GENERALIZED PEPTIDE DESIGN WITH PHYSICALLY INSPIRED DIFFUSION

Anonymous authors
Paper under double-blind review

ABSTRACT

Peptides, as short amino acid polymers, have attracted increasing attention due to their favorable pharmacological properties. Recent advances in deep learning have greatly facilitated the design and discovery of peptides. However, most existing generative approaches rely primarily on static structural representations, and the generated peptides generally adhere closely to patterns in the training data, which inherently limits the exploration of functionally diverse conformations and weakens generalization capability to unseen scenarios. Inspired by the dynamic physical process of molecular docking, where hotspot regions form key interactions and flexible regions enable adaptive fit, we propose two complementary strategies for peptide generation: (i) **Flexibility-Conditioned Out-of-Distribution Exploration**, which encourages the model to dynamically explore the conformational landscape based on structural flexibility, thereby promoting high novelty; and (ii) **Hotspot-Driven Hierarchical Affinity Guidance**, which steers the exploration process toward high affinity binding modes by prioritizing key interactions at hotspot regions. Integrating these strategies, we introduce **PhyDiff**, a physically informed diffusion-based framework that adaptively explores the chemical space while simultaneously discovering high-affinity binding patterns, thereby producing peptides that exhibit both strong binding affinity and novelty, and ultimately enhancing the model's generalization capability in real-world drug design scenarios. Extensive experiments across diverse peptide design benchmarks consistently demonstrate the effectiveness and generalization capability of our approach.

1 INTRODUCTION

Peptides, which are short chains of amino acids, hold strong therapeutic potential due to their unique biological functions and ability to bind protein targets (Petsalaki and Russell, 2008; Kaspar and Reichert, 2013). However, discovering functional and high-affinity peptides remains challenging. Conventional experimental methods, such as directed evolution and high-throughput screening (Bhardwaj et al., 2016; Cao et al., 2022), are time-consuming and inefficient, particularly given the vastness of the peptide sequence-structure space.

Deep learning has significantly transformed computational peptide design (Huang et al., 2016; Jin et al., 2021; Jumper et al., 2021). Recent methods have demonstrated impressive capabilities in generating peptide sequences and structures likely to bind to specific protein targets (Watson et al., 2022; Roel-Touris et al., 2023). In particular, two dominant paradigms have emerged: diffusion-based generative models and hallucination-based frameworks. Diffusion-based methods condition on the target protein to learn the distribution of binding peptides (Luo et al., 2022; Li et al., 2024b; Kong et al., 2025a), while hallucination-based frameworks leverage pretrained models without additional training, using gradient-based optimization to iteratively refine candidates until high model confidence or binding affinity is achieved (Pacesa et al., 2024; Wohlwend et al., 2025; Passaro et al., 2025).

Despite these advances, a critical blind spot remains: most mainstream approaches overlook peptide flexibility and the dynamic nature of protein-peptide recognition, both of which are recognized as essential for accurate binding (Boehr et al., 2009; Wright and Dyson, 2015). Specifically, peptides do not bind as rigid bodies. They form key interactions at specific hotspot regions on the protein surface while retaining enough flexibility to dynamically adapt to the target's geometry (Clackson and Wells, 1995; Bogan and Thorn, 1998). This dynamic interplay is a hallmark of high-affinity binding.

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.

Diffusion-based methods rely on static peptide information from training sets. While they produce physically plausible binders, they tend to reproduce similar patterns rather than exploring truly novel, flexible motifs (Kong et al., 2023; Li et al., 2024b). Moreover, peptide-specific datasets are far smaller than protein datasets, further limiting transferability to unseen scenarios. Hallucination-based methods depend heavily on confidence metrics of pretrained folding models and require prolonged iterative optimization (Evans et al., 2021; Mirdita et al., 2022). Several previous studies have highlighted that the predictive accuracy of existing pretrained folding or inverse-folding models degrades sharply for protein–peptide complexes that differ considerably from their training distribution (Johansson-Åkhe and Wallner, 2022; Petsalaki and Russell, 2008). Overall, these limitations severely hinder the generalization and practical utility in drug design settings.

To overcome these challenges, we draw inspiration from the physical process of molecular docking and propose **PhyDiff**, a physically informed diffusion-based framework (Lipman et al., 2022). PhyDiff integrates two complementary mechanisms: (i) Flexibility-Conditioned Out-of-Distribution Exploration (FlexOOD), which leverages structural flexibility to explore beyond the training distribution and navigate diverse geometric spaces; and (ii) Hotspot-Driven Hierarchical Affinity Guidance (HHAG), which provides hierarchical guidance at both residue-level and peptide-level, facilitating the discovery of high-affinity binding patterns and hotspot recognition during exploration. By coupling flexibility-aware exploration with hotspot-driven affinity guidance, PhyDiff adaptively explores diverse geometries while simultaneously discovering high-affinity binding patterns and recognizing key hotspot interactions. This yields peptide binders that are both high in affinity and novel, thereby enhancing generalization and real-world applicability. Our main contributions include:

- We analyze the generalization and transferability limitations of mainstream deep peptide design methods, and propose a more generalized docking-inspired paradigm, PhyDiff.
- Through FlexOOD and HHAG, PhyDiff generates high-affinity, novel peptide binders, thereby improving the practical applicability of drug design.
- Extensive experiments on de novo design and scaffold-constrained generation tasks consistently demonstrate the effectiveness and generalization capability of PhyDiff.

2 Related Work

Peptide Design. Conventional peptide/protein design methods sample residues or assemble small structural fragments from predefined libraries (Huang et al., 2016; Evans et al., 2021; Bryant and Elofsson, 2022). While physically plausible, they are computationally intensive and prone to local optima. Recent deep generative models offer more efficient modeling of the sequence-structural space (Lin et al., 2024; Wu et al., 2024). For instance, PepFlow (Li et al., 2024b) introduces a multi-modal flow matching framework for co-designing sequences and full-atom structures, and PepGLAD (Kong et al., 2024) employs geometric latent diffusion coupled with receptor-specific affine transformations to generalize across diverse binding scenarios. Beyond diffusion-based methods, hallucination-based frameworks such as BindCraft (Pacesa et al., 2024), Boltz1 (Wohlwend et al., 2025), and Boltz2 (Passaro et al., 2025) also demonstrate strong performance, which leverage powerful pretrained models and extensive iterative optimization. However, most mainstream deep models overlook peptide flexibility and the dynamic nature of protein–peptide recognition. Consequently, generated peptides tend to mirror the patterns of the training data, limiting generalization to unseen scenarios.

Molecular Flexibility. Previous approaches (Alford et al., 2017; Planas-Iglesias et al., 2022) have attempted to incorporate flexibility into protein design using classical simulation tools and energy functions, but these methods are computationally intensive and limited in their ability to generate novel structures. Recent studies have begun leveraging deep generative models conditioned on protein flexibility, with flexibility parameters derived from molecular dynamics simulations. For example, FlexPert-3D (Kouba et al., 2024) proposes a pipeline that conditions protein sequence design on MD-derived flexibility. Besides, FliPS (Viliuga et al., 2025) introduces a conditional flow matching model to generate backbones that display a target flexibility profile.

Hotspot-Driven Methods. Many studies have shown that certain residues play a crucial role in mediating strong interactions with target proteins, contributing significantly to binding affinity (Boehr et al., 2009; Wright and Dyson, 2015). Therefore, identifying key interaction residues and hotspot

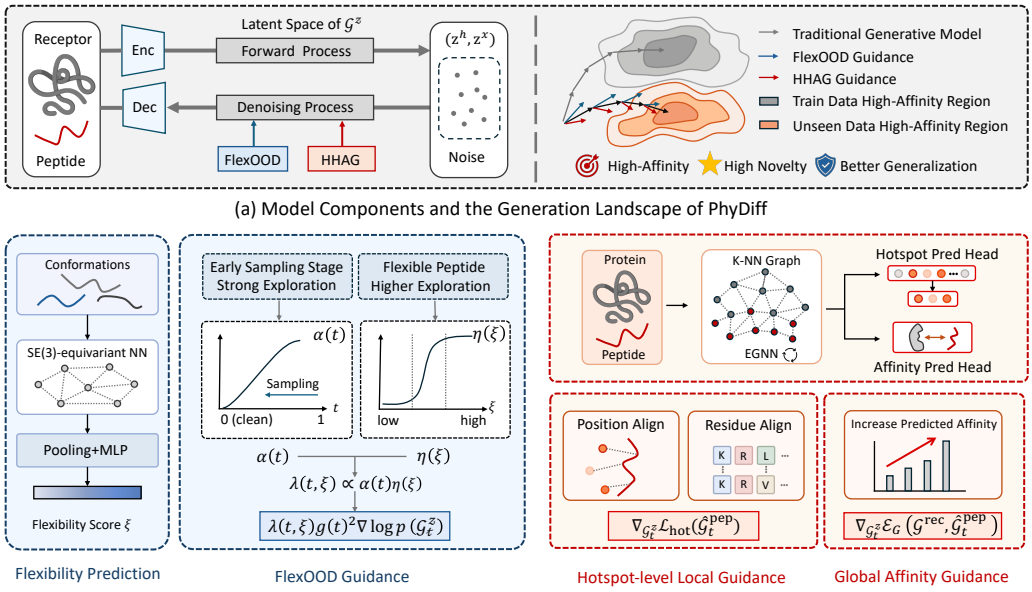


Figure 1: **The workflow of PhyDiff**.

regions is essential for accurate binding. Methods such as AlphaProteo (Zambaldi et al., 2024) and PepHAR (Li et al., 2024a) explicitly condition the generative process on hotspot regions to enhance binding affinity. However, these hotspot-driven methods generally ignore the dynamic nature of peptides and still face challenges in generating binders with both high affinity and novelty.

3 Methodology

In this section, we introduce the two core components of PhyDiff: Flexibility-Conditioned Out-of-Distribution Exploration (FlexOOD) and Hotspot-Driven Hierarchical Affinity Guidance (HHAG), as shown in Figure 1. FlexOOD allows the model to dynamically explore diverse chemical space conditioned on peptide flexibility. To identify high-affinity binding patterns and avoid implausible generative trajectories during the exploration process, we further propose HHAG. Together, these two components enable PhyDiff to adaptively explore beyond the training distribution, while effectively capturing high-affinity interaction patterns. As a result, PhyDiff achieves stronger generalization capability and advances the practical development of real-world peptide design.

3.1 Preliminary

Problem Definition. Let a peptide or receptor be represented as a residue-level geometric graph $\mathcal{G} = \{(a_i, \mathbf{X}_i)\}_{i=1}^{N}$, where N denotes the number of residues, a_i is the amino acid identity of residue i, and $\mathbf{X}_i \in \mathbb{R}^{c_i \times 3}$ denotes the full-atom coordinates of residue i, with c_i being the number of atoms in that residue. Given a receptor–peptide complex, we denote the receptor binding site by $\mathcal{G}^{\text{rec}}$ and the peptide by $\mathcal{G}^{\text{pep}}$. Following PepGLAD (Kong et al., 2024), we define the receptor binding site as the set of target-protein residues whose C_β atoms lie within 10Å of the peptide residue. Our goal is to learn a generative model p_θ for peptide generation conditioned on the receptor binding site. Specifically, given the binding site $\mathcal{G}^{\text{rec}}$, we aim to generate peptides that conform to the conditional distribution of binding peptides, i.e., $q(\mathcal{G}^{\text{pep}} \mid \mathcal{G}^{\text{rec}})$.

Latent Diffusion for Peptide Design. Considering the diverse modalities of proteins and peptides, including both sequence information and structural geometry, directly generating a valid peptide in the original space is challenging. Following PepTri (Nguyen et al., 2026) and PepGLAD, we unify peptide sequence and structure generation in an SE(3)-equivariant latent space, ensuring that rotations

and translations of the input lead to corresponding transformations in the output. Specifically, we first adopt an SE(3)-equivariant variational autoencoder (VAE) to encode the peptide in the context of the binding site into a latent representation, and then reconstruct the peptide sequence and structure from the latent embeddings. Formally, given the peptide $\mathcal{G}^{\text{pep}}$ and the binding site $\mathcal{G}^{\text{rec}}$, the encoder E maps the input into a latent state $\mathcal{G}^z$, while the decoder D reconstructs the peptide as:

$$\mathcal{G}^z = E(\mathcal{G}^{\text{pep}}, \mathcal{G}^{\text{rec}}), \quad \hat{\mathcal{G}}^{\text{pep}} = D(\mathcal{G}^z, \mathcal{G}^{\text{rec}}),$$

where $\mathcal{G}^z = \{(z_i^h, z_i^x) \mid i \in \mathcal{G}^{\text{pep}}\}$ contains the latent feature representation $z_i^h \in \mathbb{R}^{d_h}$ and geometric representation $z_i^x \in \mathbb{R}^{c_i \times 3}$ for each residue, and $\hat{\mathcal{G}}^{\text{pep}} = \{(\hat{a}_i, \hat{\mathbf{X}}_i)\}$. The VAE is trained by optimizing a reconstruction objective on both sequence and structure, along with a Kullback–Leibler (KL) regularization term, as follows:

$$\mathcal{L}_{\text{VAE}} = \sum_{i \in \mathcal{G}^{\text{pep}}} H\big(p(a_i), p(\hat{a}_i)\big) + \text{MSE}(\mathbf{X}_i, \hat{\mathbf{X}}_i) + \beta \mathcal{L}_{KL},$$

where $H(\cdot, \cdot)$ denotes the cross-entropy loss, MSE is mean squared error, and β is scalar. By leveraging the VAE, we transform peptide generation from the original complex space into a compact and structured latent space of $\mathcal{G}^z$. Let $\mathcal{G}_t^z$ denote the latent state of the peptide at diffusion step t. Inspired by EDM (Karras et al., 2022), we train the denoising network f_θ to directly predict the clean latent embedding from noisy inputs at different noise levels. Specifically, given a clean latent embedding $\mathcal{G}_0^z$, we perturb it as $\mathcal{G}_t^z = \mathcal{G}_0^z + \sigma_t \epsilon, \epsilon \sim \mathcal{N}(0, \mathbf{I})$, and train the denoiser f_θ with

$$\mathcal{L}_{\text{LDM}} := \mathbb{E}_{\mathcal{G}_0^z, \sigma_t} \left[\| f_\theta(\mathcal{G}_t^z, \sigma_t) - \mathcal{G}_0^z \|_2^2 \right].$$

For clarity, we provide more details about the VAE and the denoising network in Appendix B.1.

3.2 Flexibility-Conditioned Out-of-Distribution Exploration

Traditional diffusion-based models typically rely on static peptide representations, and the generated binders tend to resemble the patterns present in the limited training sets, making it difficult to generalize to real-world scenarios where the targets may differ substantially from the training data. To address this limitation, we propose FlexOOD that enables the model to dynamically explore chemical space beyond the training distribution based on the flexibility of each peptide, which broadens the exploration space during generation and improves generalization capability to unseen targets.

Flexibility Predictor. Proteins and peptides display dynamic behavior to perform diverse functions, such as catalysis, association, or regulation (Henzler-Wildman and Kern, 2007). To quantify flexibility, we employ the widely used Root Mean Square Fluctuation (RMSF) as a proxy to measure the fluctuations with respect to the surroundings of the peptide. Mathematically, the overall flexibility ξ of the peptide from a conformation set $\mathcal{X} \subset \mathbb{R}^{N \times 3}$ and a reference conformation $\mathbf{x}^{(\text{ref})} \in \mathbb{R}^{N \times 3}$ is:

$$\xi \equiv \sqrt{\frac{1}{|\mathcal{X}|} \sum_{\mathbf{x} \in \mathcal{X}} \left| (T_{\text{Align}} \circ \mathbf{x}) - \mathbf{x}^{(\text{ref})} \right|^2}, \tag{1}$$

where $T_{\text{Align}} \equiv \arg\min_{T \in SE(3)} \left\| T \circ \mathbf{x} - \mathbf{x}^{(\text{ref})} \right\|_2$ can be efficiently computed using the Kabsch algorithm (Kabsch, 1976). Here, $T \circ \mathbf{x}$ denotes the action of the rigid transformation on $\mathbf{x}$.

To enable accurate flexibility prediction, we employ an SE(3)-equivariant predictor. The predictor is implemented by EGNN (Satorras et al., 2021), which takes the all-atom features and positions of the peptide as input and extracts node embeddings $\mathbf{h}_i$ after message passing. We then apply a permutation-invariant pooling operator over $\{\mathbf{h}_i\}$ and leverage a Multi-Layer Perceptron (MLP) to map the global representation to $[0, \xi_{\max}]$ via a scaled sigmoid activation:

$$\xi = \mathcal{E}_\xi(\mathcal{G}^{\text{pep}}) = \xi_{\max}\, \sigma(\text{MLP}(\text{Pool}(\{\mathbf{h}_i\}))), \tag{2}$$

where $\xi_{\max}$ is a pre-defined upper bound on flexibility. We obtain ground-truth flexibility labels from conformation ensembles extracted from molecular dynamics (MD) trajectories and train the flexibility predictor on the ATLAS dataset (Vander Meersche et al., 2024).

Flexibility-Conditioned Out-of-Distribution Exploration. Following (Song et al., 2021), the reverse process of diffusion models can be formulated as the following Stochastic Differential Equation (SDE):

$$d\mathcal{G}_t^z = [\mathbf{f}(\mathcal{G}_t^z, t) - g(t)^2 \nabla_{\mathcal{G}_t^z} \log p_t(\mathcal{G}_t^z)]dt + g(t)\, d\bar{\mathbf{w}}, \tag{3}$$

where $\mathbf{f}(\cdot)$ and $g(\cdot)$ denote the drift and diffusion coefficients, respectively, $\nabla \log p_t(\cdot)$ is the score function, and $\bar{\mathbf{w}}$ denotes the reverse-time Wiener process. Recently, MOOD (Lee et al., 2023) proposed controlling the degree of OOD exploration during sampling by damping the score term with a scalar hyperparameter $\lambda \in [0, 1)$. This yields the modified reverse-time SDE as follows:

$$d\mathcal{G}_t^z = \left[\mathbf{f}(\mathcal{G}_t^z, t) - \left(1 - \sqrt{\lambda}\right) g(t)^2 \nabla_{\mathcal{G}_t^z} \log p_t(\mathcal{G}_t^z)\right] dt + g(t)\, d\bar{\mathbf{w}}. \tag{4}$$

Here, a larger λ encourages broader exploration. However, since different peptides exhibit distinct flexibility characteristics, using a fixed scalar makes it difficult to perform adaptive exploration according to the dynamic properties of each peptide. To address this limitation, we propose an adaptive flexibility-conditioned OOD strategy. Specifically, we leverage the peptide-level flexibility ξ and replace λ with a time and flexibility-dependent function $\lambda(t, \xi)$. First, we map the peptide flexibility via a sigmoid function as follows:

$$\eta(\xi) = \frac{1}{1 + \exp(-k(\xi - \xi_0))}, \tag{5}$$

where $k > 0$ controls the transition sharpness and ξ_0 is the midpoint. As shown in Figure 1(b), $\eta(\xi)$ enables adaptive exploration: rigid peptides with small ξ receive minimal extrapolation, while more flexible peptides are assigned larger OOD weights. Moreover, $\eta(\xi) \in (0, 1)$ saturates for large ξ, ensuring that the extrapolation strength remains bounded. Intuitively, in the early phase of the sampling trajectory when $t \to 1$, the state is relatively far from the data manifold and broader exploration is beneficial, whereas as $t \to 0$, the generation should converge toward physically plausible structures. Based on this insight, we further modulate $\lambda(t, \xi)$ with a cosine annealing schedule as follows:

$$\lambda(t, \xi) = \lambda_{\text{base}}\, \alpha(t)\eta(\xi), \qquad \alpha(t) = \frac{1 + \cos(\pi(1 - t))}{2}, \tag{6}$$

where λ_{base} is a scalar. Finally, the reverse-time diffusion process of FlexOOD is updated as:

$$d\mathcal{G}_t^z = \left[\mathbf{f}(\mathcal{G}_t^z, t) - \left(1 - \sqrt{\lambda(t, \xi)}\right) g(t)^2 \nabla_{\mathcal{G}_t^z} \log p_t(\mathcal{G}_t^z)\right] dt + g(t)\, d\bar{\mathbf{w}}. \tag{7}$$

To ensure conformational validity during exploration, we adopt physical losses from AlphaFold2 (Jumper et al., 2021) to penalize implausible geometries. Details can be found in Appendix B.3.

3.3 Hotspot-Driven Hierarchical Affinity Guidance

FlexOOD encourages the model to explore a more diverse chemical space. To steer this exploration process toward high-affinity and physically plausible patterns, we propose HHAG, which integrates both residue-level and peptide-level guidance. This enables the model to capture key interactions in hotspot regions while discovering patterns that further improve overall affinity.

Hotspot Labeling and Prediction. First, we label the hotspot residues on the peptide given the target receptor. Specifically, we run Rosetta alanine scanning (Alford et al., 2017; Cunningham and Wells, 1989) to compute per-residue $\Delta\Delta G$ values, defined as the change in binding free energy upon alanine substitution at each peptide residue. We then select the top K residues with the largest $\Delta\Delta G$ values as hotspots. Let $\mathcal{H} \subset \{1, \ldots, |\mathcal{G}^{\text{pep}}|\}$ denote the set of peptide hotspot indices, where $|\mathcal{H}| = K$. We define the hotspot region as $\mathcal{G}^{\text{hot}} = \{(a_i, \mathbf{X}_i)\}_{i \in \mathcal{H}}$.

In practical peptide design, we typically only have access to the receptor information $\mathcal{G}^{\text{rec}}$. Therefore, we employ an equivariant network $\mathcal{E}_\psi$ that takes $\mathcal{G}^{\text{rec}}$ as input and predicts $\hat{\mathcal{G}}^{\text{hot}} = \mathcal{E}_\psi(\mathcal{G}^{\text{rec}})$ to estimate the peptide hotspot region. Since the number of hotspot residues is usually small, i.e., $K \in [1, 3]$ (Clackson and Wells, 1995), learning $\mathcal{E}_\psi$ is relatively tractable, which facilitates accurate hotspot prediction. Besides, in real-world design scenarios, prior knowledge may directly specify key interacting residues, obviating the need to rely on $\mathcal{E}_\psi$. We will discuss it in Appendix C.1.

Hotspot Identification and Affinity Evaluation. To enable hotspot residue identification and overall affinity evaluation during generation, we follow ProteinMPNN (Dauparas et al., 2022) and represent the receptor-peptide complex $(\mathcal{G}^{\mathrm{rec}}, \mathcal{G}^{\mathrm{pep}})$ as a k-NN graph built from heavy-atom coordinates. We then use a unified EGNN $\mathcal{E}_G$ to jointly predict residue-level hotspot probabilities and the binding affinity. Concretely, $\mathcal{E}_G$ first produces embeddings $\{\mathbf{h}_i\}$ for all residues. Hotspot probabilities are obtained by applying an MLP head to each peptide residue, followed by a softmax, while the affinity score is predicted from a permutation-invariant pooling of all residue embeddings:

$$\ell_i = \mathrm{MLP}_{\mathrm{hot}}(\mathbf{h}_i),\ \hat{p}_i = \frac{\exp(\ell_i)}{\sum_{j \in V^{\mathrm{pep}}} \exp(\ell_j)},\ i \in V^{\mathrm{pep}}, \tag{8}$$

$$\hat{s} = \mathrm{MLP}_{\mathrm{aff}}\Big(\mathrm{Pool}\big(\{\mathbf{h}_i\}\big)\Big),\ i \in V^{\mathrm{comp}}, \tag{9}$$

where V^{pep} and V^{comp} represent the peptide residues and complex residues, respectively. Then, $\mathcal{E}_G$ can be trained with a multi-task loss $\mathcal{L} = -\sum_{i \in V_{\mathrm{pep}}} y_i \log \hat{p}_i + \lambda_{\mathrm{aff}}\big(\hat{s} - s\big)^2$, where $y_i \in \{0, 1\}$ is the ground-truth hotspot indicator for peptide residue i, $s \in \mathbb{R}$ is the binding affinity estimated by Rosetta (Leaver-Fay et al., 2011). Sharing the same graph encoder encourages representations that capture both residue-level interaction patterns and global binding strength.

Hierarchical Affinity Guidance. To improve the binding affinity of the generated peptides during the exploration process, we propose a hierarchical affinity-guiding scheme that considers both residue-level hotspot-driven guidance and overall peptide-level affinity guidance. Specifically, we use classifier-guided sampling (Dhariwal and Nichol, 2021) and steer the generated peptide to satisfy a specific condition y based on the Bayes rule as follows:

$$\nabla_{\mathcal{G}_t^z} \log p(\mathcal{G}_t^z | y) \approx \nabla_{\mathcal{G}_t^z} \log p(\mathcal{G}_t^z) + \gamma \nabla_{\mathcal{G}_t^z} \log p(y | \mathcal{G}_t^z), \tag{10}$$

where $\nabla_{\mathcal{G}_t^z} \log p(y|\mathcal{G}_t^z)$ is the guidance term and γ is the guiding strength.

i) ***Hotspot-level local guidance.*** To guide the model to focus on key interactions in hotspot regions, we first estimate the clean latent state and decode it to obtain the peptide representation $\hat{\mathcal{G}}_t^{\mathrm{pep}} = D(f_\theta(\mathcal{G}_t^z, \sigma_t), \mathcal{G}^{\mathrm{rec}})$. Then, we use $\mathcal{E}_G$ to identify potential hotspot residues based on the receptor–peptide complex $(\mathcal{G}^{\mathrm{rec}}, \hat{\mathcal{G}}_t^{\mathrm{pep}})$. Finally, we leverage the predicted hotspot information $\hat{\mathcal{G}}^{\mathrm{hot}}$ to steer the decoded peptide toward these hotspot-specific anchors. The hotspot-driven guidance is defined as:

$$-\nabla_{\mathcal{G}_t^z} \mathcal{L}_{\mathrm{hot}}(\hat{\mathcal{G}}_t^{\mathrm{pep}}) = -\nabla_{\mathcal{G}_t^z} \sum_{i \in V^{\mathrm{pep}}} \mathbb{I}\left[i \in \mathcal{H}\right] \Big(\|\mathbf{x}_i^{\mathrm{hot}} - \hat{\mathbf{x}}_i\|^2 + \lambda_{\mathrm{type}} \mathrm{CE}\big(\mathbf{s}_i^{\mathrm{hot}}, \hat{\mathbf{s}}_i\big) \Big), \tag{11}$$

where the predicted hotspot residue set $\mathcal{H}$ is obtained based on the hotspot probability $\hat{p}_i$ from $\mathcal{E}_G$ by top-K selection, and $\mathbf{x}_i^{\mathrm{hot}}$ and $\mathbf{s}_i^{\mathrm{hot}}$ are the hotspot-specific anchor position and residue-type distribution derived from $\hat{\mathcal{G}}^{\mathrm{hot}}$, respectively. Besides, $\hat{\mathbf{x}}_i$ and $\hat{\mathbf{s}}_i$ are peptide information of $\hat{\mathcal{G}}_t^{\mathrm{pep}}$.

ii) ***Peptide-level global affinity guidance.*** Apart from the residue-level hotspot-driven guidance, we also incorporate peptide-level global guidance to encourage the model to discover global interaction patterns that improve the overall binding affinity. We encourage high-affinity binders by performing gradient ascent as follows:

$$\nabla_{\mathcal{G}_t^z} \mathcal{E}_G\Big(\mathcal{G}^{\mathrm{rec}}, \hat{\mathcal{G}}_t^{\mathrm{pep}}\Big), \tag{12}$$

which steers $\hat{\mathcal{G}}_t^{\mathrm{pep}}$ towards peptides with higher predicted affinity under $\mathcal{E}_G$. Finally, according to (10), combining the hotspot-level guidance in (11) and the peptide-level global affinity guidance in (12), the hierarchical affinity guidance term is expressed as:

$$\mathbf{g}_t^{\mathrm{HHAG}} = \gamma_1 \nabla_{\mathcal{G}_t^z} \mathcal{E}_G(\mathcal{G}^{\mathrm{rec}}, \hat{\mathcal{G}}_t^{\mathrm{pep}}) - \gamma_2 \nabla_{\mathcal{G}_t^z} \mathcal{L}_{\mathrm{hot}}(\hat{\mathcal{G}}_t^{\mathrm{pep}}). \tag{13}$$

3.4 Overall Procedure

By leveraging FlexOOD (7) with HHAG (13), PhyDiff achieves adaptive exploration with high-affinity pattern discovery, which facilitates the generation of binders that exhibit both high novelty and predicted affinity, ultimately improving generalization and performance in unseen design scenarios. The complete reverse-time sampling process of PhyDiff is governed by the following unified SDE:

$$d\mathcal{G}_t^z = \Big[\mathbf{f}(\mathcal{G}_t^z, t) - \big(1 - \sqrt{\lambda(t,\xi)}\big)\, g(t)^2\, \nabla_{\mathcal{G}_t^z} \log p_t(\mathcal{G}_t^z) - g(t)^2\, \mathbf{g}_t^{\mathrm{HHAG}}\Big]\, dt + g(t)\, d\bar{\mathbf{w}}, \tag{14}$$

We provide a more detailed description of the PhyDiff workflow in Algorithm 1.

4 EXPERIMENT

Datasets. Following prior studies (Kong et al., 2024; Nguyen et al., 2026), we evaluate PhyDiff under both cross-domain and in-domain settings. The data mainly consist of short peptides, with a substantial proportion shorter than 30 amino acids, which matches common peptide design scenarios.

For the **cross-domain** setting, we train the model on **PepBench** (Kong et al., 2024) and evaluate on the **LNR** test set curated by (Tsaban et al., 2022), which contains 93 expert-curated protein–peptide complexes with canonical amino acids. This setting is used to assess the generalization ability of PhyDiff on unseen targets from an external distribution. For the **in-domain** setting, we use **PepBDB** (Wen et al., 2019). We split the dataset into training, validation, and test sets following the same setting as the previous methods (Nguyen et al., 2026). Besides, we extract the protein–peptide complexes with biologically meaningful lengths, i.e., receptors longer than 30 residues and peptides containing 4–25 residues. We further remove redundant complexes according to sequence identity constraints on both receptors and peptides, and apply MMseqs2 clustering for split construction. In addition, we follow PepGLAD and leverage **ProtFrag**, a collection of approximately 70,645 unlabeled protein fragments, as auxiliary unsupervised data to pretrain the VAE. The details of the datasets can be found in Appendix A.1.

Baselines. To comprehensively evaluate the effectiveness of our proposed method, we compare PhyDiff with advanced peptide design methods, such as PepGLAD (Kong et al., 2024), PepFlow (Li et al., 2024b), UniMoMo (Kong et al., 2025b), PepHAR (Li et al., 2024a), and Peptri (Nguyen et al., 2026). Besides, we also leverage the Hallucination-based BindCraft (Pacesa et al., 2024) for further comparison in Appendix C.5. We also include details of each baseline in Appendix A.2.

Evaluation Metrics. We follow PepTri and employ a comprehensive set of evaluation metrics that capture both structural quality and functional relevance. Specifically, we report **Success Rate**, defined as the fraction of generated peptides with thermodynamically stable binding; **Binding Free Energy** (ΔG), which measures binding affinity; **DockQ**, which evaluates interface quality. To further assess the novelty of the generated peptides, we also report **Sequence Diversity**, **Structure Diversity**, **Sequence Validity**, and **Consistency**, which respectively measure the diversity of generated sequences and structures, the proportion of valid sequences, and the sequence-structure consistency. We also leverage geometrical metrics, such as **Contact_F1**, **Local RMSD**, and **GDT_TS**. The calculation of the metrics can be found in Appendix A.3.

4.1 EXPERIMENTS ON DE NOVO PEPTIDE DESIGN

To evaluate the de novo peptide design capability of PhyDiff, we conduct sequence–structure co-design experiments on the widely used PepBench and PepBDB datasets. Following PepGLAD, we generate 40 candidates for each target and report Success Rate, ΔG, and DockQ, as summarized in Table 1. We additionally report per-target standard deviations. As shown in the table, PhyDiff achieves competitive performance compared with other advanced models. In particular, on the unseen LNR test set of PepBench, PhyDiff achieves strong performance across all three metrics, providing initial evidence of improved target-level generalization. Essentially, HHAG enables PhyDiff to focus more effectively on the key interactions in hotspot regions between the receptor and the peptide, guiding the generated peptides toward high-affinity patterns and thereby improving binding performance.

A desirable generative model should not only produce reasonable and high-affinity peptide candidates, but also generate diverse and novel candidates to ensure its applicability in real-world scenarios. To this end, following (Yim et al., 2023) and (Kong et al., 2024), we measure the diversity via unique clusters of sequences and structures. In addition, we also report Sequence Validity and sequence–structure Consistency. The full results are presented in Table 2, where PhyDiff achieves competitive performance on diversity and validity metrics, demonstrating its strong ability to generate diverse yet biologically plausible sequences together with coherent structures. Besides, we also include geometrical metrics, including Contact_F1, Local RMSD, and GDT_TS in Appendix C.4.

Table 1: Binding quality metrics on PepBench and PepBDB. Baseline results are taken from PepTri for fair comparison. Results are reported as *without relaxation / with relaxation*.

Dataset	Method	Success Rate (↑)	ΔG (REU) (↓)	DockQ (↑)
PepBench	PepGLAD	0.29 ± 0.19 / 0.79 ± 0.17	-15.63 ± 8.51 / -34.48 ± 12.44	0.60 ± 0.15 / 0.59 ± 0.14
	PepFlow	0.31 ± 0.19 / 0.74 ± 0.13	-17.05 ± 8.25 / -35.98 ± 18.81	0.53 ± 0.11 / 0.42 ± 0.09
	PepHAR	0.32 ± 0.16 / 0.81 ± 0.12	-16.78 ± 7.54 / -33.52 ± 14.51	0.53 ± 0.16 / 0.57 ± 0.16
	$\text{UniMoMo}_{\text{single}}$	0.34 ± 0.19 / 0.79 ± 0.13	-19.04 ± 7.17 / -30.19 ± 9.55	0.57 ± 0.23 / 0.54 ± 0.19
	PepTri	0.40 ± 0.19 / 0.83 ± 0.16	-19.39 ± 7.08 / -36.36 ± 14.27	0.63 ± 0.16 / 0.62 ± 0.15
	PhyDiff (Ours)	**0.43 ± 0.17 / 0.85 ± 0.15**	**-20.52 ± 6.73 / -36.54 ± 11.28**	**0.66 ± 0.13 / 0.67 ± 0.11**
PepBDB	PepGLAD	0.15 ± 0.12 / 0.67 ± 0.26	-14.48 ± 9.91 / -31.22 ± 13.28	0.43 ± 0.20 / 0.43 ± 0.19
	PepFlow	0.30 ± 0.15 / 0.66 ± 0.31	-17.44 ± 9.52 / -34.98 ± 25.97	0.41 ± 0.21 / 0.31 ± 0.15
	PepHAR	0.28 ± 0.18 / 0.64 ± 0.20	-18.02 ± 9.43 / -32.57 ± 17.64	0.40 ± 0.21 / 0.46 ± 0.15
	$\text{UniMoMo}_{\text{single}}$	0.30 ± 0.22 / 0.74 ± 0.25	-18.89 ± 12.59 / -34.05 ± 18.89	0.44 ± 0.20 / 0.43 ± 0.18
	PepTri	0.31 ± 0.23 / 0.74 ± 0.27	-18.15 ± 11.92 / -34.82 ± 18.20	0.49 ± 0.21 / 0.49 ± 0.19
	PhyDiff (Ours)	**0.38 ± 0.13 / 0.78 ± 0.17**	**-19.34 ± 8.55 / -35.12 ± 16.24**	**0.53 ± 0.18 / 0.55 ± 0.16**

Table 2: Design diversity metrics on PepBench and PepBDB (*without relaxation / with relaxation*).

Dataset	Method	SeqDiv (↑)	SeqValid(↑)	StrDiv(↑)	Consistency(↑)
PepBench	PepGLAD	0.92	0.27	0.54 / 0.60	0.81 / 0.84
	PepFlow	0.83	0.25	0.44 / 0.58	0.67 / 0.80
	UniMoMo_{single}	0.92	0.21	**0.62** / 0.65	0.79 / 0.84
	PepTri	0.93	0.27	0.44 / 0.63	0.80 / 0.86
	PhyDiff (Ours)	**0.95**	**0.29**	0.58 / **0.67**	**0.82 / 0.87**
PepBDB	PepGLAD	0.92	0.24	**0.81** / 0.82	0.94 / 0.96
	PepFlow	0.67	0.20	0.58 / 0.71	0.74 / 0.82
	UniMoMo_{single}	0.90	0.20	**0.81** / 0.80	0.92 / 0.94
	PepTri (Ours)	0.93	**0.28**	0.63 / 0.70	0.89 / 0.95
	PhyDiff (Ours)	**0.96**	0.26	0.80 / **0.83**	**0.96 / 0.98**

Compared to de novo peptide design, practical scenarios often provide prior knowledge about key interaction residues. Following PepHAR, we consider a scaffold generation task in Appendix C.1.

4.2 Generalization to Unseen Targets

We first evaluate the cross-domain performance of PhyDiff on LNR, where the generative models are trained on PepBench and tested on an external target set. To further assess generalization to unseen targets, we construct an independent test set, **PepPro-96**, from Propedia v2.3 (Martins et al., 2021). Compared with the in-domain test set and LNR, PepPro-96 contains a larger proportion of receptors with low similarity to the PepBench training set under the specified sequence and structural similarity criteria. The generative models are evaluated on PepPro-96 without target-specific retraining. Detailed experimental settings and the construction procedure of PepPro-96 are provided in Appendix C.2.

As shown in Table 3, PhyDiff achieves higher DockQ, Success Rate, and SeqDiv than the compared baselines. Together with the LNR evaluation, these results provide additional evidence of PhyDiff's generalization capability under the evaluated cross-domain peptide design settings.

Table 3: Performance comparison on PepPro-96.

Method	DockQ	Success Rate	SeqDiv
PepGLAD	0.599	0.495	0.913
PepTri	0.594	0.472	0.872
PhyDiff (Ours)	**0.611**	**0.568**	**0.934**

4.3 Analysis of PhyDiff

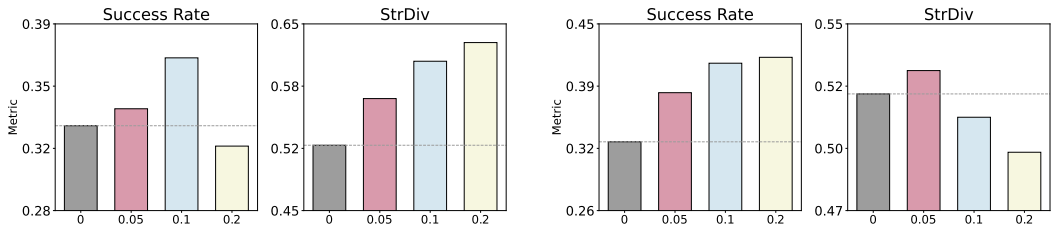
Ablation Studies. To evaluate the effect of different components in PhyDiff, we conduct ablation experiments on LNR dataset to measure their contributions to overall performance. Specifically, we compare model variants with and without FlexOOD and HHAG. The detailed ablation results are reported in Table 4. We further analyze the hyperparameter sensitivity, including the FlexOOD strength λ_{base} and HHAG guidance scales γ_1 and γ_2. In our experiments, we set $\gamma = \gamma_1 = \gamma_2$. As shown in Figure 2a, introducing an appropriate λ_{base} enables the model to explore diverse chemical

space and potentially identify high-affinity binding patterns for unseen targets. However, excessively strong OOD exploration introduces noise and may distort the sampling process.

For HHAG, the results shown in Figure 2b demonstrate that HHAG significantly improves binding performance and leads to the enhanced success rate. Nevertheless, overly strong guidance may force the generated peptides to overfit the predicted hotspot regions, thereby reducing generation diversity. We further compare FlexOOD with MOOD in Appendix C.6 to illustrate the effectiveness of incorporating peptide flexibility into exploration control. More ablation studies can be found in Appendix C.3.

Table 4: Ablation study of PhyDiff.

Component		Metrics			
FlexOOD	HHAG	Success Rate	DockQ	SeqDiv	StrDiv
✗	✗	0.33	0.57	0.89	0.52
✓	✗	0.37	0.55	**0.95**	**0.61**
✗	✓	<u>0.41</u>	<u>0.65</u>	0.86	0.51
✓	✓	**0.43**	**0.66**	**0.95**	<u>0.58</u>



(a) Model Performance under different λ_{base}.

(b) Model Performance under different γ.

Figure 2: Model Performance under different hyperparameters.

Case Studies We present the generated examples in Appendix F. These examples show that PhyDiff can generate diverse peptide candidates for a given target while maintaining high docking quality, highlighting its ability to design diverse, high-affinity binders. We further compare the hotspot regions of the native peptide with those of the generated peptides, as shown in Figure 3. We use red asterisks to indicate hotspot regions. The comparison shows that peptides generated by PhyDiff can effectively preserve key interactions around hotspot regions, thereby contributing to high binding affinity.

Model Efficiency. FlexOOD and HHAG introduce only modest additional computational time and memory overhead during sampling. As reported in Appendix C.7, the full model requires 1.30 s and 483 MB per measured batch setting, compared with 0.82 s and 259 MB for the baseline, while delivering substantial performance improvements.

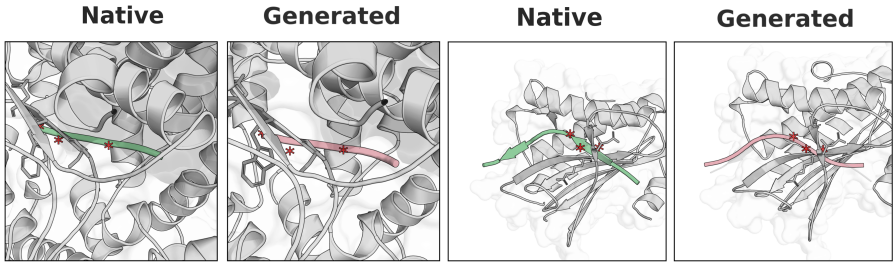


Figure 3: Hotspot regions of the native peptide and the generated peptides

5 Conclusion

In this work, we presented PhyDiff, a physically inspired diffusion framework for generalized peptide design. Motivated by the dynamic nature of protein–peptide recognition, PhyDiff explicitly incorporates peptide flexibility and hotspot-mediated interactions into the generative process. By coupling these two complementary mechanisms, PhyDiff is able to generate peptide binders that are both structurally diverse and affinity-enhanced. Extensive experiments on de novo peptide design and scaffold-constrained generation demonstrate that PhyDiff achieves competitive or superior performance across binding quality, diversity, and generalization metrics.

AI USAGE STATEMENT

We used generative AI tools primarily for language polishing, manuscript editing, and code optimization. The authors take full responsibility for the accuracy and integrity of the final manuscript, code, and reported results.

REPRODUCIBILITY STATEMENT

Appendix A provides detailed experimental settings. In addition, Appendix B includes the full details of the model configuration and training procedure.

REFERENCES

Rebecca F Alford, Andrew Leaver-Fay, Jeliazko R Jeliazkov, Matthew J O'Meara, Frank P DiMaio, Hahnbeom Park, Maxim V Shapovalov, P Douglas Renfrew, Vikram K Mulligan, Kalli Kappel, et al. The rosetta all-atom energy function for macromolecular modeling and design. *Journal of chemical theory and computation*, 13(6):3031–3048, 2017.

Sankar Basu and Björn Wallner. Dockq: a quality measure for protein-protein docking models. *PloS one*, 11(8):e0161879, 2016.

Gaurav Bhardwaj, Vikram Khipple Mulligan, Christopher D Bahl, Jason M Gilmore, Peta J Harvey, Olivier Cheneval, Garry W Buchko, Surya VSRK Pulavarti, Quentin Kaas, Alexander Eletsky, et al. Accurate de novo design of hyperstable constrained peptides. *Nature*, 538(7625):329–335, 2016.

David D Boehr, Ruth Nussinov, and Peter E Wright. The role of dynamic conformational ensembles in biomolecular recognition. *Nature chemical biology*, 5(11):789–796, 2009.

Andrew A Bogan and Kurt S Thorn. Anatomy of hot spots in protein interfaces. *Journal of molecular biology*, 280(1):1–9, 1998.

Patrick Bryant and Arne Elofsson. Evobind: in silico directed evolution of peptide binders with alphafold. *bioRxiv*, pages 2022–07, 2022.

Longxing Cao, Brian Coventry, Inna Goreshnik, Buwei Huang, William Sheffler, Joon Sung Park, Kevin M Jude, Iva Marković, Rameshwar U Kadam, Koen HG Verschueren, et al. Design of protein-binding proteins from the target structure alone. *Nature*, 605(7910):551–560, 2022.

Tim Clackson and James A Wells. A hot spot of binding energy in a hormone-receptor interface. *Science*, 267(5196):383–386, 1995.

Brian C Cunningham and James A Wells. High-resolution epitope mapping of hgh-receptor interactions by alanine-scanning mutagenesis. *Science*, 244(4908):1081–1085, 1989.

Justas Dauparas, Ivan Anishchenko, Nathaniel Bennett, Hua Bai, Robert J Ragotte, Lukas F Milles, Basile IM Wicky, Alexis Courbet, Rob J de Haas, Neville Bethel, et al. Robust deep learning–based protein sequence design using proteinmpnn. *Science*, 378(6615):49–56, 2022.

Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.

Richard Evans, Michael O'neill, Alexander Pritzel, Natasha Antropova, Andrew Senior, Tim Green, Augustin Žídek, Russ Bates, Sam Blackwell, Jason Yim, et al. Protein complex prediction with alphafold-multimer. *biorxiv*, pages 2021–10, 2021.

Katherine Henzler-Wildman and Dorothee Kern. Dynamic personalities of proteins. *Nature*, 450 (7172):964–972, 2007.

Po-Ssu Huang, Scott E Boyken, and David Baker. The coming of age of de novo protein design. *Nature*, 537(7620):320–327, 2016.

Wengong Jin, Jeremy Wohlwend, Regina Barzilay, and Tommi Jaakkola. Iterative refinement graph neural network for antibody sequence-structure co-design. *arXiv preprint arXiv:2110.04624*, 2021.

Isak Johansson-Åkhe and Björn Wallner. Improving peptide-protein docking with alphafold-multimer using forced sampling. *Frontiers in bioinformatics*, 2:959160, 2022.

John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, et al. Highly accurate protein structure prediction with alphafold. *nature*, 596(7873):583–589, 2021.

Wolfgang Kabsch. A solution for the best rotation to relate two sets of vectors. *Foundations of Crystallography*, 32(5):922–923, 1976.

Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=k7FuTOWMOc7.

Allan A Kaspar and Janice M Reichert. Future directions for peptide therapeutics development. *Drug discovery today*, 18(17-18):807–817, 2013.

Xiangzhe Kong, Wenbing Huang, and Yang Liu. End-to-end full-atom antibody design. *arXiv preprint arXiv:2302.00203*, 2023.

Xiangzhe Kong, Yinjun Jia, Wenbing Huang, and Yang Liu. Full-atom peptide design with geometric latent diffusion. *Advances in Neural Information Processing Systems*, 37:74808–74839, 2024.

Xiangzhe Kong, Rui Jiao, Haowei Lin, Ruihan Guo, Wenbing Huang, Wei-Ying Ma, Zihua Wang, Yang Liu, and Jianzhu Ma. Peptide design through binding interface mimicry with pepmimic. *Nature Biomedical Engineering*, Oct 2025a. ISSN 2157-846X. doi: 10.1038/s41551-025-01507-4. URL https://doi.org/10.1038/s41551-025-01507-4.

Xiangzhe Kong, Zishen Zhang, Ziting Zhang, Rui Jiao, Jianzhu Ma, Wenbing Huang, Kai Liu, and Yang Liu. Unimomo: Unified generative modeling of 3d molecules for de novo binder design. In *Forty-second International Conference on Machine Learning*, 2025b. URL https://openreview.net/forum?id=KUN7A7Okb6.

Petr Kouba, Joan Planas-Iglesias, Jiri Damborsky, Jiri Sedlar, Stanislav Mazurenko, and Josef Sivic. Learning to engineer protein flexibility. *arXiv preprint arXiv:2412.18275*, 2024.

Andrew Leaver-Fay, Michael Tyka, Steven M Lewis, Oliver F Lange, James Thompson, Ron Jacak, Kristian W Kaufman, P Douglas Renfrew, Colin A Smith, Will Sheffler, et al. Rosetta3: an object-oriented software suite for the simulation and design of macromolecules. In *Methods in enzymology*, volume 487, pages 545–574. Elsevier, 2011.

Seul Lee, Jaehyeong Jo, and Sung Ju Hwang. Exploring chemical space with score-based out-of-distribution generation. In *International Conference on Machine Learning*, pages 18872–18892. PMLR, 2023.

Jiahan Li, Tong Chen, Shitong Luo, Chaoran Cheng, Jiaqi Guan, Ruihan Guo, Sheng Wang, Ge Liu, Jian Peng, and Jianzhu Ma. Hotspot-driven peptide design via multi-fragment autoregressive extension. *arXiv preprint arXiv:2411.18463*, 2024a.

Jiahan Li, Chaoran Cheng, Zuofan Wu, Ruihan Guo, Shitong Luo, Zhizhou Ren, Jian Peng, and Jianzhu Ma. Full-atom peptide design based on multi-modal flow matching. *arXiv preprint arXiv:2406.00735*, 2024b.

Haitao Lin, Odin Zhang, Huifeng Zhao, Dejun Jiang, Lirong Wu, Zicheng Liu, Yufei Huang, and Stan Z Li. Ppflow: Target-aware peptide design with torsional flow matching. *arXiv preprint arXiv:2405.06642*, 2024.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

Shitong Luo, Yufeng Su, Xingang Peng, Sheng Wang, Jian Peng, and Jianzhu Ma. Antigen-specific antibody design and optimization with diffusion-based generative models for protein structures. *Advances in Neural Information Processing Systems*, 35:9754–9767, 2022.

Pedro M Martins, Lucianna H Santos, Diego Mariano, Felippe C Queiroz, Luana L Bastos, Isabela de S Gomes, Pedro HC Fischer, Rafael EO Rocha, Sabrina A Silveira, Leonardo HF de Lima, et al. Propedia: a database for protein–peptide identification based on a hybrid clustering algorithm. *BMC bioinformatics*, 22(1):1, 2021.

Milot Mirdita, Konstantin Schütze, Yoshitaka Moriwaki, Lim Heo, Sergey Ovchinnikov, and Martin Steinegger. Colabfold: making protein folding accessible to all. *Nature methods*, 19(6):679–682, 2022.

Ngoc-Quang Nguyen, Jaeyoon Jung, Seijung Kim, Sunkyu Kim, and Jaewoo Kang. Peptri: Tri-guided all-atom diffusion for peptide design via physics, evolution, and mutual information. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=yQlTgHo1um.

Martin Pacesa, Lennart Nickel, Christian Schellhaas, Joseph Schmidt, Ekaterina Pyatova, Lucas Kissling, Patrick Barendse, Jagrity Choudhury, Srajan Kapoor, Ana Alcaraz-Serna, et al. Bindcraft: one-shot design of functional protein binders. *BioRxiv*, pages 2024–09, 2024.

Saro Passaro, Gabriele Corso, Jeremy Wohlwend, Mateo Reveiz, Stephan Thaler, Vignesh Ram Somnath, Noah Getz, Tally Portnoi, Julien Roy, Hannes Stark, et al. Boltz-2: Towards accurate and efficient binding affinity prediction. *BioRxiv*, 2025.

Evangelia Petsalaki and Robert B Russell. Peptide-mediated interactions in biological systems: new discoveries and applications. *Current opinion in biotechnology*, 19(4):344–350, 2008.

Joan Planas-Iglesias, Filip Opaleny, Pavol Ulbrich, Jan Stourac, Zainab Sanusi, Gaspar P Pinto, Andrea Schenkmayerova, Jan Byska, Jiri Damborsky, Barbora Kozlikova, et al. Loopgrafter: a web tool for transplanting dynamical loops for protein engineering. *Nucleic Acids Research*, 50 (W1):W465–W473, 2022.

Jorge Roel-Touris, Marta Nadal, and Enrique Marcos. Single-chain dimers from de novo immunoglobulins as robust scaffolds for multiple binding loops. *Nature communications*, 14(1):5939, 2023.

Vıctor Garcia Satorras, Emiel Hoogeboom, and Max Welling. E (n) equivariant graph neural networks. In *International conference on machine learning*, pages 9323–9332. PMLR, 2021.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=PxTIG12RRHS.

Tomer Tsaban, Julia K Varga, Orly Avraham, Ziv Ben-Aharon, Alisa Khramushin, and Ora Schueler-Furman. Harnessing protein folding neural networks for peptide–protein docking. *Nature communications*, 13(1):176, 2022.

Yann Vander Meersche, Gabriel Cretin, Aria Gheeraert, Jean-Christophe Gelly, and Tatiana Galochkina. Atlas: protein flexibility description from atomistic molecular dynamics simulations. *Nucleic acids research*, 52(D1):D384–D392, 2024.

Vsevolod Viliuga, Leif Seute, Nicolas Wolf, Simon Wagner, Arne Elofsson, Jan Stühmer, and Frauke Gräter. Flexibility-conditioned protein structure design with flow matching. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=890gHX7ieS.

Joseph L Watson, David Juergens, Nathaniel R Bennett, Brian L Trippe, Jason Yim, Helen E Eisenach, Woody Ahern, Andrew J Borst, Robert J Ragotte, Lukas F Milles, et al. Broadly applicable and accurate protein design by integrating structure prediction networks and diffusion generative models. *BioRxiv*, pages 2022–12, 2022.

Zeyu Wen, Jiahua He, Huanyu Tao, and Sheng-You Huang. Pepbdb: a comprehensive structural database of biological peptide–protein interactions. *Bioinformatics*, 35(1):175–177, 2019.

Jeremy Wohlwend, Gabriele Corso, Saro Passaro, Noah Getz, Mateo Reveiz, Ken Leidal, Wojtek Swiderski, Liam Atkinson, Tally Portnoi, Itamar Chinn, et al. Boltz-1 democratizing biomolecular interaction modeling. *BioRxiv*, pages 2024–11, 2025.

Peter E Wright and H Jane Dyson. Intrinsically disordered proteins in cellular signalling and regulation. *Nature reviews Molecular cell biology*, 16(1):18–29, 2015.

Kevin E Wu, Kevin K Yang, Rianne Van den Berg, Sarah Alamdari, James Y Zou, Alex X Lu, and Ava P Amini. Protein structure generation via folding diffusion. *Nature communications*, 15(1): 1059, 2024.

Jason Yim, Brian L Trippe, Valentin De Bortoli, Emile Mathieu, Arnaud Doucet, Regina Barzilay, and Tommi Jaakkola. Se (3) diffusion model with application to protein backbone generation. *arXiv preprint arXiv:2302.02277*, 2023.

Vinicius Zambaldi, David La, Alexander E Chu, Harshnira Patani, Amy E Danson, Tristan OC Kwan, Thomas Frerix, Rosalia G Schneider, David Saxton, Ashok Thillaisundaram, et al. De novo design of high-affinity protein binders with alphaproteo. *arXiv preprint arXiv:2409.08022*, 2024.

A Experimental Details

A.1 Datasets

Following prior work, we use MMseqs2-based clustering to partition the dataset into training, validation, and test sets while avoiding target overlap across splits. In particular, the LNR dataset was used to evaluate cross-target generalization, and the PepBDB data were similarly split based on clustering results to ensure non-redundancy among samples. The PepBench dataset contains 6,105 non-redundant protein–peptide complexes, and the LNR dataset contains 93 complexes. In the PepBDB experiment, we follow PepTri and use the same training, validation, and test splits to ensure a fair comparison. All experiments are conducted on NVIDIA RTX 4090 GPUs with 24 GB each. We report the average results over three independent runs.

A.2 Baselines

We compare PhyDiff against both diffusion-based and hallucination-based peptide design methods:

- **PepGLAD**: Full-atom peptide design with geometric latent diffusion. Employs an SE(3)-equivariant VAE to encode peptide structures into a latent space and a DDPM denoiser conditioned on the receptor binding site. Receptor-specific affine transformations enable generalization across diverse binding scenarios.
- **PepFlow**: Multi-modal flow matching for full-atom peptide design. Factorizes the generation into backbone torsion angles and side-chain conformations, using separate flow matching processes for each modality.
- **PepHAR**: Hotspot-driven peptide design via multi-fragment autoregressive extension. Conditions generation on predicted hotspot residues and extends the peptide fragment by fragment.
- **UniMoMo**: Unified generative modeling of 3D molecules for de novo binder design, combining protein and small-molecule generation in a single framework.
- **PepTri**: Tri-guided all-atom diffusion integrating physics, evolutionary, and mutual information guidance during both training and sampling. Uses an SE(3)-equivariant latent VAE with BLOSUM-derived evolutionary priors and OpenMM force field coupling.
- **BindCraft**: One-shot design of functional protein binders via hallucination. Iteratively optimizes sequences using AlphaFold2 confidence metrics (pLDDT, pAE, pTM) as surrogate objectives, requiring no task-specific training.

A.3 Metrics

We adopt the evaluation protocol from PepTri (Nguyen et al., 2026) and PepGLAD (Kong et al., 2024), employing a comprehensive set of metrics that capture both structural quality and functional relevance.

Binding quality metrics. **Success Rate (SR)**: The fraction of generated peptides with thermodynamically stable binding, defined as Rosetta binding free energy $\Delta G < -5$ REU. **Binding Free Energy (ΔG)**: Computed via Rosetta to evaluate the binding affinity of the generated candidates. **DockQ**: A composite docking quality score combining interface RMSD (iRMSD), ligand RMSD (LRMSD), and fraction of native contacts (fnat) Basu and Wallner (2016).

Geometrical metrics. **Contact_F1** measures the accuracy of predicted inter-residue contacts using the harmonic mean of precision and recall. **Local RMSD** evaluates regional structural accuracy through a sliding-window analysis, providing a more fine-grained assessment than global RMSD. **GDT_TS** (Global Distance Test–Total Score) measures the fraction of C_α atoms that fall within distance cutoffs of 1, 2, 4, and 8 Å after optimal superposition.

Diversity and novelty metrics. **Sequence Diversity (SeqDiv)**: Measured by the number of unique sequence clusters at edit-distance threshold 0.4 via hierarchical clustering, normalized by the number

of samples. **Structure Diversity (StrDiv)**: Measured by the number of unique structural clusters at C_α RMSD threshold 4.0 Å via hierarchical clustering, normalized by the number of samples. **Sequence Validity (SeqValid)**: The proportion of generated sequences containing only valid amino acid identities (no unknown or invalid tokens). **Consistency**: Sequence–structure coherence measured via Cramér's V statistic between sequence and structure cluster assignments, capturing whether similar sequences correspond to similar structures.

B Model and Training Details

PhyDiff consists of four components: (1) an SE(3)-equivariant VAE that maps peptide structures to a compact latent space, (2) a denoising network that generates latent embeddings conditioned on the receptor, (3) a flexibility predictor for FlexOOD, and (4) a hotspot/affinity predictor for HHAG. We detail each below.

B.1 AutoEncoder (VAE)

Architecture. The encoder and decoder each consist of 3 layers of Adaptive Multi-Channel E(n) Equivariant Graph Neural Network (Kong et al., 2023). Each layer performs equivariant message passing that jointly updates node features and coordinates while preserving SE(3) equivariance. The multi-channel design handles all-atom representations by treating each atom type as a separate coordinate channel ($n_{\text{channel}} = 14$). The channel-level radial features are computed from pairwise inter-channel distances, weighted by atom channel masks, and mapped through a linear layer ($d^2_{\text{attr}} \to d_{\text{hidden}}$). Coordinate updates employ a RollerPooling operator that adaptively averages displacement messages based on the number of valid atom channels per residue. The detailed architecture of VAE is shown in Table 5.

Table 5: VAE architecture (encoder and decoder share the same structure).

Component	Specification
Embedding dimension	128
Hidden dimension	128
Latent sequence dim (d_h)	8
Latent structure channels	1 ($c_i \times 3$ per residue)
Atom channels (n_{channel})	14 (all-atom)
Channel embedding dim	32
RBF bins	32, cutoff 3.0 Å (in latent space)
Dropout	0.1

Pretraining. The VAE is pretrained on approximately 70,645 unlabeled protein fragments from ProtFrag following PepGLAD, then fine-tuned on the target dataset (PepBench or PepBDB). Training uses AdamW (lr $= 1 \times 10^{-4}$) with ReduceLROnPlateau (factor 0.8, patience 5, min_lr 5×10^{-6}) for 100 epochs. A mask ratio of 25% is used to randomly mask peptide residues during training.

B.2 Denoising Network

The denoising network f_θ operates in the latent space of the VAE. Following EDM, we perturb the clean latent embedding as $\mathcal{G}_t^z = \mathcal{G}_0^z + \sigma_t \epsilon$, $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, and train f_θ to directly predict the clean latent: $\hat{\mathcal{G}}_0^z = f_\theta(\mathcal{G}_t^z, \sigma_t)$.

Architecture. The denoiser shares the same design as the VAE encoder. It consists of 3 message-passing layers plus 1 output layer, with hidden dimension 128 and channel embedding dimension 32. The input node features are constructed by concatenating: (i) the noisy latent embedding $z_t^h \in \mathbb{R}^8$, (ii) a noise-level encoding $[\sigma_t, \sin(\sigma_t), \cos(\sigma_t)] \in \mathbb{R}^3$, and (iii) a learnable absolute position embedding $\in \mathbb{R}^8$, yielding an input dimension of 19. The network outputs the predicted clean latent $\hat{\mathcal{G}^z}_0 = (\hat{z}_0^h, \hat{z}_0^x)$ for both sequence and structure components.

Training. Following EDM, we use a geometric noise schedule with $T = 100$ discretization steps $\sigma_n = \left(\sigma_{\max}^{1/\rho} + \frac{n}{T}\left(\sigma_{\min}^{1/\rho} - \sigma_{\max}^{1/\rho}\right)\right)^{\rho}$, $\sigma_{\max} = 80, \sigma_{\min} = 0.002, \rho = 7$. This schedule concentrates more steps at low noise levels where fine structural details are resolved, while allocating fewer steps at high noise levels where the distribution is close to Gaussian. The denoiser is trained with AdamW (lr $= 1 \times 10^{-4}$) for 500 epochs. During training, noise levels σ are sampled log-uniformly from $[\sigma_{\min}, \sigma_{\max}]$, and the denoising objective is:

$$\mathcal{L}_{\text{LDM}} = \mathbb{E}_{\mathcal{G}_0^z, \sigma}\left[w_h \|\hat{z}_0^h - z_0^h\|_2^2 + w_x \|\hat{z}_0^x - z_0^x\|_2^2\right]. \tag{15}$$

To ensure conformational validity during exploration, we adopt physical losses from AlphaFold2 to penalize implausible geometries. These include bond length violations, bond angle deviations, and steric clashes, applied to the decoded peptide structure $\hat{\mathcal{G}}_t^{\text{pep}} = D(f_\theta(\mathcal{G}_t^z, \sigma_t), \mathcal{G}^{\text{rec}})$ during training.

B.3 Flexibility Predictor

The flexibility predictor $\mathcal{E}_\xi$ estimates the overall flexibility ξ of a peptide from its structural graph. We train $\mathcal{E}_\xi$ on protein structures from the ATLAS database, which provides RMSF values computed from atomistic molecular dynamics simulations. The detailed architecture of $\mathcal{E}_\xi$ is included in Table 6.

Architecture. $\mathcal{E}_\xi$ is a 5-layer SE(3)-equivariant attention MPNN. It takes C_α coordinates and one-hot amino acid features (20 dims) as input, and constructs a k-NN graph (k=12). Node features are embedded via a 3-layer MLP. Edge features consist of 16 Gaussian RBF-encoded pairwise distances. Each attention MPNN layer performs: (i) pre-LayerNorm multi-head attention (H=4, d_{head}=64) over the k-nearest neighbors, where attention weights are computed as $\alpha_{ij}^h = \text{softmax}_j\left(\frac{\mathbf{Q}_i^h \cdot \mathbf{K}_j^h}{\sqrt{d_{\text{head}}}} + b_{ij}^h\right)$ with edge bias $\mathbf{b}_{ij} = \mathbf{W}_b(\text{MLP}_{\text{edge}}(\mathbf{e}_{ij}))$ projected from RBF-encoded edge features; (ii) an equivariant coordinate update via a Tanh-gated MLP that produces scalar weights modulating coordinate displacements: $\mathbf{x}_i \leftarrow \mathbf{x}_i + \sum_j \phi(\mathbf{e}_{ij}) \cdot (\mathbf{x}_i - \mathbf{x}_j)$; and (iii) a gated residual update $\mathbf{h}_i \leftarrow \mathbf{h}_i + \sigma(\mathbf{W}_g[\mathbf{h}_i \| \mathbf{m}_i]) \odot \mathbf{m}_i$ followed by a feed-forward network with pre-LayerNorm residual connection. After message passing, the per-residue output head applies a 3-layer MLP followed by a scaled sigmoid $\xi_i = \xi_{\max}\,\sigma(\cdot)$. For global flexibility prediction, node embeddings are aggregated via masked mean pooling and mapped through a 2-layer MLP:

$$\xi = \mathcal{E}_\xi(\mathcal{G}^{\text{pep}}) = \xi_{\max}\,\sigma(\text{MLP}_{\text{global}}(\text{Pool}(\{\mathbf{h}_i\}))). \tag{16}$$

Table 6: Flexibility predictor ($\mathcal{E}_\xi$) architecture.

Component	**Specification**
Input features	One-hot amino acid (20 dims)
Node dimension	128
Hidden dimension	256
Number of layers	5
Attention heads	4 (head dim = 64)
Radial basis functions	16 bins over $[0, 20]$ Å
k-NN graph	$k = 12$
Coordinate update	Tanh-gated MLP ($256 \to 128 \to 1$)
$\xi_{\max}$	5.0 Å
Dropout	0.1

B.4 HHAG Predictor

The hotspot and affinity predictor $\mathcal{E}_G$ jointly predicts per-residue hotspot probabilities and global binding affinity from the receptor–peptide complex. Following ProteinMPNN, we represent the complex $(\mathcal{G}^{\text{rec}}, \mathcal{G}^{\text{pep}})$ as a k-NN graph and use a unified attention MPNN $\mathcal{E}_G$ to produce shared residue-level embeddings from which both task-specific outputs are derived. Details are included in Table 7.

Architecture. $\mathcal{E}_G$ is a 5-layer attention MPNN. Receptor and peptide residues are embedded separately through 3-layer MLPs. The node embeddings are concatenated to form a unified graph $\{\mathbf{h}_i\}_{i\in\mathcal{G}^{\text{rec}}\cup\mathcal{G}^{\text{pep}}}$. A k-NN graph (k=10) is constructed over all residue C_α positions. Each layer follows the same attention MPNN design as the flexibility predictor, with wider dimensions: multi-head attention (H=6, d_{head}=64) over k-nearest neighbors with edge features constructed from 24 Gaussian RBF-encoded distances concatenated with the cross-chain indicator, projected to the hidden dimension 256. Messages are aggregated via gated residual connections, followed by a feed-forward network with pre-LayerNorm.

Table 7: HHAG predictor ($\mathcal{E}_G$) architecture.

Component	Specification
Input features	One-hot amino acid (20 dims) + type embedding
Node dimension	128
Hidden dimension	256
Number of layers	5
Attention heads	6 (head dim = 64)
Radial basis functions	24 bins over $[0, 20]$ Å
k-NN graph	$k = 10$
Dropout	0.1

Output heads. $\mathcal{E}_G$ produces embeddings $\{\mathbf{h}_i\}$ for all residues, from which two task-specific outputs are derived: 1) **Hotspot head**: A 3-layer MLP is applied per peptide residue to produce logits $\ell_i = \text{MLP}_{\text{hot}}(\mathbf{h}_i)$, followed by softmax normalization over peptide residues to obtain hotspot probabilities $\hat{p}_i = \frac{\exp(\ell_i)}{\sum_{j\in V^{\text{pep}}}\exp(\ell_j)}, \quad i \in V^{\text{pep}}$. 2) **Affinity head**: Receptor and peptide embeddings are pooled separately via masked mean pooling and concatenated. The concatenated representation is processed by a 3-layer MLP to predict the scalar binding affinity $\hat{s} = \text{MLP}_{\text{aff}}(\text{Pool}(\{\mathbf{h}_i\}_{i\in\mathcal{G}^{\text{rec}}})\|\text{Pool}(\{\mathbf{h}_j\}_{j\in\mathcal{G}^{\text{pep}}}))$.

Training. We use AdamW (lr $= 3\times10^{-4}$, weight decay 10^{-4}) with cosine annealing over 60 epochs. Coordinate noise augmentation ($\sigma = 0.1$ Å) is applied during training. Gradient norms are clipped to 5.0, and early stopping with patience 10 is used. Sharing the same graph encoder for both heads encourages representations that capture both residue-level interaction patterns and global binding strength.

B.5 Prediction Performance of Different Predictors

FlexOOD and HHAG rely on learned predictors to provide sampling-time guidance, so we separately evaluate the quality of these signals. For flexibility prediction, we follow prior flexibility-modeling practice and train on the ATLAS database, which provides RMSF labels derived from molecular dynamics trajectories. During diffusion sampling, we do not apply the flexibility predictor directly to the noisy latent/intermediate conformation. Instead, at each step we first use the denoiser to estimate the clean t=0 state, decode this estimate, and evaluate flexibility on the resulting denoised structure. This reduces the distribution mismatch between noisy diffusion states and the native-like structural inputs used to train the predictor.

Table 8: Standalone prediction performance of the guidance models used by FlexOOD and HHAG.

Metric	Value
Flexibility MAE (FlexOOD) (↓)	0.17
Affinity MAE (HHAG) (↓)	0.21
Hotspot-region Accuracy (HHAG) (↑)	0.88

Compared with modeling the full high-dimensional chemical space, the scalar flexibility and affinity targets and the residue-level hotspot classification task are more constrained prediction problems. Their standalone performance is reported in Table 8. These results support the use of the predictors as guidance signals, while they should be interpreted as learned approximations of physically meaningful properties rather than exact physical simulators.

Algorithm 1 PhyDiff Sampling with FlexOOD and HHAG Guidance

Require: Receptor $\mathcal{G}^{\text{rec}}$, steps T, noise schedule $\{\sigma_t\}_{t=0}^{T}$, FlexOOD $(\lambda_{\text{base}},\, \xi_0)$, HHAG $(\gamma_1,\, \gamma_2)$
Ensure: Generated peptide $\hat{\mathcal{G}}^{\text{pep}}$

$\mathcal{G}_T^z \sim \mathcal{N}(\mathbf{0},\, \sigma_T^2\, \mathbf{I})$ ▷ Sample from prior at $\sigma_{\max}$
for $t = T$ **down to** 1 **do**
  // EDM denoiser prediction
  $\hat{\mathcal{G}}_0^z \leftarrow f_\theta\big(\mathcal{G}_t^z,\, \sigma_t\big)$ ▷ Predict denoised latent $D_\theta(\mathbf{x};\, \sigma_t)$
  $\hat{\mathcal{G}}_t^{\text{pep}} \leftarrow D\big(\hat{\mathcal{G}}_0^z,\, \mathcal{G}^{\text{rec}}\big)$ ▷ Decode for guidance computation
  // FlexOOD: flexibility-conditioned exploration
  $\xi \leftarrow \mathcal{E}_\xi\big(\hat{\mathcal{G}}_t^{\text{pep}}\big)$ ▷ Predict flexibility score
  $\alpha_{t/T} \leftarrow \dfrac{1 + \cos\big(\pi \cdot (1 - t/T)\,\big)}{2}$ ▷ Cosine annealing: large at high σ_t, vanishes near σ_0
  $\lambda_t \leftarrow \lambda_{\text{base}} \cdot \alpha_{t/T} \cdot \eta(\xi)$ ▷ Time- and flex-adaptive weight
  // HHAG: hierarchical affinity guidance
  $\mathbf{g}_t \leftarrow \gamma_1\, \nabla_{\mathcal{G}_t^z} \mathcal{E}_G\big(\mathcal{G}^{\text{rec}},\, \hat{\mathcal{G}}_t^{\text{pep}}\big) \;-\; \gamma_2\, \nabla_{\mathcal{G}_t^z} \mathcal{L}_{\text{hot}}\big(\hat{\mathcal{G}}_t^{\text{pep}}\big)$
  $\mathbf{g}_t \leftarrow \mathbf{g}_t \cdot \min\big(1,\; c \,/\, \|\mathbf{g}_t\|\big)$ ▷ Gradient clipping ($c{=}6.0$)
  // Probability-flow ODE Euler step: $\sigma_t \to \sigma_{t-1}$
  $\mathbf{d}_t \leftarrow \big(\hat{\mathcal{G}}_0^z - \mathcal{G}_t^z\big) \,/\, \sigma_t$ ▷ Denoising direction (noisy $\to$ clean)
  $\mathbf{d}_t \leftarrow \big(1 - \sqrt{\lambda_t}\big)\, \mathbf{d}_t \;+\; \sigma_t\, \mathbf{g}_t$ ▷ FlexOOD attenuation + HHAG guidance
  $\mathcal{G}_{t-1}^z \leftarrow \mathcal{G}_t^z + (\sigma_t - \sigma_{t-1}) \cdot \mathbf{d}_t$ ▷ Euler update ($\sigma_t - \sigma_{t-1} > 0$)
end for
$\hat{\mathcal{G}}^{\text{pep}} \leftarrow D\big(\hat{\mathcal{G}}_0^z,\, \mathcal{G}^{\text{rec}}\big)$ ▷ Final decode
return $\hat{\mathcal{G}}^{\text{pep}}$

C Additional Experiments

C.1 Experiments on Scaffold Generation

Compared to de novo peptide design, practical scenarios often provide prior knowledge about key interaction residues. Following PepHAR, we consider a scaffold generation task in which hotspot residues are fixed, and the model is required to complete the entire peptide. We identify hotspots using the procedure in Section 3.3, where Rosetta computes residue-level energies for selection. In this setting, we replace the predicted $\hat{\mathcal{G}}^{\text{hot}}$ with ground-truth hotspots. Results on the LNR dataset are shown in Table 9, where we set $K = 3$ and report results without relaxation.

Table 9: Experiments on the scaffold generation tasks.

Method	Success Rate	ΔG	SeqValid	StrDiv
PepFlow	0.32	-16.87	0.28	0.40
PepGLAD	0.31	-17.35	0.27	0.52
PepHAR	0.37	-16.91	0.27	**0.61**
PepTri	<u>0.38</u>	<u>-19.18</u>	<u>0.30</u>	0.45
PhyDiff	**0.45**	**-20.98**	**0.32**	<u>0.59</u>

PhyDiff demonstrates strong performance on both diversity and binding metrics, further validating its effectiveness on the scaffold generation tasks. Moreover, compared with the de novo design setting, introducing ground-truth hotspot information leads to improved binding performance for PhyDiff, further underscoring the importance of hotspot regions for overall binding quality.

C.2 Additional Experiments on Generalization to Unseen Targets

To further illustrate the effectiveness and generalization ability of our method in unseen scenarios, we additionally conducted a cross-domain evaluation. Specifically, we constructed the **PepPro-96** benchmark from **Propedia v2.3**. Starting from 19,813 entries, we retained only complexes with peptide lengths between 4 and 25 amino acids and receptor lengths between 30 and 1200 amino acids, containing only standard residues. We further removed any PDB IDs that appeared in PepBench, PepBDB, or LNR, ensuring that all targets in PepPro-96 were unseen during training and benchmark construction. Finally, this process yielded 96 peptide–protein complexes. All models were trained on PepBench and evaluated on PepPro-96.

To quantify the degree of distribution shift between PepPro-96 and the training set, we measured the similarity between each test receptor and all training receptors along two axes, retaining the maximum similarity in each case. Specifically, we considered: (i) **sequence similarity**, computed using Smith–Waterman local alignment with BLOSUM62, where sequence identity was normalized by the length of the shorter chain; and (ii) **structural similarity**, computed using TM-align on C_α atoms, where the TM-score was normalized by the query chain. We define **near homologs** as samples with maximum sequence similarity ≥ 0.40 or maximum TM-score ≥ 0.80 to any training receptor; the remaining targets are regarded as **remote homologs**, representing more strongly shifted cases. As shown in the table, the newly constructed PepPro-96 exhibits a more severe distribution shift than LNR in terms of OOD difficulty, and both LNR and PepPro-96 display substantially stronger OOD characteristics than the in-domain test set.

Table 10: Distribution shift comparison across datasets.

Data	Near-homolog	Remote-homolog
In-domain test data	58.5%	41.5%
LNR-93	40.9%	59.1%
PepPro-96	27.0%	73.0%

The performance of different methods on PepPro-96 is summarized below, further demonstrating the generalization capability of **PhyDiff** and its effectiveness in unseen scenarios.

Table 11: Performance comparison on PepPro-96.

Method	DockQ	Success Rate	SeqDiv
PepGLAD	0.599	0.495	0.913
PepTri	0.594	0.472	0.872
PhyDiff (Ours)	**0.611**	**0.568**	**0.934**

C.3 More Ablation Studies of PhyDiff

Ablation studies on the cosine annealing strategy. The cosine annealing schedule in FlexOOD is inspired by the time-dependent sampling schedule used in diffusion models. Intuitively, stronger exploration is more suitable in the early stage of sampling, when the generated sample is still far from the data manifold, whereas the sampling trajectory should gradually converge to physically plausible structures near the end. To validate this design, we further compared the full method with a variant that removes cosine annealing and directly applies flexibility-based exploration throughout the entire sampling process.

Ablation studies on the ξ_0 of FlexOOD. We intentionally avoid introducing many additional tunable parameters. The base scale parameter λ_{base} controls the overall strength of FlexOOD and serves to absorb the global magnitude of exploration. We further vary the midpoint ξ_0, which controls where the sigmoid flexibility response changes most rapidly. As shown in Table 13, $\xi_0 = 2$ provides the best balance between binding quality and diversity, while $\xi_0 = 3$ favors stronger diversity at a modest cost in binding metrics.

Table 12: Ablation study on the cosine annealing strategy in FlexOOD.

Methods	DockQ	Success Rate	SeqDiv
without FlexOOD	0.57	0.33	0.89
FlexOOD without annealing	0.51	0.32	0.94
full FlexOOD	0.55	0.37	0.95

Table 13: Ablation study on the flexibility midpoint ξ_0 in FlexOOD.

ξ_0	DockQ	Success Rate	SeqDiv
1	0.63	0.42	0.94
2	**0.66**	**0.43**	0.95
3	0.64	0.41	**0.97**

C.4 More Results on the Geometrical and Other Metrics.

We additionally report geometrical and other metrics of different models on PepBench and PepBDB, including Contact_F1, Local RMSD, and GDT_TS. As shown in Table 14, our method achieves consistently strong performance across three metrics, further demonstrating its effectiveness.

Table 14: Geometrical metrics on PepBench and PepBDB (*without relaxation / with relaxation*).

Dataset	**Method**	**Contact_F1 (↑)**	**Local RMSD (Å) (↓)**	**GDT-TS (↑)**
PepBench	PepGLAD	0.80 ± 0.26 / 0.80 ± 0.24	1.22 ± 0.49 / 1.21 ± 0.43	0.72 ± 0.20 / 0.73 ± 0.19
	PepFlow	0.80 ± 0.25 / 0.82 ± 0.19	1.07 ± 0.40 / 1.06 ± 0.48	0.74 ± 0.20 / 0.74 ± 0.21
	UniMoMo$_{single}$	0.61 ± 0.35 / 0.67 ± 0.32	1.98 ± 1.64 / 1.37 ± 1.27	0.62 ± 0.27 / 0.62 ± 0.27
	PepTri	0.83 ± 0.23 / 0.84 ± 0.20	1.18 ± 0.42 / 1.10 ± 0.46	0.75 ± 0.18 / 0.76 ± 0.19
	PhyDiff(Ours)	**0.85 ± 0.24 / 0.87 ± 0.21**	**1.03 ± 0.38 / 1.05 ± 0.34**	**0.77 ± 0.23 / 0.77 ± 0.22**
PepBDB	PepGLAD	0.52 ± 0.36 / 0.60 ± 0.35	1.92 ± 1.83 / 1.46 ± 0.48	0.52 ± 0.26 / 0.60 ± 0.25
	PepFlow	0.71 ± 0.30 / 0.72 ± 0.28	**1.27 ± 0.45** / 1.35 ± 0.44	0.65 ± 0.21 / 0.63 ± 0.22
	UniMoMo$_{single}$	0.48 ± 0.36 / 0.55 ± 0.35	2.78 ± 1.50 / 1.52 ± 0.65	0.48 ± 0.24 / 0.56 ± 0.24
	PepTri	0.75 ± 0.29 / 0.77 ± 0.26	1.34 ± 1.43 / 1.29 ± 0.41	**0.66 ± 0.20** / 0.67 ± 0.20
	PhyDiff(Ours)	**0.78 ± 0.31 / 0.79 ± 0.25**	1.30 ± 0.89 / **1.26 ± 0.47**	0.65 ± 0.19 / **0.69 ± 0.22**

Orthogonal geometric evaluation. Because the HHAG affinity predictor is trained using Rosetta-derived labels while ΔG and Success Rate also rely on Rosetta, we additionally report geometry-based metrics that do not use the Rosetta ΔG criterion. Following the evaluation protocol used by UniMoMo, we consider complex RMSD (C-RMSD), ligand RMSD (L-RMSD), intra-chain and inter-chain clash rates, and Jensen–Shannon divergence of backbone and side-chain geometry distributions. As shown in Table 15, PhyDiff remains competitive across these orthogonal metrics, which reduces the concern that the observed improvements arise solely from optimizing toward the same Rosetta-based oracle used for evaluation.

C.5 Comparison between Hallucination-based BindCraft

Hallucination-based approaches use powerful folding or inverse-folding models, such as AlphaFold and ProteinMPNN, to iteratively optimize designed binders and have demonstrated strong performance in protein binder design. However, peptide design poses additional challenges due to limited training data and the short length of peptides, which are typically fewer than 30 amino acids. For comparison, we select 6 targets from the LNR dataset (1elr, 1t4f, 2f31, 3n55, 3r42, and 4x3h), for which BindCraft can generate relatively stable outputs. We adopt the default setting of BindCraft, i.e., AF2-based hallucination with pLDDT/pAE optimization. It is worth noting that, in our experiments, BindCraft often fails to produce stable structures for very short peptides or receptors with small interaction interfaces. Such designs are automatically discarded by its filtering pipeline. The results for these targets are reported in Table 16. Compared with diffusion-based methods, BindCraft performs

Table 15: Orthogonal geometric evaluation independent of the Rosetta ΔG criterion (lower is better).

Method	C-RMSD	L-RMSD	Clash_{in}	$\text{Clash}_{\text{out}}$	JSD_{bb}	JSD_{sc}
PepFlow	2.87	1.79	2.72%	4.62%	0.240	0.693
PepGLAD	2.74	1.60	1.82%	1.66%	0.474	0.398
UniMoMo$_{single}$	2.48	1.48	1.53%	**0.94%**	0.390	0.365
PhyDiff (Ours)	**2.23**	**1.18**	**1.34%**	0.96%	**0.218**	**0.223**

worse on the DockQ metric, but achieves competitive binding performance in terms of ΔG, which, to some extent, demonstrates its effectiveness.

Table 16: Experiments on BindCraft. The results of diffusion-based models are with relaxation.

Method	**PepGLAD**	**BindCraft**	**PhyDiff**
DockQ (↑)	0.488	0.218	**0.584**
ΔG (↓)	-28.4	-31.4	**-32.8**

C.6 Comparison between MOOD

To further demonstrate the effectiveness of FlexOOD, we compare it with MOOD. Unlike MOOD, which uses a fixed constant to control out-of-distribution exploration, FlexOOD effectively takes the flexibility of different peptides into account and enables adaptive exploration. The detailed results are shown in Table 17. It can be observed that introducing FlexOOD not only improves diversity but can also enhance binding performance. In contrast, MOOD may introduce much noise into the generation process, which ultimately has a negative impact on binding performance and validity.

Table 17: Experiments on FlexOOD and MOOD.

λ	**Methods**	**Success Rate**	ΔG	**SeqValid**	**SeqDiv**
0	Baseline	0.33	-16.79	**0.28**	0.52
0.05	FlexOOD	0.34	-17.35	0.27	0.57
	MOOD	0.31	-15.63	0.26	0.58
0.1	FlexOOD	**0.37**	**-18.12**	0.27	**0.61**
	MOOD	0.30	-15.27	0.24	0.59

C.7 Model Efficiency

FlexOOD and HHAG require additional predictor evaluations during sampling, so we explicitly measure their computational overhead. We use batch size 4 and generate peptides for 5 LNR targets, averaging all measurements over three independent runs. As shown in Table 18, the baseline requires 0.82 s and 259 MB peak memory, while the full FlexOOD+HHAG model requires 1.30 s and 483 MB. The corresponding absolute overhead is 0.48 s and 224 MB in this setting. FlexOOD alone costs 1.01 s/393 MB and HHAG alone costs 1.22 s/470 MB, indicating that most of the additional cost comes from sampling-time predictor guidance rather than from changing the denoising backbone. Although this overhead is non-negligible and can matter for very high-throughput screening, PhyDiff still performs direct diffusion sampling and avoids the repeated folding/optimization loops used by hallucination-based pipelines. We therefore view efficiency as a practical trade-off rather than claiming that the guidance is cost-free.

D Limitations

PhyDiff is physically inspired rather than a first-principles physical simulator: flexibility and hotspot/affinity information enter through learned predictors and sampling-time guidance, and the

Table 18: Experiments on the model efficiency.

Method	Time (s)	Peak Memory (MB)
Baseline	0.82	259
FlexOOD only	1.01	393
HHAG only	1.22	470
FlexOOD + HHAG	1.30	483

method does not explicitly integrate molecular dynamics or a physical energy function into the diffusion dynamics. Consequently, predictor calibration and distribution shift can affect guidance quality. Our OOD evidence is based on two external peptide benchmarks (LNR and PepPro-96), so we do not claim universal generalization to arbitrary biochemical regimes. In addition, FlexOOD and HHAG introduce non-negligible sampling-time overhead. Finally, dynamic behavior is not unique to peptides; proteins and other binders also exhibit inherent flexibility. Due to computational resource constraints, our experiments focus on peptide design and do not extend to these broader settings. We leave explicit physics integration, wider-domain validation, and further efficiency improvements to future work.

E Impact Statement

PhyDiff advances computational peptide design by enabling the generation of diverse, high-affinity peptide binders. This has the potential to accelerate therapeutic peptide discovery, reduce experimental costs, and expand the druggable proteome by targeting protein–protein interactions that are challenging for small molecules. As with all generative models for biomolecules, there is a potential for misuse in designing peptides with harmful biological activities (e.g., toxins or pathogenic agents).

F VISUALIZATION

We provide generated samples of PhyDiff in Figure 4, which further demonstrate its effectiveness in designing diverse and high-affinity peptide binders.

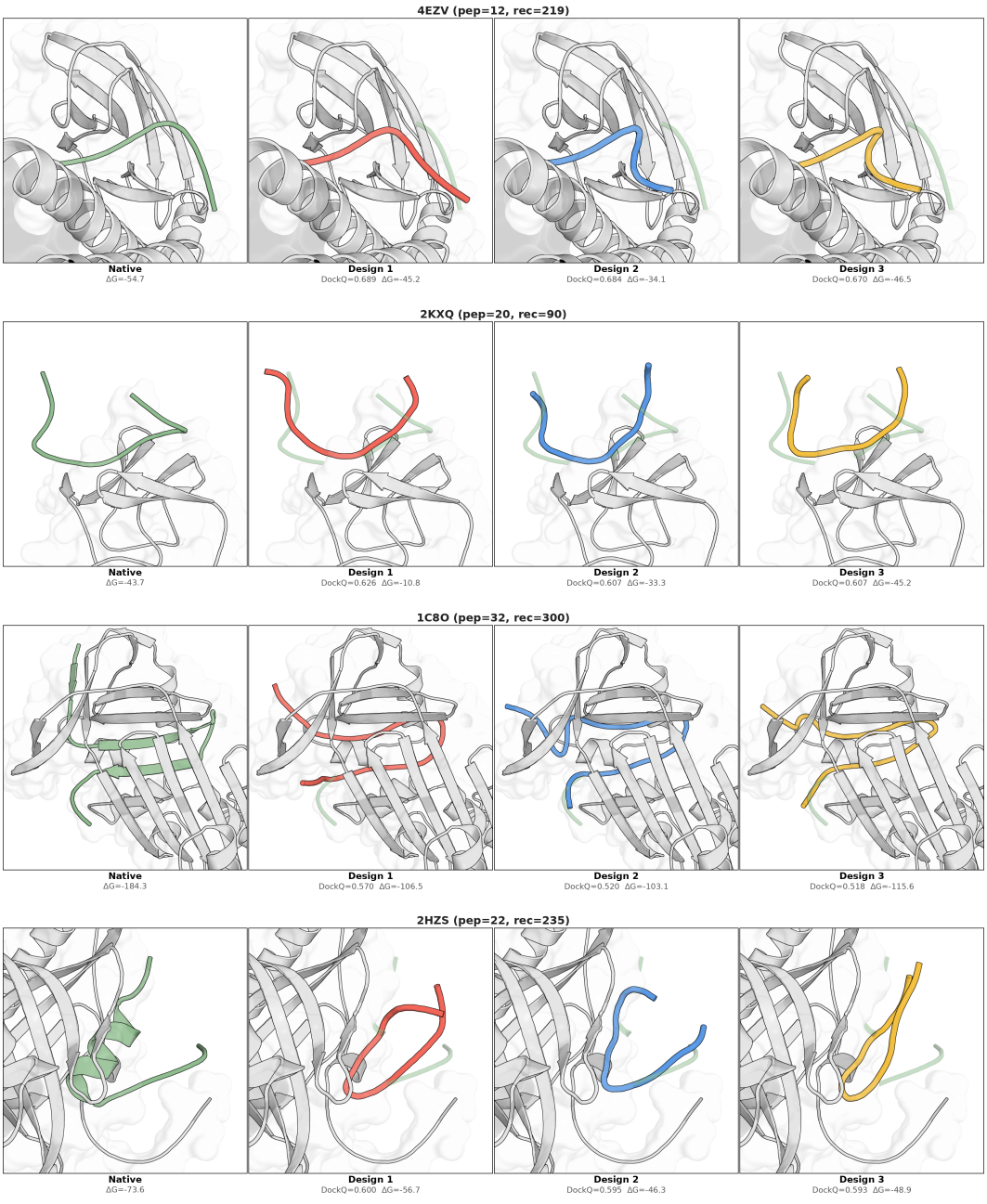


Figure 4: **Generated samples of PhyDiff**.