CupOFLATTE: Coupled Objective-Guided Discrete Flows via Linker Assembly for Targeted PROTAC Engineering

Anonymous Authors[1]

Abstract

Proteolysis-targeting chimeras (PROTACs) are heterobifunctional molecules whose function emerges from three coupled components: a warhead that binds a protein of interest, a linker, and an E3 ligand that recruits the cellular degradation machinery. Most generative approaches design the linker in isolation and assemble the ternary molecule only after generation, ignoring cross-component dependencies and often producing PROTACs that fall outside the empirical degrader distribution. We introduce **C**o**up**led **O**bjective-Guided Discrete **F**lows via **L**inker **A**ssembly for **T**argeted pro**T**ac **E**ngineering (**CupOFLATTE**), a discrete generative framework that restricts generative trajectories to a chemically feasible conjugate manifold and biases local transitions using soft objectives derived from PROTAC-DB chemistry, keeping warhead, linker, and E3 ligand components compatible throughout generation. Across assembly validity, predicted degradation activity (DegradeMaster), and PROTAC-like physicochemical property ranges, CupOFLATTE outperforms staged baselines including LinkerNet, DiffLinker, and DiffPROTACs, demonstrating that explicit coupling and constraint enforcement are sufficient to recover functional degraders in settings where function emerges only at the level of the assembled ternary system.

1. Introduction

Many disease-associated proteins lack a well-defined active site and are considered undruggable by traditional small-molecule inhibitors that rely on occupancy-based pharmacology (Dang et al., 2017). Targeted protein degradation sidesteps this limitation by hijacking the cell's ubiquitin-proteasome system to eliminate such proteins entirely (Békés et al., 2022; Chirnomas et al., 2023). The leading modality is the proteolysis-targeting chimera (PROTAC): a heterobifunctional small molecule with three covalently joined components: a *warhead* binding the protein of interest, an *E3 ligand* recruiting an E3 ubiquitin ligase such as cereblon (CRBN) or von Hippel-Lindau (VHL), and a chemical *linker* tethering the two so simultaneous binding induces a productive ternary complex in which the E3 ligase ubiquitinates the target for proteasomal degradation (Vikal et al., 2025; Wang et al., 2024a; Maniaci et al., 2017).

The linker controls whether the assembled PROTAC is functional (Békés et al., 2022): its length, composition, rigidity, and physicochemical character govern ternary complex geometry, cooperativity, permeability, and degradation efficacy (Troup et al., 2020; Cao et al., 2022), and small perturbations can convert potent degraders into inactive binders, making linker design a natural target for generative modeling. Existing methods, including SMILES decoders (Imrie et al., 2020; Yang et al., 2020), RL-augmented generators (Li et al., 2023; Murakami et al., 2025), and 3D diffusion models (Igashov et al., 2024; Li et al., 2024; Guan et al., 2023), achieve strong linker-level validity but treat the linker in isolation. Parallel progress in generative sequence modeling (Madani et al., 2020; Ferruz et al., 2022; Wang et al., 2024b; Alamdari et al., 2023; Lee et al., 2025) and discrete flow matching (Campbell et al., 2024; Stark et al., 2024; Davis et al., 2024; Tang et al., 2025b; Chen et al., 2025b;a) has enabled few-step sampling with principled guidance, but linker-only validity does not guarantee that the assembled conjugate falls within the physicochemical ranges characteristic of functional degraders (Pike et al., 2020), nor that it corresponds to a productive degrader of the intended target (Liu et al., 2025).

This gap motivates a different formulation. Rather than designing linkers in isolation and stitching them between fixed warheads and E3 ligands, we ask whether PROTACs can be generated directly as coupled three-component systems, with chemical feasibility and functional compatibility enforced throughout generation. We introduce **CupOFLATTE** (**Coup**led **O**bjective-Guided Discrete **F**lows via **L**inker **A**ssembly for **T**argeted pro**T**ac

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

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

Engineering), a discrete generative framework that jointly samples linker tokens under explicit assembly constraints with the chosen warhead and E3 ligand, biased by soft objectives reflecting canonical PROTAC chemistry. Operating on discrete flows rather than post-hoc attachment, CupOFLATTE enables anticipatory assembly in which intermediate states remain chemically meaningful and linker decisions account for downstream attachment and conjugate-level properties.

Our contributions are threefold:

1. **Coupled discrete flows for PROTAC assembly.** We formulate PROTAC linker generation as a coupled discrete flow that jointly evolves linker tokens with their attachment to the warhead and E3 ligand, ensuring assembly-level validity throughout generation.
2. **Score-guided sampling under hard validity constraints.** We introduce an objective-guided sampling scheme that tilts each local transition toward arbitrary user-specified soft objectives while enforcing hard chemical and cross-component validity constraints. The sampler is agnostic to the specific objective: any differentiable or black-box property predictor over molecular graphs can be used as a guidance signal.
3. **State-of-the-art PROTAC linker design.** Across three PROTAC systems spanning two E3 ligases (EGFR–VHL, BRD9–VHL, CDK6–CRBN), CupOFLATTE produces chemically valid, fully assembled conjugates with high predicted degradation activity (DegradeMaster) and PROTAC-like physicochemical profiles, substantially outperforming LinkerNet, DiffLinker, and DiffPROTACs.

2. Related Works

Constrained Generation for Biomolecule Design. Generative models for biomolecule design build on a growing toolkit of discrete generative methods. Discrete flow matching has emerged as a principled approach for sampling from complex discrete spaces, with jump-process models learning time-dependent transition rates that induce continuous-time Markov chains with prescribed marginals (Gat et al., 2024; Campbell et al., 2024), continuous-time simplex methods diffusing discrete data through embeddings over the probability simplex (Tang et al., 2025b; Davis et al., 2024; Stark et al., 2024), and rectified formulations such as ReDi (Yoo et al.) and AReUReDi (Chen et al., 2025a) simplifying probability paths and supporting multi-objective Pareto-guided sampling. Most biomolecular applications optimize linear objectives via reward-weighted sampling, classifier guidance, or guided diffusion (Nisonoff et al., 2024; Gruver et al., 2023), yet many design requirements are better expressed as binary feasibility constraints, including SMILES validity (Jin et al., 2018; Tang et al., 2025a; Chen et al., 2025a) and developability thresholds (Zeng et al., 2024). For diffusion-based generation, gradient-based constrained sampling (Kumar et al., 2022), projected diffusion (Christopher et al., 2024), and Constrained Discrete Diffusion (Cardei et al., 2025) bias iterates toward feasible regions, but most rely on soft guidance and do not guarantee constraint satisfaction. Analogous hard-constraint mechanisms remain underdeveloped for discrete flow matching, which has focused on controllable generation via reward reweighting.

Generative PROTAC Linker Design. Generative methods for PROTAC linker design have evolved along two axes. SMILES-based approaches treat linker generation as a sequence completion task: SyntaLinker (Yang et al., 2020) uses a conditional transformer for fragment linking, PROTAC-INVENT (Li et al., 2023) couples SMILES generation with reinforcement learning over 2D and 3D properties, and PROTAC-TS (Murakami et al., 2025) adds a permeability predictor as an RL reward. 3D structure-based approaches generate linker atoms in space conditioned on warhead and E3-ligand fragments: DeLinker (Imrie et al., 2020) introduced graph-based 3D-aware generation, DiffLinker (Igashov et al., 2024) uses an E(3)-equivariant diffusion model that links arbitrary numbers of fragments, DiffPROTACs (Li et al., 2024) adapts diffusion to the PROTAC distribution, and LinkerNet (Guan et al., 2023) jointly co-designs fragment poses and linker geometry. These methods optimize linker plausibility or local geometric fit, but none enforce assembly-level chemical validity or guide generation by desired properties, which is precisely the gap CupOFLATTE addresses.

3. Preliminaries

We introduce the mathematical objects required to define coupled discrete flows on structured molecular graph spaces and to formulate objective-guided transport for PROTAC linker generation. Throughout, $\mathcal{A}$ denotes a finite vocabulary of atom types augmented with an empty token (e.g., $\{\varnothing, \mathrm{C}, \mathrm{N}, \mathrm{O}, \mathrm{S}, \mathrm{F}, \mathrm{Cl}\}$), $\mathcal{B}$ denotes a finite vocabulary of bond types augmented with an empty token (e.g., $\{\varnothing, \text{single}, \text{double}, \text{triple}, \text{aromatic}\}$), and $\mathcal{G}_N$ denotes the set of molecular graphs on N atom slots. The empty tokens enable atom and bond *insertion* and *deletion* to be expressed uniformly as substitutions, supporting a fixed-budget formulation in which N is fixed but the effective molecular size varies. In this work, PROTACs are represented as discrete molecular graphs, consistent with prior molecular graph generation frameworks such as DiGress, DeFoG, and DiffLinker (Vignac et al., 2023; Qin et al., 2025; Igashov et al., 2024). While we present notation for a single molecular graph, subsequent sections consider structured states composed of fixed warhead and E3-ligand subgraphs coupled

with a generated linker subgraph.

3.1. Molecular Graphs as States in a Discrete Space

A PROTAC molecule is represented as a discrete graph state

$$x = (V, E) \in \mathcal{G}_N, \tag{1}$$

where $V \in \mathcal{A}^N$ is a tuple of node (atom) labels $V = (v_1, \ldots, v_N)$ with $v_i \in \mathcal{A}$, and $E \in \mathcal{B}^{N\times N}$ is a symmetric adjacency matrix of edge (bond) labels with $E_{ij} \in \mathcal{B}$. A slot i with $v_i = \varnothing$ is an unused atom slot; an entry $E_{ij} = \varnothing$ indicates the absence of a bond between slots i and j. The effective molecular graph is recovered by restricting to slots with $v_i \neq \varnothing$ and bonds with $E_{ij} \neq \varnothing$.

Fragment conditioning is encoded by partitioning the slot indices into fixed indices $\mathcal{I}_{\text{frag}}$ (warhead and E3 ligand) and free indices $\mathcal{I}_{\text{linker}}$, with the constraint that v_i for $i \in \mathcal{I}_{\text{frag}}$ and E_{ij} for $i, j \in \mathcal{I}_{\text{frag}}$ remain unchanged throughout transport.

Transitions between molecular graphs occur through local graph edit operations restricted to linker indices. We define the edit neighborhood

$$\mathcal{N}(x) := \left\{x' \in \mathcal{G}_N \,:\, d_{\text{link}}(x, x') = 1\right\}, \tag{2}$$

where a local edit corresponds to either an atom-type substitution at a free node ($v_i \to v_i'$ for $i \in \mathcal{I}_{\text{linker}}$) or a bond-type substitution on an edge incident to at least one free node ($E_{ij} \to E_{ij}'$ for $i \in \mathcal{I}_{\text{linker}}$ or $j \in \mathcal{I}_{\text{linker}}$). Substitutions involving the empty token $\varnothing$ realize atom and bond insertion (when $\varnothing$ is replaced by a concrete type) and deletion (when a concrete type is replaced by $\varnothing$).

3.2. Reference Dynamics

To define transport in discrete molecular graph space, we introduce a simple reference process that captures uninformed local exploration. We consider a continuous-time Markov chain (CTMC) with generator R_0 defined by

$$R_0(x, x') = \begin{cases} \dfrac{1}{|\mathcal{N}(x)|} & x' \in \mathcal{N}(x), \\ -\displaystyle\sum_{y \neq x} R_0(x, y) & x' = x, \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

This reference dynamics defines a random walk over molecular graphs that performs small, unbiased atom-type and bond-type edits on the linker region, and serves as a neutral baseline relative to which structured transport is learned.

3.3. Discrete Flow

Rather than modeling infinitesimal stochastic dynamics, we focus on learning *discrete flows* that directly transport molecular graph states between times. For $0 \leq s < t \leq T$, a discrete flow is a function

$$\Phi_\theta^{s\to t} : \mathcal{G}_N \to \mathcal{G}_N, \tag{4}$$

parameterized by θ, which maps a molecular graph at time s to a molecular graph at time t while preserving the fixed fragment indices $\mathcal{I}_{\text{frag}}$. These maps are intended to approximate the solution operator of an underlying probability flow, enabling one- or few-step generation without explicit simulation of a continuous-time process.

3.4. Objective Functions

To guide transport toward functionally relevant regions of molecular graph space, we assume access to one or more objective functions

$$S : \mathcal{G}_N \to \mathbb{R}, \tag{5}$$

which score molecular graphs according to downstream criteria such as synthetic accessibility (SA), terminus chemistry compatibility, or similarity to canonical PROTAC linker chemotypes (PEG, alkyl, piperazine, triazole). Importantly, these objectives are evaluated on the *current transported molecular state* and do not define a probability density. Instead, they provide directional guidance that biases transport toward desirable configurations.

3.5. Generative Sampling

After training, sampling proceeds by drawing an initial state $X_0 \sim \mu_0$ from a simple molecular-graph prior conditioned on the fixed warhead and E3-ligand subgraphs, and applying the learned flow:

$$X_t = \Phi_\theta^{0\to t}(X_0), \qquad 0 < t \leq T. \tag{6}$$

In the multistep regime, we choose a time grid $0 = t_0 < t_1 < \cdots < t_K = T$ and compose maps sequentially. Iterating this procedure from 0 to T yields samples from the learned objective-guided transport, producing valid molecular graphs corresponding to fully assembled PROTACs.

4. CupOFLATTE

We now formalize CupOFLATTE as a coupled generative transport problem on discrete molecular graphs representing PROTACs. The key objects are (i) structured discrete states that represent an assembled PROTAC with fixed warhead and E3-ligand subgraphs and a generated linker subgraph, (ii) a neutral notion of local atom and bond edits restricted to the linker region, and (iii) a family of *discrete flows* that transport a simple prior distribution to an empirical data distribution while incorporating objective guidance and hard validity constraints. Our goal is to learn a parametric family

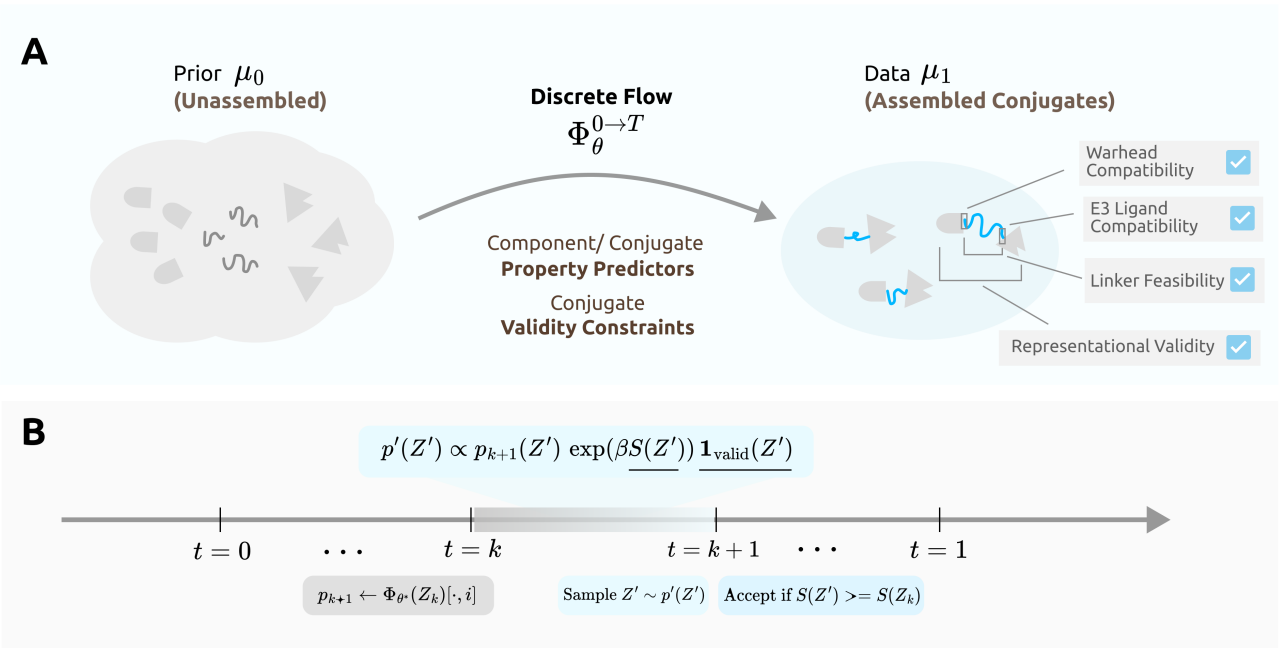

Figure 1. **Overview of CupOFLATTE for PROTAC linker generation. (A)** CupOFLATTE guides a pre-trained discrete flow model from a fixed warhead and E3 ligand toward valid, fully assembled PROTACs by generating the connecting linker, incorporating objective-guided optimization (synthetic accessibility, similarity to canonical PROTAC linker chemotypes, and PROTAC-DB physicochemical closeness) and hard validity constraints (terminus chemistry compatibility, single-component connectivity, fragment preservation, and representational validity). **(B)** Guided sampling with validity-constrained acceptance. At each timestep, candidate linker edits are sampled from a reweighted distribution incorporating objective scores and validity. Samples are accepted if they meet or exceed the current score, ensuring monotonic improvement toward high-scoring, valid PROTACs.

of flows Φ_θ^t whose induced transport produces chemically valid PROTACs and supports objective-guided linker assembly.

4.1. PROTAC State Space and Data Distribution

Let $\mathcal{A}$ and $\mathcal{B}$ denote the atom-type and bond-type vocabularies (each including an empty token $\varnothing$), and fix a maximum number of atom slots $N \in \mathbb{N}$. We write the molecular graph space as

$$\mathcal{X} := \mathcal{A}^N \times \mathcal{B}^{N\times N}, \tag{7}$$

where $x = (V, E) \in \mathcal{X}$ consists of node labels $V = (v_1, \ldots, v_N)$ with $v_i \in \mathcal{A}$ and a symmetric adjacency matrix E with $E_{ij} \in \mathcal{B}$.

In CupOFLATTE, a PROTAC is represented as a structured state composed of three coupled regions of a single molecular graph: the warhead, the linker, and the E3 ligand. The slot indices $\{1, \ldots, N\}$ are partitioned into three disjoint subsets $\mathcal{I}_{\text{wh}}, \mathcal{I}_{\text{link}}$, and $\mathcal{I}_{\text{e3}}$, inducing the decomposition

$$\begin{aligned} z &= \left(x^{\text{wh}}, x^{\text{link}}, x^{\text{e3}}\right) \in \mathcal{Z} \\ \text{with} \qquad \mathcal{Z} &:= \mathcal{X}_{\text{wh}} \times \mathcal{X}_{\text{link}} \times \mathcal{X}_{\text{e3}}, \end{aligned} \tag{8}$$

where x^{wh} and x^{e3} are the warhead and E3-ligand subgraphs (fixed throughout transport) and x^{link} is the linker subgraph (the region the model edits). Cross-component bonds at the warhead–linker and linker–E3-ligand junctions are included in the linker subgraph and are subject to learning. Although the formulation applies broadly to molecular conjugates, we focus on PROTACs where the state explicitly couples a warhead, a linker, and an E3 ligand.

We assume access to an unknown data distribution μ_1 over $\mathcal{Z}$ that describes the empirical law of valid assembled PROTACs. In practice, we observe an i.i.d. dataset

$$\mathcal{D}_{\text{data}} = \{z^{(i)}\}_{i=1}^{N_{\text{data}}}, \qquad z^{(i)} \sim \mu_1, \tag{9}$$

constructed from PROTAC-DB by extracting (warhead, linker, E3-ligand) decompositions of known PROTACs.

We also specify a simple prior distribution μ_0 on $\mathcal{Z}$ obtained by fixing the warhead and E3 ligand of a target PROTAC system and initializing the linker subgraph as the *empty graph*: every linker slot $i \in \mathcal{I}_{\text{link}}$ is set to $v_i = \varnothing$, and every linker bond entry is set to $E_{ij} = \varnothing$. The flow then introduces linker atoms and bonds during transport via empty-token substitutions. This prior serves as the starting distribution for the generative process.

The learning objective is to construct a transport mechanism

from μ_0 to μ_1:

$$Z_0 \sim \mu_0, \qquad Z_T \sim \mu_1, \tag{10}$$

where intermediate states preserve the fixed warhead and E3-ligand subgraphs, and the terminal state achieves cross-component connectivity through the assembled linker.

4.2. Fixed Termini and Attachment Convention

A central design choice in CupOFLATTE is that the *terminal atoms* of the warhead and E3 ligand are treated as a fixed input, not as something the model must discover. For a target PROTAC system we specify two distinguished slots: the warhead anchor $a^{\mathrm{wh}} \in \mathcal{I}_{\mathrm{wh}}$ and the E3-ligand anchor $a^{\mathrm{e3}} \in \mathcal{I}_{\mathrm{e3}}$, at which conjugating bond chemistry occurs in the assembled PROTAC. For example, in the EGFR/VHL system, the warhead is a gefitinib-derived EGFR ligand with a free -COOH at a^{wh} and the E3 ligand is VH032 with a free -NH_2 at a^{e3}; the BRD9/VHL and CDK6/CRBN systems specify analogous anchors per Appendix Table 9, and all three assemble via amide-bond closure at the anchors.

The fixed-termini convention is enforced at three levels:

- **State.** The atom types $v_{a^{\mathrm{wh}}}$ and $v_{a^{\mathrm{e3}}}$ are immutable and consistent with the fragment chemistry (e.g., the acyl carbon and amine nitrogen for amide-coupled systems).
- **Edit neighborhood.** Bonds incident to a^{wh} and a^{e3} are treated as linker edges: the model may add or remove the bonds connecting the linker to these anchors, but it cannot modify the anchor atoms themselves.
- **Validity.** A PROTAC state is considered valid only if it passes representational sanitization, contains no chemical liability substructures, and exposes at least two reactive termini compatible with the warhead/E3 amide-coupling chemistry; see Section 4.5.

Fixing the anchors while learning the bonds and intermediate atoms that connect them is a substantially easier learning problem than discovering valid termini de novo, which are jointly rare under unconditioned base-model sampling.

4.3. Reference Edit Structure and Neighborhoods

We lift the edit neighborhood of Section 3 to the structured PROTAC state space. For $z \in \mathcal{Z}$, define

$$\mathcal{N}(z) := \big\{ z' \in \mathcal{Z} : d_{\mathrm{link}}(z, z') = 1, \; x'^{\mathrm{wh}} = x^{\mathrm{wh}}, \; x'^{\mathrm{e3}} = x^{\mathrm{e3}} \big\}. \tag{11}$$

where d_{link} counts admissible single atom- or bond-type substitutions on linker indices, including bonds incident to the anchors a^{wh} and a^{e3} (which are themselves immutable; see Section 4.2). This neighborhood provides the reference notion of "small" changes in PROTAC space used for feasibility constraints and training signals.

4.4. Discrete Flows and Transport

Rather than learning stepwise stochastic dynamics, we learn a *discrete flow* that transports samples from the prior to the data distribution through a sequence of deterministic maps. Let

$$0 = t_0 < t_1 < \cdots < t_K = T$$

denote a discrete time grid. We parameterize a family of transport maps

$$\Phi_\theta^{t_k \to t_{k+1}} : \mathcal{Z} \to \mathcal{Z}, \qquad k = 0, \ldots, K-1, \tag{12}$$

where each map advances a state from time t_k to t_{k+1} while preserving the fixed warhead and E3-ligand subgraphs and the anchor atoms.

Starting from $z_0 \sim \mu_0$, the discrete flow evolves by composition,

$$z_{k+1} = \Phi_\theta^{t_k \to t_{k+1}}(z_k), \tag{13}$$

so that the state at time t_k is given by

$$z_k = \Phi_\theta^{t_{k-1} \to t_k} \circ \cdots \circ \Phi_\theta^{t_0 \to t_1}(z_0).$$

The distribution induced at time t_k is the pushforward of the prior,

$$\mu_{t_k}^\theta := \left(\Phi_\theta^{t_{k-1} \to t_k} \circ \cdots \circ \Phi_\theta^{t_0 \to t_1} \right)_{\#} \mu_0, \tag{14}$$

where $\#$ denotes the pushforward of measures. The terminal distribution $\mu_T^\theta = \mu_{t_K}^\theta$ defines the model distribution over assembled PROTACs.

This formulation mirrors continuous flows, where a complex target distribution is obtained by composing simple, learnable transport maps. From a flow matching perspective, each map $\Phi_\theta^{t_k \to t_{k+1}}$ approximates the finite-time transport induced by an underlying continuous probability flow on graphs, enabling few-step generation by discretizing and composing these learned transports.

4.5. Constraints and Objective Guidance

PROTAC assembly imposes hard validity constraints coupling the linker with the fixed warhead and E3 ligand, encoded via a binary classifier $\mathbf{1}_{\mathrm{valid}}(z) \in \{0, 1\}$ that returns 1 iff z satisfies all of:

(a) **Representational validity.** The molecule passes RDKit sanitization and the SMILES round-trip $\mathrm{MolFromSmiles}(\mathrm{MolToSmiles}(z))$ recovers a valid molecule; states with malformed valences or aromaticity are rejected.

(b) **Liability filter.** No SMARTS match against hydrazides (`[NX3][NX3]`), azo groups (`[#7]=[#7]`), N-oxides (`[N+][O-]`), N–O single bonds (`[#7]-[OX2]`), or carbamates (`[NX3]C(=[OX1])[OX2H1]`), which decompose, exhibit toxicity, or fail amide-coupled assembly.

(c) **Reactive termini count.** At least two carbon-bonded carboxylic acids (`[#6][CX3](=[OX1])[OX2H1]`) or aldehydes (`[#6][CX3H1](=[OX1])`); fewer renders warhead/E3 amide-coupling impossible.

Candidates with $\mathbf{1}_{\text{valid}}(z) = 0$ receive zero probability under the sampling distribution.

Beyond the hard gate, we use a continuous aggregate objective

$$S(z) \;=\; \sum_{j=1}^{4} w_j \, S_j(z) \;\in\; \mathbb{R}, \tag{15}$$

combining four soft scorers with default weights $(w_1, w_2, w_3, w_4) = (1.0,\ 1.0,\ 1.0,\ 4.0)$:

- S_1 (**PROTAC-likeness**): logistic regression over Morgan fingerprints of the linker (radius 2, 1024 bits), trained on PROTAC-DB linkers against negative chemotypes; output in $[0, 1]$. Gated to $3 \leq |\text{heavy atoms}| \leq 20$, with liability-flagged candidates mapped to zero.
- S_2 (**Synthetic accessibility**): a linear ramp on RDKit's SA score, 1.0 at $\text{SA} = 1$ and 0.0 at $\text{SA} = 6$, with $\text{SA} > 6$ mapped to zero.
- S_3 (**Chemotype closeness**): a trapezoidal windowed score from PROTAC-DB statistics averaging ether count, $\log P$, linker MW, and rotatable-bond count, each mapped into $[0, 1]$ via a window centered on the empirical PROTAC-DB range (assembled mass $[771, 1108]$ Da minus fixed warhead/E3 offsets). Liability-flagged candidates are mapped to zero.
- S_4 (**Connectivity**): returns 1.0 for a single connected component, otherwise $\frac{|\text{largest fragment}|}{|\text{total atoms}| \cdot |\text{fragments}|}$, halved if no path exists through x^{link} between a^{wh} and a^{e3}. Unlike the other terms, S_4 is evaluated even on partially assembled states with $\mathbf{1}_{\text{valid}}(z) = 0$; the elevated w_4 ensures the topological gradient toward connected, anchor-bridging linkers dominates early in transport.

The aggregate S provides directional guidance toward functionally relevant regions of $\mathcal{Z}$ but does not define a likelihood model for μ_1.

4.6. Endpoint Couplings and Interpolation

Learning discrete flows benefits from relating prior samples to data samples through an endpoint coupling. We assume a coupling γ between μ_0 and μ_1:

$$\gamma \in \Pi(\mu_0, \mu_1), \tag{16}$$

where $\Pi(\mu_0, \mu_1)$ denotes the set of joint distributions on $\mathcal{Z} \times \mathcal{Z}$ with marginals μ_0 and μ_1. In practice, γ is constructed by pairing each empty-linker prior sample with a PROTAC from the training set that shares the same warhead and E3 ligand (and therefore the same anchors $a^{\text{wh}}, a^{\text{e3}}$), breaking ties by independent pairing when multiple candidates exist.

For each sampled pair $(z_0, z_1) \sim \gamma$ and time $t \in [0, T]$, we construct an intermediate state z_t by applying a bounded number of admissible atom or bond edits on the linker subgraph that move from z_0 toward z_1 while preserving the anchor atoms. This interpolation provides training triples (z_0, z_t, z_1) that expose intermediate PROTAC configurations and support learning of maps for arbitrary time intervals.

4.7. Generative Sampling

Generation under the trained flow proceeds by drawing $Z_0 \sim \mu_0$ and applying the composition of Section 4.4 on the chosen time grid, yielding $Z_T \sim \mu_T^{\theta}$ after K steps. The number of steps K controls a tradeoff between computational cost and sample quality.

4.8. Warmup Steps

Early in transport, the linker subgraph is empty or near-empty and the guided sampler spends most of its budget on edits that grow the linker outward from the anchors — atom and bond insertions that branch into unoccupied slots. In this regime, the soft objectives carry little discriminative signal: most candidate edits produce small, partially assembled linkers that score similarly under S_1–S_3, and the connectivity term S_4 dominates simply by rewarding any growth toward a single connected component. Spending compute on exponential-tilting and acceptance checks during this phase yields limited benefit.

We therefore introduce a warmup phase of $K_{\text{warm}} < K$ steps during which the sampler draws directly from the pre-trained flow without guidance:

$$Z_{k+1} \;=\; \Phi_{\theta^*}^{t_k \to t_{k+1}}(Z_k), \qquad k = 0, \ldots, K_{\text{warm}} - 1, \tag{17}$$

bypassing the tilted resampling and acceptance criterion of Algorithm 1. This produces an initial branching structure for the linker under the prior induced by the pre-trained model. After warmup, guided sampling with exponential tilting and validity-constrained acceptance proceeds for the remaining $K - K_{\text{warm}}$ steps, at which point candidate edits

Algorithm 1 Exponential Tilting with Warmup and Acceptance Criterion

Input: trained discrete flow Φ_{θ^*}, prior μ_0 on $\mathcal{Z}$
time horizon T, total steps K, warmup steps K_{warm}
guidance score $S(z)$, strength β
Define time grid $0 = t_0 < t_1 < \cdots < t_K = T$
Sample $Z_0 \sim \mu_0$ s.t. $\mathbf{1}_{\text{valid}}(Z_0) = 1$; $Z^* \leftarrow Z_0$
(Warmup: unguided sampling from the pre-trained flow)
for $k = 0$ **to** $K_{\text{warm}} - 1$ **do**
  Select a free linker site i
  $p_{k+1} \leftarrow \Phi_{\theta^*}(Z_k)[\cdot, i]$
  Sample $z^\dagger \sim p_{k+1}$ (no tilting, no acceptance)
  $Z_{k+1} \leftarrow Z_k$ with site i set to $z^\dagger$
  if $\mathbf{1}_{\text{valid}}(Z_{k+1}) = 1$ **and** $S(Z_{k+1}) > S(Z^*)$ **then**
    $Z^* \leftarrow Z_{k+1}$ (track best-so-far during warmup)
  end if
end for
(Guided phase: exponential tilting with hard validity mask)
for $k = K_{\text{warm}}$ **to** $K - 1$ **do**
  Select a free linker site i (atom slot in $\mathcal{I}_{\text{link}}$ or bond entry incident to it)
  $p_{k+1} \leftarrow \Phi_{\theta^*}(Z_k)[\cdot, i]$ (predicted distribution at site i)
  for each $z' \in \text{TopK}(p_{k+1})$ **do**
    $p'(z') \propto p_{k+1}(z') \exp(\beta\, S(z'))\, \mathbf{1}_{\text{valid}}(z')$
  end for
  Sample $z^\dagger \sim p'$
  $Z' \leftarrow Z_k$ with site i set to $z^\dagger$ (atom-type or bond-type edit)
  if $S(Z') \geq S(Z_k)$ **then**
    $Z_{k+1} \leftarrow Z'$ (acceptance criterion)
    if $S(Z') > S(Z^*)$ **then**
      $Z^* \leftarrow Z'$
    end if
  else
    $Z_{k+1} \leftarrow Z_k$ (reject, keep current)
  end if
end for
return Z^* (best PROTAC graph found)

begin to meaningfully differentiate under the soft objectives. Empirically, K_{warm} is set as a fixed fraction (≤ 0.5) of K and reduces wall-clock sampling time without measurable degradation in the terminal score, since the warmup phase explores configurations that the guided sampler would have accepted anyway.

5. Results

For evaluation of CupOFLATTE, we focus on three key questions: (1) whether discrete-flow guidance with hard validity constraints produces assembled, valid PROTACs at higher rates than existing linker-generation baselines; (2) whether the generated linkers maintain competitive PROTAC-like physicochemical profiles and predicted degradation activity; and (3) whether CupOFLATTE produces diverse, non-memorized linkers across structurally distinct PROTAC systems.

We evaluate on three benchmark systems spanning two E3 ligases and three target classes: EGFR–VHL (kinase target), BRD9–VHL (bromodomain target), and CDK6–CRBN (kinase target with a non-VHL E3). For each system we generate 50 linkers per method and report molecular metrics over the resulting valid assemblies. We compare against three published PROTAC linker generation baselines: LinkerNet (Guan et al., 2023), DiffPROTACs (Li et al., 2024), and DiffLinker (Igashov et al., 2024).

5.1. Assembly Validity Across PROTAC Systems

PROTAC linker generation imposes a stricter validity bar than generic molecular generation: a generated linker must not only be a parseable molecule, but must also bridge the warhead and E3-ligand termini under chemically compatible bond chemistry. We separate two notions of validity. *Standard validity* is the fraction of generated SMILES that RDKit can parse. *Assembly validity* is the fraction of generations that produce a fully assembled PROTAC passing canonical attachment and analyzer checks (representational validity, liability filtering, and reactive-termini compatibility; see Section 4.5).

Table 1. Assembly validity across PROTAC systems. Std. / Asm. Valid (%): RDKit-parseable SMILES / canonical attachment + analyzer pass. $N = 50$ per cell.

	LinkerNet	DiffPROTACs	DiffLinker	**Ours**
EGFR–VHL				
Std. Valid	98.0	98.0	86.0	**100.0**
Asm. Valid	46.0	80.0	0.0	**88.0**
BRD9–VHL				
Std. Valid	**100.0**	**100.0**	74.0	94.0
Asm. Valid	16.0	0.0	0.0	**72.0**
CDK6–CRBN				
Std. Valid	**100.0**	86.0	92.0	94.0
Asm. Valid	**96.0**	74.0	0.0	74.0

Table 1 reveals a sharp split between standard and assembly validity for the baselines. DiffLinker, despite producing parseable SMILES at 74–92%, fails assembly entirely (0.0% across all three systems): the generated linkers do not expose chemistry compatible with the warhead and E3 termini for canonical amide-coupling. DiffPROTACs assembles well on EGFR–VHL (80.0%) and CDK6–CRBN (74.0%) but collapses to 0.0% on BRD9–VHL. LinkerNet, a retrieval-augmented baseline, achieves 96.0% assembly validity on CDK6–CRBN but only 46.0% on EGFR–VHL and 16.0% on BRD9–VHL.

CupOFLATTE achieves the highest assembly validity on EGFR–VHL (88.0%) and BRD9–VHL (72.0%), and matches DiffPROTACs on CDK6–CRBN (74.0%). The validity gain on BRD9–VHL is particularly notable: where DiffPROTACs and DiffLinker both fail entirely and LinkerNet achieves only 16.0%, CupOFLATTE's hard-validity gating during sampling produces 72.0% assembled PROTACs. This reflects the fact that CupOFLATTE enforces termini compatibility and liability filtering as a zero-probability mask during sampling rather than relying on the base model to implicitly learn these constraints.

Because DiffLinker produces 0.0% assembly validity across all three systems, it cannot be evaluated on any downstream property, activity, or memorization metric defined over assembled PROTACs. We therefore exclude DiffLinker from all subsequent comparisons in Sections 5.2–5.4 and report only the assembly validity numbers from Table 1.

5.2. Molecular Properties of Generated PROTACs

We next examine the physicochemical properties of valid generations across methods. We report standard drug-likeness metrics (synthetic accessibility (SA), QED, the fraction within PROTAC-DB physicochemical windows for $\log P$, MW, and rotatable bonds (RotB)) along with uniqueness within each generated batch. Per the exclusion noted in Section 5.1, DiffLinker is omitted from all property tables. Methods that fail assembly on a particular system (DiffPROTACs on BRD9–VHL) are shown as dashes for that system's entries.

Table 2. Molecular properties of valid PROTACs targeting EGFR–VHL. SA ↓, all others ↑. SA and QED reported as mean ± std. The $\log P$, MW, and RotB rows report the fraction of valid PROTACs falling within PROTAC-DB-derived windows ($\log P \in [2.5, 7.4]$, MW $\in [771, 1108]$ Da, RotB $\in [12, 25]$: mean ± std).

Property	LinkerNet	DiffPROTACs	Ours
SA score	**1.89 ± 0.46**	2.95 ± 0.91	3.14 ± 0.68
QED	0.487 ± 0.072	0.424 ± 0.133	**0.490 ± 0.170**
$\log P$ in range	**87.0%**	75.0%	70.5%
MW in range	**100.0%**	80.0%	63.6%
RotB in range	**100.0%**	67.5%	81.8%
Uniqueness	65.2%	97.5%	**97.7%**

Table 3. Molecular properties of valid PROTACs targeting BRD9–VHL. DiffPROTACs entries are dashes due to 0% assembly validity on this system. Same conventions as Table 2.

Property	LinkerNet	DiffPROTACs	Ours
SA score	**1.94 ± 0.52**	—	3.01 ± 0.61
QED	0.457 ± 0.072	—	**0.471 ± 0.134**
$\log P$ in range	87.5%	—	**97.2%**
MW in range	**100.0%**	—	91.7%
RotB in range	**100.0%**	—	97.2%
Uniqueness	**100.0%**	—	97.2%

Table 4. Molecular properties of valid PROTACs targeting CDK6–CRBN. SA ↓, all others ↑. SA and QED reported as mean ± std. The $\log P$, MW, and RotB rows report the fraction of valid PROTACs falling within PROTAC-DB-derived windows ($\log P \in [2.5, 7.4]$, MW $\in [771, 1108]$ Da, RotB $\in [12, 25]$). DiffLinker omitted (0% assembly-valid).

Property	LinkerNet	DiffPROTACs	Ours
SA score	**1.97 ± 0.44**	2.58 ± 0.79	3.03 ± 0.57
QED	0.484 ± 0.078	0.513 ± 0.106	**0.544 ± 0.140**
$\log P$ in range	87.5%	**91.9%**	86.5%
MW in range	**100.0%**	**100.0%**	**100.0%**
RotB in range	89.6%	**97.3%**	78.4%
Uniqueness	77.1%	**100.0%**	97.3%

Across all three systems, CupOFLATTE achieves the highest QED among methods that produce valid assemblies, indicating favorable drug-likeness in the generated linkers. SA scores are higher (less favorable) than LinkerNet's, which is expected: LinkerNet retrieves linkers from a curated training set of known PROTAC chemotypes, which biases it toward simpler, easier-to-synthesize fragments already known to be synthetically tractable. CupOFLATTE, by contrast, generates linkers *de novo* under the soft SA scorer S_2 (Section 4.5), trading some synthetic simplicity for broader chemical exploration. Physicochemical window hit rates are competitive with DiffPROTACs across systems and match or exceed it on BRD9–VHL.

5.3. Predicted Degradation Activity

Beyond physicochemical drug-likeness, the central question for PROTAC design is whether the generated molecule is predicted to function as a degrader. We use DegradeMaster (Liu et al., 2025), an independent neural classifier trained to predict whether a candidate PROTAC induces target degradation, and report mean predicted probability and the fraction exceeding thresholds of 0.3, 0.5, and 0.7. Note that this serves as external validation, as CupOfLATTE isn't guided by this metric.

CupOFLATTE achieves substantially higher predicted degradation probability on EGFR–VHL and BRD9–VHL, the two systems where LinkerNet struggles to assemble valid PROTACs. On EGFR–VHL, CupOFLATTE's mean DegradeMaster probability (0.529) is 1.9× that of LinkerNet (0.278) and 1.7× that of DiffPROTACs (0.306); the fraction exceeding 0.5 is 57.1% versus 0.0% for LinkerNet and 20.0% for DiffPROTACs. On BRD9–VHL, CupOFLATTE is the only method producing any generations above the 0.5 threshold (30.6%).

CDK6–CRBN inverts this picture: LinkerNet achieves the highest DegradeMaster scores (0.925 mean, 100.0% above 0.7), followed by DiffPROTACs and then CupOFLATTE. This reflects the narrower chemotype space of CRBN PRO-

Table 5. DegradeMaster predicted degradation probability on valid PROTACs. "mean" is mean ± std over valid generations; threshold rows report the fraction with DegradeMaster $\geq \tau$. DiffLinker is omitted (see Section 5.1); dashes indicate methods that produce 0% valid assemblies on the corresponding system.

	LinkerNet	DiffPROTACs	**Ours**
EGFR–VHL			
mean	0.278 ± 0.051	0.306 ± 0.236	$\mathbf{0.529 \pm 0.277}$
≥ 0.3	34.8%	48.6%	**81.0%**
≥ 0.5	0.0%	20.0%	**57.1%**
≥ 0.7	0.0%	8.6%	**23.8%**
BRD9–VHL			
mean	0.172 ± 0.053	—	$\mathbf{0.289 \pm 0.259}$
≥ 0.3	0.0%	—	**36.1%**
≥ 0.5	0.0%	—	**30.6%**
≥ 0.7	0.0%	—	**8.3%**
CDK6–CRBN			
mean	$\mathbf{0.925 \pm 0.055}$	0.882 ± 0.185	0.758 ± 0.290
≥ 0.3	**100.0%**	96.7%	88.6%
≥ 0.5	**100.0%**	96.7%	82.9%
≥ 0.7	**100.0%**	93.3%	74.3%

Table 6. Diversity and memorization. Diversity is mean pairwise Tanimoto within the generated batch (lower = more diverse). Memorization is mean max-Tanimoto to PROTAC-8K (lower = less memorized). "Mem > 0.80" is the fraction exceeding the memorization threshold. DiffLinker is omitted (see Section 5.1); dashes indicate methods that produce 0% valid assemblies on the corresponding system.

	LinkerNet	DiffPROTACs	**Ours**
EGFR–VHL			
Diversity	0.895	0.763	**0.752**
Memorization	0.816	0.779	**0.623**
Mem > 0.80	78.3%	35.0%	**0.0%**
BRD9–VHL			
Diversity	0.829	—	**0.732**
Memorization	0.825	—	**0.768**
Mem > 0.80	75.0%	—	**16.7%**
CDK6–CRBN			
Diversity	0.833	0.735	**0.700**
Memorization	0.776	0.694	**0.541**
Mem > 0.80	45.8%	2.7%	**0.0%**

TACs: LinkerNet's training set contains many CRBN-targeted PROTACs whose linker patterns are recognizable to DegradeMaster, and retrieval-based generation produces near-copies of these patterns. The novelty and memorization analysis in Section 5.4 clarifies this tradeoff.

5.4. Diversity, Novelty, and Memorization

A method that achieves high DegradeMaster by reproducing training linkers offers little design value. We assess the originality of generated linkers using two metrics: *Intra-set diversity* (mean pairwise Tanimoto similarity within the generated batch; lower indicates more diverse generations), and *Memorization* (mean maximum Tanimoto similarity from each generation to its nearest neighbor in PROTAC-8K, paired with the percentage of generations exceeding the standard memorization threshold of 0.80).

Table 6 shows that CupOFLATTE produces the most diverse and least memorized generations across all three systems. On EGFR–VHL, 78.3% of LinkerNet generations exceed the 0.80 memorization threshold to PROTAC-8K, compared to 35.0% for DiffPROTACs and 0.0% for CupOFLATTE. The pattern holds on BRD9–VHL (75.0% vs. 16.7%) and CDK6–CRBN (45.8% vs. 0.0%).

This contextualizes the CDK6–CRBN DegradeMaster results in Table 5: LinkerNet's 0.925 mean DegradeMaster probability is achieved alongside 45.8% of generations being near-copies (> 0.80 Tanimoto) of training PROTACs. CupOFLATTE's 0.758 DegradeMaster on the same system is achieved with 0.0% memorization and 100.0% novelty: every generation is a linker not present in the training set. Across the three systems, CupOFLATTE achieves 100.0% novelty (no exact training matches) while maintaining higher DegradeMaster scores than baselines on the two more challenging targets (EGFR–VHL and BRD9–VHL) and competitive scores on CDK6–CRBN. We interpret this as evidence that CupOFLATTE produces *de novo* linkers with predicted degradation activity, rather than retrieving variations on known PROTACs.

6. Discussion

We present **CupOFLATTE**, an objective-guided discrete-flow framework that generates PROTAC linkers under soft chemical objectives and hard validity constraints. Across three benchmark systems, it outperforms existing baselines on assembly rate, achieves 100% novelty with low memorization, and delivers higher predicted degradation activity on EGFR-VHL and BRD9-VHL. Still, all evaluations are *in silico*; DegradeMaster scores on sparse target–E3 pairs are noisy, so we emphasize relative orderings; and QED reflects distributional drift more than drug-likeness gaps. Our next steps include ternary-complex and ADMET scorers, joint warhead/linker/E3 generation, and experimental validation.

Impact Statement

CupOFLATTE contributes generative methodology for PROTAC linker design, with potential to accelerate degrader discovery for previously undruggable targets. Predicted activity is not a substitute for experimental validation, and wet-lab characterization and toxicology remain essential before any generated molecule is advanced. We see no novel dual-use risks beyond those already present in small-molecule generative modeling.

References

Alamdari, S., Thakkar, N., van den Berg, R., Lu, A., Fusi, N., Amini, A., and Yang, K. Protein generation with evolutionary diffusion: Sequence is all you need. *bioRxiv*, 2023. doi: 10.1101/2023.09.11.556673. URL https://www.biorxiv.org/content/10.1101/2023.09.11.556673v2.full-text. Preprint.

Békés, M., Langley, D. R., and Crews, C. M. PROTAC targeted protein degraders: the past is prologue. *Nature Reviews Drug Discovery*, 21(3):181–200, 2022. doi: 10.1038/s41573-021-00371-6.

Campbell, A., Yim, J., Barzilay, R., Rainforth, T., and Jaakkola, T. Generative flows on discrete state-spaces: Enabling multimodal flows with applications to protein co-design. In *International Conference on Machine Learning*, pp. 5453–5512. PMLR, 2024.

Cao, C., He, M., Wang, L., He, Y., and Rao, Y. Chemistries of bifunctional PROTAC degraders. *Chemical Society Reviews*, 51(16):7066–7114, 2022. doi: 10.1039/D2CS00220E.

Cardei, M., Christopher, J. K., Kailkhura, B., Hartvigsen, T., and Fioretto, F. Constrained molecular generation with discrete diffusion for drug discovery. In *NeurIPS 2025 Workshop on AI Virtual Cells and Instruments: A New Era in Drug Discovery and Development*, 2025.

Chen, T., Zhang, Y., and Chatterjee, P. Areuredi: Annealed rectified updates for refining discrete flows with multi-objective guidance, 2025a. URL https://arxiv.org/abs/2510.00352.

Chen, T., Zhang, Y., Tang, S., and Chatterjee, P. Multi-objective-guided discrete flow matching for controllable biological sequence design, 2025b. URL https://arxiv.org/abs/2505.07086.

Chirnomas, D., Hornberger, K. R., and Crews, C. M. Protein degraders enter the clinic — a new approach to cancer therapy. *Nature Reviews Clinical Oncology*, 20(4):265–278, 2023. doi: 10.1038/s41571-023-00736-3.

Christopher, J. K., Baek, S., and Fioretto, N. Constrained synthesis with projected diffusion models. *Advances in Neural Information Processing Systems*, 37:89307–89333, 2024.

Dang, C. V., Reddy, E. P., Shokat, K. M., and Soucek, L. Drugging the 'undruggable' cancer targets. *Nature Reviews Cancer*, 17(8):502–508, 2017. doi: 10.1038/nrc.2017.36.

Davis, O., Kessler, S., Petrache, M., Ceylan, İ. İ., Bronstein, M., and Bose, A. J. Fisher flow matching for generative modeling over discrete data. *Advances in Neural Information Processing Systems*, 37:139054–139084, 2024.

Ferruz, N., Schmidt, S., and Höcker, B. Protgpt2 is a deep unsupervised language model for protein design. *Nature Communications*, 13, 2022. doi: 10.1038/s41467-022-32007-7. URL https://www.nature.com/articles/s41467-022-32007-7.

Gat, I., Remez, T., Shaul, N., Kreuk, F., Chen, R. T., Synnaeve, G., Adi, Y., and Lipman, Y. Discrete flow matching. *Advances in Neural Information Processing Systems*, 37:133345–133385, 2024.

Gruver, N., Stanton, S., Frey, N. C., Rudner, T. G. J., Hotzel, I., Lafrance-Vanasse, J., Rajpal, A., Cho, K., and Wilson, A. G. Protein design with guided discrete diffusion, 2023. URL https://arxiv.org/abs/2305.20009.

Guan, J., Peng, X., Jiang, P., Luo, Y., Peng, J., and Ma, J. LinkerNet: Fragment poses and linker co-design with 3D equivariant diffusion. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Igashov, I., Stärk, H., Vignac, C., Schneuing, A., Satorras, V. G., Frossard, P., Welling, M., Bronstein, M., and Correia, B. Equivariant 3D-conditional diffusion model for molecular linker design. *Nature Machine Intelligence*, 6(4):417–427, 2024. doi: 10.1038/s42256-024-00815-9.

Imrie, F., Bradley, A. R., van der Schaar, M., and Deane, C. M. Deep generative models for 3D linker design. *Journal of Chemical Information and Modeling*, 60(4):1983–1995, 2020. doi: 10.1021/acs.jcim.9b01120.

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.

Kumar, S., Paria, B., and Tsvetkov, Y. Gradient-based constrained sampling from language models. *arXiv preprint arXiv:2205.12558*, 2022.

Landrum, G., Tosco, P., Kelley, B., Rodriguez, R., Cosgrove, D., Vianello, R., Gedeck, P., Jones, G., Kawashima, E., Nealschneider, D., et al. rdkit/rdkit: 2025_03_1 (q1 2025) release. *Zenodo*, 2025.

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.

Li, B., Ran, T., and Chen, H. 3D-based generative PROTAC linker design with reinforcement learning. *Briefings in Bioinformatics*, 24(5):bbad323, 2023. doi: 10.1093/bib/bbad323.

Li, F., Hu, Q., Zhou, Y., Yang, H., and Bai, F. DiffPROTACs is a deep learning-based generator for proteolysis targeting chimeras. *Briefings in Bioinformatics*, 25(5): bbae358, 2024. doi: 10.1093/bib/bbae358.

Liu, J., Roy, M. J., Isbel, L., and Li, F. Accurate PROTAC-targeted degradation prediction with DegradeMaster. *Bioinformatics*, 41(Supplement_1):i342–i350, 2025. doi: 10.1093/bioinformatics/btaf191.

Madani, A., McCann, B., Naik, N., Keskar, N. S., Anand, N., Eguchi, R. R., Huang, P.-S., and Socher, R. Progen: Language modeling for protein generation. *bioRxiv*, 2020. doi: 10.1101/2020.03.07.982272. URL https://www.biorxiv.org/content/10.1101/2020.03.07.982272v2.full-text. Preprint.

Maniaci, C., Hughes, S. J., Testa, A., Chen, W., Lamont, D. J., Rocha, S., Alessi, D. R., Romeo, R., and Ciulli, A. Homo-PROTACs: bivalent small-molecule dimerizers of the VHL E3 ubiquitin ligase to induce self-degradation. *Nature Communications*, 8(1):830, 2017. doi: 10.1038/s41467-017-00954-1.

Murakami, Y., Ishida, S., Cho, N., Yuki, H., Ohta, M., Honma, T., Demizu, Y., and Terayama, K. Data-driven design of PROTAC linkers to improve PROTAC cell membrane permeability. *ChemRxiv preprint*, 2025. doi: 10.26434/chemrxiv-2025-24kkf. Preprint, August 2025.

Nisonoff, H., Xiong, J., Allenspach, S., and Listgarten, J. Unlocking guidance for discrete state-space diffusion and flow models. *arXiv preprint arXiv:2406.01572*, 2024.

Pike, A., Williamson, B., Harlfinger, S., Martin, S., and McGinnity, D. F. Optimising proteolysis-targeting chimeras (PROTACs) for oral drug delivery: a drug metabolism and pharmacokinetics perspective. *Drug Discovery Today*, 25(10):1793–1800, 2020. doi: 10.1016/j.drudis.2020.07.013.

Qin, Y., Madeira, M., Thanou, D., and Frossard, P. Defog: Discrete flow matching for graph generation. In *Proceedings of the 42nd International Conference on Machine Learning*, 2025. URL https://proceedings.mlr.press/v267/qin25d.html.

Stark, H., Jing, B., Wang, C., Corso, G., Berger, B., Barzilay, R., and Jaakkola, T. Dirichlet flow matching with applications to dna sequence design. *arXiv preprint arXiv:2402.05841*, 2024.

Tang, S., Zhang, Y., and Chatterjee, P. Peptune: De novo generation of therapeutic peptides with multi-objective-guided discrete diffusion, 2025a. URL https://arxiv.org/abs/2412.17780.

Tang, S., Zhang, Y., Tong, A., and Chatterjee, P. Gumbel-softmax score and flow matching for discrete biological sequence generation. In *ICLR 2025 Workshop on AI for Nucleic Acids*, 2025b. URL https://openreview.net/forum?id=ITpCmDhSfu.

Troup, R. I., Fallan, C., and Baud, M. G. J. Current strategies for the design of PROTAC linkers: a critical review. *Exploration of Targeted Anti-Tumor Therapy*, 1(5):273–312, 2020. doi: 10.37349/etat.2020.00018.

Vignac, C., Krawczuk, I., Siraudin, A., Wang, B., Cevher, V., and Frossard, P. Digress: Discrete denoising diffusion for graph generation. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=UaAD-Nu86WX.

Vikal, A., Maurya, R., Patel, B. B., Sharma, R., Patel, P., Patil, U. K., and Das Kurmi, B. Protacs in cancer therapy: mechanisms, design, clinical trials, and future directions. *Drug delivery and translational research*, 15(6):1801–1827, 2025.

Wang, C., Zhang, Y., Chen, W., Wu, Y., and Xing, D. New-generation advanced protacs as potential therapeutic agents in cancer therapy. *Molecular cancer*, 23(1):110, 2024a.

Wang, X., Zheng, Z., Ye, F., Xue, D., Huang, S., and Gu, Q. Diffusion language models are versatile protein learners. In Salakhutdinov, R., Kolter, Z., Heller, K., Weller, A., Oliver, N., Scarlett, J., and Berkenkamp, F. (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 52309–52333. PMLR, 2024b. URL https://proceedings.mlr.press/v235/wang24ct.html.

Yang, Y., Zheng, S., Su, S., Zhao, C., Xu, J., and Chen, H. SyntaLinker: automatic fragment linking with deep conditional transformer neural networks. *Chemical Science*, 11(31):8312–8322, 2020. doi: 10.1039/D0SC03126G.

Yoo, J., Kim, W., and Hong, S. Redi: Rectified discrete flow. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

Zeng, Y., Elliott, H., Maffettone, P., Greenside, P., Bastani, O., and Gardner, J. R. Antibody design with constrained bayesian optimization. In *GEM workshop, ICLR 2024*, 2024.

Appendix

A. Generated PROTACs Visualization

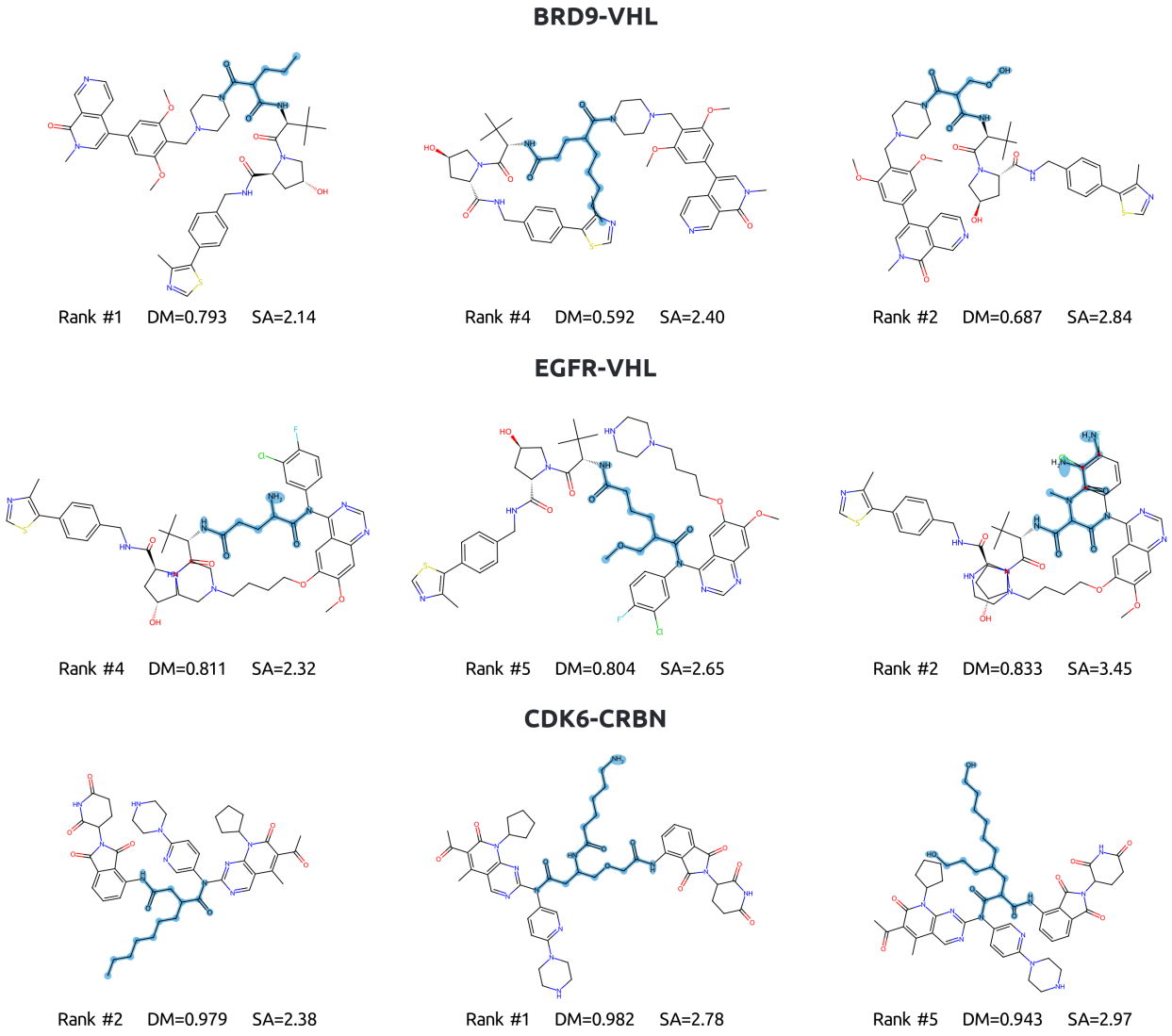

Figure A1. **Top three PROTACs generated for each target.** For each target system (BRD9–VHL, EGFR–VHL, CDK6–CRBN), the three PROTACs with the lowest synthetic-accessibility (SA) score among the top five by DegradMaster (DM) probability are shown horizontally. Each panel depicts the assembled conjugate with the *de novo*–generated linker highlighted in blue; the warhead and E3-ligand fragments are the fixed inputs. DM denotes the predicted degradation-activity probability (higher is better) and SA the synthetic-accessibility score (lower is better, range 1–10).

B. Theoretical Proofs

We provide a minimal theoretical justification for the objective-guided sampling mechanism used in CupOFLATTE. In particular, we show that the exponential tilting rule employed during discrete flow sampling arises as the unique solution to a KL-regularized objective improvement problem under hard validity constraints.

B.1. Notation and Setup

We adopt the notation from the main text. Let $\mathcal{Z}$ denote the finite discrete state space of PROTAC molecular graphs with fixed warhead and E3-ligand subgraphs. A discrete flow step is represented by a Markov kernel $K(\cdot \mid z)$ on $\mathcal{Z}$, where $K(z' \mid z)$ denotes the probability of proposing state z' given current state z. Deterministic updates are recovered as the special case where $K(\cdot \mid z)$ is a point mass.

Validity is encoded by an indicator function $\mathbf{1}_{\text{valid}} : \mathcal{Z} \to \{0, 1\}$, which specifies whether a candidate state satisfies the chemical and representational constraints in Section 4.5. Objective guidance is defined by a score function $S : \mathcal{Z} \to \mathbb{R}$ and a guidance strength $\beta > 0$.

B.2. Objective Guidance as KL-Regularized Exponential Tilting

We show that exponential tilting is the unique optimal way to incorporate objective information into a discrete flow step while remaining close to a base kernel and enforcing hard validity constraints.

Theorem B.1 (Exponential tilting solves KL-regularized objective improvement)**.** *Fix a base kernel $K(\cdot \mid z)$ on $\mathcal{Z}$ and a score function $S : \mathcal{Z} \to \mathbb{R}$. For a fixed input state z, consider the optimization problem*

$$\max_{Q} \left\{ \mathbb{E}_{Z' \sim Q}[S(Z')] - \frac{1}{\beta} \mathrm{KL}(Q \,\|\, K(\cdot \mid z)) \right\} \quad \textit{subject to} \quad Q(z') = 0 \;\; \textit{if} \;\; \mathbf{1}_{\text{valid}}(z') = 0, \tag{18}$$

where $\beta > 0$. Then the unique maximizer is

$$Q^\star_\beta(z' \mid z) = \frac{K(z' \mid z)\, \mathbf{1}_{\text{valid}}(z') \, \exp\big(\beta S(z')\big)}{\sum_{y \in \mathcal{Z}} K(y \mid z)\, \mathbf{1}_{\text{valid}}(y) \, \exp\big(\beta S(y)\big)}. \tag{19}$$

Proof. Let $\mathcal{Z}_{\text{ok}} := \{z' \in \mathcal{Z} : \mathbf{1}_{\text{valid}}(z') = 1\}$ and restrict Q to be supported on $\mathcal{Z}_{\text{ok}}$. The objective in (18) can be written as

$$\sum_{z' \in \mathcal{Z}_{\text{ok}}} Q(z')\, S(z') - \frac{1}{\beta} \sum_{z' \in \mathcal{Z}_{\text{ok}}} Q(z') \log \frac{Q(z')}{K(z' \mid z)}. \tag{20}$$

Introduce a Lagrange multiplier λ enforcing $\sum_{z' \in \mathcal{Z}_{\text{ok}}} Q(z') = 1$. Differentiating with respect to $Q(z')$ and setting the derivative to zero yields

$$S(z') - \frac{1}{\beta}(\log Q(z') - \log K(z' \mid z) + 1) - \lambda = 0. \tag{21}$$

Rearranging gives

$$\log Q(z') = \log K(z' \mid z) + \beta S(z') + c, \tag{22}$$

for a constant c, implying

$$Q(z') \propto K(z' \mid z) \exp\big(\beta S(z')\big) \quad \text{on } \mathcal{Z}_{\text{ok}}. \tag{23}$$

Normalizing over $\mathcal{Z}_{\text{ok}}$ yields (19). The objective is strictly concave in Q, so the maximizer is unique. □

Corollary B.2 (Limits of guided kernels)**.** *Assume $K(\cdot \mid z)$ assigns positive mass to at least one valid state. Then: (i) as $\beta \to 0$, $Q^\star_\beta(\cdot \mid z)$ converges to $K(\cdot \mid z)$ restricted and renormalized to valid states; (ii) as $\beta \to \infty$, $Q^\star_\beta(\cdot \mid z)$ concentrates on valid maximizers of S within the support of $K(\cdot \mid z)$.*

Proof. Both statements follow directly from standard properties of the log-sum-exp normalization in (19). □

C. Base Model and Scorer Architectures

C.1. DeFoG Base Model

We use the pre-trained DeFoG discrete flow matching model (Qin et al., 2025) as our base PROTAC linker generator. DeFoG operates directly on molecular graphs rather than SMILES strings, representing nodes (atoms) and edges (bonds) as separate discrete random variables and learning a continuous-time discrete flow between a structured prior and the data distribution. The model uses a graph transformer architecture with separate node and edge update streams. We use the released pre-trained weights without further fine-tuning; the linker subgraph is the only region the flow modifies, with warhead and E3-ligand subgraphs held fixed throughout sampling (Section 4.4).

C.2. PROTAC-Likeness Classifier

The soft scorer S_1 (PROTAC-likeness; Section 4.5) is a logistic regression model over Morgan fingerprints of the linker subgraph. Linkers are fingerprinted at radius 2 with 1024 bits using RDKit (Landrum et al., 2025), and the logistic regression is trained to discriminate PROTAC-DB linkers from a chemotype-matched negative set. The trained model is loaded from `protac_linker_clf.pkl`; predictions are gated to linkers with $3 \leq |\text{heavy atoms}| \leq 20$, with liability-flagged candidates mapped to zero. Configuration in Table 7.

Table 7. PROTAC-likeness classifier configuration.

Parameter	Value
Fingerprint type	Morgan (RDKit)
Radius	2
Bit length	1024
Classifier	Logistic regression (sklearn)
Heavy-atom gate	$[3, 20]$
Liability filter	Hydrazide, azo, N-oxide, N–O, carbamate

C.3. DegradeMaster Activity Predictor

For evaluation in Section 5.3 we use DegradeMaster (Liu et al., 2025), an independent neural classifier trained to predict whether a candidate PROTAC induces target degradation. DegradeMaster takes the assembled PROTAC SMILES as input and returns a probability in $[0, 1]$. We use the released pre-trained model and report mean predicted probability along with the fraction exceeding thresholds $0.3, 0.5, 0.7$. DegradeMaster is used solely for evaluation in this work; it is not used as a guidance objective during sampling.

D. PROTAC Validity Constraints

We detail the implementation of the validity classifier $\mathbf{1}_{\text{valid}}$ and the soft connectivity scorer S_4 from Section 4.5. All checks are implemented in RDKit (Landrum et al., 2025).

D.1. Hard Validity Gate

A generated PROTAC is accepted as valid only if all three checks in Table 8 pass. Failure on any check returns $\mathbf{1}_{\text{valid}}(z) = 0$, which masks the candidate to zero probability under the tilted distribution during sampling.

Check	Pattern / Procedure	Failure mode
Representational validity	`Chem.SanitizeMol(mol)` succeeds AND `MolFromSmiles(MolToSmiles(mol))` round-trips	Bad valences or aromaticity; molecule cannot be persisted
Liability filter	`[NX3][NX3]`, `[#7]=[#7]`, `[N+][O-]`, `[#7]-[OX2]`, `[NX3]C(=[OX1])[OX2H1]`	Hydrazide, azo, N-oxide, N–O bond, carbamate — chemistries that decompose, are toxic, or fail amide assembly
Reactive termini count ≥ 2	Count of `[#6][CX3](=[OX1])[OX2H1]` (C-COOH) + `[#6][CX3H1](=[OX1])` (C-CHO) ≥ 2	Fewer than two reactive termini; warhead/E3 amide-coupling assembly impossible

Table 8. Hard validity gate checks. All three must pass for $\mathbf{1}_{\text{valid}}(z) = 1$.

D.2. Soft Connectivity Scorer

The connectivity scorer S_4 is computed even on states with $\mathbf{1}_{\text{valid}}(z) = 0$, providing a gradient toward single-component linkers during early sampling. Let $|\text{frag}|$ denote the number of connected fragments in the assembled graph, $|\text{largest}|$ the

size of the largest fragment, and |total| the total atom count. Then

$$S_4(z) = \begin{cases} 1.0 & \text{if } |\text{frag}| = 1, \\ \dfrac{|\text{largest}|}{|\text{total}| \cdot |\text{frag}|} & \text{otherwise.} \end{cases} \tag{24}$$

The score is halved if no path exists through the linker subgraph between the warhead anchor a^{wh} and the E3-ligand anchor a^{e3}.

D.3. Termini Conventions for Benchmark Systems

Each PROTAC system in our experiments specifies fixed warhead and E3-ligand anchor atoms at which amide-coupling assembly occurs. The conventions used are listed in Table 9.

System	Warhead anchor	E3 anchor
EGFR–VHL	Gefitinib-piperazine NH	VH032 NH_2 (tert-leucine)
BRD9–VHL	BI-7273-piperazine NH	VH032 NH_2 (tert-leucine)
CDK6–CRBN	Palbociclib-piperazine NH	Pomalidomide aryl-NH_2

Table 9. Fixed termini conventions for benchmark PROTAC systems.

E. Sampling Implementation

E.1. Notation

Table 10. Notation used in the sampling algorithm.

Symbol	Description
Φ_{θ^*}	Pre-trained discrete flow model with parameters θ^*
$\mathcal{Z}$	Discrete state space (PROTAC molecular graphs)
μ_0	Prior distribution (empty linker, fixed warhead/E3)
K	Total number of guided sampling steps
K_{warm}	Number of unguided warmup steps
$S(z)$	Aggregate guidance score (Section 4.5)
β	Guidance strength (temperature parameter)
$\mathbf{1}_{\text{valid}}(z)$	Hard validity indicator
Z^*	Best PROTAC graph found during sampling

E.2. Two-Phase Sampling Framework

Sampling proceeds in two phases (Algorithm 1). The first phase performs unguided sampling from the pre-trained flow for K_{warm} steps, growing the linker outward from the anchors during the early-branching regime where the soft objectives provide limited discriminative signal (Section 4.8). The second phase applies exponential tilting with hard validity masking and an acceptance criterion for the remaining $K - K_{\text{warm}}$ steps, optimizing the aggregate objective $S(z)$ while maintaining chemical validity.

E.2.1. Phase 1: Warmup

The warmup phase samples directly from the pre-trained flow: $Z_{k+1} \sim \Phi_{\theta^*}(Z_k)$ for $k = 0, \ldots, K_{\text{warm}} - 1$. No tilting and no acceptance check are applied. The best-so-far state Z^* is tracked over warmup steps so that any high-scoring intermediate configurations are retained.

E.2.2. Phase 2: Guided Optimization

The guided phase applies exponential tilting with hard validity masking and an acceptance criterion. At each step, a free linker site i (atom slot in $\mathcal{I}_{\text{link}}$ or bond entry incident to it) is selected, and the model's predicted distribution p_{k+1} at site i is

tilted by the aggregate score S and masked by validity.

Table 11. Two-phase sampling parameters.

Parameter	Value	Description
Total steps (K)	1500	Total sampling steps (warmup + guided)
Warmup steps (K_{warm})	100	Unguided sampling from pre-trained flow
Guided steps	1400	$K - K_{\text{warm}}$, with tilting + acceptance
Guidance strength (β)	5.0	Exponential tilting temperature
Top-K	50	Candidates per site for tilting
Temperature	1.0	Softmax temperature for predictions
Position selection	Random	Uniform random free linker site each step
`use_acceptance`	Enabled	Reject proposals with $S(Z') < S(Z_k)$

Log-Space Computation. For numerical stability, we compute the tilted distribution in log-space:

$$\tilde{p}(z') = \log p_k(z') + \beta \cdot S(z') + \log \mathbf{1}_{\text{valid}}(z'), \tag{25}$$

$$\tilde{p}_{\text{stable}}(z') = \tilde{p}(z') - \max_{z''} \tilde{p}(z''), \tag{26}$$

$$p'(z') = \frac{\exp(\tilde{p}_{\text{stable}}(z'))}{\sum_{z''} \exp(\tilde{p}_{\text{stable}}(z''))}. \tag{27}$$

Tilting and validity masking are restricted to the TopK candidates under p_{k+1} to limit per-step scorer evaluation cost.

Fallback Behavior. If exponential tilting produces invalid probabilities (all candidates fail validity, or numerical underflow), we fall back to sampling from the base distribution p_{k+1} restricted to valid candidates. If no valid candidate exists in the TopK, the step is rejected and $Z_{k+1} = Z_k$.

Acceptance Criterion. After sampling $z^\dagger \sim p'$ and constructing the candidate state Z', we apply the acceptance criterion

$$Z_{k+1} = \begin{cases} Z' & \text{if } S(Z') \geq S(Z_k), \\ Z_k & \text{otherwise.} \end{cases} \tag{28}$$

This ensures monotonic non-decrease of S over the guided phase and, combined with tracking Z^*, returns the highest-scoring valid PROTAC encountered during sampling.

E.3. Objective Weights

The aggregate objective $S(z) = \sum_{j=1}^{4} w_j S_j(z)$ uses the weights in Table 12. The connectivity weight $w_4 = 4.0$ is elevated relative to the other scorers so that the topological gradient toward connected, anchor-bridging linkers dominates early in sampling, where partial assemblies score similarly under S_1–S_3.

Table 12. Soft objective weights for the aggregate score $S(z)$.

Scorer	Description	Weight
S_1	PROTAC-likeness (logreg over Morgan FP)	1.0
S_2	Synthetic accessibility ramp	1.0
S_3	PROTAC-DB chemotype closeness	1.0
S_4	Soft connectivity gradient	4.0

E.4. Computational Requirements

Sampling experiments were conducted on a single NVIDIA A6000 GPU with 48 GB of VRAM. Generation produces 50 PROTACs per target as a single parallel batch of size 50; sampling timings below are for the entire batch (not per PROTAC).

For each batch, the warmup phase (100 steps of unguided DeFoG denoising) completes in approximately 30–45 seconds, and the guided phase (1400 steps of one-position-per-step exponential tilting with hard validity gating) takes approximately 7–10 minutes on an unloaded host. Total wall-clock time per target+method is approximately 8–11 minutes under typical load.

Throughput is CPU-bound rather than GPU-bound: each guided step evaluates the classifier + soft-scorer stack (RDKit sanitization, SMARTS pattern matching, sklearn LogReg, SA score, MW/logP/RotBonds windows, connectivity check) on every candidate type, with GPU forward passes consuming under 20% of per-step wall-clock.