNAS Nexus*, 4(11):pgaf329, 2025.
- Vamathevan, J., Clark, D., Czodrowski, P., Dunham, I., Ferran, E., Lee, G., Li, B., Madabhushi, A., Shah, P., Spitzer, M., et al. Applications of machine learning in drug discovery and development. *Nature Reviews Drug Discovery*, 18(6):463–477, 2019.
- Vignac, C., Krawczuk, I., Siraudin, A., Wang, B., Cevher, V., and Frossard, P. Digress: Discrete denoising diffusion for graph generation. In *International Conference on Learning Representations*, 2023.
- Wang, Z., Nie, W., Qiao, Z., Xiao, C., Baraniuk, R., and Anandkumar, A. Retrieval-based controllable molecule generation. *arXiv preprint arXiv:2208.11126*, 2022.
- Weininger, D. Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules. *Journal of Chemical Information and Computer Sciences*, 28(1):31–36, 1988.
- Wu, J.-N., Wang, T., Chen, Y., Tang, L.-J., Wu, H.-L., and Yu, R.-Q. t-smiles: a fragment-based molecular representation framework for de novo ligand design. *Nature Communications*, 15:4993, 2024.
- Xie, Y., Shi, C., Zhou, H., Yang, Y., Zhang, W., Yu, Y., and Li, L. Mars: Markov molecular sampling for multi-objective drug discovery. In *International Conference on Learning Representations*, 2021a.
- Xie, Y., Xu, Z., Ma, J., and Mei, Q. How much space has been explored? measuring the chemical space covered by databases and machine-generated molecules. *arXiv preprint arXiv:2112.12542*, 2021b.
- Yang, S., Hwang, D., Lee, S., Yan, S., and Kim, S. Hit and lead discovery with explorative rl and fragment-based molecule generation. In *Advances in Neural Information Processing Systems*, volume 34, pp. 7927–7939, 2021.
- Yang, X., Zhang, J., Yoshizoe, K., Terayama, K., and Tsuda, K. Chemts: an efficient python library for de novo molecular generation. *Science and technology of advanced materials*, 18(1):972–976, 2017.
- Zheng, K., Chen, Y., Mao, H., Liu, M., Zhu, J., and Zhang, Q. Masked diffusion models are secretly time-agnostic masked models and exploit inaccurate categorical sampling. *arXiv preprint arXiv:2409.02908*, 2024.
- Zhou, Z., Kearnes, S., Li, L., Zare, R. N., and Riley, P. Optimization of molecules via deep reinforcement learning. *Scientific Reports*, 9(1):1–10, 2019.

A. Related Work

A.1. String-based Molecular Representations

Molecular string representations have evolved from atom-level linearizations to more structured forms, encompassing syntax-constrained strings and rule-based fragmentation schemes. While SMILES (Weininger, 1988) serves as the standard linearization for its compactness, it lacks syntactic robustness, frequently leading to invalid valencies or unclosed rings in generative outputs (Kusner et al., 2017). Although SELFIES (Krenn et al., 2020) guarantees validity via a recursive derivation system, this robustness often compromises sequence compactness and interpretability.

To explicitly capture local chemical semantics, rule-based decomposition strategies have been widely adopted. Approaches such as SAFE (Noutahi et al., 2024), JT-VAE (Jin et al., 2018), and HierVAE (Jin et al., 2020a) decompose molecules into motif-level subgraphs (e.g., rings and linkers) based on chemical axioms like BRICS (Degen et al., 2008) or RECAP (Lewell et al., 1998). These methods ensure chemical validity of sub-components but suffer from rigid fragmentation rules and inflated sequence lengths due to auxiliary connectivity tokens (Figure 1). Conversely, data-driven tokenization methods like SPE (Li et al., 2021) and t-SMILES (Wu et al., 2024) attempt to learn substructures statistically. However, their vocabularies remain static after training, lacking the flexibility to dynamically adjust granularity during inference.

Different from these discrete paradigms that depend on explicit chemical axioms or static vocabularies, we investigate a rule-free block representation strategy. By partitioning sequences into generic fixed-length segments, this approach removes the reliance on heuristic fragmentation and auxiliary connectivity tokens. This shifts the paradigm from rigid, rule-based preprocessing to a computation-centric representation, delegating the learning of chemical syntax and semantics to the generative model itself rather than enforcing it through static tokenization rules.

A.2. Generative Models for Molecular Design

Autoregressive Models. Autoregressive architectures represent the canonical framework for molecular string generation, wherein the joint distribution is factorized into a sequence of conditional probabilities. While high validity is achieved by Transformer-decoder architectures such as MolGPT (Bagal et al., 2021), a fundamental inductive bias mismatch is introduced by their strictly causal factorization: molecules are intrinsically undirected graphs governed by bidirectional interactions rather than unidirectional sequence histories (Jin et al., 2018; De Cao & Kipf, 2018). Consequently, the modeling of local chemical contexts is impeded, as the validity of topological substructures depends on their complete neighborhood (Atz et al., 2021; Vignac et al., 2023). Furthermore, a significant computational bottleneck is created by the $\mathcal{O}(L)$ latency inherent to sequential token-by-token generation.

Diffusion Models. Discrete diffusion constitutes a powerful non-autoregressive paradigm, enabling bidirectional context modeling and parallel generation. Although this framework has been adapted for molecular strings (Lee et al., 2025; Sahoo et al., 2024), a structural misalignment is introduced by the standard diffusion requirement for fixed-dimensional tensors. This constraint conflicts with the intrinsic nature of small molecules, which are discrete entities of variable length and strict topology (Sterling & Irwin, 2015). Consequently, computational efficiency is compromised by the necessity for extensive padding or complex length-modeling priors, which are suboptimal for the bounded and highly structured chemical space (Kusner et al., 2017; Vignac et al., 2023).

Block Diffusion. Block diffusion architectures (Han et al., 2022; Arriola et al., 2025) offer a hybrid solution by combining autoregressive block planning with parallel intra-block refinement. This paradigm is particularly well-suited for molecular generation, as it reconciles the need for variable-length generation (autoregressive) with the requirement for modeling rigorous local substructures (diffusion). By mitigating the limitations of pure AR and diffusion approaches, this hybrid framework enables efficient, structured generation without reliance on rigid fragmentation rules.

A.3. Target-specific Molecular Design and Search

Reinforcement Learning and Genetic Algorithms. Optimization in chemical space is frequently formulated as a search problem guided by generative models. Reinforcement learning (RL) strategies, notably REINVENT (Olivecrona et al., 2017), cast molecule generation as a sequential decision process, fine-tuning policies via gradient-based updates. While potent, these methods require delicate scalarization of multi-objective rewards (combining affinities with pharmacological constraints) to avert mode collapse and syntactic degradation. This reliance often manifests as parameter sensitivity (Goel et al., 2021) or "reward hacking," where models exploit scoring function artifacts to produce invalid high-scoring structures (Tano et al.,

2024; Tom et al., 2025). Alternatively, evolutionary algorithms like Graph-GA (Jensen, 2019) apply heuristic mutation operators directly to molecular graphs. While sample-efficient, their exploration is inherently constrained to the local chemical neighborhood of the seed population.

Fragment-Library and Retrieval-Based Methods. A complementary paradigm restricts the search space to recombinations of a pre-existing fragment library. Frameworks such as GEAM (Lee et al., 2024a) and f -RAG (Lee et al., 2024b) construct candidates by assembling or retrieving motifs from large corpora. This approach significantly enhances sample efficiency and ensures local chemical validity. However, the achievable chemical variety is fundamentally bounded by the granularity and coverage of the underlying library: the step size is rigidly determined by the fragmentation scheme (e.g., BRICS), impeding the discovery of novel scaffolds absent from the predefined vocabulary.

Monte Carlo Tree Search (MCTS). MCTS offers a principled framework for balancing exploration and exploitation in sequential decision-making (Chaslot et al., 2008). In molecular design, ChemTS (Yang et al., 2017) pioneered token-level MCTS for SMILES, while MolDQN (Zhou et al., 2019) formulated optimization as a graph MDP. More recently, Trio (Ji et al., 2025) introduced fragment-level expansion guided by a pretrained prior. However, existing methods face a granularity dilemma: atom-level search induces intractably deep trees, whereas rigid fragment-based MCTS restricts the reachable manifold. SoftMol addresses this by leveraging the soft-fragment representation to define a *tunable, generative action space*. Our framework enables open-ended exploration with adjustable block granularity, while a **tunable feasibility gate** explicitly decouples constraint satisfaction from property optimization, efficiently guiding search toward high-affinity regions without engaging computationally expensive docking for infeasible candidates.

B. Attention Mask Design

To enable structured learning across both corrupted and clean views of the input, we employ a custom block-aware attention scheme (Arriola et al., 2025). At each training step, we concatenate the noised sequence \mathbf{x}_t and the clean sequence \mathbf{x} into a single input of total length $2L$, then apply a hybrid attention pattern defined via a binary mask $\mathcal{M}_{\text{full}} \in \{0, 1\}^{2L \times 2L}$.

The overall attention mask is decomposed into sub-components corresponding to specific dependency structures:

$$\mathcal{M}_{\text{full}} = \begin{bmatrix} \mathcal{M}_{\text{BD}} & \mathcal{M}_{\text{OBC}} \\ \mathbf{0} & \mathcal{M}_{\text{BC}} \end{bmatrix}, \quad (11)$$

where each component is defined as follows:

- **Block-Diagonal Mask** (\mathcal{M}_{BD}): Enables bidirectional self-attention among tokens within the same block in the noised sequence \mathbf{x}_t . This facilitates local structure healing and refinement:

$$[\mathcal{M}_{\text{BD}}]_{ij} = \begin{cases} 1 & \text{if } i, j \text{ belong to the same block in } \mathbf{x}_t, \\ 0 & \text{otherwise.} \end{cases}$$

- **Offset Block-Causal Mask** (\mathcal{M}_{OBC}): Allows each noised token in \mathbf{x}_t to attend to tokens from strictly preceding blocks in the clean sequence \mathbf{x} , providing global autoregressive conditioning:

$$[\mathcal{M}_{\text{OBC}}]_{ij} = \begin{cases} 1 & \text{if token } j \text{ is in a block strictly preceding that of } i, \\ 0 & \text{otherwise.} \end{cases}$$

- **Block-Causal Mask** (\mathcal{M}_{BC}): Maintains a standard causal mask over the clean sequence \mathbf{x} , ensuring that the autoregressive backbone representations are updated correctly:

$$[\mathcal{M}_{\text{BC}}]_{ij} = \begin{cases} 1 & \text{if token } j \text{ is in the same or an earlier block as } i, \\ 0 & \text{otherwise.} \end{cases}$$

Inference-time Optimization. During inference, we adopt a simplified causal attention mechanism that reuses decoded blocks as a frozen prefix context. As illustrated in Figure 9 (b), previously generated blocks from history $\mathbf{x}^{<b}$ are cached to avoid redundant computation. Only the current noised block \mathbf{x}_t^b is actively refined in each step; it attends bidirectionally within itself (similar to \mathcal{M}_{BD}) while attending causally to the unmasked tokens in the cached history. This design significantly reduces the memory footprint and computational cost by restricting active attention to the current block scale $\mathcal{O}(K_{\text{sample}}^2)$ rather than the full sequence length operations.

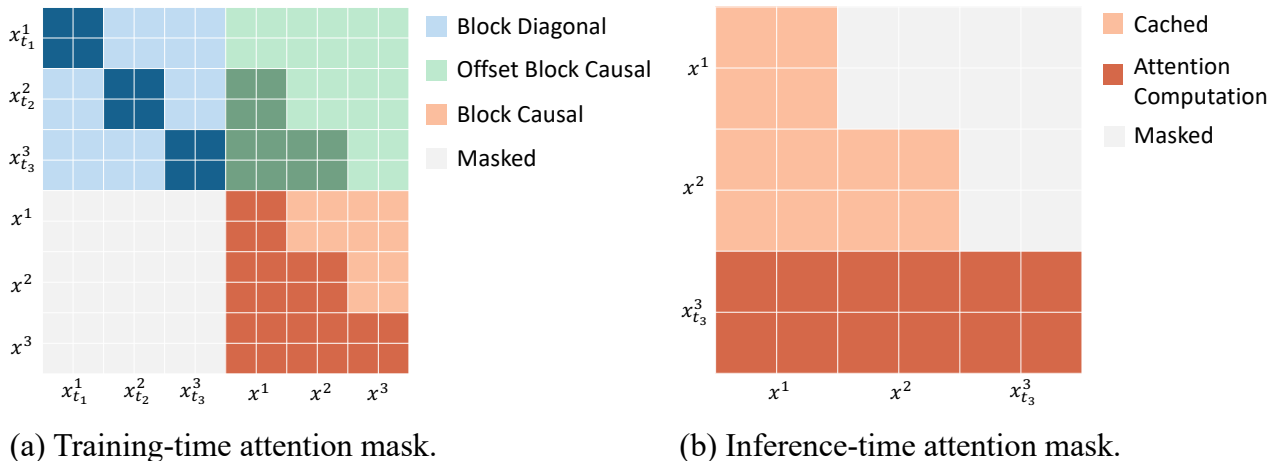


Figure 9. **Attention mask design.** (a) Training-time mask ($\mathcal{M}_{\text{full}}$) combining intra-block bidirectional attention (\mathcal{M}_{BD}), offset block-causal context (\mathcal{M}_{OBC}), and causal clean history (\mathcal{M}_{BC}). (b) Inference-time mask leveraging history caching for blocks $\mathbf{x}^{<b}$ while restricting active computation to the current bidirectional block \mathbf{x}_t^b .

C. Details of Training Objective

For completeness, we summarize the training objective used in SoftBD. While implementation operates in discrete time steps, we derive the objective in the continuous-time limit ($T \rightarrow \infty$) for theoretical completeness, following Sahoo et al. (2024); Arriola et al. (2025). We adopt the Negative Evidence Lower Bound (NELBO) formulation with the *SUBS-parameterization*.

Consider a molecular sequence \mathbf{x} partitioned into B soft-fragments $\{\mathbf{x}^b\}_{b=1}^B$. The joint likelihood is factorized autoregressively over blocks:

$$\log p_{\theta}(\mathbf{x}) = \sum_{b=1}^B \log p_{\theta}(\mathbf{x}^b | \mathbf{x}^{<b}). \quad (12)$$

Each conditional term $\log p_{\theta}(\mathbf{x}^b | \mathbf{x}^{<b})$ is modeled via a discrete diffusion process. By applying the standard variational bound and taking the continuous-time limit ($T \rightarrow \infty$), the objective can be decomposed into prior, reconstruction, and diffusion loss terms. Crucially, SoftMol utilizes the *SUBS-parameterization* (Arriola et al., 2025), which enforces that unmasked tokens are never re-masked during the reverse process ($s < t$). Specifically, for any token index i in block b , we set:

$$p_{\theta}(\mathbf{x}_{s,i}^b = \mathbf{x}_{t,i}^b | \mathbf{x}_{t,i}^b \neq [\text{MASK}]) = 1. \quad (13)$$

Under this constraint, the prior and reconstruction terms vanish.

The resulting objective simplifies to a weighted integral of the cross-entropy loss over time $t \in [0, 1]$:

$$\mathcal{L}_{\text{BD}}(\mathbf{x}^b | \mathbf{x}^{<b}) = \mathbb{E}_{t \sim [0,1]} \mathbb{E}_{q(\mathbf{x}_t^b | \mathbf{x}^b)} \left[-\frac{\alpha'_t}{1 - \alpha_t} \mathcal{L}_{\text{CE}}(\mathbf{x}^b, p_{\theta}(\cdot | \mathbf{x}_t^b, \mathbf{x}^{<b})) \right], \quad (14)$$

where α_t represents the noise schedule (probability of a token remaining unmasked, decaying from 1 to 0) and \mathcal{L}_{CE} is the cross-entropy loss on the masked tokens ($x_{t,i}^b = [\text{MASK}]$). The final training objective is the sum over all blocks:

$$\mathcal{L}_{\text{BD}}(\mathbf{x}; \theta) = \sum_{b=1}^B \mathcal{L}_{\text{BD}}(\mathbf{x}^b | \mathbf{x}^{<b}). \quad (15)$$

This formulation allows us to train the model to predict clean soft-fragments from their corrupted versions, conditioned on the clean history of previous blocks.

D. Experimental Details

D.1. ZINC-Curated Dataset Preprocessing

We constructed the ZINC-Curated dataset by collecting molecules from ZINC-22 (Sterling & Irwin, 2015) and applying a four-stage curation pipeline to ensure pharmaceutical relevance and structural diversity. The procedure, summarized in Algorithm 1, proceeds as follows. First, we perform Physicochemical Filtering to discard molecules with QED ≤ 0.5 or SA ≥ 5 , corresponding to poor drug-likeness or low synthetic accessibility. Second, we enforce Structural Validity by removing compounds that contain undesirable elements such as Si or Sn, carry non-neutral charges, include free radicals, or exhibit overly complex topologies with more than two bridgehead atoms, rings larger than eight members, or more than ten rotatable bonds. We further exclude molecules with TPSA > 140 or known toxic or PAINS substructures. Third, we impose Medicinal Chemistry Rules via Lipinski’s Rule of Five, constraining molecular weight to $100 \leq MW \leq 500$, lipophilicity to $\text{LogP} \leq 5$, and hydrogen bond donor and acceptor counts. Finally, to mitigate dataset bias, we apply Diversity-Aware Stratification by grouping molecules by heavy-atom count between 4 and 49 and retaining only those whose Tanimoto similarity to previously accepted molecules in the same bucket is below 0.5.

D.2. Model Architecture

SoftBD instantiates a discrete block-diffusion Transformer over soft-fragment tokens. Concretely, we follow the DDiT backbone of BD3-LM (Arriola et al., 2025): a stack of Transformer decoder blocks with masked self-attention over the concatenation of previously generated blocks and the current block, combined with position-wise feed-forward layers and residual connections. The model vocabulary \mathcal{V} consists of the atom-level SMILES character set from the training corpus and the control symbols $\{[\text{BOS}], [\text{EOS}], [\text{PAD}], [\text{MASK}]\}$. Input sequences are constructed by wrapping tokenized SMILES strings with start/end tokens as $([\text{BOS}], s_1, \dots, s_n, [\text{EOS}])$ and padding with $[\text{PAD}]$ to a fixed length L (e.g., $L = 72$). Soft-fragments are then obtained by slicing this full fixed-length tensor into blocks as described in Section 3.1. We tie the input token embeddings and output projection weights to reduce the parameter count and use dropout of 0.1 in all layers.

Across experiments, we use a family of SoftBD backbones ranging from 55M to 624M parameters. All models use a 128-dimensional conditioning embedding to encode the diffusion timestep, and employ the scale-by- σ parameterization from BD3-LM. The 89M-parameter model is used for all main *de novo* generation and target-specific molecular design experiments, while the 55M/72M models are used for corpus-sensitivity studies and the 116M/624M variants appear only in the scaling-law analysis in Section 4. Table 6 summarizes the architectural hyperparameters of each backbone.

Table 6. SoftBD backbone configurations used in this work. The maximum context length refers to the model’s positional capacity; all experiments in this paper use SMILES sequences of length $L = 72$.

Params	Layers	Hidden size	Heads	Max context length
55M	10	640	10	512
72M	10	736	8	512
89M	11	784	8	512
116M	15	768	12	512
624M	25	1408	16	1024

D.3. Training Details

All SoftBD backbones in this work share the same optimization and training strategy. We train with the AdamW optimizer (learning rate 3×10^{-4} , $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 10^{-8}$, weight decay = 0) and apply a linear warmup over the first 2,500 steps, after which the learning rate is kept constant. We use mixed-precision training with bfloat16 activations and FP32 master weights via automatic mixed precision, and apply a dropout rate of 0.1 in all Transformer layers. To destabilize training and reduce gradient noise, we maintain an exponential moving average (EMA) of the model parameters with decay 0.9999 and employ antithetic sampling of diffusion noise. We do not use gradient accumulation; all results are obtained with a single update per batch.

D.4. Adaptive Confidence Decoding for SoftBD Details

We detail the algorithmic formulation of the three efficient inference mechanisms introduced in Section 3.3. Algorithm 2 summarizes their integration into a unified generation pipeline.

Analytic First-Hitting Schedule. Methodologically, First-Hitting Sampling acts as the temporal driver describing *when* to sample. By sampling the next event time from the order statistics of the remaining mask count m_t , it determines strictly necessary evaluation points, skipping intervals where no tokens would be unmasked. In the algorithm, this corresponds to the time update step $t \leftarrow t \cdot u^{1/m}$, ensuring that every forward pass contributes to resolving at least one token.

Greedy Confidence Decoding Implementation. At each denoising step, given probability tensor $\mathbf{P} \in \mathbb{R}^{N_{\text{batch}} \times K_{\text{sample}} \times |\mathcal{V}|}$ for the active block, we deterministically select the position-token pair with maximum confidence. Specifically, we compute the confidence $c_j = \max_v \mathbf{P}_{:,j,v}$ for each masked position j , then select $j^* = \operatorname{argmax}_j c_j$ and reveal the corresponding argmax token. This ensures the model commits first to high-certainty substructures before resolving ambiguous features.

Batched block-wise inference with caching. To enable efficient variable-length generation without memory overhead, we maintain a global sequence tensor $\mathbf{X}_{\text{accum}} \in \mathbb{R}^{N_{\text{batch}} \times L_{\text{max}}}$ updated in-place. For block b , we extract a sliding context window $[bK_{\text{sample}} - W, (b + 1)K_{\text{sample}})$ from $\mathbf{X}_{\text{accum}}$, ensuring $O(W^2)$ attention complexity independent of total sequence length. To handle asynchronous termination, we track completion status per sequence: once an [EOS] token is generated, all subsequent positions in that sequence are frozen to [EOS] and excluded from updates. Generation terminates only when all sequences have produced at least one [EOS], ensuring correct tensor alignment across variable-length molecules.

D.5. Hardware Infrastructure

Computations are distributed across hardware configurations tailored to specific task requirements:

- **Pretraining:** SoftBD models exceeding 116M parameters are trained on 8×NVIDIA H100 GPUs, while smaller configurations ($\leq 116\text{M}$) utilize 8×NVIDIA RTX 4090 GPUs.
- **De Novo Generation:** All distribution learning evaluations and inference speed benchmarks (including all baselines) are strictly executed on a single NVIDIA H100 GPU to ensure a controlled comparison.
- **Target-Specific Optimization:** Tasks involving the full SoftMol framework (SoftBD generation coupled with Gated MCTS) leverage a cluster of NVIDIA RTX 4090 GPUs to support parallelized sampling and docking simulations.

Ablation studies adhere to this partition: generative configurations are evaluated on the single H100, while search parameter sweeps utilize the RTX 4090 cluster.

D.6. Evaluation Protocols

We employ standardized metrics to evaluate molecular quality and diversity. Specifically, we use the RDKit library (Landrum et al., 2016) to obtain Morgan fingerprints, while measures of QED, SA, and diversity are calculated using the Therapeutics Data Commons (TDC) library (Huang et al., 2021a). For binding affinity estimation, docking simulations are performed using QuickVina 2 (Alhossary et al., 2015). All candidate molecules are converted to PDBQT format using OpenBabel prior to docking.

D.7. De Novo Generation Details

Model Configuration. We utilize the 89M-parameter SoftBD model trained on the ZINC-Curated dataset. The model was trained for 6 epochs, and we selected the checkpoint exhibiting the lowest validation loss for evaluation.

Inference Strategy. We employ our *Adaptive Confidence Decoding* with a fine-grained inference setting of $K_{\text{sample}} = 2$ to ensure high-fidelity generation. Sampling parameters are configured with nucleus sampling probability $p = 0.95$ and temperature $\tau = 0.9$ specifically for the efficiency analysis presented in Figure 3.

Baselines. When comparing against Safe-GPT and GenMol, we use their official implementations and recommended hyperparameters to ensure fair reproduction.

D.8. Target-specific Molecular Design Details

Task Setup. We target five protein binding sites defined in the TDS benchmark. The search space centers and box sizes are listed in Table 7. We adopt the median docking scores of known active molecules (Table 8) as strict baselines for defining high-affinity hits.

MCTS Configuration. We employ the SoftMol search algorithm (Algorithm 3) utilizing the same SoftBD checkpoint as in Appendix D.7. We use a coarse-grained expansion strategy ($K_{\text{sample}} = 8$). As shown in Appendix D.9, this setting offers an optimal trade-off: it is computationally cheaper than fine-grained generation while providing sufficient structural exploration for scaffold hopping. Table 9 lists the key hyperparameters. We use a Children-Adaptive widening strategy where the expansion width dynamically scales based on the reward dispersion of valid children.

Feasibility Gate & Evaluation. During simulation, we apply a lightweight filter ($\text{QED} > 0.5$, $\text{SA} < 5.0$) before invoking the costly docking oracle. Candidates failing this gate receive a penalty score. For the final evaluation, valid candidates are those that pass the gate, are unique SMILES, and successfully generate a 3D conformer via OpenBabel.

Baselines. To rigorously evaluate the performance of SoftMol, we differentiate between reported and reproduced baseline results:

- **Reported Results:** For the docking scores presented in Table 2, we directly report the values from Lee et al. (2024b) for all methods except GenMol, as the experimental conditions (target proteins and search budget) are consistent.
- **Reproduced Results:** For GenMol and all distributional analyses (Hit Ratio, DS Distribution in Figure 4; Uniqueness in Figure 5; Circle Diversity in Table 3), we fully reproduced the results using the official implementations. This was necessary because distributional metrics and diversity statistics are not uniformly reported in the literature or depend on the specific sampling scale (3,000 attempts in our setting).

We generated 3,000 molecules per run across 3 independent runs for all reproduced baselines to ensure statistical significance.

Table 7. **Docking Experiment Protocol.** Grid box parameters for QuickVina 2 docking. All lengths are in Ångströms.

Target Protein	PDB Source / ID	Box Center (x, y, z)	Box Size (x, y, z)
parp1	4r6e	(26.413, 11.282, 27.238)	(18.521, 17.479, 19.995)
fa7	1kl1	(10.131, 41.879, 32.097)	(20.673, 20.198, 21.362)
5ht1b	4iar	(−26.602, 5.277, 17.898)	(22.500, 22.500, 22.500)
braf	3og7	(84.194, 6.949, −7.081)	(22.032, 19.211, 14.106)
jak2	3ugc	(114.758, 65.496, 11.345)	(19.033, 17.929, 20.283)

Hyperparameter Summary. Table 9 summarizes the MCTS configuration used in target-specific molecular design experiments. The UCT selection mechanism employs exploration constant C and weight λ to balance average reward and peak performance. Search widths are controlled by a Children-Adaptive widening strategy: the root node initializes with C_{init} children; non-root nodes use a base width C_{base} , which the adaptive mechanism dynamically adjusts between C_{min} and C_{max} based on the reward dispersion $I(s)$ and scaling factor β . Expansion generates M candidate next-block soft-fragments per node using batched SoftBD sampling, while simulation completes n_{sim} rollouts to assess node values.

Table 8. **Binding Affinity Thresholds.** The median docking scores (kcal/mol) of known active molecules used as thresholds for defining novel hits. Consistent with Lee et al. (2023), candidates with scores lower than these thresholds are considered high-affinity binders.

Target Protein	Threshold (kcal/mol)
parp1	−10.0
fa7	−8.5
5ht1b	−8.8
braf	−10.3
jak2	−9.1

D.9. Analysis of Soft-Fragment Length Details

For the granularity grid analysis in Section 5, we train independent 89M-parameter SoftBD models on ZINC-Curated ($L = 72$) for 1 epoch and select the final checkpoint. This covers the full spectrum of $K_{\text{train}}, K_{\text{sample}} \in \{1, 2, 3, 4, 6, 8, 12, 24, 36, 72\}$, yielding 100 configurations. For each pair, we generate 1,000 molecules per run using nucleus sampling ($p = 0.95$) and temperature $\tau = 1.0$. All metrics (Validity, Uniqueness, Quality, Diversity, Docking-Filter, and Sampling Time) are averaged over 3 independent runs with seeds 42, 43, and 44. The resulting heatmaps (Figure 6) visualize the full performance landscape.

Table 9. Hyperparameters for target-specific molecular design experiments.

Hyperparameter	Value
MCTS Parameters	
Search budget (N_{\max})	10000
Exploration constant (C)	2.1
UCT weight (λ)	0.5
Children-Adaptive scaling (β)	2
Root initial width (C_{init})	20
Base node width (C_{base})	8
Minimum adaptive width (C_{min})	8
Maximum adaptive width (C_{max})	10
Batch expansion size (M)	64
Rollouts per expansion (n_{sim})	1
Max tree depth (D_{max})	100
Feasibility Gate	
QED threshold (τ_{QED})	0.5
SA threshold (τ_{SA})	5.0
Unconstrained setting	$\tau_{\text{QED}} = 0, \tau_{\text{SA}} = +\infty$
SoftBD Sampling	
Temperature (τ)	1.1
Nucleus probability (p)	1.0
Soft-fragment length (K_{sample})	8
Max context length (L_{max})	512
Diffusion steps (T)	128
Random seeds	42, 43, 44

D.10. Ablation Studies Details

Adaptive Confidence Decoding (Table 4). We utilize the same fully-converged SoftBD checkpoint used in the *de novo* (Appendix D.7) and target-specific molecular design experiments. Inference is performed with $K_{\text{sample}} = 8$, $L = 72$, nucleus sampling $p = 0.95$, and temperature $\tau = 0.9$.

Pretraining Corpus and Model Scaling (Table 5). All models presented in this ablation are trained for 1 epoch, and we select the final checkpoint for evaluation. Sampling is performed with $K_{\text{sample}} = 8$, $L = 72$, $p = 0.95$, and $\tau = 0.9$.

Gated MCTS (Figure 7). We utilize the same checkpoint as in the target-specific molecular design experiments. Due to computational constraints, we generate 50 molecules per target for high-budget configurations ($N_{\text{max}} \in \{5000, 10000\}$), while all other settings generate 100 molecules per target.

E. SoftMol Search Pseudo-code

Algorithm 3 presents the complete SoftMol search procedure. Each iteration follows the standard MCTS lifecycle:

- **Selection:** Traverse the tree using UCT (Eq. 8) to balance exploitation and exploration.
- **Expansion:** Generate M candidate soft-fragments via SoftBD ($K_{\text{sample}} = 8$), filter duplicates, and instantiate a new child.
- **Simulation:** Complete n_{sim} rollouts to [EOS]. Assign $R_{\text{pen}} = -1.0$ for failures; otherwise, reward is $-\text{VINAScore}(\mathbf{x}, P)$.
- **Backpropagation:** Update statistics $N(s)$, $\bar{R}(s)$, and $R^{\text{max}}(s)$ along the path.

The root node initializes with $C_{\text{init}} = 20$ children, while non-root nodes use a base width $C_{\text{base}} = 8$ (adaptively widened up to $C_{\text{max}} = 10$). For SoftMol (Unconstrained), we set $(\tau_{\text{QED}}, \tau_{\text{SA}}) = (0, +\infty)$ to bypass the feasibility gate and probe the affinity ceiling.

Table 10. Prefix completion from syntactically broken SMILES. The model is conditioned on incomplete SMILES prefixes representing common truncation artifacts, including unclosed branches/rings, dangling bonds, and incomplete stereochemical/charge specifications. For each prefix, 1,000 samples are generated across three independent runs ($K_{\text{sample}} = 2$, $\tau = 1.0$, $p = 0.95$).

Prefix	Syntactic Challenge	Validity (%)	Uniqueness (%)	Example Completion
O=C (Unclosed Branch	100.0 ± 0.0	99.6 ± 0.3	O=C (NCC=Cc1ccc (Br) cc1) c1cnn2c1OCCCC2
C1CCCCC	Unclosed Aliphatic Ring	100.0 ± 0.0	99.1 ± 0.0	C1CCCCC1Nc1ccc (C#N) cc1 [N+] (=O) [O-]
c1ccccc	Unclosed Aromatic Ring	100.0 ± 0.0	88.9 ± 1.3	c1ccccc1Nc1ccc (C2=NCCO2) cc1
c1cc (Branch within Aromatic System	100.0 ± 0.0	94.2 ± 0.4	c1cc (NC [C@@H] 2CCSC2) ncn1
C1CC1	Small Ring Strain (3-mem)	99.7 ± 0.1	85.4 ± 0.1	C1CC1N=C (O) c1cc (Br) ccc1O
c1nc2	Unclosed Fused Heterocycle	99.6 ± 0.1	74.3 ± 0.7	c1nc2c (c (=NC3CCSCC3) [nH]1) CCCC2
CC=	Dangling Double Bond	100.0 ± 0.0	96.2 ± 0.4	CC=CC (=O) c1ccc2c (c1) N=C (O) CCO2
CC#	Dangling Triple Bond	99.5 ± 0.3	68.7 ± 1.2	CC#N . COC (=O) c1ccc (F) cc1F
S (=O) (Hypervalent Sulfur State	100.0 ± 0.0	74.6 ± 0.1	S (=O) (=O) NC1CCc2ccccc2C1
N [C@@H] (Unclosed Stereocenter	100.0 ± 0.0	63.9 ± 0.2	N [C@@H] (CO) c1ccc2c (c1) N=C (O) CS2
c1ccc ([N+] (=O)	Charge Balance & Bracket Closure	100.0 ± 0.0	48.7 ± 1.8	c1ccc ([N+] (=O) [O-]) c (OC2CCNC2) c1

F. Additional Experimental Results

F.1. Robustness to Syntactic Discontinuities

Fixed-length segmentation inevitably partitions chemically meaningful substructures, potentially yielding unclosed branches, unmatched ring indices, or incomplete bracketed tokens. To evaluate the model’s capacity for restoring such syntactic discontinuities, a controlled prefix-completion experiment was conducted. Generation was conditioned on a curated set of chemically broken SMILES prefixes designed to mimic characteristic boundary truncations. These prefixes were selected to systematically evaluate the model’s ability to recover various syntactic discontinuities, covering unclosed branches and rings, incomplete complex structures such as fused heterocycles, dangling bonds representing valency violations, and fine-grained stereo- or charge-related specifications. For each prefix, 1,000 completions were sampled across three independent runs. Validity was defined as parsability by RDKit, and strict adherence to the prefix constraint was verified via exact string matching.

As summarized in Table 10, the model demonstrates near-perfect chemical validity across all distinct syntactic challenges. Despite the disruptive nature of fixed-length segmentation, the architecture reliably recovers local syntax and generates diverse, valid molecular continuations. It is observed that uniqueness naturally correlates with the specificity of the constraint; highly restrictive prefixes narrow the feasible chemical space, thereby reducing uniqueness while maintaining validity. Crucially, these results provide strong empirical evidence that the soft-fragment representation does not compromise molecular semantic integrity. Our SoftBD model effectively mitigates the arbitrary nature of fixed-length segmentation, ensuring that the intrinsic chemical logic remains intact.

F.2. Analysis of Property Distributions

To analyze the exploration characteristics of different methods, we compare the property distributions of the unique molecules derived from 3,000 generated samples per method in Figure 10. As expected, SoftMol (Unconstrained) achieves the **weakest** binding affinities by exploiting the unconstrained search space. However, SoftMol effectively concentrates the generated candidates within the drug-like manifold, resulting in distributions that align with the feasibility requirements. In contrast, baselines such as GenMol exhibit significantly lower QED and SA scores, indicating a trade-off where affinity optimization compromises molecular quality.

F.3. Ablation of diffusion budget T

We analyze the sensitivity of generation performance to the diffusion budget T (Table 11). Under the default efficient inference strategy, performance exhibits a sharp phase transition at $T = K_{\text{sample}}$: for $T < K_{\text{sample}}$, the budget is insufficient to resolve all block tokens, leading to validity collapse; for $T \geq K_{\text{sample}}$, metrics saturate immediately at near-optimal levels (Validity 100%, Quality $\approx 81.9\%$) with negligible variance. The First-Hitting mechanism effectively decouples computational cost from the budget T , maintaining low latency (≈ 10 s) even as T increases.

In contrast, disabling our inference strategy reveals the limitations of standard diffusion decoding. While increasing T gradually improves validity, it incurs a prohibitive linear computational cost. Crucially, even at extreme budgets ($T = 5000$), the baseline fails to match the drug-likeness of the default strategy ($T = 8$), plateauing at significantly lower Quality (57.0% vs. 81.9%) and Docking-Filter scores. This suggests that FH and GCD function not merely as accelerators, but as critical

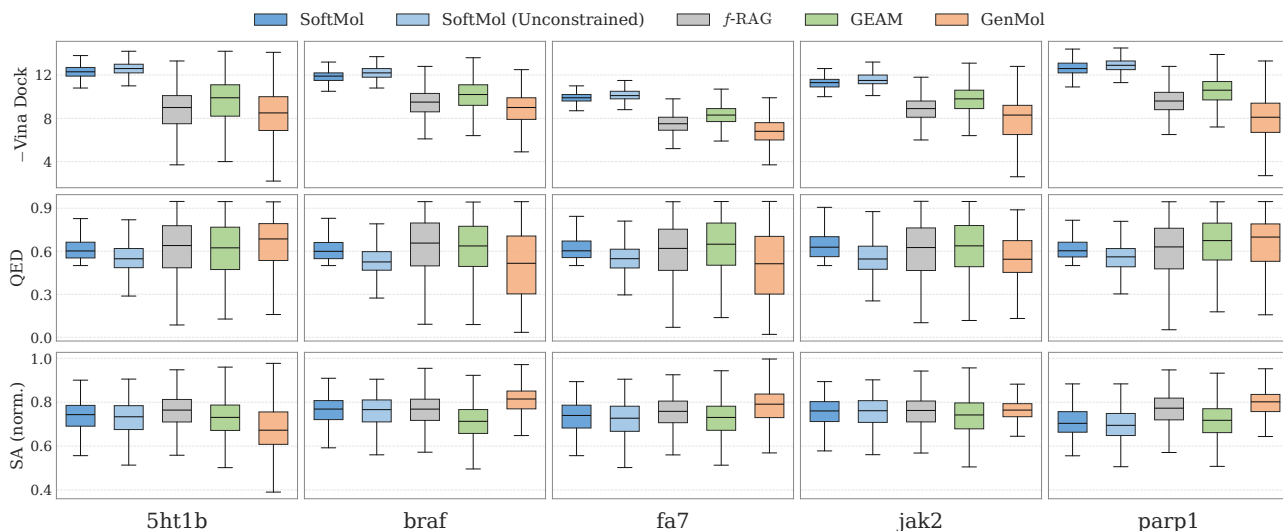


Figure 10. **Property Distributions.** Box plots of negative Vina Docking Score ($-Vina\ Dock$), QED, and normalized SA ($(10-SA)/9$) for unique molecules identified from 3,000 generated samples per method. SoftMol effectively constrains candidates to the high-quality drug-like region, whereas SoftMol (Unconstrained) pushes the affinity boundary.

Strategy	T	Validity (%)	Uniqueness (%)	Quality (%)	Docking-Filter (%)	Diversity	Time (s)
Default (FH+GCD+Batch)	7	0.0 \pm 0.0	0.0 \pm 0.0	0.0 \pm 0.0	0.0 \pm 0.0	0.000 \pm 0.000	8.922 \pm 0.572
	8	100.0 \pm 0.0	100.0 \pm 0.0	81.9 \pm 2.0	98.5 \pm 0.3	0.845 \pm 0.001	9.490 \pm 0.440
	100	100.0 \pm 0.0	100.0 \pm 0.0	81.9 \pm 2.0	98.5 \pm 0.3	0.845 \pm 0.001	9.864 \pm 0.728
	300	100.0 \pm 0.0	100.0 \pm 0.0	81.9 \pm 2.0	98.5 \pm 0.3	0.845 \pm 0.001	10.766 \pm 0.514
	500	100.0 \pm 0.0	100.0 \pm 0.0	81.9 \pm 2.0	98.5 \pm 0.3	0.845 \pm 0.001	10.928 \pm 0.632
	1000	100.0 \pm 0.0	100.0 \pm 0.0	81.9 \pm 2.0	98.5 \pm 0.3	0.845 \pm 0.001	10.779 \pm 0.877
	3000	100.0 \pm 0.0	100.0 \pm 0.0	81.9 \pm 2.0	98.5 \pm 0.3	0.845 \pm 0.001	11.157 \pm 1.456
5000	100.0 \pm 0.0	100.0 \pm 0.0	81.9 \pm 2.0	98.5 \pm 0.3	0.845 \pm 0.001	11.222 \pm 0.660	
w/o FH+GCD (Batch)	7	9.7 \pm 0.4	100.0 \pm 0.0	6.2 \pm 0.2	8.7 \pm 0.1	0.876 \pm 0.003	10.326 \pm 0.497
	8	14.9 \pm 0.9	100.0 \pm 0.0	9.5 \pm 0.2	12.7 \pm 0.6	0.874 \pm 0.004	10.852 \pm 0.895
	100	86.2 \pm 0.1	100.0 \pm 0.0	52.6 \pm 0.8	73.1 \pm 0.4	0.855 \pm 0.001	65.642 \pm 0.991
	300	89.1 \pm 0.5	100.0 \pm 0.0	55.8 \pm 0.2	76.8 \pm 0.4	0.855 \pm 0.001	189.697 \pm 0.411
	500	89.8 \pm 0.6	100.0 \pm 0.0	55.0 \pm 1.2	77.4 \pm 1.4	0.855 \pm 0.001	313.211 \pm 0.863
	1000	90.7 \pm 0.4	100.0 \pm 0.0	55.9 \pm 1.1	77.9 \pm 0.9	0.855 \pm 0.000	623.491 \pm 0.898
	3000	90.3 \pm 0.8	100.0 \pm 0.0	56.9 \pm 0.5	78.5 \pm 0.3	0.855 \pm 0.001	1726.290 \pm 0.347
5000	90.5 \pm 0.5	100.0 \pm 0.0	57.0 \pm 0.8	78.1 \pm 0.8	0.854 \pm 0.001	2476.522 \pm 4.262	

Table 11. Effect of the diffusion budget T under the default efficient inference strategy (Section 3.3) and its ablation. Results follow the ablation configuration (Section D.10) with 1,000 molecules.

correction mechanisms that actively enforce chemical feasibility.

F.4. Extended MCTS Ablations

MCTS Hyperparameters. Figure 11 analyzes the search parameters. Performance improves monotonically with search budget N_{max} . For node expansion width (candidates per node), a trade-off emerges: excessive width dilutes the search budget across shallow branches, while insufficient width limits exploration. Similarly, the exploration constant C balances exploring new chemotypes versus exploiting high-scoring branches; overly aggressive exploration ($C = 5$) degrades performance by favoring uncertain, incomplete paths.

SoftBD Model Configuration. Figure 12 examines the generative prior. Increasing the expansion batch size $M > 1$ enhances efficiency and diversity. Disabling nucleus sampling ($p = 1$) oddly improves performance, suggesting that truncating the distribution tail may hinder the discovery of strictly high-affinity novel structures. Softmax temperature τ shows negligible impact, indicating the robustness of the pretrained prior. Across all settings, SoftMol maintains a Hit Ratio near 100%, validating the effectiveness of the feasibility gate.

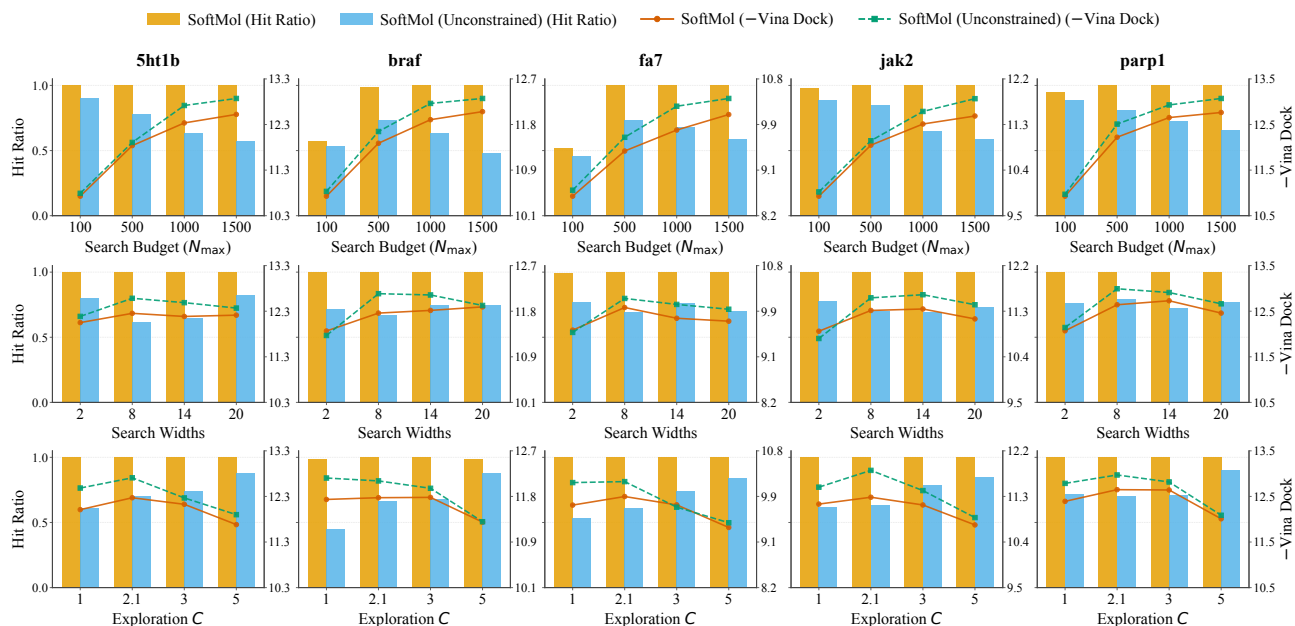


Figure 11. **Ablation on MCTS Hyperparameters.** Impact of Search Budget (N_{max}), Node Expansion Width, and Exploration Constant C . Results are averaged across 5 targets from 100 independent runs.

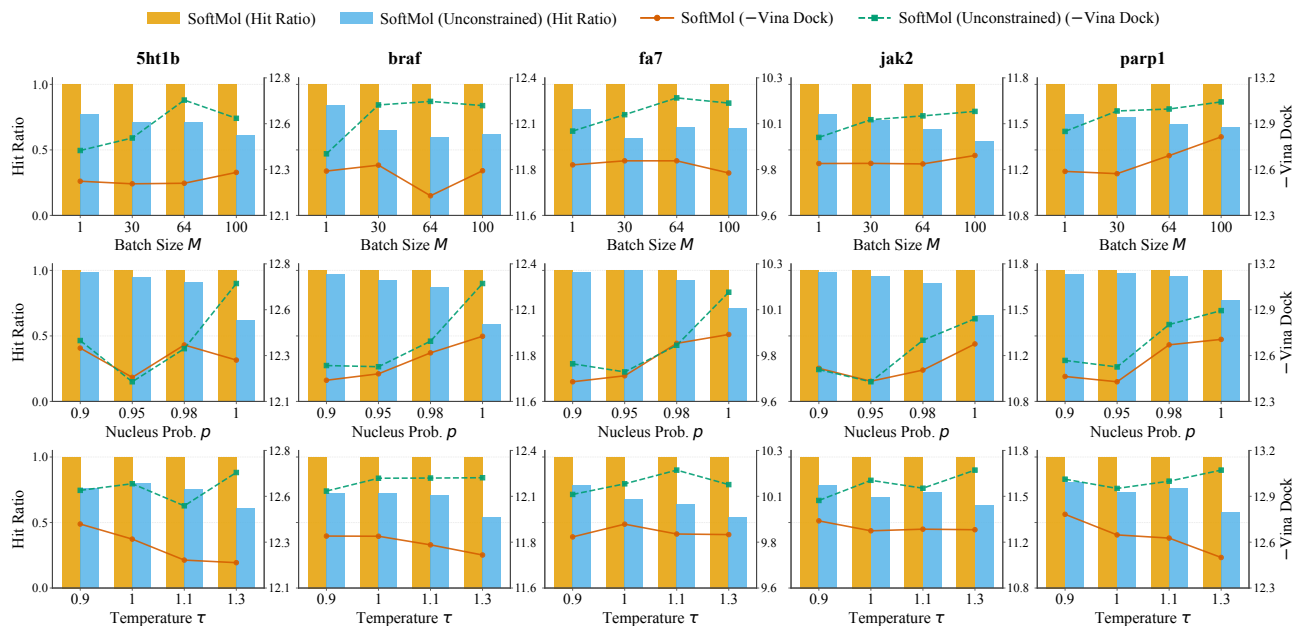


Figure 12. **Ablation on SoftBD Model Parameters.** Impact of Expansion Batch Size M , Nucleus Sampling Probability p , and Softmax Temperature τ . Results are averaged across 5 targets from 100 independent runs.

Diversity Analysis. Figure 13 quantifies the internal diversity of generated molecules across the full hyperparameter sweep. We first observe that SoftMol maintains high diversity comparable to the unconstrained baseline across all configurations, suggesting that pharmacophoric constraints do not significantly hinder chemical space exploration. Three key observations emerge. First, nucleus sampling probability p exerts the strongest influence: reducing p from 1.0 to 0.9 substantially increases diversity by broadening the generative sampling distribution, allowing exploration of lower-probability chemotypes. Second, temperature τ and batch size M exhibit secondary effects: higher τ introduces stochasticity that diversifies outcomes, while larger M amplifies the duplicate-filtering mechanism in batched expansion, directly increasing the fingerprint dissimilarity of instantiated children. Third, MCTS parameters show minimal impact: search budget N_{\max} , exploration constant C , and node widths yield near-constant diversity, confirming that molecular dissimilarity is primarily governed by the generative prior rather than search configuration. These results validate that SoftMol’s batched generation mechanism effectively translates sampling stochasticity into molecular diversity, enabling the discovery of chemically dissimilar high-affinity candidates.

Algorithm 1 ZINC-Curated curation pipeline

```

1: Input: Raw SMILES dataset  $\mathcal{D}_{\text{raw}}$  (ZINC-22)
2: Output: Curated dataset  $\mathcal{D}_{\text{curated}}$ 
3: Initialize buckets  $\mathcal{H} \leftarrow \{\mathcal{B}_4, \dots, \mathcal{B}_{49}\}$ 
4: Define forbidden substructures  $\mathcal{S}_{\text{ban}} \leftarrow \text{Toxic} \cup \text{PAINS} \cup \text{ForbiddenFragments}$ 
5: for each molecule  $m \in \mathcal{D}_{\text{raw}}$  do
6:   {1. PHYSICOCHEMICAL FILTERING}
7:   if  $\text{QED}(m) \leq 0.5 \vee \text{SA}(m) \geq 5$  then
8:     continue
9:   end if
10:  {2. STRUCTURAL VALIDITY}
11:  if  $\text{Elements}(m) \cap \{\text{Si}, \text{Sn}\} \neq \emptyset \vee \text{Charge}(m) \neq 0 \vee \text{Radicals}(m) > 0$  then
12:    continue
13:  end if
14:  if  $\text{Bridgehead}(m) > 2 \vee \text{MaxRing}(m) > 8 \vee \text{RotBonds}(m) > 10$  then
15:    continue
16:  end if
17:  if  $\text{TPSA}(m) > 140 \vee \exists s \in \mathcal{S}_{\text{ban}} : s \subseteq m$  then
18:    continue
19:  end if
20:  if  $\exists p \in \text{Atoms}(m, \text{P}) : \neg \text{HasSubstructure}(p, [\text{P}] (=O) (*))$  then
21:    continue
22:  end if
23:  {3. MEDICINAL CHEMISTRY RULES (LIPINSKI)}
24:  if  $\text{LogP}(m) > 5 \vee \text{MW}(m) \notin [100, 500]$  then
25:    continue
26:  end if
27:  if  $\text{HBD}(m) > 5 \vee \text{HBA}(m) > 10$  then
28:    continue
29:  end if
30:  {4. DIVERSITY-AWARE STRATIFICATION}
31:   $h \leftarrow \text{HeavyAtomCount}(m)$ 
32:  if  $h \in [4, 49]$  and  $\max_{m' \in \mathcal{B}_h} \text{Tanimoto}(m, m') < 0.5$  then
33:     $\mathcal{B}_h \leftarrow \mathcal{B}_h \cup \{m\}$ 
34:  end if
35: end for
36:  $\mathcal{D}_{\text{curated}} \leftarrow \bigcup_{h=4}^{49} \mathcal{B}_h$ 

```

Algorithm 2 Adaptive Confidence Decoding for SoftBD

```

1: Input: Prior  $p_\theta$ , Batch size  $N$ , Block size  $K_{\text{sample}}$ , Context Window  $W$ , Max blocks  $S$ 
2: Output: Generated molecules  $\mathbf{X}_{\text{accum}}$ 
3: Initialize:  $\mathbf{X}_{\text{accum}} \leftarrow [\text{BOS}] \times N$  (padded to max length)
4: Initialize:  $\mathbf{m}_{\text{done}} \leftarrow \mathbf{0}_N$  (False)
5: for block  $b = 0$  to  $S - 1$  do
6:   if  $\forall i, \mathbf{m}_{\text{done}}^{(i)} = \text{True}$  then
7:     break
8:   end if
9:   Define active block indices:  $\mathcal{I}_{\text{act}} \leftarrow [\max(1, bK_{\text{sample}}), (b + 1)K_{\text{sample}})$ 
10:  Define context window indices:  $\mathcal{I}_{\text{ctx}} \leftarrow [\max(0, bK_{\text{sample}} - W), (b + 1)K_{\text{sample}})$ 
11:  Initialize active block in  $\mathbf{X}_{\text{accum}}$  with  $[\text{MASK}]$ 
12:  Reset diffusion time per sequence:  $\mathbf{t} \leftarrow \mathbf{1}_N$ 
13:  loop
14:    Count masked tokens per sequence:  $\mathbf{m} \leftarrow \sum_{k \in \mathcal{I}_{\text{act}}} \mathbb{I}(\mathbf{X}_{\text{accum}}[:, k] == [\text{MASK}])$ 
15:    if  $\mathbf{m} == 0$  (all resolved) then
16:      break
17:    end if
18:    {1. Analytic First-Hitting Update}
19:    Sample  $\mathbf{u} \sim \mathcal{U}(0, 1)^N$ 
20:    Update time:  $\mathbf{t} \leftarrow \mathbf{t} \cdot \mathbf{u}^{1/\mathbf{m}}$ 
21:    {2. Sliding Window Prediction}
22:    Slice input:  $\mathbf{X}_{\text{in}} \leftarrow \mathbf{X}_{\text{accum}}[:, \mathcal{I}_{\text{ctx}}]$ 
23:    Compute logits:  $\mathbf{L} \leftarrow p_\theta(\mathbf{X}_{\text{in}}, \mathbf{t})$ 
24:    Extract probabilities for active block:  $\mathbf{P} \leftarrow \text{Softmax}(\mathbf{L}, -K_{\text{sample}}, :)$ 
25:    {3. Greedy Confidence Update}
26:    Compute best confidence per pos:  $\mathbf{C} \leftarrow \max_v \mathbf{P}$ 
27:    Mask already revealed positions in  $\mathbf{C}$ 
28:    Identify target position:  $j^* \leftarrow \text{argmax} \mathbf{C}$  (per sequence)
29:    Identify target token:  $v^* \leftarrow \text{argmax} \mathbf{P}_{:, j^*, :}$  (per sequence)
30:    Update tensor:  $\mathbf{X}_{\text{accum}}[:, bK + j^*] \leftarrow v^*$ 
31:    {4. Asynchronous Termination}
32:    if  $\exists i : v_i^* = [\text{EOS}]$  then
33:      Update  $\mathbf{m}_{\text{done}}$  for finished sequences
34:      Freeze remaining tokens in finished rows to  $[\text{EOS}]$ 
35:    end if
36:  end loop
37: end for
38: return  $\mathbf{X}_{\text{accum}}$ 

```

Algorithm 3 SoftMol target-specific optimization

```

1: Input: Protein  $P$ , SoftBD prior  $p_\theta$ , search budget  $N_{\max}$ , batch size  $M$ , rollouts  $n_{\text{sim}}$ , search widths
   ( $C_{\text{init}}, C_{\text{base}}, C_{\text{min}}, C_{\text{max}}$ ), feasibility gate ( $\tau_{\text{QED}}, \tau_{\text{SA}}$ ), exploration params ( $\lambda, C, \beta$ )
2: Output: Best molecule  $\mathbf{x}^*$ 
3: {Initialize root with  $C_{\text{init}}$  children capacity}
4:  $root \leftarrow \text{CreateNode}([\text{BOS}], \text{capacity} \leftarrow C_{\text{init}})$ 
5: for  $i = 1$  to  $N_{\max}$  do
6:   {Phase 1: Selection - traverse tree via UCT}
7:    $node \leftarrow root, \text{expanded} \leftarrow 0$ 
8:   while not  $\text{IsTerminal}(node)$  and  $\text{expanded} = 0$  do
9:      $\text{UpdateSearchWidth}(node, C_{\text{min}}, C_{\text{max}})$  {Adaptive widening via Eq. 9}
10:    if not  $\text{IsFullyExpanded}(node)$  then
11:      {Phase 2: Expansion - sample and add child}
12:       $candidates \leftarrow \text{BatchSampleNextBlock}(p_\theta, node, M)$  {Sample batch to ensure novelty}
13:       $candidates \leftarrow \text{FilterSiblingDuplicates}(candidates, node)$ 
14:       $child \leftarrow \text{CreateNode}(\text{RandomSelect}(candidates), \text{capacity} \leftarrow C_{\text{base}})$  {Select one to expand}
15:       $\text{AddChild}(node, child); node \leftarrow child$ 
16:       $\text{expanded} \leftarrow 1$ 
17:    else
18:       $node \leftarrow \text{BestUCTChild}(node)$  {Select via Eq. 8}
19:      if  $node$  is null then
20:        break
21:      end if
22:    end if
23:  end while
24:  {Phase 3: Simulation - rollout and evaluate}
25:  if  $\text{expanded} = 1$  then
26:     $\text{reward} \leftarrow -1.0$ 
27:    for  $t = 1$  to  $n_{\text{sim}}$  do
28:       $\mathbf{x}_t \leftarrow \text{RolloutToEOS}(p_\theta, node)$  {Complete to [EOS]}
29:       $\text{reward}_t \leftarrow -1.0$ 
30:      if  $\text{IsValidSmiles}(\mathbf{x}_t)$  and  $\text{PassFeasibilityGate}(\mathbf{x}_t, \tau_{\text{QED}}, \tau_{\text{SA}})$  then
31:         $\text{reward}_t \leftarrow -\text{VinaScore}(\mathbf{x}_t, P)$  {Negate for reward}
32:      end if
33:       $\text{reward} \leftarrow \max(\text{reward}, \text{reward}_t)$  {Take best rollout}
34:    end for
35:  else
36:     $\text{reward} \leftarrow \text{CachedReward}(node)$  {No expansion, use cached}
37:  end if
38:  {Phase 4: Backpropagation}
39:   $\text{Backpropagate}(node, \text{reward})$  {Update  $N, \bar{R}, R^{\max}$  to root}
40: end for
41: return  $\text{BestRollout}(root)$  {Return best molecule found}

```

From Tokens to Blocks: A Block-Diffusion Perspective on Molecular Generation

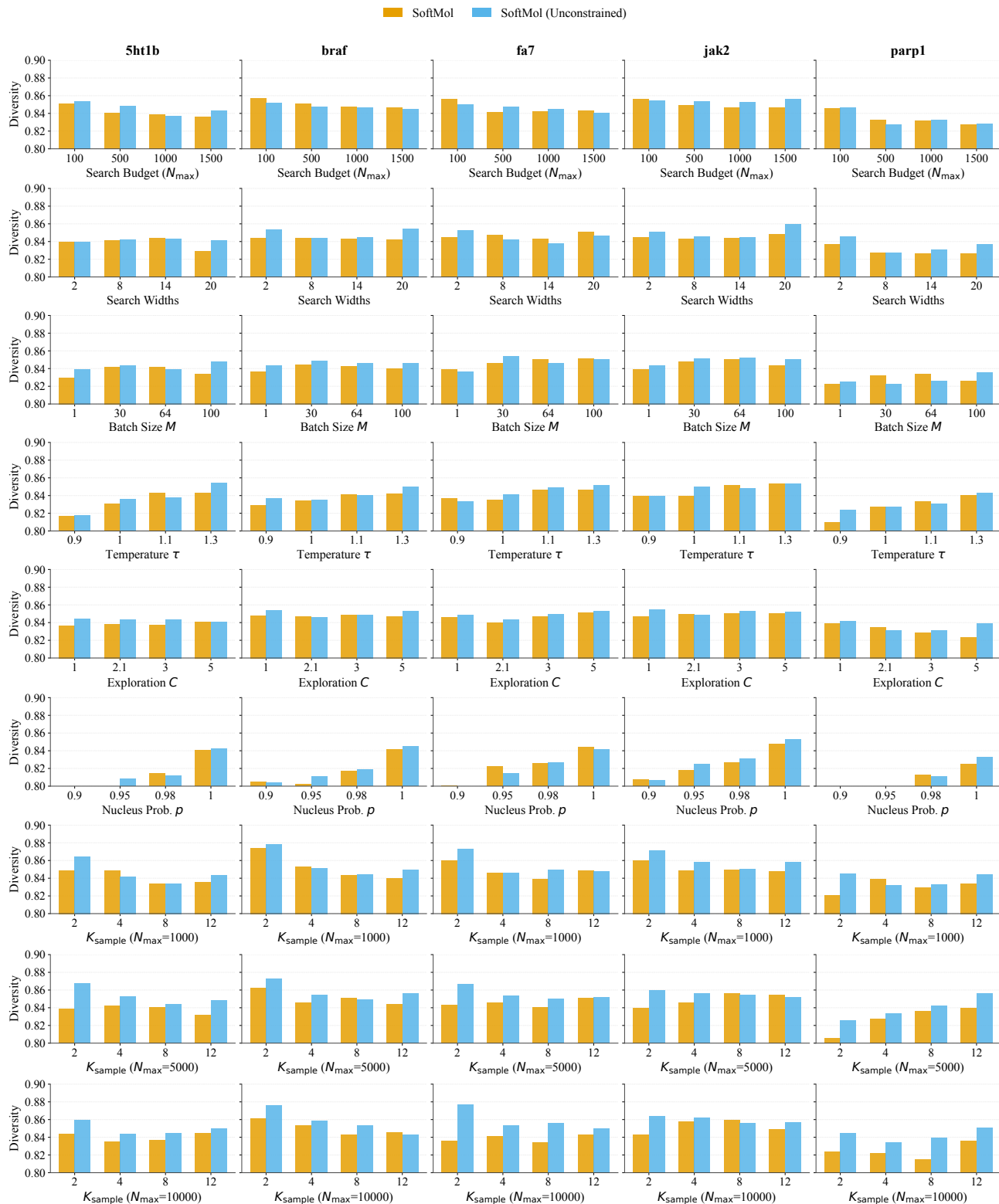


Figure 13. Impact of Hyperparameters on Generative Diversity. Internal diversity ($1 - \text{mean pairwise similarity}$) evaluated for SoftMol and the unconstrained baseline across five targets. Parameters are varied individually relative to the default configuration (Table 9). Results are averaged across 5 targets from 100 independent runs (50 runs for Search Budget $\in \{5000, 10000\}$ due to computational constraints).

G. Limitations

While SoftMol presents a robust framework for target-aware molecular generation, we acknowledge the following limitations:

(1) The 2D-3D Gap: Our generative policy operates on 1D representations to maximize search throughput. Although we incorporate 3D docking during evaluation, the generation process itself lacks intrinsic awareness of 3D steric constraints. This may occasionally yield candidates that are topologically valid but conformationally energetically unfavorable within the binding pocket.

(2) Reliance on Computational Proxies: As with most *in silico* benchmarks, we rely on Vina docking and heuristic scores (QED, SA) as optimization objectives. These are efficient but imperfect proxies for complex thermodynamic binding affinities and wet-lab synthetic feasibility.

(3) Computational Cost and Search Budget: While SoftBD achieves high inference speed, the full Gated MCTS procedure requires iterative rollout and oracle interaction. This represents a deliberate trade-off: investing higher computational cost per molecule to navigate the rugged energy landscapes of difficult protein targets.

(4) Scope Limitations: Our soft-fragment assumption is optimized for small drug-like molecules. Its effectiveness on macro-molecules or polymers, where structural dependencies span much larger logical distances, remains to be validated.