AReUReDi: Annealed Rectified Updates for Refining Discrete Flows with Multi-Objective Guidance

Anonymous authors
Paper under double-blind review

Abstract

Biomolecular design requires individual candidates to combine strong target activity with favorable developability. Multi-objective generation must therefore address both trade-off exploration and the weakest properties within each candidate. We introduce **AReUReDi** (**A**nnealed **Re**ctified **U**pdates for **Re**fining **Di**screte Flows), an inference-time method for preference-controlled refinement with frozen discrete generators. AReUReDi combines generator-informed token proposals with annealed Tchebycheff guidance to target limiting properties and a weighted-utility acceptance rule to preserve aggregate quality throughout refinement. Across five predicted binding and developability objectives, AReUReDi outperformed all five comparators in candidate-level balance on both peptide benchmarks, achieving stronger worst-objective scores and more even property profiles. Preference variation enabled controllable affinity, and its leading balance on PPP5 persisted under predictor refitting. Experiments across amino-acid sequences and chemically modified peptide SMILES demonstrated broad applicability to peptide design. AReUReDi provides a unified refinement framework for designing peptide candidates with balanced predicted binding and developability profiles and tunable property priorities.

1 Introduction

Peptide candidates must combine target binding with properties that support formulation, persistence, and tolerability (Fosgerau & Hoffmann, 2015; Swanson, 2014; Tang et al., 2025b). High affinity alone does not establish a useful developability profile. Multi-objective optimization is therefore central to computational peptide design, where each sequence must satisfy several competing criteria (Naseri & Koffas, 2020). The relevant design unit is an individual candidate, and its weakest property can limit its value for further evaluation.

Evolutionary search and Bayesian optimization explore competing objectives through populations, surrogate models, and sequential evaluations (Zitzler & Thiele, 1998; Deb, 2011; Frisby & Langmead, 2021). Generative methods further use learned molecular distributions to guide multi-objective search (Li et al., 2018; Sousa et al., 2021; Yao et al., 2024). Pareto coverage characterizes the range of trade-offs in a returned set. Within-candidate balance characterizes whether a single sequence combines favorable properties. These complementary criteria require separate measurements: a diverse frontier can contain candidates with weak individual properties, and a compact group of balanced candidates can provide limited frontier coverage.

Discrete generative models provide token probabilities that can guide refinement in categorical sequence spaces (Campbell et al., 2024; Gat et al., 2024; Sahoo et al., 2024). Existing guidance methods support property-directed sampling (Nisonoff et al., 2025; Tang et al., 2025a), and MOG-DFM incorporates multiple objectives through transition reweighting and directional filtering (Chen et al., 2025). PepTune combines discrete diffusion and tree search for multi-objective peptide SMILES design (Tang et al., 2025b). These advances motivate a local refinement mechanism that uses learned sequence information to propose edits, responds to limiting properties, and controls the aggregate utility of accepted candidates.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Rectified Discrete Flow (ReDi) improves the generator used for this refinement by updating source–target couplings and reducing factorization error in discrete transitions (Yoo et al., 2025). Its continuous counterpart, rectified flow, learns straighter transport paths for few-step generation (Liu et al., 2023). We use frozen ReDi token probabilities to guide candidate substitutions and evaluate each substitution with fixed property scores. Separating pretrained sequence modeling from inference-time refinement allows the same generator to support different target and developability priorities.

Here, we introduce **AReUReDi**, which combines proposals guided by the weakest weighted property with acceptance that preserves weighted-sum utility. Annealed Tchebycheff guidance increasingly favors candidate edits with stronger limiting scores, and a shared preference vector controls both proposal guidance and acceptance. This construction maintains sensitivity to weak properties while preventing decreases in the specified aggregate utility. On two five-objective peptide benchmarks, AReUReDi led all three candidate-balance metrics among six methods. Preference control increased affinity and coverage, and experiments with amino-acid sequences and chemically modified peptide SMILES established the same refinement construction across representations.

Our key contributions are:

1. We introduce generator-informed refinement that couples annealed Tchebycheff proposals with weighted-utility acceptance to jointly optimize properties within individual sequences, without retraining the generator for each objective.
2. We prove nondecreasing weighted utility for the implemented refinement rule and establish invariance, concentration, and Pareto representability for an associated exact Metropolis–Hastings formulation under explicit assumptions.
3. We demonstrate the strongest candidate-level balance among six methods on both peptide benchmarks, preference-dependent affinity and coverage gains, and refinement across amino-acid and chemically modified peptide representations.

2 PRELIMINARIES

2.1 DISCRETE FLOW MATCHING

Let $\mathcal{S} = \mathcal{V}^L$ denote the discrete state space, where $\mathcal{V}$ is a vocabulary of size K and each $x = (x^1, \ldots, x^L) \in \mathcal{S}$ is a sequence of tokens. A *discrete flow matching (DFM)* model (Campbell et al., 2024; Gat et al., 2024; Dunn & Koes, 2024) defines a probability path $\{p_t\}_{t\in[0,1]}$ interpolating between a simple source distribution p_0 and a target distribution p_1 by means of a coupling $\pi(x_0, x_1)$ and conditional bridge distributions $p_t(x_t \mid x_0, x_1)$. For x_t^i denoting token i at time t, the model is trained to approximate conditional transitions $p_{s|t}(x_s \mid x_t)$ for $0 \le t < s \le 1$.

Finite-step implementations approximate joint transitions over L coordinates by a factorization

$$p_{s|t}(x_s \mid x_t) \;\approx\; \prod_{i=1}^{L} p_{s|t}\big(x_s^i \mid x_t\big)\,,$$

which introduces a discrepancy measured by the conditional total correlation, with KL denoting Kullback–Leibler divergence,

$$\mathrm{TC}_{s|t} \;=\; \mathbb{E}_{X_t\sim p_t}\left[\mathrm{KL}\left(p_{s|t}(\cdot \mid X_t) \;\middle\|\; \bigotimes_{i=1}^{L} p_{s|t}(\cdot^i \mid X_t)\right)\right].$$

Here $\bigotimes$ denotes the product of coordinate marginals. This quantity measures the inter-dimensional dependencies neglected by factorized transitions in few-step generation (Yoo et al., 2025). Reducing this transition error is central to improving few-step discrete generation.

2.2 RECTIFIED DISCRETE FLOW

To mitigate factorization error, **Rectified Discrete Flow (ReDi)** (Yoo et al., 2025) introduces an iterative rectification of the coupling π. Let $r \ge 0$ index rectification rounds. Starting from an

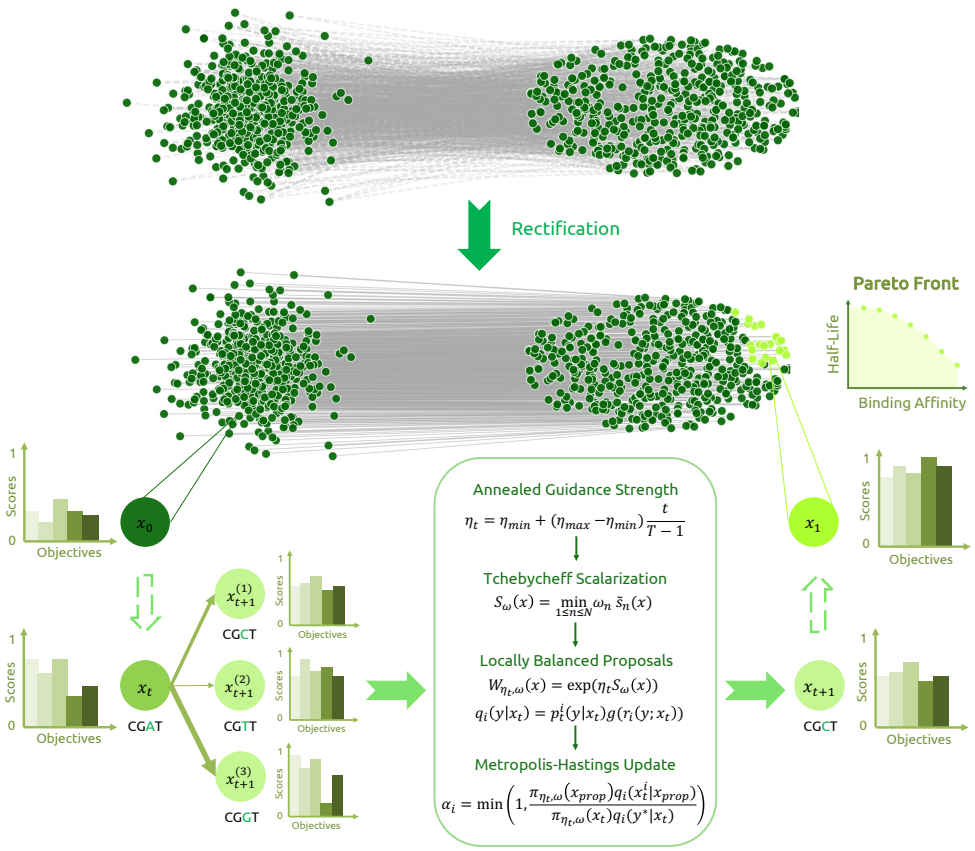

Figure 1: **AReUReDi overview.** A discrete generator returns candidate-token probabilities. Annealed Tchebycheff guidance reweights single-position proposals. The displayed proposal expression gives unnormalized weights; sampling and the MH ratio use normalized proposal probabilities. The displayed MH update represents the exact-target theoretical formulation. In the main experiments, the proposal construction uses a weighted-sum monotone gate: accept if $U_\omega(x_{\text{prop}}) \geq U_\omega(x)$ (Eq. 4). The plotted Pareto front is schematic. In the graphic, t and $\tilde{s}_n$ correspond to refinement index k and guidance score h_n in the text.

initial coupling $\pi^{(0)}(x_0, x_1)$, a DFM is trained under $\pi^{(r)}$ to produce new source–target pairs, with parameters $\theta^{(r)}$. The next coupling consists of pairs sampled from the source and the trained generator,

$$\pi^{(r+1)}(x_0, x_1) = p_0(x_0)\, p_{\theta^{(r)}}(x_1 \mid x_0).$$

Yoo et al. (2025) analyze this coupling update through reductions in conditional total correlation. We use the rectified generator to obtain token probabilities for multi-objective refinement.

2.3 Multi-Objective Setup

Let $\mathcal{S}_0 \subseteq \mathcal{S}$ denote the finite search support and let $h_n : \mathcal{S}_0 \to \mathbb{R}$, $n = 1, \ldots, N$, be fixed, higher-is-better guidance scores. These scores include the experiment-specific transformations in Appendix F. For nonnegative coefficients $\omega = (\omega_1, \ldots, \omega_N)$ with $\sum_n \omega_n > 0$, the weighted max–min score is

$$S_\omega(x) = \min_{1 \le n \le N} \omega_n h_n(x). \tag{1}$$

This Tchebycheff-type scalarization rewards the smallest weighted score (Bowman Jr, 1976; Miettinen, 1999). We use normalized preferences in $\Delta^{N-1} = \{\omega \in \mathbb{R}^N_{\geq 0} : \sum_n \omega_n = 1\}$ for the five-property amino-acid experiments; additional analyzer coefficients for SMILES are specified separately. A state is Pareto optimal if no feasible competitor is at least as good in all objectives and strictly better in one. For evaluation, $\tilde{s}_n(x)$ denotes a normalized favorable property score, distinct from any additional feasibility score used during guidance.

3 AREUREDI: PREFERENCE-CONTROLLED SEQUENCE REFINEMENT

AReUReDi refines a sequence through two complementary decisions. A generator-informed proposal favors substitutions with stronger limiting weighted scores, and the acceptance rule retains substitutions that preserve aggregate utility (Figure 1). Both decisions use the same preference vector, and the trajectory returns the final sequence x_T. Algorithm 1 specifies the implemented update.

3.1 GENERATOR-INFORMED CANDIDATE EDITS

At iteration k, the current sequence is $x_k \in \mathcal{S}_0$. We select an editable position i uniformly and obtain token probabilities $b_{k,i}(y \mid x_k)$ from a frozen generator at model time $t_k \in [0, 1]$. Replacing token i by y produces $x_k^{(i \leftarrow y)}$. The candidate set $\mathcal{C}_i(x_k)$ excludes the current token and task-specific forbidden tokens; nucleus filtering and a SMILES top-200 cap limit candidate evaluations (Appendix F). We evaluate every candidate in this set with the fixed guidance scores.

3.2 ANNEALED TCHEBYCHEFF GUIDANCE

Guidance compares each candidate's limiting weighted score with that of the current sequence. Let $T > 1$ be the refinement budget and $0 \le \eta_{\min} \le \eta_{\max}$ the guidance endpoints. We increase guidance over refinement through

$$\eta_k = \eta_{\min} + (\eta_{\max} - \eta_{\min}) \frac{k}{T-1}, \qquad k = 0, \dots, T-1. \tag{2}$$

Small initial guidance gives greater relative influence to learned token probabilities. Larger guidance increasingly favors positive changes in S_ω. Preferences remain fixed within a trajectory and vary across runs to generate different property profiles.

3.3 BARKER-BALANCED PROPOSALS

For candidate token $y \in \mathcal{C}_i(x_k)$, define the guidance ratio and normalized proposal

$$\begin{aligned} r_{k,i}(y; x_k) &= \exp\Big\{ \eta_k \big[S_\omega(x_k^{(i \leftarrow y)}) - S_\omega(x_k) \big] \Big\}, \\ q_{k,i}(y \mid x_k) &= \frac{b_{k,i}(y \mid x_k) g(r_{k,i}(y; x_k))}{\sum_{z \in \mathcal{C}_i(x_k)} b_{k,i}(z \mid x_k) g(r_{k,i}(z; x_k))}, \qquad g(u) = \frac{u}{1+u}. \end{aligned} \tag{3}$$

Barker's balancing function satisfies $g(u) = ug(1/u)$, a standard relation for locally informed proposals (Zanella, 2017). Its bounded multiplier favors positive score changes and weights each token by its generator probability. We sample $y^\star \sim q_{k,i}(\cdot \mid x_k)$ and set $x' = x_k^{(i \leftarrow y^\star)}$.

3.4 ACCEPTANCE THAT PRESERVES WEIGHTED UTILITY

The proposal score prioritizes limiting properties. Acceptance evaluates the complete weighted profile through

$$U_\omega(x) = \sum_{n=1}^{N} \omega_n h_n(x), \qquad x_{k+1} = \begin{cases} x' & U_\omega(x') \ge U_\omega(x_k), \\ x_k & \text{otherwise.} \end{cases} \tag{4}$$

Every update preserves or increases U_ω, including accepted moves with equal utility. This permits property trade-offs and plateau transitions without reducing aggregate utility. The same ω influences both the Tchebycheff bottleneck and weighted-sum acceptance. The two scores have distinct roles: the guarantee concerns U_ω, and S_ω need not increase at every accepted step.

Appendix A.3 proves utility monotonicity and a finite bound on strictly improving updates. Appendix A also analyzes an associated exact Metropolis–Hastings (MH) kernel (Hastings, 1970), including fixed-target invariance, high-guidance concentration, and positive-score Pareto representability. That kernel requires exact base-density ratios and normalized reverse proposals. Our experiments use Eq. 4; the ungated ablation accepts every guided proposal.

Table 1: **Candidate-level balance on two five-objective peptide benchmarks.** Each metric is computed within a candidate, then averaged over 100 candidates. Min.: worst normalized property; Disp.: population standard deviation across properties. Bold marks the best mean per target. Diversity and objective-level scores are reported in Tables S16 and S2. Score transformations are defined in Appendix H.

Target	Method	Min. ↑	Range ↓	Disp. ↓
1B8Q	AReUReDi	**0.516**	**0.483**	**0.186**
	MOPSO	0.440	0.551	0.216
	NSGA-III	0.386	0.605	0.225
	SMS-EMOA	0.335	0.654	0.250
	SPEA2	0.292	0.685	0.254
	PepTune+DPLM	0.103	0.752	0.266
PPP5	AReUReDi	**0.613**	**0.387**	**0.148**
	MOPSO	0.440	0.552	0.212
	NSGA-III	0.532	0.461	0.173
	SMS-EMOA	0.375	0.608	0.235
	SPEA2	0.321	0.655	0.245
	PepTune+DPLM	0.084	0.801	0.294

4 EXPERIMENTS

We evaluated whether AReUReDi could jointly improve predicted binding and developability within individual candidates, control the resulting property trade-offs, and transfer across amino-acid and chemically modified peptide representations. Frozen PepReDi and SMILESReDi generators provided token probabilities (Appendix B), and fixed predictors scored binding affinity, non-hemolysis, non-fouling, solubility, and, for amino-acid peptides, half-life. The main comparison used 100 outputs per method and target under method-specific generation budgets. We measured each candidate's worst normalized property, objective range, and dispersion, together with property means, sequence diversity, and set-level coverage. These measurements separated within-candidate property balance from the breadth of trade-offs represented by the returned set.

4.1 AREUREDI JOINTLY OPTIMIZES BINDING AND DEVELOPABILITY

We first tested whether AReUReDi improved the balance of binding and developability within individual peptide candidates. We compared AReUReDi with MOPSO (Coello & Lechuga, 2002), NSGA-III (Deb & Jain, 2013), SMS-EMOA (Beume et al., 2007), SPEA2 (Zitzler et al., 2001), and PepTune+DPLM on 1B8Q and PPP5 (Zhang et al., 1999; Yang et al., 2004). Among the six methods, AReUReDi achieved the highest mean worst-objective score on both targets, reaching 0.516 on 1B8Q and 0.613 on PPP5 compared with the strongest alternative values of 0.440 and 0.532, respectively (Table 1). It also attained the smallest mean objective range and dispersion on both targets. Mean non-fouling and solubility exceeded 0.86, and sequence diversity remained 0.881 and 0.899 on the two benchmarks (Tables S2 and S16). AReUReDi therefore led all three candidate-level balance metrics on both benchmarks, combining a stronger weakest property with more even predicted property profiles.

For the generative comparison, we adapted PepTune's multi-objective tree search (Tang et al., 2025b) to the DPLM amino-acid backbone (Wang et al., 2024). Relative to PepTune+DPLM, AReUReDi achieved higher mean values for all five objectives on both targets (Table S2). Its mean worst-objective scores of 0.516 and 0.613 substantially exceeded PepTune+DPLM's 0.103 and 0.084, and its objective ranges and dispersions were smaller on both benchmarks (Table 1). The comparison therefore favored AReUReDi at both the individual-property and within-candidate balance levels.

We next tested whether the joint profiles depended on guidance for each constituent property. On 7LUL, removing solubility guidance reduced mean solubility from 0.9398 to 0.4013, and removing

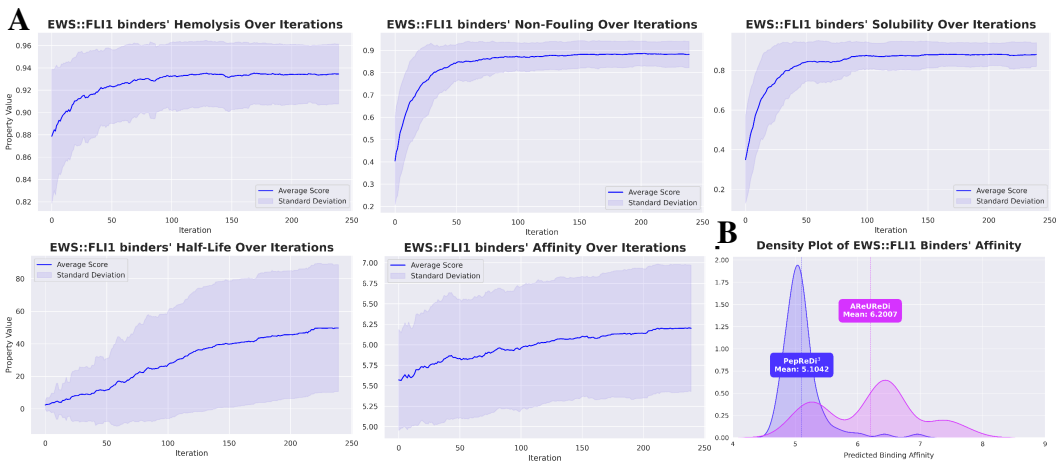

Figure 2: **Guided peptide refinement for EWS::FLI1.** Panels A and B show predicted-property trajectories with standard-deviation bands across 100 length-12 candidates and the affinity distribution of guided versus unconditional PepReDi[3] samples, respectively. The hemolysis score is oriented toward non-hemolysis.

affinity guidance reduced affinity from 6.2559 to 5.2611 (Table S7). On CLK1, removing non-fouling guidance reduced non-fouling from 0.8285 to 0.2902 even as predicted half-life increased from 74.04 to 96.59 h; removing half-life guidance reduced predicted half-life to 1.33 h (Table S8). These objective-removal experiments demonstrated that the favorable joint profiles depended on explicit guidance for the corresponding binding and developability properties in these tasks.

4.2 AReUReDi generates peptide candidates across targets and lengths

We next evaluated the breadth of amino-acid peptide refinement across eight proteins: structured targets with known binders (1B8Q, 5AZ8, 7JVS), additional structured targets (AMHR2, OX1R, DUSP12), and intrinsically disordered targets (EWS::FLI1, MYC). Across every evaluated setting, mean non-hemolysis exceeded 0.91, non-fouling exceeded 0.86, and solubility exceeded 0.85 (Table S5). AMHR2 and EWS::FLI1 were each evaluated at lengths 8, 12, and 16, and these developability thresholds were maintained at every tested length. AReUReDi therefore produced consistently favorable predicted developability profiles across all eight targets and all evaluated target–length settings.

We assessed selected candidates with AlphaFold3 interface predicted TM-score (ipTM) (Abramson et al., 2024) and AutoDock Vina (Trott & Olson, 2010), neither of which was used as an optimization objective. For 1B8Q, 5AZ8, and 7JVS, the selected AReUReDi candidates had higher predicted non-hemolysis, non-fouling, solubility, and half-life than the displayed pre-existing binders (Figures S1 and S2). Predicted affinity was also higher for the selected 1B8Q and 7JVS candidates, whereas the 5AZ8 pre-existing binder had higher predicted affinity. All three selected designs had more favorable AutoDock Vina scores than their corresponding pre-existing binders; ipTM was higher for the 7JVS design, matched for 5AZ8, and lower for 1B8Q. We also obtained predicted complexes and docking assessments for AMHR2, EWS::FLI1, DUSP12, and MYC, where no characterized binder was used for comparison (Figure S3). These out-of-objective assessments provided additional computational evidence for prioritizing candidates for experimental testing.

To measure how refinement changed the candidate population, we tracked five predicted properties across 100 length-12 EWS::FLI1 candidates at every iteration (Figure 2). The mean predicted property trajectories increased during refinement, with solubility and non-fouling rising from approximately 0.35–0.40 to 0.88–0.90. The final affinity distribution also shifted toward higher values relative to unconditional PepReDi[3] samples. AReUReDi therefore improved predicted binding and developability jointly across the EWS::FLI1 candidate population rather than only in selected endpoints.

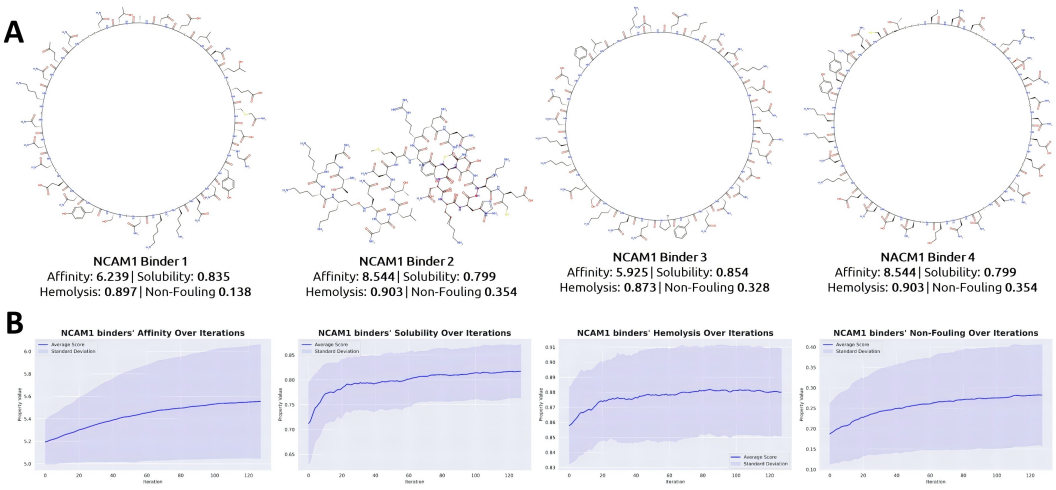

Figure 3: **(A)** Example 2D SMILES structure of AReUReDi-designed peptide binders with four property scores. **(B)** Plots showing the mean scores for each property across the number of iterations during AReUReDi's design of binders of length 200 for NCAM1.

4.3 Preference control and predictor-refit evaluation

We next tested whether the preference vector controlled the property profiles produced by AReUReDi. Increasing the affinity-designated coefficient from $\omega_{\text{aff}} = 0.2$ to 0.5 and 0.8 increased mean normalized affinity monotonically from 0.526 to 0.609 and 0.653 on 1B8Q, and from 0.615 to 0.694 and 0.818 on PPP5 (Table S18). We also compared uniform simplex sampling, boundary-focused sampling, and deterministic preference covering using 36 five-objective preferences and 108 outputs per strategy and target. The Dirichlet(1) sweep returned higher reported hypervolume than the equal-weight configuration, increasing from 0.576 to 0.616 on 1B8Q and from 0.695 to 0.817 on PPP5, with corresponding reductions in empirical IGD^+ (Tables S17 and S19). SMS-EMOA retained higher empirical set-level coverage in the reported comparison, separating AReUReDi's candidate-balance advantage from frontier-coverage performance. The preference experiments established direct control over predicted affinity and showed that preference variation broadened the reported trade-off coverage relative to equal weighting.

We then tested sensitivity to predictor fitting by training four bootstrap/initialization replicas per objective and rescoring 3,063 frozen candidates from AReUReDi and five baselines. Generated-population rank agreement was highest for non-fouling, affinity, and non-hemolysis, with Spearman ranges of 0.900–0.942, 0.813–0.938, and 0.769–0.843, respectively; solubility and half-life were more sensitive to predictor refitting (Table S20). Across 100 joint predictor combinations, AReUReDi retained the best mean worst-objective score, range, and dispersion on PPP5, including a mean worst-objective score of 0.560 compared with 0.477 for the strongest alternative. On 1B8Q, MOPSO ranked ahead after refitting, with a mean worst-objective score of 0.512 compared with 0.494 for AReUReDi. Predictor refitting therefore preserved AReUReDi's candidate-balance advantage on PPP5 and identified target-specific predictor sensitivity on 1B8Q. Complete target- and objective-specific results are reported in Appendix H.3.

4.4 AReUReDi generates chemically modified peptide SMILES

We tested whether the same refinement construction extended to chemically modified peptide representations using the rectified SMILESReDi generator. We designed candidates for five targets: glucagon-like peptide-1 receptor (GLP1), transferrin receptor (TfR), neural cell adhesion molecule 1 (NCAM1), glutamate transporter GLAST, and anti-Müllerian hormone receptor type 2 (AMHR2). Generation optimized predicted target affinity, non-hemolysis, solubility, and non-fouling. Across 100 candidates per target, refinement increased affinity and non-fouling for all five targets (Figures 3, S4, and S5), extending coordinated property refinement from amino-acid sequences to chemically modified peptide SMILES.

AReUReDi generated SMILES with validity and uniqueness of 1.000. Relative to PepTune, it increased diversity from 0.677 to 0.789 and reduced similarity to the nearest training neighbor from 0.486 to 0.392 while matching its 1.000 validity and uniqueness (Table S4). We further tested requested lengths of 100–400 tokenizer tokens at a fixed 128-update budget. All 120 outputs passed the peptide-validity assessment, with no output truncated in this evaluation; mean runtime increased from 301 to 582 s and peak memory from 16.8 to 26.3 GiB across the tested range (Table S23). AReUReDi therefore combined complete observed validity with greater diversity and lower training-set similarity than PepTune, and remained operational throughout the tested 100–400-token range.

4.5 Ablations identify the contributions of proposals, acceptance, and annealing

We first examined how the frozen generator and proposal distribution affected refinement. Comparing PepDFM, the base PepReDi model, and three-round PepReDi^3 showed target-dependent effects of rectification (Table S9). PepReDi^3 achieved the highest mean values for all five properties on AMHR2 and the highest predicted half-life on 5AZ8, although it did not lead every 5AZ8 property. The proposal distribution produced a clearer effect: replacing learned PepReDi probabilities with a uniform proposal reduced four of five mean property scores on 1B8Q and all five on PPP5 (Tables S12 and S13). Across 180 additional trajectories, PepReDi^3 proposals also produced the lowest endpoint cross-entropy proxy under the common frozen generator on both targets (Table S22). Learned generator probabilities therefore improved most optimized properties and continued to influence the distribution of refined endpoints.

We next isolated the two principal refinement choices. Under the same refinement-step budget, weighted-sum monotone acceptance improved all five mean objective scores over ungated guided proposals on both 8CN1 and 4EBP2 (Table S6). Annealed guidance produced the highest solubility and predicted half-life on 1DDV and the highest mean values for all five properties on P53 among the tested fixed and annealed schedules (Table S10). Finally, under matched objective-evaluation budgets, single-token proposals achieved higher hypervolume, higher worst-objective scores, higher affinity, and lower runtime than the tested paired-token construction on both targets (Table S24). These ablations support the implemented combination of generator-informed single-token proposals, annealed guidance, and monotone weighted-utility acceptance.

5 Related Work

Multi-objective search and scalarization. Evolutionary algorithms use nondominated sorting, strength-based selection, particle updates, or hypervolume to search trade-offs (Deb et al., 2002; Zitzler et al., 2001; Coello & Lechuga, 2002; Beume et al., 2007). Bayesian optimization uses learned surrogates (Shahriari et al., 2015), including biological sequence models (Stanton et al., 2022; Frisby & Langmead, 2021), expected hypervolume improvement (Emmerich & Klinkenberg, 2008), information gain (Belakaria et al., 2021), and diverse batch selection (Ahmadianshalchi et al., 2024). Scalarizations connect preferences to optimization through ParEGO, MOEA/D, and random scalarization (Knowles, 2006; Zhang & Li, 2007; Paria et al., 2020). Tchebycheff extensions include smooth objectives, online optimization, and set scalarization (Lin et al., 2024b; Liu et al., 2024; Lin et al., 2024a); R_2 utilities provide another preference-based construction (Tu et al., 2023). AReUReDi uses fixed predictors and couples a limiting-property proposal score to an aggregate-utility acceptance rule.

Guided discrete generation. Guided diffusion supports protein design (Gruver et al., 2023), and multi-objective GFlowNets learn distributions over high-reward candidates (Jain et al., 2023). Discrete flow models include categorical, simplex-based, and molecular constructions (Campbell et al., 2024; Gat et al., 2024; Stark et al., 2024; Davis et al., 2024; Dunn & Koes, 2024; Tang et al., 2025a). MOG-DFM reweights transitions and filters directions for multi-objective sequence generation (Chen et al., 2025). PepTune combines masked diffusion and tree search for peptide SMILES (Tang et al., 2025b). Our local refinement construction uses Barker-balanced proposals (Zanella, 2017) with frozen generator probabilities and acceptance that preserves weighted utility.

Continuous and molecular optimization. ParetoFlow, PROUD, and preference-guided diffusion address multi-objective generation in continuous or latent representations (Yuan et al., 2024; Yao et al.,

2024; Annadani et al., 2025). EGD and MUDM study multi-property 3D molecular generation (Sun et al., 2025; Han et al., 2023). Continuous global optimization and discrete relaxations address related search difficulties (Beliakov & Lim, 2007; Michael et al., 2024). AReUReDi directly refines amino-acid and peptide-SMILES tokens, with candidate-level balance as the primary empirical criterion. Appendix I discusses adjacent optimization tasks and property-modeling approaches represented in the related literature.

6 Discussion

AReUReDi enables preference-controlled multi-objective refinement of individual biomolecular candidates generated by pretrained rectified discrete flows. The method combines annealed Tchebycheff guidance, locally balanced proposals, and weighted-sum acceptance to improve limiting properties while maintaining high aggregate utility. Under the reference predictors, AReUReDi achieved the strongest candidate-level balance among six methods on both five-objective peptide benchmarks. Preference sweeps shifted the resulting property profiles and increased affinity when affinity received greater weight, establishing direct control over the trade-offs expressed by the refined candidates. These results establish rectified discrete flows as effective starting distributions for post-training multi-objective refinement under explicit property preferences.

Candidate-level balance is particularly relevant to peptide design because binding and developability must coexist within the same molecule. A candidate with high affinity can remain unsuitable if other properties such as hemolysis, solubility, or stability are poor. AReUReDi directly optimizes this joint requirement at the candidate level, and the resulting peptides achieved stronger worst-property balance than the evaluated alternatives on both targets. The same refinement procedure also improved candidates represented as amino-acid sequences and chemically modified peptide SMILES, demonstrating applicability across two distinct biomolecular tokenizations. Learned token proposals improved several individual property scores, and evaluation with refitted predictors retained the leading mean candidate-level balance on PPP5. Together, these experiments support preference-controlled refinement as a practical strategy for generating candidates with jointly favorable predicted property profiles.

The present evidence remains computational and depends on learned property predictors. Predictor refitting tested sensitivity to the objective models, but experimental measurements are required to determine whether the predicted improvements translate to binding and developability in physical assays. Candidate evaluation also accounts for a substantial fraction of the optimization cost when objective predictors are expensive. Future work will therefore test selected candidates experimentally, evaluate transfer across additional targets and molecular representations, and reduce evaluation cost through more selective proposal and scoring strategies. Within the scope of the current benchmarks, AReUReDi establishes that pretrained rectified discrete flows can be refined after training to generate individual biomolecular candidates with controllable and balanced multi-property profiles.

AI Use Statement

We acknowledge the use of AI to assist in polishing and editing parts of this manuscript. AI was used to refine phrasing, improve clarity, and ensure consistency of style across sections. All technical content, experiments, analyses, and conclusions were developed by the authors, with AI support limited to language refinement and editorial improvements.

Reproducibility Statement

Algorithm 1 specifies the practical refinement rule, and Appendix A states the assumptions and proofs. Appendices C, E, and F describe the generators, data sources, property predictors, and sampling configurations. Additional evaluation definitions, predictor-refit results, and computational comparisons appear in Appendices H and J.

Ethics Statement

This study used existing biological datasets and computational sequence generation, with no new human-subject or animal experiments. Generated peptides are candidate designs whose predicted properties require experimental assessment before biological or therapeutic use. We recognize potential dual-use risks in accelerating biomolecular optimization and emphasize responsible access, appropriate safety screening, and compliance with the licenses of reused datasets and models. Predictor bias and limited training coverage can affect candidate prioritization, as discussed in Appendix J.

References

Osama Abdin, Satra Nim, Han Wen, and Philip M Kim. Pepnn: a deep attention model for the identification of peptide binding sites. *Communications biology*, 5(1):503, 2022.

Josh Abramson, Jonas Adler, Jack Dunger, Richard Evans, Tim Green, Alexander Pritzel, Olaf Ronneberger, Lindsay Willmore, Andrew J Ballard, Joshua Bambrick, et al. Accurate structure prediction of biomolecular interactions with alphafold 3. *Nature*, pp. 1–3, 2024.

Alaleh Ahmadianshalchi, Syrine Belakaria, and Janardhan Rao Doppa. Pareto front-diverse batch multi-objective bayesian optimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 10784–10794, 2024.

Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. Optuna: A next-generation hyperparameter optimization framework. In *International Conference on Knowledge Discovery and Data Mining*, pp. 2623–2631, 2019.

Yashas Annadani, Syrine Belakaria, Stefano Ermon, Stefan Bauer, and Barbara E Engelhardt. Preference-guided diffusion for multi-objective offline optimization. *arXiv preprint arXiv:2503.17299*, 2025.

Valentin Artemyev, Anna Gubaeva, Anastasiia Iu Paremskaia, Amina A Dzhioeva, Andrei Deviatkin, Sofya G Feoktistova, Olga Mityaeva, and Pavel Yu Volchkov. Synthetic promoters in gene therapy: Design approaches, features and applications. *Cells*, 13(23):1963, 2024.

Zeynep Kalender Atak, Ibrahim Ihsan Taskiran, Jonas Demeulemeester, Christopher Flerin, David Mauduit, Liesbeth Minnoye, Gert Hulselmans, Valerie Christiaens, Ghanem-Elias Ghanem, Jasper Wouters, et al. Interpretation of allele-specific chromatin accessibility using cell state–aware deep learning. *Genome research*, 31(6):1082–1096, 2021.

Syrine Belakaria, Aryan Deshwal, and Janardhan Rao Doppa. Output space entropy search framework for multi-objective bayesian optimization. *Journal of artificial intelligence research*, 72:667–715, 2021.

Gleb Beliakov and Kieran F Lim. Challenges of continuous global optimization in molecular structure prediction. *European journal of operational research*, 181(3):1198–1213, 2007.

Nicola Beume, Boris Naujoks, and Michael Emmerich. Sms-emoa: Multiobjective selection based on dominated hypervolume. *European journal of operational research*, 181(3):1653–1669, 2007.

Alessandra Bigi, Eva Lombardo, Roberta Cascella, and Cristina Cecchi. The toxicity of protein aggregates: new insights into the mechanisms, 2023.

Jenny Bostrom, Chingwei V Lee, Lauric Haber, and Germaine Fuh. Improving antibody binding affinity and specificity for therapeutic development. In *Therapeutic Antibodies: Methods and Protocols*, pp. 353–376. Springer, 2008.

V Joseph Bowman Jr. On the relationship of the tchebycheff norm and the efficient frontier of multiple-criteria objectives. In *Multiple Criteria Decision Making: Proceedings of a Conference Jouy-en-Josas, France May 21–23, 1975*, pp. 76–86. Springer, 1976.

Jason D Buenrostro, Paul G Giresi, Lisa C Zaba, Howard Y Chang, and William J Greenleaf. Transposition of native chromatin for fast and sensitive epigenomic profiling of open chromatin, dna-binding proteins and nucleosome position. *Nature methods*, 10(12):1213–1218, 2013.

Anton Bushuiev, Roman Bushuiev, Petr Kouba, Anatolii Filkin, Marketa Gabrielova, Michal Gabriel, Jiri Sedlar, Tomas Pluskal, Jiri Damborsky, Stanislav Mazurenko, et al. Learning to design protein-protein interactions with enhanced generalization. *arXiv preprint arXiv:2310.18515*, 2023.

Andrew Campbell, Jason Yim, Regina Barzilay, Tom Rainforth, and Tommi Jaakkola. Generative flows on discrete state-spaces: Enabling multimodal flows with applications to protein co-design. In *Forty-first International Conference on Machine Learning*, 2024. URL https://openreview.net/forum?id=kQwSbv0BR4.

Shengfu Chen, Zhiqiang Cao, and Shaoyi Jiang. Ultra-low fouling peptide surfaces derived from natural amino acids. *Biomaterials*, 30(29):5892–5896, 2009.

Tong Chen, Yinuo Zhang, Sophia Tang, and Pranam Chatterjee. Multi-objective-guided discrete flow matching for controllable biological sequence design. *arXiv preprint arXiv:2505.07086*, 2025. URL https://arxiv.org/abs/2505.07086.

CA Coello Coello and Maximino Salazar Lechuga. Mopso: A proposal for multiple objective particle swarm optimization. In *Proceedings of the 2002 Congress on Evolutionary Computation. CEC'02 (Cat. No. 02TH8600)*, volume 2, pp. 1051–1056. IEEE, 2002.

Indraneel Das and John E Dennis. Normal-boundary intersection: A new method for generating the pareto surface in nonlinear multicriteria optimization problems. *SIAM journal on optimization*, 8 (3):631–657, 1998.

Oscar Davis, Samuel Kessler, Mircea Petrache, Ismail Ceylan, Michael Bronstein, and Joey Bose. Fisher flow matching for generative modeling over discrete data. *Advances in Neural Information Processing Systems*, 37:139054–139084, 2024.

Kalyanmoy Deb. Multi-objective optimisation using evolutionary algorithms: an introduction. In *Multi-objective evolutionary optimisation for product design and manufacturing*, pp. 3–34. Springer, 2011.

Kalyanmoy Deb and Himanshu Jain. An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part i: solving problems with box constraints. *IEEE transactions on evolutionary computation*, 18(4):577–601, 2013.

Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and TAMT Meyarivan. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE transactions on evolutionary computation*, 6(2):182–197, 2002.

Ian Dunn and David Ryan Koes. Exploring discrete flow matching for 3d de novo molecule generation. *ArXiv*, pp. arXiv–2411, 2024.

Vera D'Aloisio, Paolo Dognini, Gillian A Hutcheon, and Christopher R Coxon. Peptherdia: database and structural composition analysis of approved peptide therapeutics and diagnostics. *Drug Discovery Today*, 26(6):1409–1419, 2021.

Michael Emmerich and Jan-willem Klinkenberg. The computation of the expected improvement in dominated hypervolume of pareto front approximations. *Rapport technique, Leiden University*, 34: 7–3, 2008.

Keld Fosgerau and Torsten Hoffmann. Peptide therapeutics: current status and future directions. *Drug discovery today*, 20(1):122–128, 2015.

Trevor S Frisby and Christopher James Langmead. Bayesian optimization with evolutionary and structure-based regularization for directed protein evolution. *Algorithms for Molecular Biology*, 16 (1):13, 2021.

Itai Gat, Tal Remez, Neta Shaul, Felix Kreuk, Ricky TQ Chen, Gabriel Synnaeve, Yossi Adi, and Yaron Lipman. Discrete flow matching. *Advances in Neural Information Processing Systems*, 37: 133345–133385, 2024.

Nate Gruver, Samuel Stanton, Nathan Frey, Tim GJ Rudner, Isidro Hotzel, Julien Lafrance-Vanasse, Arvind Rajpal, Kyunghyun Cho, and Andrew G Wilson. Protein design with guided discrete diffusion. *Advances in neural information processing systems*, 36:12489–12517, 2023.

Chakradhar Guntuboina, Adrita Das, Parisa Mollaei, Seongwon Kim, and Amir Barati Farimani. Peptidebert: A language model based on transformers for peptide property prediction. *The Journal of Physical Chemistry Letters*, 14(46):10427–10434, 2023.

Xu Han, Caihua Shan, Yifei Shen, Can Xu, Han Yang, Xiang Li, and Dongsheng Li. Training-free multi-objective diffusion model for 3d molecule generation. In *The Twelfth International Conference on Learning Representations*, 2023.

W Keith Hastings. Monte carlo sampling methods using markov chains and their applications. 1970.

Moksh Jain, Sharath Chandra Raparthy, Alex Hernández-Garcıa, Jarrid Rector-Brooks, Yoshua Bengio, Santiago Miret, and Emmanuel Bengio. Multi-objective gflownets. In *International conference on machine learning*, pp. 14631–14653. PMLR, 2023.

Shipra Jain, Srijanee Gupta, Sumeet Patiyal, and Gajendra PS Raghava. Thpdb2: compilation of fda approved therapeutic peptides and proteins. *Drug Discovery Today*, pp. 104047, 2024.

Joshua Knowles. Parego: A hybrid algorithm with on-line landscape approximation for expensive multiobjective optimization problems. *IEEE transactions on evolutionary computation*, 10(1): 50–66, 2006.

Evan Komp, Christian Phillips, Lauren M Lee, Shayna M Fallin, Humood N Alanzi, Marlo Zorman, Michelle E McCully, and David AC Beck. Neural network conditioned to produce thermophilic protein sequences can increase thermal stability. *Scientific Reports*, 15(1):14124, 2025.

Ryan P Kreiser, Aidan K Wright, Natalie R Block, Jared E Hollows, Lam T Nguyen, Kathleen LeForte, Benedetta Mannini, Michele Vendruscolo, and Ryan Limbocker. Therapeutic strategies to reduce the toxicity of misfolded protein oligomers. *International journal of molecular sciences*, 21(22):8651, 2020.

Jinsen Li, Tsu-Pei Chiu, and Remo Rohs. Predicting dna structure using a deep learning method. *Nature communications*, 15(1):1243, 2024.

Krinos Li, Xianglu Xiao, Zijun Zhong, and Guang Yang. Accurate and generalizable protein-ligand binding affinity prediction with geometric deep learning. *arXiv preprint arXiv:2504.16261*, 2025.

Yibo Li, Liangren Zhang, and Zhenming Liu. Multi-objective de novo drug design with conditional graph generative model. *Journal of cheminformatics*, 10:1–24, 2018.

Xi Lin, Yilu Liu, Xiaoyuan Zhang, Fei Liu, Zhenkun Wang, and Qingfu Zhang. Few for many: Tchebycheff set scalarization for many-objective optimization. *arXiv preprint arXiv:2405.19650*, 2024a.

Xi Lin, Xiaoyuan Zhang, Zhiyuan Yang, Fei Liu, Zhenkun Wang, and Qingfu Zhang. Smooth tchebycheff scalarization for multi-objective optimization. *arXiv preprint arXiv:2402.19078*, 2024b.

Zeming Lin, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, Nikita Smetanin, Robert Verkuil, Ori Kabeli, Yaniv Shmueli, et al. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637):1123–1130, 2023.

Meitong Liu, Xiaoyuan Zhang, Chulin Xie, Kate Donahue, and Han Zhao. Online mirror descent for tchebycheff scalarization in multi-objective optimization. *arXiv preprint arXiv:2410.21764*, 2024.

Xingchao Liu, Chengyue Gong, and qiang liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=XVjTT1nw5z.

Deepika Mathur, Satya Prakash, Priya Anand, Harpreet Kaur, Piyush Agrawal, Ayesha Mehta, Rajesh Kumar, Sandeep Singh, and Gajendra PS Raghava. Peplife: a repository of the half-life of peptides. *Scientific reports*, 6(1):36617, 2016.

Richard Michael, Simon Bartels, Miguel González-Duque, Yevgen Zainchkovskyy, Jes Frellsen, Søren Hauberg, and Wouter Boomsma. A continuous relaxation for discrete bayesian optimization. *arXiv preprint arXiv:2404.17452*, 2024.

K. Miettinen. *Nonlinear multiobjective optimization*. Kluwer, Boston, USA, 1999.

Stephanie E Mohr, Yanhui Hu, Benjamin Ewen-Campen, Benjamin E Housden, Raghuvir Viswanatha, and Norbert Perrimon. Crispr guide rna design for research applications. *The FEBS journal*, 283 (17):3232–3238, 2016.

Gita Naseri and Mattheos AG Koffas. Application of combinatorial optimization strategies in synthetic biology. *Nature communications*, 11(1):2446, 2020.

Atef Nehdi, Nosaibah Samman, Vanessa Aguilar-Sánchez, Azer Farah, Emre Yurdusev, Mohamed Boudjelal, and Jonathan Perreault. Novel strategies to optimize the amplification of single-stranded dna. *Frontiers in Bioengineering and Biotechnology*, 8:401, 2020.

Nhat Khang Ngo, Thanh VT Tran, Viet Thanh Duy Nguyen, and Truong Son Hy. Latent-based directed evolution accelerated by gradient ascent for protein sequence design. *bioRxiv*, pp. 2024–04, 2024.

Hunter Nisonoff, Junhao Xiong, Stephan Allenspach, and Jennifer Listgarten. Unlocking guidance for discrete state-space diffusion and flow models. *Proceedings of the 13th International Conference on Learning Representations (ICLR)*, 2025.

Biswajit Paria, Kirthevasan Kandasamy, and Barnabás Póczos. A flexible framework for multi-objective bayesian optimization using random scalarizations. In *Uncertainty in Artificial Intelligence*, pp. 766–776. PMLR, 2020.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. Scikit-learn: Machine learning in python. *J. Mach. Learn. Res.*, 12(null):2825–2830, November 2011. ISSN 1532-4435.

William Peebles and Saining Xie. Scalable diffusion models with transformers. *arXiv preprint arXiv:2212.09748*, 2022.

M Pirtskhalava, B Vishnepolsky, and M Grigolava. Transmembrane and antimicrobial peptides. hydrophobicity, amphiphilicity and propensity to aggregation. *arXiv preprint arXiv:1307.6160*, 2013.

Anand Singh Rathore, Shubham Choudhury, Akanksha Arora, Purva Tijare, and Gajendra PS Raghava. Toxinpred 3.0: An improved method for predicting the toxicity of peptides. *Computers in Biology and Medicine*, 179:108926, 2024.

Dillon J Rinauro, Fabrizio Chiti, Michele Vendruscolo, and Ryan Limbocker. Misfolded protein oligomers: Mechanisms of formation, cytotoxic effects, and pharmacological approaches against protein misfolding diseases. *Molecular Neurodegeneration*, 19(1):20, 2024.

Sandra Romero-Molina, Yasser B Ruiz-Blanco, Joel Mieres-Perez, Mirja Harms, Jan Munch, Michael Ehrmann, and Elsa Sanchez-Garcia. Ppi-affinity: A web tool for the prediction and optimization of protein–peptide and protein–protein binding affinity. *Journal of proteome research*, 21(8): 1829–1841, 2022.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, pp. 234–241. Springer, 2015.

Subham Sekhar Sahoo, Marianne Arriola, Yair Schiff, Aaron Gokaslan, Edgar Marroquin, Justin T Chiu, Alexander Rush, and Volodymyr Kuleshov. Simple and effective masked diffusion language models. *Advances in Neural Information Processing Systems*, 2024.

Henri Schmidt, Minsi Zhang, Dimitar Chakarov, Vineet Bansal, Haralambos Mourelatos, Francisco J Sánchez-Rivera, Scott W Lowe, Andrea Ventura, Christina S Leslie, and Yuri Pritykin. Genome-wide crispr guide rna design and specificity analysis with guidescan2. *Genome biology*, 26(1): 1–25, 2025.

Bobak Shahriari, Kevin Swersky, Ziyu Wang, Ryan P Adams, and Nando De Freitas. Taking the human out of the loop: A review of bayesian optimization. *Proceedings of the IEEE*, 104(1): 148–175, 2015.

Neelam Sharma, Leimarembi Devi Naorem, Shipra Jain, and Gajendra PS Raghava. Toxinpred2: an improved method for predicting toxicity of proteins. *Briefings in bioinformatics*, 23(5):bbac174, 2022.

Tiago Sousa, João Correia, Vitor Pereira, and Miguel Rocha. Combining multi-objective evolutionary algorithms with deep generative models towards focused molecular design. In *Applications of Evolutionary Computation: 24th International Conference, EvoApplications 2021, Held as Part of EvoStar 2021, Virtual Event, April 7–9, 2021, Proceedings 24*, pp. 81–96. Springer, 2021.

Samuel Stanton, Wesley Maddox, Nate Gruver, Phillip Maffettone, Emily Delaney, Peyton Greenside, and Andrew Gordon Wilson. Accelerating bayesian optimization for biological sequence design with denoising autoencoders. In *International conference on machine learning*, pp. 20459–20478. PMLR, 2022.

Hannes Stark, Bowen Jing, Chenyu Wang, Gabriele Corso, Bonnie Berger, Regina Barzilay, and Tommi Jaakkola. Dirichlet flow matching with applications to dna sequence design. *Proceedings of the 41st International Conference on Machine Learning (ICML)*, 2024.

Marta M Stepniewska-Dziubinska, Piotr Zielenkiewicz, and Pawel Siedlecki. Development and evaluation of a deep learning model for protein–ligand binding affinity prediction. *Bioinformatics*, 34(21):3666–3674, 2018.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

Ruiqing Sun, Dawei Feng, Sen Yang, Yijie Wang, and Huaimin Wang. Evolutionary training-free guidance in diffusion model for 3d multi-objective molecular generation. *arXiv preprint arXiv:2505.11037*, 2025.

Kyle Swanson, Parker Walther, Jeremy Leitz, Souhrid Mukherjee, Joseph C Wu, Rabindra V Shivnaraine, and James Zou. Admet-ai: a machine learning admet platform for evaluation of large-scale chemical libraries. *Bioinformatics*, 40(7):btae416, 2024.

R Swanson. Long live peptides—evolution of peptide half-life extension technologies and emerging hybrid approaches. *Drug Discovery World*, 15:57–61, 2014.

Sophia Tang, Yinuo Zhang, and Pranam Chatterjee. Gumbel-softmax flow matching with straight-through guidance for controllable biological sequence generation. *arXiv preprint arXiv:2503.17361*, 2025a.

Sophia Tang, Yinuo Zhang, and Pranam Chatterjee. Peptune: De novo generation of therapeutic peptides with multi-objective-guided discrete diffusion. *Proceedings of the 41st International Conference on Machine Learning (ICML)*, 2025b.

Masahiro Tominaga, Yoko Shima, Kenta Nozaki, Yoichiro Ito, Masataka Someda, Yuji Shoya, Noritaka Hashii, Chihiro Obata, Miho Matsumoto-Kitano, Kohei Suematsu, et al. Designing strong inducible synthetic promoters in yeasts. *Nature Communications*, 15(1):10653, 2024.

Oleg Trott and Arthur J Olson. Autodock vina: improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading. *Journal of computational chemistry*, 31(2):455–461, 2010.

Kotaro Tsuboyama, Justas Dauparas, Jonathan Chen, Elodie Laine, Yasser Mohseni Behbahani, Jonathan J Weinstein, Niall M Mangan, Sergey Ovchinnikov, and Gabriel J Rocklin. Mega-scale experimental analysis of protein folding stability in biology and design. *Nature*, 620(7973): 434–444, 2023.

Ben Tu, Nikolas Kantas, Robert M Lee, and Behrang Shafei. Multi-objective optimisation via the r2 utilities. *arXiv preprint arXiv:2305.11774*, 2023.

Tsuyoshi Ueno, Trevor David Rhone, Zhufeng Hou, Teruyasu Mizoguchi, and Koji Tsuda. Combo: An efficient bayesian optimization library for materials science. *Materials discovery*, 4:18–21, 2016.

Xinyou Wang, Zaixiang Zheng, Fei Ye, Dongyu Xue, Shujian Huang, and Quanquan Gu. Diffusion language models are versatile protein learners. *arXiv preprint arXiv:2402.18567*, 2024.

Jing Yang, S Mark Roe, Matthew J Cliff, Mark A Williams, John E Ladbury, Patricia T W Cohen, and David Barford. Molecular basis for tpr domain-mediated regulation of protein phosphatase 5. *The EMBO Journal*, 24(1):1–10, December 2004. ISSN 1460-2075. doi: 10.1038/sj.emboj.7600496. URL http://dx.doi.org/10.1038/sj.emboj.7600496.

Yinghua Yao, Yuangang Pan, Jing Li, Ivor Tsang, and Xin Yao. Proud: Pareto-guided diffusion model for multi-objective generation. *Machine Learning*, 113(9):6511–6538, 2024.

Jaehoon Yoo, Wonjung Kim, and Seunghoon Hong. Redi: Rectified discrete flow. *arXiv preprint arXiv:2507.15897*, 2025.

Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. *Advances in neural information processing systems*, 33: 5824–5836, 2020.

Ye Yuan, Can Chen, Christopher Pal, and Xue Liu. Paretoflow: Guided flows in multi-objective optimization. *arXiv preprint arXiv:2412.03718*, 2024.

Giacomo Zanella. Informed proposals for local MCMC in discrete spaces. *arXiv preprint arXiv:1711.07424*, 2017. URL https://arxiv.org/abs/1711.07424.

Chengxin Zhang, Xi Zhang, Peter L Freddolino, and Yang Zhang. Biolip2: an updated structure database for biologically relevant ligand–protein interactions. *Nucleic Acids Research*, 52(D1): D404–D412, 2024.

Mingjie Zhang, Hidehito Tochio, Qiang Zhang, Pravat Mandal, and Ming Li. Solution structure of the extended neuronal nitric oxide synthase pdz domain complexed with an associated peptide. *Nature Structural Biology*, 6(5):417–421, May 1999. ISSN 1072-8368. doi: 10.1038/8216. URL http://dx.doi.org/10.1038/8216.

Qingfu Zhang and Hui Li. Moea/d: A multiobjective evolutionary algorithm based on decomposition. *IEEE Transactions on evolutionary computation*, 11(6):712–731, 2007.

Ruochi Zhang, Haoran Wu, Yuting Xiu, Kewei Li, Ningning Chen, Yu Wang, Yan Wang, Xin Gao, and Fengfeng Zhou. Pepland: a large-scale pre-trained peptide representation model for a comprehensive landscape of both canonical and non-canonical amino acids. *arXiv preprint arXiv:2311.04419*, 2023.

Ruimin Zhou, Zhaoyan Jiang, Chen Yang, Jianwei Yu, Jirui Feng, Muhammad Abdullah Adil, Dan Deng, Wenjun Zou, Jianqi Zhang, Kun Lu, et al. All-small-molecule organic solar cells with over 14% efficiency by optimizing hierarchical morphologies. *Nature communications*, 10(1):5393, 2019.

Eckart Zitzler and Lothar Thiele. Multiobjective optimization using evolutionary algorithms—a comparative case study. In *International conference on parallel problem solving from nature*, pp. 292–301. Springer, 1998.

Eckart Zitzler, Marco Laumanns, and Lothar Thiele. Spea2: Improving the strength pareto evolutionary algorithm. *TIK report*, 103, 2001.

A THEORETICAL GUARANTEES

We analyze practical refinement and an associated exact MH kernel separately. The search support $\mathcal{S}_0$ is finite, and all guidance scores h_n are fixed and deterministic. For the exact formulation, let $p_1(x) > 0$ be a base probability mass on $\mathcal{S}_0$. At fixed guidance strength $\eta \geq 0$ and positive preference vector ω, define

$$\pi_{\eta,\omega}(x) = \frac{p_1(x)e^{\eta S_\omega(x)}}{Z_{\eta,\omega}}, \qquad Z_{\eta,\omega} = \sum_{z\in\mathcal{S}_0} p_1(z)e^{\eta S_\omega(z)}. \tag{5}$$

For states x, x' differing only at coordinate i, the full MH acceptance probability is

$$\alpha_{k,i}(x, x') = \min\left\{1, \frac{p_1(x')e^{\eta S_\omega(x')}q_{k,i}(x^i \mid x')}{p_1(x)e^{\eta S_\omega(x)}q_{k,i}(x'^i \mid x)}\right\}. \tag{6}$$

The forward and reverse proposals use the same model time and guidance strength. A reverse token outside its candidate set has probability zero. This distributional formulation is separate from the practical gate and the ungated experimental ablation.

A.1 FIXED-TARGET INVARIANCE

Proposition A.1 (Detailed balance)**.** *Fix η and ω. For each selected coordinate, use the normalized proposal in Eq. 3 and the full acceptance ratio in Eq. 6. Reject a proposed move if its reverse proposal probability is zero. Then the MH kernel leaves $\pi_{\eta,\omega}$ invariant. If the proposal kernel is held fixed and the resulting MH kernel is irreducible and aperiodic, repeated updates converge to that distribution.*

Proof. For distinct states x, x' differing at coordinate i, let $q_i(x' \mid x)$ denote the token proposal probability. Uniform coordinate selection has the same probability in both directions. The off-diagonal probability flux is

$$\pi(x)P(x, x') = \frac{1}{L}\min\{\pi(x)q_i(x' \mid x), \pi(x')q_i(x \mid x')\},$$

which is symmetric in x, x'. This proves detailed balance, and summing over x proves invariance. The convergence conclusion follows from the finite-state irreducible, aperiodic Markov-chain theorem. A positive-probability self-loop in an irreducible chain ensures aperiodicity; connectivity must be checked for the actual feasible candidate sets. □

The identity $g(u) = ug(1/u)$ motivates locally balanced proposals, but does not remove the generator probabilities or the forward and reverse normalization constants from the acceptance ratio. Neither Barker nor square-root balancing makes acceptance identically one in general. Candidate truncation and validity filtering are part of the proposal definition and must also be included when evaluating reverse probabilities. Exact MH additionally requires an evaluable base-density ratio; token probabilities alone do not determine that ratio.

A.2 CONCENTRATION AND PARETO REPRESENTABILITY

Let $M_\omega = \arg\max_{x\in\mathcal{S}_0} S_\omega(x)$ and $S^* = \max_x S_\omega(x)$. Write $p_1(A) = \sum_{x\in A} p_1(x)$.

Proposition A.2 (High-guidance concentration)**.** *If $M_\omega \neq \mathcal{S}_0$, define $\Delta = S^* - \max_{x\notin M_\omega} S_\omega(x) > 0$. Then*

$$\pi_{\eta,\omega}(\mathcal{S}_0 \setminus M_\omega) \leq \frac{p_1(\mathcal{S}_0 \setminus M_\omega)}{p_1(M_\omega)}e^{-\eta\Delta}.$$

In particular, $\pi_{\eta,\omega}(M_\omega) \to 1$. Within M_ω, limiting state probabilities are proportional to p_1.

Proof. Bound each numerator term outside M_ω by $p_1(x)e^{\eta(S^*-\Delta)}$ and bound the denominator below by $p_1(M_\omega)e^{\eta S^*}$. For states in M_ω, their relative masses equal their base-mass ratios for every η. The case $M_\omega = \mathcal{S}_0$ is immediate. □

A weakly Pareto-optimal state has no feasible competitor strictly better in every objective. A Pareto-optimal state has no competitor weakly better in all objectives and strictly better in at least one.

Proposition A.3 (Efficiency and representability)**.** *For positive ω, every global maximizer of S_ω is weakly Pareto optimal. A unique maximizing objective vector is Pareto optimal. Conversely, any Pareto-optimal objective vector $v = (h_1(x^*), \ldots, h_N(x^*))$ with all $v_n > 0$ is the unique maximizing objective vector for weights*

$$\omega_n = \frac{1/v_n}{\sum_j 1/v_j}.$$

Multiple states with vector v may share this maximum.

Proof. If a competitor strictly improves every objective, positive weights make its minimum strictly larger, contradicting maximality. If a distinct vector weakly dominates a maximizing vector, its minimum is at least as large, contradicting uniqueness. For the converse, all weighted components of v equal $c = (\sum_j 1/v_j)^{-1}$. A vector with weighted minimum at least c must be componentwise at least v. Pareto optimality requires equality of every component, proving uniqueness at the objective-vector level. □

Because the feasible space is finite, the strict gap between v and other objective vectors persists in a neighborhood of these weights. A preference distribution with full support on the interior of the simplex assigns that neighborhood positive probability. This gives representability and positive probability of selecting each supported, strictly positive Pareto objective vector under ideal optimization or equilibrium concentration. It does not guarantee coverage within a finite number of trajectories.

The positivity and uniqueness qualifications matter. For vectors $(0.5, 0.5)$ and $(0.5, 0.8)$, the first can tie for the maximum despite being dominated. For vectors $(0, 1)$ and $(0.5, 0.5)$, the first is Pareto optimal but cannot maximize the unshifted minimum with strictly positive weights. Thus the unshifted scalarization does not provide an unconditional full-front guarantee. Annealing also changes the target: fixed-target invariance alone does not establish that a finite, time-inhomogeneous trajectory tracks equilibrium.

A.3 Weighted-sum monotone refinement

Proposition A.4 (Nondecreasing utility and finite strict progress)**.** *Under Eq. 4, $U_\omega(x_{k+1}) \geq U_\omega(x_k)$ at every iteration. Let $D = |\{U_\omega(x) : x \in \mathcal{S}_0\}|$ be the number of distinct attainable utility values. At most $D - 1 \leq |\mathcal{S}_0| - 1$ updates strictly increase utility. Equal-utility state changes may continue indefinitely.*

Proof. An accepted proposal has utility at least that of the current state, and rejection retains the current utility. Every strictly improving update reaches a larger member of a finite ordered set with D elements. There can therefore be at most $D - 1$ such updates. The acceptance test permits transitions among states with equal utility, so it does not bound their number. □

Finite strict progress does not imply finite termination at a local maximizer. For example, two adjacent states with equal utility may alternate indefinitely. The result gives no global optimum or invariance for Eq. 5. A global maximizer of a positive weighted sum is Pareto optimal: a dominating state would have strictly larger weighted utility. The practical finite-budget algorithm need not reach such a maximizer, and an accepted update need not increase S_ω.

The proposal assumes a nonempty candidate set with positive total weight. The inspected implementation has no explicit empty-set fallback. SMILES guidance includes an analyzer score, as defined in Appendix F; the acceptance rule does not impose a separate hard peptide-validity constraint. The ungated ablation uses the same proposals and unconditionally assigns $x \leftarrow x'$.

Algorithm 1 AReUReDi practical refinement

Input: initial sequence $x \in \mathcal{S}_0$, generator probabilities $b_{k,i}$, fixed scores h_n, nonnegative weights ω with positive sum, budget $T > 1$, strengths $\eta_{\min}, \eta_{\max}$.
for $k = 0, \ldots, T-1$ **do**
  Set η_k by Eq. 2; choose an editable position i uniformly.
  Form $\mathcal{C}_i(x)$ using task-specific token filters (Appendix F).
  Evaluate $h_n(x^{(i \leftarrow y)})$ for every $y \in \mathcal{C}_i(x)$ and score index n.
  Compute the Barker-balanced distribution $q_{k,i}(\cdot \mid x)$ by Eq. 3.
  Draw $y \sim q_{k,i}(\cdot \mid x)$ and set $x' = x^{(i \leftarrow y)}$.
  if $U_\omega(x') \geq U_\omega(x)$ **then**
    Set $x \leftarrow x'$.
  end if
end for
Return: final candidate x.

B Rectification Improves Peptide Generator Validation and SMILES Validity

We developed **PepReDi**, an unconditional peptide generator based on ReDi, to obtain token proposals for peptide refinement. The model backbone of PepReDi is a Diffusion Transformer (DiT) architecture (Peebles & Xie, 2022). We trained the base generator on a dataset comprising approximately 15,000 peptides from the PepNN and BioLip2 datasets, as well as sequences from the PPIRef dataset, with lengths ranging from 6 to 49 amino acids (Abdin et al., 2022; Zhang et al., 2024; Bushuiev et al., 2023). Using this trained model, we generated new data couplings containing 10,000 sequences for each peptide length and used them to fine-tune PepReDi in an iterative rectification procedure. Three rounds of rectification reduced validation NLL from 1.6458 to 1.3548 and PPL from 5.19 to 3.88 (Table S3). Estimated conditional TC increased after the first round, then declined over rounds two and three; the final estimate remained above the unrectified value. These empirical estimates do not establish monotonic TC reduction across all trained models.

SMILESReDi uses the PepReDi backbone with Rotary Positional Embeddings (RoPE) to encode relative token positions in peptide SMILES (Su et al., 2024). We incorporated a time-dependent noising schedule for peptide-bond tokens (Appendix C.2). Training used the same dataset as the peptide SMILES diffusion model PepMDLM (Tang et al., 2025b). After two training epochs, SMILESReDi reached a validation NLL of 0.722 and a sampling validity of 76.3% with 16 generation steps. We generated 100 SMILES sequences for each length from 4 to 1035 to construct the next training coupling. The rectified checkpoint used for the original SMILES experiments had validation NLL 0.6084, and the sampling validity increased to 98.6% with 16 steps and 100% with 32 steps (Table S4). SMILESReDi[1] achieved higher validity than PepMDLM at the reported sampling configuration, with diversity and SNN reported separately in Table S4.

C Base Model Details

C.1 PepReDi

Model Architecture. PepReDi uses a Diffusion Transformer (DiT) backbone within a Masked Diffusion Language Model (MDLM) architecture (Peebles & Xie, 2022; Sahoo et al., 2024). Input amino acid sequences are transformed to discrete tokens using the ESM-2-650M tokenizer (Lin et al., 2023). Tokenized amino acid sequences and time-steps are converted to continuous embedding vectors using two separate layers, which are then fused and processed by stacked DiT transformer blocks equipped with multi-head self-attention to capture long-range dependencies in the amino-acid sequence. Residual connections and layer normalization stabilize the training dynamics, and a final projection layer outputs token logits for each position.

Dataset Curation. We curated the PepReDi training data from PepNN, BioLip2, and PPIRef (Abdin et al., 2022; Zhang et al., 2024; Bushuiev et al., 2023). The data included all peptides from PepNN and

BioLip2 and PPIRef sequences of 6–49 amino acids. We divided the dataset into training, validation, and test sets at an 80/10/10 ratio.

Training Strategy. We trained the model for 100 epochs on a GPU node using the SLURM workload manager. Adam used a learning rate of 1e-4 and weight decay of 1e-5. A learning rate scheduler with 10 warm-up epochs and cosine decay was used, with initial and minimum learning rates both 1e-5. The network architecture included a model dimension of 512, 6 transformer layers, and 8 attention heads, with a vocabulary size of 24 and a maximum sequence length of 100 tokens. We estimated conditional total correlation using 20 batches and 50 samples per batch, and selected the checkpoint with the lowest estimate. Rectification training started from the pretrained checkpoint with the same hyperparameters and weight decay increased to 2e-5.

Dynamic Batching. We used dynamic batching for variable-length sequences. Following the ESM-2 batching approach (Lin et al., 2023), we sorted peptides by length and used a maximum token budget of 100 per GPU.

Rectification. We generated 10,000 sequences for each peptide length from 6 to 49 using 16 sampling steps and temperature 1. The generated sequences were grouped by dynamic batching for rectification training.

C.2 SMILESREDI

Model Architecture. SMILESReDi uses a Diffusion Transformer (DiT) backbone within a Masked Diffusion Language Model (MDLM) architecture to generate molecular SMILES sequences (Peebles & Xie, 2022; Sahoo et al., 2024). Input SMILES sequences are transformed to discrete tokens using the PeptideCLM-23M tokenizer. Tokenized SMILES sequences and time-steps are converted to continuous embedding vectors using two separate layers. Both embeddings are then fused and processed by stacked DiT transformer blocks that incorporate Rotary Positional Embeddings (RoPE) and multi-head attention modules to capture long-range structural dependencies while preserving positional information (Su et al., 2024). A final layer normalization and linear projection produce token logits for each position.

Time-dependent bond-aware noising schedule. Peptide SMILES share a conserved backbone of alternating carbonyl and amide groups connected by chemically constrained peptide bonds, while their side chains remain highly diverse. We used a bond-aware interpolation schedule with a different transition rate for backbone and side-chain tokens. Specifically, for each position j with a bond indicator $b_j \in \{0, 1\}$ and δ_z denoting point mass at token z, the time-t marginal of the probability path is

$$p_t(x_t^{(j)} \mid x_0^{(j)}, x_1^{(j)}) = \left[\, b_j t^\gamma + (1 - b_j)t \,\right] \delta_{x_1^{(j)}} + \left[\, 1 - b_j t^\gamma - (1 - b_j)t \,\right] \delta_{x_0^{(j)}}, \quad t \in [0, 1],\ \gamma > 1,$$

so each token is equal to $x_1^{(j)}$ with the indicated mixture coefficient and to $x_0^{(j)}$ otherwise, ensuring that backbone tokens ($b_j = 1$) transition more slowly than non-bond tokens along the DFM probability path.

Training Strategy. We trained the model on a four-GPU NVIDIA system with 48 GB of VRAM per device for 5 epochs. We selected the checkpoint with the lowest evaluation loss. Optimization used Adam with a learning rate of 1e-4. A learning rate scheduler with warm-up over 10% of total training steps and cosine decay was used, with initial and minimum learning rates both 1e-5. The network architecture included a model dimension of 768, 8 transformer layers, and 8 attention heads. Gradient clip value was set to 1.0 and γ to 2.0 in the time-dependent bond-aware noising schedule. Rectification training started from the pretrained checkpoint, used the same hyperparameters, and ran for 10 epochs.

Rectification. We generated 100 sequences for each length from 4 to 1035 using 100 sampling steps and temperature 1. The generated sequences were grouped by dynamic batching for rectification training.

Evaluation metrics. SMILES validity in Table S4 is the fraction of all attempted strings that RDKit parses successfully. Peptide validity, evaluated separately in Table S23, additionally assesses the peptide structure. Uniqueness is the fraction of valid strings with distinct canonical SMILES. Diversity and similarity to nearest neighbor (SNN) use unique valid molecules and binary Morgan

fingerprints with radius 3 and 2048 bits. For bit vectors f and g, define

$$\tau(f, g) = \frac{f^\top g}{\|f\|_1 + \|g\|_1 - f^\top g}.$$

Let G be the set of unique valid generated molecules, $m = |G| \geq 2$, and R the nonempty set of valid molecules in the full reference dataset. With $f(x)$ denoting a molecule's fingerprint,

$$\text{Diversity}(G) = 1 - \frac{2}{m(m-1)} \sum_{\substack{x, x' \in G \\ x < x'}} \tau(f(x), f(x')),$$

$$\text{SNN}(G, R) = \frac{1}{m} \sum_{x \in G} \max_{r \in R} \tau(f(x), f(r)).$$

The ordering $x < x'$ enumerates unordered distinct pairs once. The reference was the full peptide-SMILES dataset, stored as `30K_all.csv`, after excluding invalid reference strings. A larger diversity indicates less similar generated structures; a smaller SNN indicates less similarity to the reference set.

C.3 PepDFM

Model Architecture. The base model is a time-dependent architecture based on U-Net (Ronneberger et al., 2015). It uses two separate embedding layers for sequence and time, followed by five convolutional blocks with varying dilation rates to capture temporal dependencies, while incorporating time-conditioning through dense layers. The final output layer generates logits for each token. We used a polynomial convex schedule with a polynomial exponent of 2.0 for the mixture discrete probability path in the discrete flow matching.

Dataset Curation. We curated the PepDFM training data from PepNN, BioLip2, and PPIRef (Abdin et al., 2022; Zhang et al., 2024; Bushuiev et al., 2023). The data included all peptides from PepNN and BioLip2 and PPIRef sequences of 6–49 amino acids. We divided the dataset into training, validation, and test sets at an 80/10/10 ratio.

Training Strategy. We trained the model on a 2xH100 NVIDIA NVL GPU system with 94 GB of VRAM for 200 epochs with batch size 512. We selected the checkpoint with the lowest evaluation loss. Optimization used Adam with a learning rate of 1e-4. A learning rate scheduler with 20 warm-up epochs and cosine decay was used, with initial and minimum learning rates both 1e-5. The embedding dimension and hidden dimension were set to be 512 and 256 respectively for the base model.

Performance. PepDFM achieved a validation loss of 3.1051. This evaluation used the generalized KL objective in Gat et al. (2024).

D Objective Description

We considered five predicted properties: non-hemolysis, non-fouling, solubility, half-life, and binding affinity. Non-hemolysis denotes reduced red-blood-cell lysis risk (Pirtskhalava et al., 2013). Non-fouling denotes resistance to nonspecific biomolecular interactions (Chen et al., 2009). Solubility affects formulation and delivery (Fosgerau & Hoffmann, 2015), and half-life characterizes persistence in circulation (Swanson, 2014). Binding affinity measures interaction strength with the target. These are complementary design criteria; their predicted values do not establish efficacy or safety.

E Score Model Details

For chemically modified peptides, we used the score models of Tang et al. (2025b). For the original amino-acid study, we trained property predictors using hemolysis (9,316), non-fouling (17,185), solubility (18,453), and binding affinity (1,781) data from PepLand and PeptideBERT (Zhang et al., 2023; Guntuboina et al., 2023). These sequences contained standard L-amino acids, and we used ESM-2 to tokenize and represent them (Lin et al., 2023).

The original predictor development and the later PeptiVerse evaluation used different model families. The historical sampling code used the XGBoost non-hemolysis and non-fouling models, a hydrophobic-residue solubility heuristic, the log-half-life predictor, and the affinity predictor. The trained solubility classifier described below was not the solubility function called by that historical sampler. Later preference and refit experiments used PeptiVerse predictors with a CNN solubility head, pooled cross-attention affinity model, and log1p half-life model. Their validation measurements and score transformations are reported separately in Appendices H.3 and F.

E.1 Boosted Trees for Classification

For hemolysis, non-fouling, and solubility classification, we trained XGBoost boosted tree models for logistic regression. We used a stratified 0.8/0.2 train/validation split from scikit-learn (Pedregosa et al., 2011) and generated mean-pooled ESM-2-650M (Lin et al., 2023) embeddings as input features to the model. Hyperparameter selection used 50 trials of OPTUNA (Akiba et al., 2019) search over XGBoost settings (Table S1), with validation F1 as the selection criterion. The selected models reached validation F1 scores of 0.58, 0.71, and 0.68, respectively.

Table S1: XGBoost Hyperparameters for Classification

Hyperparameter	**Value/Range**
Objective	`binary:logistic`
Lambda	$[1e{-}8, 10.0]$
Alpha	$[1e{-}8, 10.0]$
Colsample by Tree	$[0.1, 1.0]$
Subsample	$[0.1, 1.0]$
Learning Rate	$[0.01, 0.3]$
Max Depth	$[2, 30]$
Min Child Weight	$[1, 20]$
Tree Method	`hist`

E.2 Binding Affinity Score Model

We developed an unpooled reciprocal attention transformer model to predict protein-peptide binding affinity, using latent representations from the ESM-2 650M protein language model (Lin et al., 2023). The model retains unpooled token-level embeddings from ESM-2, which are passed through convolutional layers followed by cross-attention layers. The binding affinity data were split into a 0.8/0.2 ratio, maintaining similar affinity score distributions across splits. We used OPTUNA (Akiba et al., 2019) for hyperparameter optimization, selecting hyperparameters by validation correlation. The final model was trained for 50 epochs with a learning rate of 3.84e-5, a dropout rate of 0.15, 3 initial CNN kernel layers (dimension 384), 4 cross-attention layers (dimension 2048), and a shared prediction head (dimension 1024). The regressor achieved a validation Spearman correlation of 0.64.

E.3 Half-Life Score Model

Dataset Curation. We curated the half-life dataset from three public sources: PEPLife, PepTherDia, and THPdb2 (Mathur et al., 2016; D'Aloisio et al., 2021; Jain et al., 2024). We selected human half-life records and excluded entries with missing values. After deduplication, the dataset contained 105 entries.

Pre-training on stability data. To initialize the predictor before fine-tuning on 105 half-life records, we pretrained a score model on a larger peptide stability dataset (Tsuboyama et al., 2023). The model consists of three linear layers with ReLU activation functions, and a dropout rate of 0.3 was applied. The model was trained on a 2xH100 NVIDIA NVL GPU system with 94 GB of VRAM for 50 epochs. The Adam optimizer was employed with a learning rate of 1e-2. A learning rate scheduler with 5 warm-up epochs and cosine decay was used, with initial and minimum learning rates both 1e-3. After training, the model achieved a validation Spearman's correlation of 0.7915 and an R^2 value of 0.6864 on the validation split.

Fine-tuning on half-life data. We fine-tuned the pretrained stability predictor on the half-life dataset. Since half-life values span a wide range, the model was adapted to predict the base-10 logarithm of the half-life (h) values to stabilize the learning process. After fine-tuning, the model achieved a validation Spearman's correlation of 0.8581 and an R^2 value of 0.5977.

F Sampling Details

Acceptance and candidate sets. Main experiments used the non-strict gate $U_\omega(x') \geq U_\omega(x)$ with the same coefficients in S_ω and U_ω. Barker balancing was used in all inspected refinement scripts. We sampled positions uniformly among editable indices, excluded the current token and task-specific special tokens, and formed nucleus candidate sets from the remaining model probabilities. Amino-acid scripts used a nucleus threshold of 0.9 by default, and the five-objective affinity sweep used 1.0. Every retained candidate was scored. No explicit empty-candidate fallback was implemented.

Original amino-acid scores. Non-hemolysis was $1 - p_{\mathrm{hem}}$, where p_{hem} is the classifier's hemolysis probability, and non-fouling was its favorable class probability. Historical solubility guidance was one minus the hydrophobic-residue fraction. For predicted log-half-life $\hat{\ell}$ and affinity $\hat{a}$, the guidance scores were

$$h_{\mathrm{HL}}(x) = \frac{\mathrm{clip}(\hat{\ell}(x), 0, 2)}{2}, \qquad h_{\mathrm{aff}}(x) = \frac{\hat{a}(x)}{10}.$$

Here $\mathrm{clip}(v, l, u) = \min(u, \max(l, v))$, and $\hat{\ell}$ predicts $\log_{10}$ half-life in hours. Historical AReUReDi half-life displays used $10^{2h_{\mathrm{HL}}}$ hours and affinity displays used $10h_{\mathrm{aff}}$. We retained that original display convention for the corresponding results.

PeptiVerse scores. The later affinity-preference, preference-strategy, and refit experiments used the PeptiVerse reference models. Non-hemolysis was one minus the hemolysis probability; non-fouling and solubility were favorable model scores. Half-life guidance was $h_{\mathrm{HL}} = \mathrm{clip}(\hat{H}, 0, 10)/10$, where $\hat{H} = \mathrm{expm1}(\hat{\ell}_{\mathrm{PV}})$ is the model's hours prediction and $\hat{\ell}_{\mathrm{PV}}$ is its log1p output. Affinity used $\hat{a}/10$. These normalized values were used directly in the later evaluations. The historical $10^{2h_{\mathrm{HL}}}$ display transformation is not an inverse of the PeptiVerse normalization.

Amino-acid refinement. We initialized amino-acid tokens uniformly and used $T = 20L$ updates for a length-L sequence. Default five-property weights were $\omega_n = 1/5$, with guidance strengths increasing from 1 to 20. For the affinity sweep, $\omega_{\mathrm{aff}} \in \{0.2, 0.5, 0.8\}$ and every other coefficient was $(1 - \omega_{\mathrm{aff}})/4$. The three-property tables retain the reported coefficients, including the equal-coefficient row $(0.3, 0.3, 0.3)$. A common coefficient rescaling leaves weighted-sum acceptance unchanged but changes the effective Tchebycheff guidance strength.

Chemically modified peptides. We initialized SMILES with SMILESReDi[1] using 16 generation steps, then applied 128 refinement updates. We retained at most 200 tokens, ordered by generator probability, after nucleus and forbidden-token filtering from the 586-token vocabulary. The four property coefficients were 0.7 for affinity and 0.1 each for non-hemolysis, non-fouling, and solubility. Guidance also included an analyzer score $h_{\mathrm{val}} \in \{0, 1\}$ with coefficient 1, giving $N = 5$ scores in Eqs. 1 and 4. The analyzer tested RDKit parsing and peptide-bond patterns, including N-methyl bonds. This was an additional scored objective; the code did not impose a separate hard rejection of invalid peptide SMILES. Affinity guidance was $\min(\hat{a}/10, 1)$, and the other property scores were favorable classifier outputs.

Checkpoint provenance. The original SMILES generation results used rectified checkpoint `v1_2`, with validation NLL 0.6084. The length stress test used checkpoint `v1`, with validation NLL 0.6225. The base model was trained for five epochs, and its minimum-NLL checkpoint was selected after epoch two (NLL 0.7221).

The core runtime comparisons used method-specific generation budgets. The additional matched-evaluation tests are described in Appendix H.

Table S2: **Five-objective peptide design on 1B8Q and PPP5.** Means over 100 candidates per method. NH: non-hemolysis; NF: non-fouling; Sol.: solubility; HL: predicted half-life (h); Aff.: affinity score. All property scores favor larger values. Time is seconds per candidate (smaller is faster). Sampling configurations are detailed in Appendix F. Bold marks the largest property mean within each target.

Target	Method	Time	NH	NF	Sol.	HL	Aff.
1B8Q	MOPSO	8.54	0.8934	0.4763	0.4684	4.45	6.0594
	NSGA-III	33.13	0.9138	0.5715	0.5825	7.32	7.2178
	SMS-EMOA	8.21	0.8804	0.3450	0.3511	3.02	5.9550
	SPEA2	17.48	0.9181	0.4973	0.5057	4.13	**7.3240**
	PepTune+DPLM	2.46	0.8547	0.3085	0.3213	1.17	5.2398
	AReUReDi	55	**0.9214**	**0.8680**	**0.8654**	**42.63**	5.7130
PPP5	MOPSO	11.34	0.9117	0.4711	0.4255	1.77	6.6958
	NSGA-III	37.30	**0.9521**	0.7138	0.7066	2.90	7.3789
	SMS-EMOA	8.43	0.8758	0.4269	0.4334	1.03	6.2854
	SPEA2	19.02	0.9445	0.6221	0.6098	2.61	**7.6253**
	PepTune+DPLM	4.80	0.8816	0.2752	0.2636	1.27	5.8454
	AReUReDi	195	0.9412	**0.8960**	**0.8832**	**38.28**	6.7186

Table S3: Training and validation performance of PepReDi over successive rectification rounds. Each row reports the training loss, validation negative log-likelihood (NLL), validation perplexity (PPL), and conditional total correlation (TC). PepReDi without superscript denotes the base model, while PepReDi^1, PepReDi^2, PepReDi^3 indicate the first, second, and third rounds of rectification, respectively.

	Train Loss	**Val NLL**	**Val PPL**	**Conditional TC**
PepReDi	1.6567	1.6458	5.19	**10.6027**
PepReDi^1	1.6170	1.6101	5.00	12.6250
PepReDi^2	1.5347	1.5238	4.59	11.7279
PepReDi^3	**1.3538**	**1.3548**	**3.88**	11.2339

G Extended Results and Ablations

Full objective profiles and ablation measurements are reported here. NH, NF, Sol., HL, and Aff. abbreviate non-hemolysis, non-fouling, solubility, predicted half-life, and predicted affinity. Higher values are favorable for these property columns. Absolute affinity scales differ between the amino-acid and SMILES predictors. Property means and balance metrics answer different questions and are reported separately. Table S2 reports the full five-property comparison and runtime; Tables S3 and S4 report generator quality. Target–length summaries appear in Table S5. Controlled comparisons assess the acceptance gate (Table S6), objective removal (Tables S7 and S8), generator choice (Table S9), and annealing (Table S10).

Increasing the update budget improved non-fouling, solubility, and half-life on MYC and all four properties on NCAM1, with longer runtimes (Table S11). Learned and uniform proposals were compared in Tables S12 and S13. Three-property weight changes raised the corresponding favored score for CLK1 and GFAP (Tables S14 and S15). These measurements connected the available computation and property priorities to the returned candidate profiles.

Table S4: Evaluation metrics for the generative quality of peptide SMILES sequences of max token length set to 200. SMILESReDi denotes the base model; SMILESReDi[1] denotes the rectified model. The first three model rows report unconditional generation, and the last two report guided outputs. Bold marks the larger diversity and smaller SNN in the guided comparison. Validity is RDKit parse validity.

Model	Validity (↑)	Uniqueness (↑)	Diversity (↑)	SNN (↓)
Data	1.000	1.000	0.885	1.000
PepMDLM	0.450	1.000	0.705	0.513
SMILESReDi	0.763	1.000	0.719	0.593
SMILESReDi[1]	0.986	1.000	0.665	0.579
PepTune	1.000	1.000	0.677	0.486
AReUReDi	1.000	1.000	**0.789**	**0.392**

Table S5: AReUReDi generates amino-acid peptide candidates for 8 diverse protein targets, optimizing five predicted binding and developability properties. Values are the reported means for runs requesting 100 candidates per setting.

Name	Binder Length	NH	NF	Sol.	HL (h)	Aff.
AMHR2	8	0.9156	0.8613	0.8564	45.73	7.0608
AMHR2	12	0.9384	0.8872	0.8810	52.52	7.2284
AMHR2	16	0.9420	0.8914	0.8755	63.34	7.2533
EWS::FLI1	8	0.9186	0.8630	0.8619	44.77	5.8424
EWS::FLI1	12	0.9345	0.8819	0.8796	59.11	6.2007
EWS::FLI1	16	0.9416	0.8875	0.8807	64.32	6.4195
MYC	8	0.9180	0.8627	0.8627	44.13	6.4082
OX1R	10	0.9302	0.8687	0.8563	50.14	7.1882
DUSP12	9	0.9240	0.8669	0.8633	48.14	6.1276
1B8Q	8	0.9214	0.8680	0.8654	42.63	5.7130
5AZ8	11	0.9293	0.8732	0.8605	58.33	6.2792
7JVS	11	0.9313	0.8840	0.8743	56.49	6.8449

Table S6: **Acceptance-rule comparison.** Ungated guided proposals and weighted-sum monotone acceptance used the same number of refinement steps on 8CN1 and 4EBP2. Each value is the mean over 100 candidates. Bold marks the larger property mean.

Target	Method	NH	NF	Sol.	HL (h)	Aff.
8CN1	Ungated	0.8650	0.4782	0.4627	2.54	5.2412
	Monotone	**0.9213**	**0.8676**	**0.8697**	**44.70**	**5.5143**
4EBP2	Ungated	0.8879	0.4288	0.4257	1.8781	5.7132
	Monotone	**0.9356**	**0.8767**	**0.8692**	**53.95**	**6.4571**

Table S7: Ablation results for amino-acid peptide candidate design targeting PDB 7LUL with different guidance settings. For each setting, 100 candidates of length 7 were designed.

Guidance Settings			NH	Sol.	Aff.
Hemolysis	Solubility	Affinity			
✓	✓	✓	0.9389	0.9398	6.2559
×	✓	✓	0.8964	0.9465	6.3272
✓	×	✓	0.9502	0.4013	6.9798
✓	✓	×	0.9535	0.9642	5.2611
×	×	✓	0.8812	0.2877	7.5057
×	✓	×	0.9036	0.9725	5.2449
✓	×	×	0.9802	0.6135	5.0985
×	×	×	0.8431	0.5810	4.8919

Table S8: Ablation results for amino-acid peptide candidate design targeting PDB CLK1 with different guidance settings. For each setting, 100 candidates of length 12 were designed.

Guidance Settings			Non-Fouling	Half-Life (h)	Aff.
Non-Fouling	Half-Life (h)	Affinity			
✓	✓	✓	0.8285	74.04	6.8099
×	✓	✓	0.2902	96.59	7.3906
✓	×	✓	0.9365	1.33	7.2029
✓	✓	×	0.9479	75.68	6.3437
×	×	✓	0.9625	1.23	6.2319
×	✓	×	0.3540	100.00	6.4116
✓	×	×	0.2531	2.96	8.6580
×	×	×	0.4988	1.82	5.4739

Table S9: **Rectification of the base generation model improves AReUReDi's performance.** Amino-acid candidates for two protein targets (PDB 5AZ8 and AMHR2) were generated using AReUReDi with three different base models: PepDFM, PepReDi (without rectification), and PepReDi^3 (with three rounds of rectification). The table reports the average score for each objective, calculated from 100 generated candidates per setting. The best score for each objective is highlighted in bold.

Target	Base Model	NH	NF	Sol.	HL (h)	Aff.
5AZ8	PepDFM	0.9296	**0.8867**	**0.8743**	37.30	6.2291
	PepReDi	**0.9326**	0.8759	0.8572	50.16	**6.4391**
	PepReDi^3	0.9293	0.8732	0.8605	**58.33**	6.2792
AMHR2	PepDFM	0.9412	0.8774	0.8612	47.84	7.2373
	PepReDi	0.9127	0.8602	0.8460	50.92	7.0101
	PepReDi^3	**0.9420**	**0.8914**	**0.8755**	**63.34**	**7.2533**

Table S10: **Annealed guidance strength improves AReUReDi's performance.** Amino-acid candidates for two protein targets (PDB 1DDV and P53) were generated under four guidance schedules: (1) fixed at the minimum strength $\eta_{min} = 1.0$, (2) fixed at the maximum strength $\eta_{max} = 20.0$, (3) fixed at the midpoint $\frac{1}{2}(\eta_{min} + \eta_{max}) = 10.5$, and (4) an annealed schedule where η_k increases from η_{min} to η_{max} over optimization steps. The table reports the average score for each objective, calculated from 100 generated candidates per setting. The best score for each objective is highlighted in bold.

Target	Method	NH	NF	Sol.	HL (h)	Aff.
1DDV	$\eta = \eta_{min}$	0.9130	0.8575	0.8429	38.70	5.3554
	$\eta = \eta_{max}$	**0.9156**	0.8512	0.8479	40.27	5.4359
	$\eta = \frac{1}{2}(\eta_{min} + \eta_{max})$	0.9108	**0.8641**	0.8544	40.43	**5.5396**
	$\eta_k = \eta_{\min} + (\eta_{\max} - \eta_{\min}) \frac{k}{T-1}$	0.9128	0.8545	**0.8565**	**44.73**	5.4482
P53	$\eta = \eta_{min}$	0.9335	0.8800	0.8706	49.97	6.2538
	$\eta = \eta_{max}$	0.9293	0.8693	0.8657	61.76	6.3043
	$\eta = \frac{1}{2}(\eta_{min} + \eta_{max})$	0.9294	0.8713	0.8653	59.43	6.3060
	$\eta_k = \eta_{\min} + (\eta_{\max} - \eta_{\min}) \frac{k}{T-1}$	**0.9353**	**0.8818**	**0.8785**	**62.83**	**6.3508**

Table S11: **Property scores and runtime across generation-step budgets.** We generated 100 candidates for MYC (12-mer amino-acid peptides) and NCAM1 (chemically-modified peptides of length 200) using different numbers of generation steps. The table reports mean property scores and time in seconds per candidate. Half-life is not optimized for NCAM1 and is indicated by "*".

Target	# Steps	NH	NF	Sol.	HL (h)	Aff.	Time
MYC	64	0.9279	0.8571	0.8519	5.49	6.5167	67
	128	0.9301	0.8721	0.8627	16.54	6.5811	131
	256	0.9357	0.8820	0.8740	34.83	6.5293	265
NCAM1	64	0.8801	0.2468	0.7954	*	5.3936	112
	128	0.8840	0.2657	0.8109	*	5.4377	198
	256	0.8900	0.3015	0.8202	*	5.5929	423

Table S12: **Learned versus uniform token proposals for amino-acid peptide design.** 100 amino-acid candidates were designed for PDB 1B8Q (8-mer) and PPP5 (16-mer), respectively. The table reports the average score for each objective. The best score for each objective is highlighted in bold.

Target	Prior	NH	NF	Sol.	HL (h)	Aff.
1B8Q	Uniform Prior	0.9009	0.8191	0.8049	14.20	**5.8432**
	PepReDi Prior	**0.9214**	**0.8680**	**0.8654**	**42.63**	5.7130
PPP5	Uniform Prior	0.9265	0.8263	0.7993	17.52	6.7122
	PepReDi Prior	**0.9412**	**0.896**	**0.8832**	**38.28**	**6.7186**

Table S13: **Learned versus uniform token proposals for modified peptide design.** For each setting, 100 chemically-modified candidates of length 200 were designed. The table reports the average score for each objective. The best score for each objective is highlighted in bold.

Target	Prior	NH	NF	Sol.	Aff.
TfR	Uniform Prior	0.8652	0.2381	**0.7777**	5.5535
	SMILESReDi Prior	**0.8665**	**0.3234**	0.7408	**6.1271**
GLP1	Uniform Prior	0.8341	0.2123	**0.7777**	7.5731
	SMILESReDi Prior	**0.8743**	**0.3438**	0.7661	**8.3414**

Table S14: Ablation results for amino-acid peptide candidate design targeting CLK1 with different weight vector settings. For each setting, 100 candidates of length 12 were designed. The table reports the average score for each objective.

Weight Vectors			NF	HL (h)	Aff.
Non-Fouling	**Half-Life**	**Affinity**			
0.3	0.3	0.3	0.8285	74.04	6.8099
0.8	0.1	0.1	0.9367	6.94	6.5231
0.1	0.8	0.1	0.5642	85.47	6.3649
0.1	0.1	0.8	0.6698	48.94	7.4922

Table S15: Ablation results for chemically-modified peptide candidate design targeting GFAP with different weight vector settings. For each setting, 100 candidates of length 200 were designed. The table reports the average score for each objective.

Weight Vectors			NF	Sol.	Aff.
Non-Fouling	**Solubility**	**Affinity**			
0.3	0.3	0.3	0.2754	0.8169	5.3011
0.8	0.1	0.1	0.3322	0.7528	5.3487
0.1	0.8	0.1	0.2273	0.8327	5.3378
0.1	0.1	0.8	0.2498	0.7910	5.8827

H Additional Evaluation

H.1 Candidate balance and empirical coverage

For each candidate with normalized favorable evaluation scores $\tilde{s}_1, \ldots, \tilde{s}_N \in [0,1]$, we computed its range $\max_n \tilde{s}_n - \min_n \tilde{s}_n$, population standard deviation across objectives, and minimum $\min_n \tilde{s}_n$. Tables report means across candidates. Smaller range and dispersion indicate greater within-candidate balance; a larger minimum indicates a stronger worst-scoring objective. These metrics complement objective means and Pareto-set coverage.

Table 1 reports the candidate-balance measurements in the main text. Table S16 retains the corresponding sequence-diversity summaries.

For the original candidate-balance analysis, the evaluation code used the stored favorable normalized columns. AReUReDi's historical half-life score used clipped log-half-life divided by 2, and affinity used the predictor score divided by 10. The PepTune+DPLM loader used its stored half-life power clipped to $[0,1]$ when available; otherwise it used $\min(\log_{10}(H)/2, 1)$ for reported half-life $H > 0$. Its affinity was divided by 10. No fitted global rescaling was applied to the stored AReUReDi columns.

For coverage, let $A \subseteq [0,1]^5$ be a target's generated objective vectors after componentwise clipping and let $z_j \sim \mathrm{Unif}([0,1]^5)$, $j = 1, \ldots, M$, be Monte Carlo points. Hypervolume relative to the origin was estimated as

$$\widehat{\mathrm{HV}}(A) = \frac{1}{M} \sum_{j=1}^{M} \mathbf{1}\{\exists a \in A : \ a_n \geq z_{j,n} \text{ for all } n\}, \qquad M = 200{,}000,$$

using random seed 0. Let R be the nondominated union of evaluated methods' objective vectors for the same target. Empirical IGD^+ for maximization was

$$\mathrm{IGD}^+(A, R) = \frac{1}{|R|} \sum_{r \in R} \min_{a \in A} \left(\sum_{n=1}^{5} \max(r_n - a_n, 0)^2 \right)^{1/2}.$$

Nondomination used numerical tolerance 10^{-12}. Larger HV and smaller IGD^+ indicate better empirical coverage. Equal-weight AReUReDi used 100 outputs; preference sweeps used 108 outputs from 36 preferences and three seeds. The comparison used the scores stored by each experiment's predictor configuration, including the later PeptiVerse normalization for preference sweeps. It therefore measures coverage in the reported score coordinates, with both the generation configuration and predictor configuration contributing to cross-experiment differences.

H.2 Preference control

Increasing the affinity-designated weight from 0.2 to 0.8 increased mean normalized affinity on both targets (Table S18). In the five-objective sweep, Dirichlet(1), boundary-focused Dirichlet(0.2), and deterministic covering each used 36 preferences and 108 outputs per target. The strategies produced distinct coverage–diversity profiles (Table S19). The original three-objective weight experiments appear in Tables S14 and S15; their entries retain the reported weight values, including the reported equal-coefficient row. A large coefficient in a weighted minimum need not have the same preference interpretation as a large coefficient in a weighted sum.

Table S16: Reported sequence diversity for the reference-predictor candidate populations. Larger values indicate greater diversity.

Method	1B8Q	PPP5
AReUReDi	0.881	0.899
MOPSO	0.840	0.903
NSGA-III	0.595	0.664
SMS-EMOA	0.656	0.662
SPEA2	0.681	0.739
PepTune+DPLM	0.841	0.899

Table S17: Empirical coverage under equal weighting, preference sweeps, and selected baselines.

Target	Method	HV	IGD^+
1B8Q	AReUReDi, equal	0.576	0.109
	AReUReDi, sweep	0.616	0.094
	SMS-EMOA	0.725	0.013
	PepTune+DPLM	0.151	0.407
PPP5	AReUReDi, equal	0.695	0.110
	AReUReDi, sweep	0.817	0.026
	SMS-EMOA	0.834	0.013
	PepTune+DPLM	0.158	0.474

Table S18: Mean normalized affinity across the affinity-weight sweep.

Target	$\omega_{\text{aff}} = 0.2$	$\omega_{\text{aff}} = 0.5$	$\omega_{\text{aff}} = 0.8$
1B8Q	0.526	0.609	0.653
PPP5	0.615	0.694	0.818

H.3 Predictor-refit evaluation

We trained four bootstrap/initialization replicas for each objective and rescored 3,063 frozen candidates from AReUReDi and the five comparators. Table S20 reports held-out performance and rank agreement between each replica and the reference predictor on the generated population. The reference values describe the PeptiVerse models, which differ from the original predictors in Appendix E. Solubility used a CNN head, affinity used pooled cross-attention, and half-life used a transformer trained on log1p hours. The half-life refit evaluation reconstructed an 80/20 split with seed 1986 because the original split was unavailable. The frozen population contained 3,063 rows and 2,980 unique sequences from equal-weight and affinity-preference AReUReDi runs and five comparison methods across both targets.

The generated-population rankings were more stable for non-fouling, affinity, and non-hemolysis than for solubility and half-life. For each joint combination, we independently sampled one of five predictors per objective (the reference plus four replicas), with replacement across combinations, using random seed 0. We evaluated 100 combinations, applying matched choices across methods for the comparisons (Table S21). Averaged over these combinations, AReUReDi retained the best range, dispersion, and minimum on PPP5. On 1B8Q, MOPSO had a smaller range and dispersion and a larger minimum than AReUReDi. Refitting retained the balance advantage on PPP5 and identified a target-specific ranking reversal on 1B8Q. Rescoring frozen candidates measures evaluation stability; it does not reproduce generation under each refitted model or independently validate biological properties.

H.4 Generator influence during refinement

We compared uniform token proposals, PepDFM proposals, and PepReDi^3 proposals over 180 refinement trajectories. Every endpoint was evaluated with a common frozen PepReDi^3 cross-entropy proxy (Table S22). This proxy is not an exact sequence log-likelihood. Its decrease under PepReDi^3 proposals, together with the endpoint SNN measurements, supports a continuing influence of the

Table S19: Five-objective preference strategies at 108 outputs per setting.

Target	Preferences	HV	IGD^+	Diversity
1B8Q	Dirichlet(1)	0.616	0.094	0.862
	Dirichlet(0.2)	0.592	0.112	0.864
	Deterministic	0.603	0.086	0.871
PPP5	Dirichlet(1)	0.817	0.026	0.872
	Dirichlet(0.2)	0.798	0.057	0.890
	Deterministic	0.804	0.057	0.885

Table S20: Predictor-refit performance. Held-out metric is F1 for the three classifiers and Spearman correlation for the two regressors. Generated rank agreement is Spearman correlation with the reference predictor on the frozen candidate population.

Objective	Reference	Replica range	Generated rank
Non-hemolysis	0.587	0.535–0.543	0.769–0.843
Non-fouling	0.730	0.727–0.735	0.900–0.942
Solubility	0.745	0.726–0.735	0.350–0.705
Half-life	0.726	0.601–0.659	0.224–0.398
Affinity	0.566	0.484–0.529	0.813–0.938

Table S21: Balance averaged over 100 joint predictor combinations.

Target	Method	Range	Disp.	Min.
1B8Q	AReUReDi	0.503	0.198	0.494
	MOPSO	0.483	0.184	0.512
	NSGA-III	0.652	0.248	0.337
	PepTune+DPLM	0.620	0.231	0.345
	SMS-EMOA	0.718	0.274	0.273
	SPEA2	0.667	0.250	0.317
PPP5	AReUReDi	0.438	0.170	0.560
	MOPSO	0.516	0.194	0.477
	NSGA-III	0.691	0.262	0.299
	PepTune+DPLM	0.641	0.237	0.316
	SMS-EMOA	0.658	0.250	0.335
	SPEA2	0.672	0.255	0.309

learned generator. It does not establish biological realism or invariance of the monotone sampler to a base distribution.

H.5 SMILES LENGTH AND COMPUTATIONAL SCALING

We generated 120 peptide SMILES across requested tokenizer lengths of 100, 200, 300, and 400 at 128 refinement updates. No sequence was truncated in this evaluation. Every output passed the peptide-validity filter. Runtime and memory increased with length (Table S23); these measurements support operation over the tested range without establishing asymptotic linear scaling. An additional setting used $T = \max(32, \lfloor L/2 \rfloor)$, including 200 updates at length 400; completed runs used seeds 42, 43, and 44 with five candidates per seed.

H.6 PROPOSAL GRANULARITY AND ACCEPTANCE

We compared single-token proposals with a paired-token construction at a matched total objective-evaluation budget. The paired construction selected the top five tokens at each of two positions and evaluated up to 25 joint candidates, compared with 20 candidates for a single-token update. The single-token configuration achieved higher HV and minimum scores at lower runtime in this

Table S22: Endpoint cross-entropy proxy, change from initialization, and similarity to nearest training neighbor (SNN), evaluated with a common frozen generator.

Target	Proposal	Endpoint CE	Δ CE	SNN
1B8Q	Uniform	2.995	+0.015	0.463
	PepDFM	2.975	-0.002	0.458
	PepReDi[3]	2.940	-0.037	0.483
PPP5	Uniform	3.054	+0.029	0.298
	PepDFM	3.036	+0.011	0.300
	PepReDi[3]	2.970	-0.055	0.321

Table S23: SMILES length stress test with checkpoint `v1` at 128 refinement updates. Requested and realized lengths are tokenizer-token counts; time is per candidate, and validity is the peptide-validity assessment. The 120 outputs cover all four requested lengths.

Requested	Mean realized	Time (s)	s/update	Peak GiB	Valid
100	96.5	301	2.35	16.8	100%
200	196.3	417	3.25	19.1	100%
300	296.8	493	3.85	22.0	100%
400	396.3	582	4.55	26.3	100%

Table S24: Single-token and paired-token proposals at matched objective-evaluation budgets.

Target	Proposal	HV	Min.	Affinity	Time (s)	Accept.
1B8Q	Single	0.631	0.601	0.601	30.9	7.45%
	Paired	0.492	0.465	0.565	54.6	6.89%
PPP5	Single	0.807	0.753	0.753	74.3	8.18%
	Paired	0.682	0.615	0.621	166	7.40%

comparison (Table S24). This comparison evaluated the specified paired-token construction at the stated budget.

The acceptance-rule comparison in Table S6 used the same step budget and 100 outputs per setting. The monotone variant improved the reported mean objectives on both targets. Its finite-budget optimization performance is separate from the equilibrium guarantees of exact MH. Those guarantees require the complete ratio and support conditions in Appendix A.

I Related Optimization and Property-Modeling Context

Biological design covers distinct sequence tasks. Synthetic promoter studies address inducibility and expression control (Tominaga et al., 2024; Artemyev et al., 2024), and guide-RNA design studies address activity and specificity (Mohr et al., 2016; Schmidt et al., 2025). DNA amplification (Nehdi et al., 2020), DNA-structure prediction (Li et al., 2024), and cell-state-aware interpretation of allele-specific chromatin accessibility (Atak et al., 2021) require different experimental readouts. ATAC-seq provides an assay of accessible chromatin (Buenrostro et al., 2013). These applications illustrate distinct biological objectives and measurements; our experiments evaluated peptide design.

Property prediction and optimization also differ in their biological scope. PPI-affinity addresses protein–peptide and protein–protein affinity (Romero-Molina et al., 2022). Deep learning approaches for protein–ligand affinity use other molecular representations and interaction settings (Li et al., 2025; Stepniewska-Dziubinska et al., 2018). ADMET-AI evaluates chemical-library properties (Swanson et al., 2024), and ToxinPred2 and ToxinPred 3.0 predict protein and peptide toxicity, respectively (Sharma et al., 2022; Rathore et al., 2024). Antibody affinity engineering (Bostrom et al., 2008), thermophilic sequence generation (Komp et al., 2025), and latent-space directed evolution (Ngo et al., 2024) address additional property-design problems. These studies motivate explicit distinctions between the property predicted, the molecular representation, and the experiment needed to confirm a candidate.

Protein-aggregate toxicity and misfolded oligomers illustrate why molecular interactions require biological context (Bigi et al., 2023; Rinauro et al., 2024). Strategies for reducing oligomer toxicity concern a different endpoint from our non-hemolysis score (Kreiser et al., 2020). We therefore report the optimized property names and reserve biological safety and efficacy claims for experimental validation.

Related optimization tools span materials and learning problems. COMBO supports Bayesian optimization in materials science (Ueno et al., 2016), and morphology optimization has improved organic solar-cell performance (Zhou et al., 2019). Gradient surgery addresses conflicts among training objectives (Yu et al., 2020); our guidance uses fixed predictors at inference time. Normal-boundary intersection generates Pareto-surface representations (Das & Dennis, 1998), complementing

the scalarization and coverage approaches discussed in the main text. These examples establish the wider optimization context without serving as peptide-design baselines.

J Scope and Limitations

Balance and coverage. AReUReDi produced the strongest reference-predictor balance metrics on the two main comparison targets. SMS-EMOA achieved higher HV and lower empirical IGD^+ in the reported coverage comparison. Candidate balance and coverage of extreme trade-offs are distinct design goals; the results support the former advantage without claiming dominance in all multi-objective metrics. Preference sweeps broadened AReUReDi's empirical coverage, but the finite experiments do not establish recovery of the full Pareto front.

Prediction and validation. All optimized biological properties were predictions. The refit evaluation identified greater ranking sensitivity for half-life and solubility, and the 1B8Q balance ranking changed under refitting. The half-life fine-tuning dataset contained 105 entries, limiting its coverage. AlphaFold3 ipTM and docking provided complementary computational assessments for selected candidates; neither confirms binding, safety, or therapeutic activity. SMILES validity reflects the stated parsing assessment and does not guarantee synthesis. Experimental validation remains necessary for biological conclusions.

Computation and comparisons. The main methods used different generation budgets, and AReUReDi required more wall-clock time per candidate than the listed evolutionary methods and the adapted PepTune baseline. The paired-token experiment controlled objective-evaluation counts for the tested proposal construction. Runtime and memory results are specific to the reported implementations and tested lengths.

Generator interpretation and comparison scope. Conditional factorization error concerns the transition distribution and does not imply independence of residues in the terminal sequence. Comparisons with feedback-driven optimization or additional generative methods require compatible generators, predictors, evaluation protocols, and budgets. The amino-acid PepTune comparison used the stated DPLM adaptation and does not characterize every PepTune configuration.

Theory and implementation. The exact MH results apply to exact density ratios, normalized reverse proposals, fixed targets, and the stated support assumptions. Main experiments used weighted-sum monotone acceptance, whose guarantee concerns utility improvement. The refinement construction also supports unrectified generators. Neither fixed-target concentration nor utility monotonicity proves that the implemented finite-budget sampler converges to the global Pareto front.

K Additional Candidate Visualizations

Selected structures and predicted property trajectories from the original experiments are reproduced below. "Hemolysis" labels in the original graphics refer to the favorable non-hemolysis output. Interface and docking scores are computational assessments, and displayed structures are selected examples.

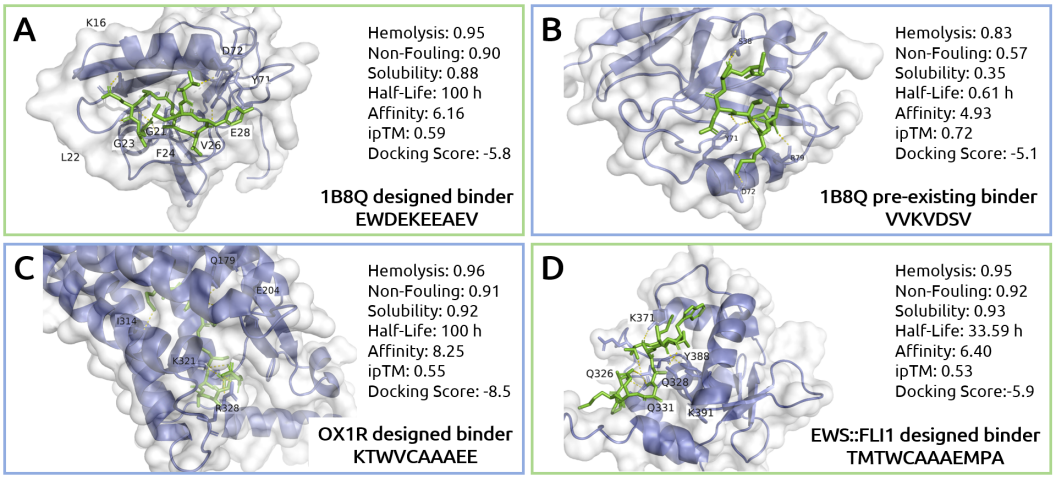

Figure S1: Predicted complexes of 1B8Q with an AReUReDi candidate (A) and a pre-existing binder (B), and OX1R (C) and EWS::FLI1 (D) with designed candidates. Property predictions, AlphaFold3 ipTM, and AutoDock Vina scores are shown.

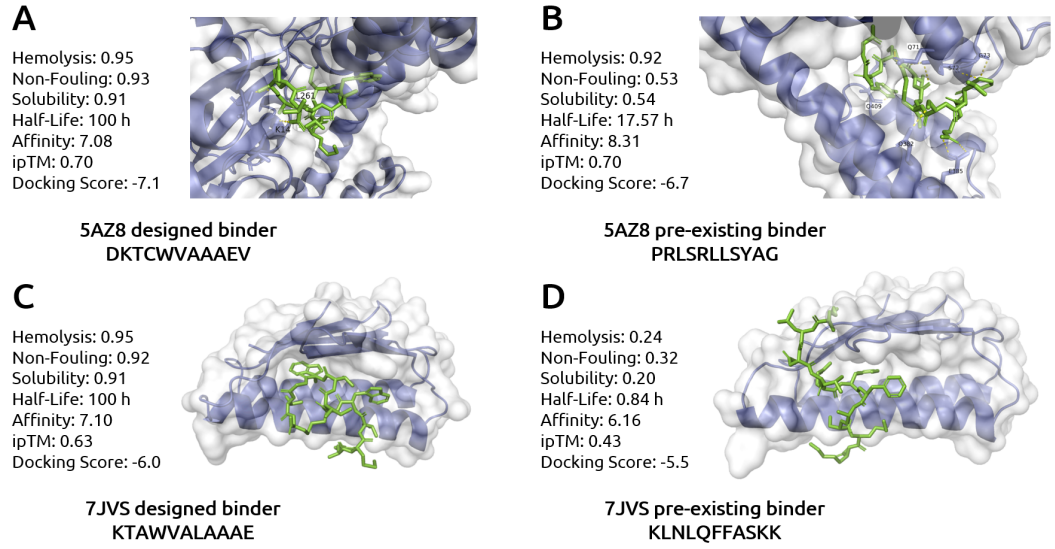

Figure S2: Predicted complexes for 5AZ8 (A–B) and 7JVS (C–D), comparing selected AReUReDi candidates with pre-existing binders. Property predictions, ipTM, and docking scores are shown.

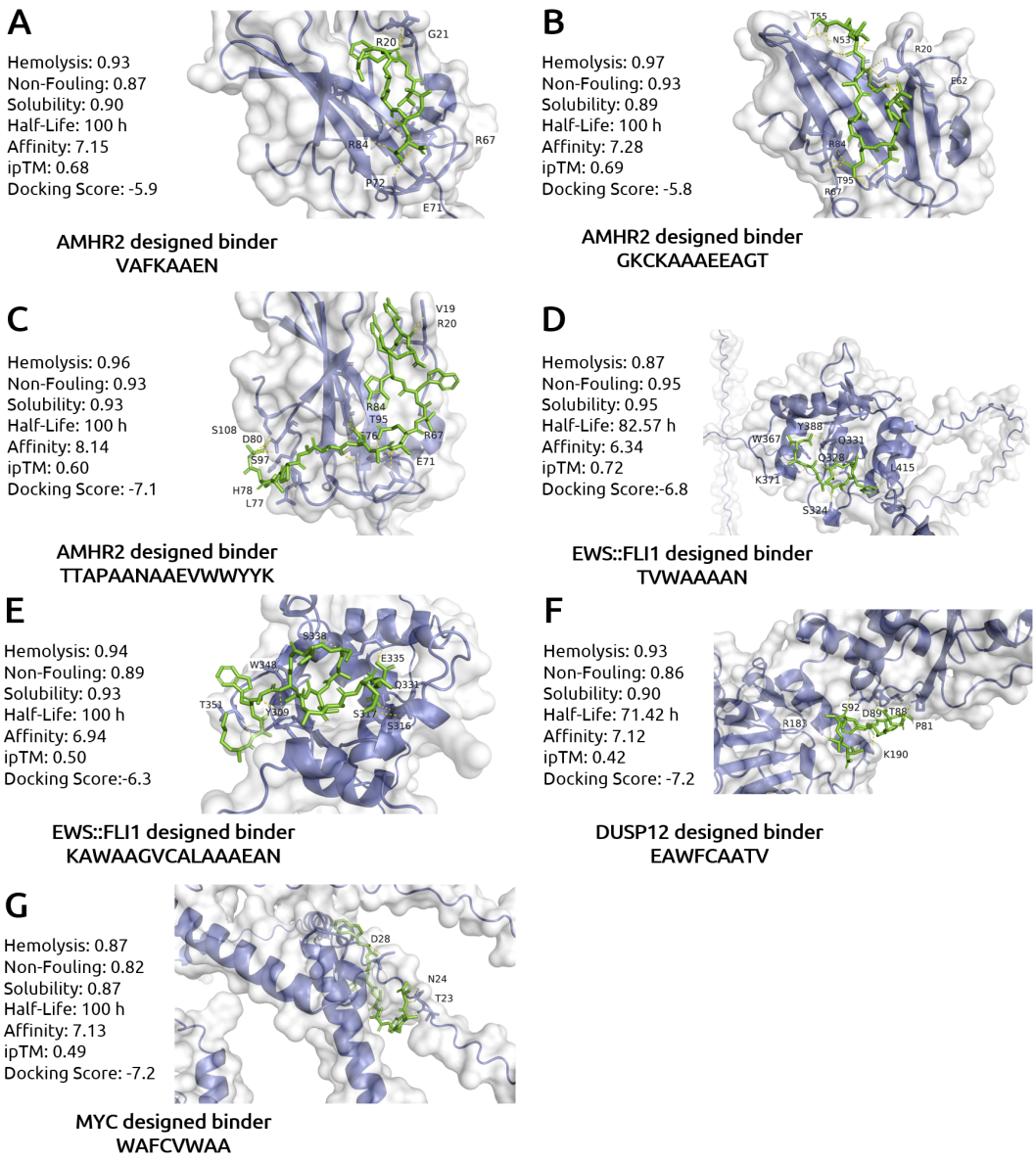

Figure S3: Predicted complexes with designed candidates for AMHR2 (A–C), EWS::FLI1 (D–E), DUSP12 (F), and MYC (G), with property predictions, ipTM, and docking scores.

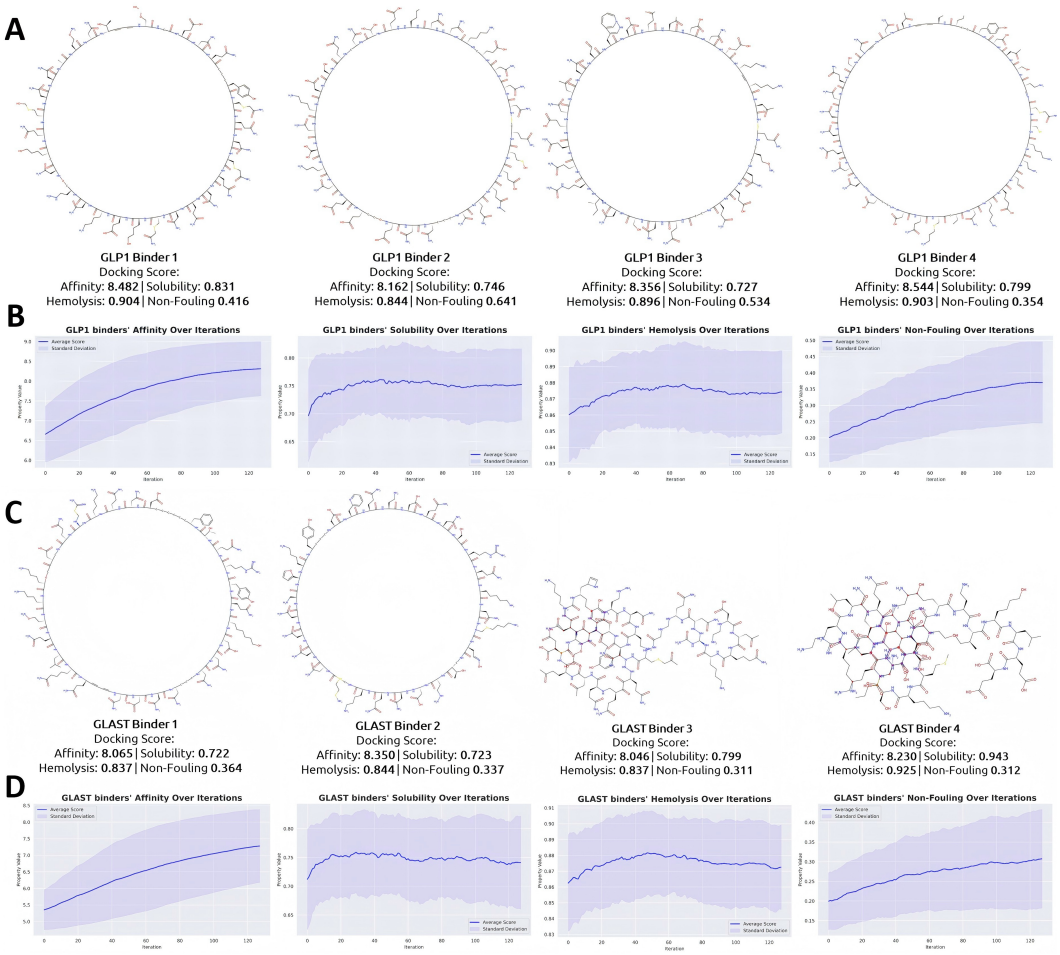

Figure S4: GLP1 and GLAST peptide SMILES: selected candidate structures (A, C) and mean property trajectories across 100 candidates per target at requested length 200 tokenizer tokens (B, D).



Figure S5: TfR and AMHR2 peptide SMILES: selected candidate structures (A, C) and mean property trajectories across 100 candidates per target at requested length 200 tokenizer tokens (B, D).