PEPSAGE: HIERARCHICAL ALL-ATOM PEPTIDE GENERATION WITH CONTINUOUS LOCAL-GEOMETRY STATES

Anonymous authors
Paper under double-blind review

ABSTRACT

Peptides can engage protein surfaces that are difficult to target with small molecules, making pocket-conditioned peptide design a valuable yet challenging therapeutic goal. Generating such peptides at all-atom resolution requires the joint modeling of global backbone pose, amino-acid identity, and residue-internal atomic conformation. Existing representations face a persistent tension between expressive local detail and geometric validity: Cartesian coordinates expose rigid-body redundancy, torsion variables are chemically grounded but less convenient as a uniform learned interface across residue types, and discrete structural tokens may discard fine conformational variation. We introduce a geometry-state generation formulation for pocket-conditioned all-atom peptide design, instantiated as PepSAGE (Peptide Structure-Aware Generation). Rather than directly generating unconstrained atom coordinates, relying on residue-specific torsions, or discretizing local structure into tokens, PepSAGE represents residue-internal conformation with a shared six-dimensional continuous state ($\boldsymbol{z} \in \mathbb{R}^6$), while modeling residue pose and amino-acid identity through separate channels. This state is learned through a reconstruction interface that is frozen during generation, allowing PepSAGE to generate continuous pre-quantization local geometry and recover all-atom structures through a geometry-constrained reconstruction pathway. Under a unified training budget and leakage-controlled evaluation, PepSAGE achieves the best backbone, all-atom, and side-chain RMSD, the lowest clash, and the highest rotamer recovery among competitive baselines, with additional analyses showing a tunable quality–diversity trade-off. Controlled ablations show that the continuous local-geometry channel drives side-chain recovery, while geometry-constrained reconstruction improves geometric validity, supporting continuous local geometry as an effective internal state for structured all-atom peptide generation.

1 INTRODUCTION

Peptides can bind extended protein interfaces that are often difficult to target with small molecules, making pocket-conditioned peptide design a promising therapeutic direction (Craik et al., 2013; Fosgerau & Hoffmann, 2015; Henninot et al., 2018; Muttenthaler et al., 2021). Recent generative methods address this task directly, but all-atom peptide generation remains challenging because the model must jointly produce global backbone pose, amino-acid identity, side-chain conformation, and chemically valid local geometry (Cao et al., 2022; Chen et al., 2023; Lin et al., 2024; Li et al., 2024; Kong et al., 2024; Wang et al., 2024; Li et al., 2025; Qian et al., 2026).

The representation of residue-internal geometry is central to this challenge. Cartesian coordinates are expressive but entangle internal conformation with rigid-body motion and provide no validity guarantee (Chu et al., 2024). Torsion-based representations are chemically meaningful, yet their dimensionality and semantics vary across residue types (Zhang et al., 2023; Li et al., 2024; Qian et al., 2026). Learned latents or discrete structural tokens offer fixed-size interfaces, but free latents may decode to invalid structures and discrete tokens can lose fine-grained conformational variation (Kong et al., 2024; Geffner et al., 2025; Yuan et al., 2025; Giri et al., 2026). These trade-offs suggest that the key design choice is not only the generative backbone, but also the object being generated: an

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.

internal state that is continuous enough to preserve local variation, compact enough to be shared across residue types, and constrained enough to decode into valid all-atom geometry.

We present PepSAGE as an instantiation of this geometry-state generation view. Each residue i is described by

$$\mathcal{S}_i = \big(\boldsymbol{t}_i,\ \boldsymbol{R}_i,\ \boldsymbol{z}_i,\ a_i\big). \tag{1}$$

Here, $\boldsymbol{t}_i$ and $\boldsymbol{R}_i$ define residue pose, a_i defines amino-acid identity, and $\boldsymbol{z}_i \in \mathbb{R}^6$ is a shared continuous local-geometry state. The $\boldsymbol{z}_i$ space is learned by a reconstruction stage over local-frame atoms and then frozen during generation. PepSAGE predicts the continuous pre-quantization value of this state, and a fixed FSQ projection with geometry-constrained reconstruction converts the generated hierarchy into all-atom coordinates. This design places PepSAGE between coarse discrete tokens and unconstrained free latents: it preserves continuous local variation while linking generated states to explicit geometric reconstruction. The resulting formulation emphasizes the residue-level object being generated, rather than only the stochastic process used to generate it. The code weights can be obtained from the anonymous repository: https://anonymous.4open.science/r/PepSAGE/

We make the following contributions:

- **A geometry-state generation formulation.** We formulate all-atom peptide generation around a compact, low-dimensional continuous state for local atomic geometry, learned through a reconstruction bottleneck over local-frame atoms. Rather than using this state as an external token or conditioning feature, PepSAGE treats its pre-quantization value as an internal generated component of each residue state.
- **A hierarchical generator coupled to a fixed reconstruction interface.** We develop PepSAGE, which jointly generates residue translations, frames, amino-acid identities, and the continuous local-geometry state, then recovers all-atom structure through a fixed finite-scalar-quantization projection, a trainable atom-coordinate head, and geometry-constrained reconstruction.
- **Accurate generation and controlled architectural validation.** Under a unified training and evaluation protocol, PepSAGE achieves leading structural accuracy, with the lowest clash and the highest rotamer recovery among competitive baselines. Controlled ablations show that the gains are tied to generating the continuous pre-quantization local-geometry state, rather than merely adding a compact structural code or relying on the reconstruction pathway alone, while additional analyses examine the quality–diversity trade-off.

2 Related Work

2.1 Pocket-Conditioned Peptide Generation

Recent methods have moved peptide design from sequence-only generation toward pocket-conditioned sequence–structure co-design. Earlier and complementary protein-design systems optimize or sample binders from target structures, sequence models, or inverse-folding pipelines (Cao et al., 2022; Dauparas et al., 2022). PPFlow models peptide backbone torsions with conditional flow matching on torus manifolds (Lin et al., 2024); DiffPepBuilder uses an $\mathrm{SE}(3)$-equivariant diffusion model to co-design peptide sequences and structures (Wang et al., 2024); and PepHAR autoregressively extends fragments from sampled interaction hotspots (Li et al., 2025). At all-atom resolution, PepFlow represents peptide geometry using rigid backbone frames and side-chain torsions and applies flow matching to the resulting Euclidean, manifold, and categorical variables (Li et al., 2024). PepBFN adopts a related explicit parameterization but formulates generation through Bayesian flows, including specialized distributions for residue identities, orientations, and side-chain torsions (Qian et al., 2026). In contrast, PepGLAD encodes variable-size all-atom residues into fixed-dimensional latent variables and performs receptor-conditioned diffusion in the learned latent space (Kong et al., 2024). These approaches establish the value of jointly generating sequence and all-atom structure, while exposing a central representation choice between explicit geometric variables and learned latent states.

2.2 All-Atom Protein Representations

All-atom generative models differ in how they divide global rigid-body motion from residue-local structure. Cartesian-coordinate models retain maximal atomic detail but must handle pose and internal geometry simultaneously, as in the direct all-atom generation of Protpardelle (Chu et al., 2024). Equivariant geometric networks provide a common substrate for such 3D modeling problems (Jing et al., 2020). Residue-frame representations describe global geometry through translations and rotations and are widely used in backbone generation (Yim et al., 2023; Watson et al., 2023). All-atom models then complement frames with explicit side-chain torsions, as in PepFlow and PepBFN (Li et al., 2024; Qian et al., 2026), while dedicated packing methods such as SCWRL4, DLPacker, and DiffPack model side-chain placement or torsional uncertainty using rotameric or learned predictors (Krivov et al., 2009; Misiura et al., 2022; Zhang et al., 2023). Latent representations instead absorb variable-size atomic detail into fixed-dimensional residue features, simplifying the generative state at the cost of relying on a learned reconstruction map (Kong et al., 2024; Geffner et al., 2025). Overall, these representations trade explicit geometric control against compactness and uniformity across residue types.

2.3 Latent and Tokenized Structural Representations

Learned structural bottlenecks provide a uniform interface for protein geometry. PepGLAD uses a variational autoencoder to diffuse all-atom peptide residues in a geometric latent space (Kong et al., 2024), while La-Proteina jointly generates an explicit coarse backbone and continuous per-residue latents containing sequence and atomistic detail (Geffner et al., 2025). Structural tokenizers instead quantize continuous geometry into discrete symbols for reconstruction or multimodal sequence modeling; recent examples study both the fidelity of such representations and their use in sequence–structure generation (van den Oord et al., 2017; Gao et al., 2025; Liu et al., 2024; Yuan et al., 2025; Giri et al., 2026). Thus, learned structural bottlenecks balance compression and modeling convenience against continuity and reconstruction fidelity.

2.4 Continuous Generative Models

Diffusion models learn iterative denoising processes from progressively corrupted data (Ho et al., 2020; Song et al., 2020; Hoogeboom et al., 2022), whereas flow matching trains continuous vector fields along prescribed probability paths (Lipman et al., 2022; Liu et al., 2022; Chen & Lipman, 2023; Bose et al., 2024). Bayesian Flow Networks (BFNs) instead maintain and update parameters of input distributions through a sequence of Bayesian updates (Graves et al., 2023). These frameworks can be extended to mixed Euclidean, categorical, and manifold-valued molecular variables, as demonstrated by PepFlow and PepBFN (Li et al., 2024; Qian et al., 2026). Collectively, they provide complementary mechanisms for modeling heterogeneous geometric and categorical variables.

3 Method

3.1 Problem Formulation

We consider pocket-conditioned peptide generation at all-atom resolution. The input is a receptor pocket $\mathcal{P}$ and a prescribed peptide length N; the output is a bound peptide $\mathcal{G}$ with amino-acid identities and heavy-atom coordinates. PepSAGE follows the geometry-state generation formulation introduced above: instead of treating every atom coordinate as an unconstrained endpoint, it predicts residue pose, amino-acid identity, and an internal local-geometry state that is later decoded into atoms. Formally, PepSAGE models

$$p_\theta(\mathcal{G} \mid \mathcal{P}, N). \tag{2}$$

In the model input, the observed pocket is combined with N masked peptide residue slots. Pocket residues remain fixed, while peptide identities, translations, rotations, and local atomic states are generated conditionally on the receptor context. Full notation for the all-atom complex is given in the supplementary material.

3.2 HIERARCHICAL RESIDUE STATE

Figure 1 summarizes the overall architecture. The central state variable in PepSAGE is a residue-level hierarchy $(\boldsymbol{t}_i, \boldsymbol{R}_i, \boldsymbol{z}_i, a_i)$. The translation $\boldsymbol{t}_i$ is the C_α position, $\boldsymbol{R}_i \in \mathrm{SO}(3)$ is a backbone-derived local frame, a_i is the amino-acid identity, and $\boldsymbol{z}_i \in \mathbb{R}^6$ is a continuous local-geometry state. This hierarchy makes the generated object explicit: global placement, residue identity, and residue-internal geometry are predicted as separate but coupled channels by a shared pocket-conditioned network. It also avoids forcing a single representation to explain both rigid-body motion and residue-internal geometry: the pose variables place each residue in the binding site, the sequence variable determines residue topology, and the continuous local state captures conformational variation within that topology. This separation is useful for peptide design because side-chain accuracy depends on local atomic detail, whereas binding-site compatibility depends on coordinated changes across the whole peptide and receptor context.

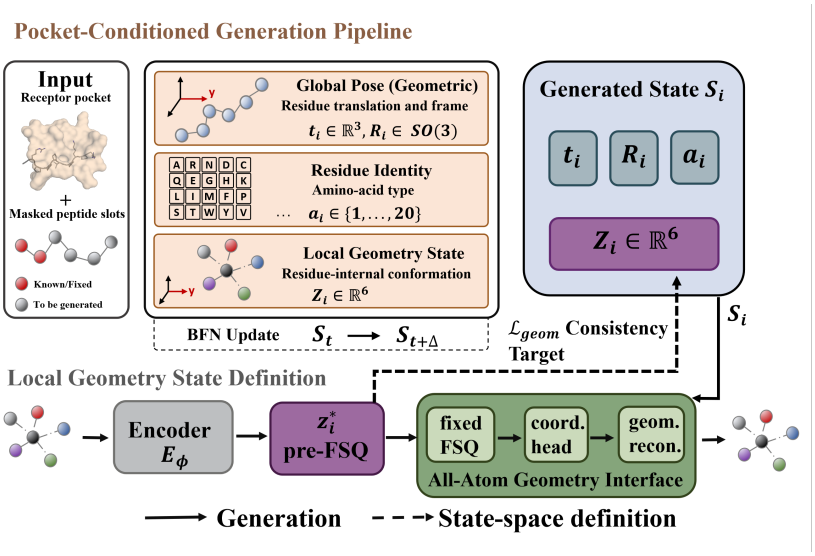

Figure 1: Overview of PepSAGE. Conditioned on a fixed receptor pocket, PepSAGE jointly generates residue pose, identity, and continuous local geometry, then reconstructs all-atom peptide coordinates.

3.3 CONTINUOUS LOCAL-GEOMETRY INTERFACE

The six-dimensional state $\boldsymbol{z}_i$ is defined by a reconstruction-learned interface trained before the generative model. This interface turns local atom geometry into the internal variable that PepSAGE will later generate. Each residue is first expressed in its backbone-derived local frame, and a coordinate-only encoder maps the valid heavy-atom coordinates, serialized in canonical atom14 order, to a pre-quantization local state. The interface receives no residue-type, atom-type, or atom-slot labels, so the bottleneck is learned from local geometry rather than from discrete topology cues.

After interface training, the encoder, FSQ projection, and output projection are frozen. Ground-truth structures are mapped to fixed supervision targets for PepSAGE, and generated states are decoded through the same frozen interface before all-atom reconstruction. Thus, $\boldsymbol{z}_i$ is not an unconstrained latent whose semantics are learned only by the generator; it is a continuous state with a prescribed reconstruction pathway. The coordinate transform, encoder pooling, FSQ projection, and interface reconstruction losses are provided in the supplementary material.

During generation, PepSAGE predicts the continuous pre-quantization state rather than a discrete structural token. This is the main representational choice of the method: the generated variable remains uniform across residue types and retains fine-grained local variation, while the frozen projection ties it to a valid reconstruction interface.

3.4 JOINT GENERATIVE MODEL

PepSAGE models the conditional distribution

$$p_\theta(\boldsymbol{t}, \boldsymbol{R}, \boldsymbol{z}, \boldsymbol{a} \mid \mathcal{P}, N) \tag{3}$$

on a unified graph containing observed pocket residues and masked peptide slots. Initial features are constructed to avoid target leakage: pocket sequence and structure remain visible, peptide identities are replaced by an unknown token, and peptide coordinate-derived features are hidden. Cross-chain sequence offsets and target-dependent pocket–peptide geometric features are also excluded.

Pocket and peptide residues interact through a shared IPA–Transformer trunk. At each BFN step, the trunk combines leakage-free context, the current generated state, and a time embedding; it updates peptide features and frames while keeping pocket states fixed. The output heads predict clean endpoints $\widehat{\boldsymbol{t}}_i$, $\widehat{\boldsymbol{R}}_i$, and $\widehat{\boldsymbol{z}}_i$, together with amino-acid logits $\ell_i \in \mathbb{R}^{20}$. Sequence prediction is parallel rather than autoregressive:

$$p_\theta(a_i \mid \mathcal{P}, N, \mathcal{S}_t) = \mathrm{softmax}(\ell_i). \tag{4}$$

Cross-entropy is applied only to generated residues. At the end of sampling, $\arg\max \ell_i$ determines the residue identity, valid heavy-atom slots, and atomic topology. Amino-acid identity is thus a jointly predicted output, not a component of the continuous local-geometry state.

The three geometric channels use state spaces matched to their variables. Translations and local-geometry states follow Gaussian BFN paths, while rotations use a Matrix–Fisher Bayesian flow on $\mathrm{SO}(3)$ (Graves et al., 2023; Qian et al., 2026). At inference, peptide translations and local states are initialized at zero and rotations are sampled uniformly. The network iteratively predicts clean endpoints, resamples the corresponding BFN states, and returns the final endpoint predictions after 100 updates. The Gaussian paths, Matrix–Fisher perturbation, loss weights, and sampling details are given in the supplementary material.

3.5 LOCAL-GEOMETRY RECONSTRUCTION

Generated continuous local states are converted to atom coordinates through the frozen FSQ projection and a trainable atom-coordinate head. The predicted amino-acid identity selects the valid atom slots and topology. Local atom coordinates are transformed by the generated residue frame, and torsions extracted from the coordinate-head output are combined with rigid-group templates and reference bond geometry to produce the final atom14 structure. This geometry-constrained reconstruction imposes standard bond lengths, bond angles, and rigid-group stereochemistry after generation, without adding bond, angle, or clash penalties to the training loss. Projection equations, atom-slot handling, and reconstruction details are provided in the supplementary material.

The reconstruction module therefore serves two roles: it decodes the generated continuous local state into residue-specific atoms, and it regularizes the final structure by rebuilding atoms through chemically meaningful rigid groups. This keeps the generative model focused on predicting pose, identity, and local state, while the final conversion step enforces the atom-level geometry used for evaluation.

3.6 TRAINING OBJECTIVES

The local-geometry interface is trained independently and then frozen to define targets and the fixed projection used by PepSAGE. The generation model minimizes

$$\begin{aligned}\mathcal{L} = \lambda_t \mathcal{L}_{\mathrm{trans}} + \lambda_R \mathcal{L}_{\mathrm{rot}} + \lambda_z \mathcal{L}_z \\ + \lambda_a \mathcal{L}_{\mathrm{seq}} + \lambda_s \mathcal{L}_{\mathrm{struct}}.\end{aligned} \tag{5}$$

Here, $\mathcal{L}_{\mathrm{trans}}$ and $\mathcal{L}_z$ are weighted Gaussian-BFN endpoint losses, $\mathcal{L}_{\mathrm{rot}}$ is the Matrix–Fisher rotation loss, $\mathcal{L}_{\mathrm{seq}}$ is amino-acid cross-entropy, and $\mathcal{L}_{\mathrm{struct}}$ is the MSE between predicted and target local atom coordinates over valid generated atoms. Loss weights and architecture hyperparameters are listed in the supplementary material.

All losses are evaluated only on generated peptide residues, while receptor features provide conditioning context. The frozen local interface defines the training targets for $\boldsymbol{z}$ and the reconstruction pathway used at inference, so PepSAGE is optimized against the same state representation that it later samples.

4 EXPERIMENTAL SETUP

4.1 DATASET AND LEAKAGE-CONTROLLED EVALUATION

We construct the dataset from experimentally resolved peptide–protein complexes collected by PepBDB (Wen et al., 2019) and Q-BioLiP (Wei et al., 2024). We retain peptides containing 3–25 residues and represent each example as a target peptide chain together with its receptor pocket. After preprocessing, the common evaluation pool contains 187 unique targets indexed by their `PDB_chain` identifiers.

We further audit the split for exact train–test overlap. A test target is removed if its `PDB_chain` identifier occurs in either the PepSAGE or the PepFlow training set. Taking the union of these two training-set overlap lists removes 67 of the 187 evaluation targets and leaves 120 leakage-controlled targets. The filtering rule is determined solely from dataset identifiers, before inspecting any generated structures or evaluation scores, and the same retained target set is used for every method. The excluded identifiers and filtering script are provided in the supplementary material.

4.2 BASELINES

We compare PepSAGE with three representative all-atom peptide generators. PepFlow (Li et al., 2024) uses multi-modal flow matching over residue-level structural and sequence variables. PepGLAD (Kong et al., 2024) performs diffusion in a learned geometric latent space, whereas PepBFN (Qian et al., 2026) jointly models Euclidean residue positions, rotations, amino-acid types, and torsional geometry with Bayesian flow networks. We retrain all three methods on the same processed dataset using the same comparison budget, rather than using their released checkpoints. Hereafter, *PepFlow*, *PepGLAD*, and *PepBFN* refer exclusively to these retrained models. They are evaluated on the same chain-resolved targets with the same structure-parsing and metric implementations.

We additionally include a **Discrete Token** baseline as an internal representation control. It models an FSQ-derived structural code categorically, replacing PepSAGE's continuous pre-quantization local-geometry state while retaining the same pocket-conditioned generation setting. We use this model to isolate the effect of a continuous local state from that of a discrete structural target; it is not treated as an external published method.

4.3 EVALUATION METRICS

We evaluate generated peptides against their native complexes along five dimensions. **Structural accuracy** is measured by Cα backbone, all-atom, and side-chain RMSD over matched heavy atoms; for local side-chain analysis we additionally report backbone-aligned side-chain RMSD (BB-SC RMSD). **Pocket consistency** is measured by contact recovery, using receptor residues within 10 Å of the peptide as the native contact set. **Local conformation** is evaluated by rotamer recovery and joint χ accuracy, with a χ angle counted as recovered when its circular error is at most 20°. **Geometric validity** is summarized by bond-length and bond-angle deviations together with a heavy-atom clash score. Finally, **sample coverage** is measured by diversity, novelty, and amino-acid recovery (AAR); TM-scores used in these metrics are computed with TM-align (Zhang & Skolnick, 2005). Full metric definitions are provided in the supplementary material.

4.4 STATISTICAL PROTOCOL

All metrics are aggregated at the target-complex level. For each metric, we first summarize all candidates for a target into one value and then report the unweighted mean over the 120 leakage-controlled targets; diversity is computed as a within-target pairwise statistic.

For method comparisons, targets are matched by their `PDB_chain` identifiers and paired target-level differences are bootstrapped with 10,000 resamples and random seed 1234. We report the mean paired difference and 95% percentile bootstrap confidence interval, treating an interval that excludes zero as significant; differences are defined as PepSAGE minus the baseline. Training stability is summarized from three PepSAGE runs with seeds 1234, 4321, and 8989 by reporting the mean and sample standard deviation of their target-level means.

Table 1: Overall metrics on 120 leakage-controlled targets. Disc. denotes the discrete-token representation control.

Metric	PepFlow	PepGLAD	PepBFN	Disc.	PepSAGE
Backbone RMSD ↓	3.922	4.357	5.319	5.488	**3.360**
All-atom RMSD ↓	7.380	7.639	7.682	8.152	**6.060**
Side-chain RMSD ↓	8.118	8.449	8.509	8.857	**6.900**
Contact ↑	0.780	**0.895**	0.710	0.682	0.840
Diversity ↑	0.534	**0.575**	0.470	0.328	0.397
Bond angle dev. ↓	**9.058**	38.011	12.001	13.561	10.458
Clash ↓	0.014	0.023	0.015	0.079	**0.013**
Rotamer recovery ↑	0.122	0.138	0.106	0.122	**0.163**

4.5 Sampling and Evaluation Protocol

For each target complex, we generate 64 candidate peptides. PepSAGE uses 100 BFN updates followed by a final endpoint prediction at $t = 1$; the receptor pocket remains fixed while only masked peptide states are updated. All methods are evaluated with the same number of candidates, target set, structure parsers, and metric implementations. Candidate-level measurements are summarized within each target before target-level aggregation and paired statistical analysis. Further serialization and endpoint-decoding details are provided in the supplementary material.

4.6 Implementation Details

PepSAGE uses six IPA–Transformer blocks with 128-dimensional node features and 64-dimensional pair features. The local-geometry state has dimension six and is projected with FSQ levels $[4, 4, 4, 4, 4, 4]$; the atom-coordinate head predicts up to 15 heavy-atom slots. Unless otherwise stated, reported checkpoints use the common comparison budget. Complete architecture and optimization hyperparameters are provided in the supplementary material.

5 Results

5.1 Overall Generation Quality

Table 1 reports accuracy and local geometry on the 120 leakage-controlled targets. PepSAGE attains the best backbone, all-atom, and side-chain RMSD, improving over the strongest baseline (PepFlow) by 0.56, 1.32, and 1.22 Å, respectively; these gains are significant under the paired target-level bootstrap. PepSAGE also obtains the lowest clash score and highest rotamer recovery, indicating that improved coordinate accuracy is accompanied by better local steric and conformational quality.

The gains are not uniform across all desiderata: PepGLAD has higher contact recovery, PepFlow and PepGLAD produce more diverse samples, and PepFlow has a slightly lower bond-angle deviation. A temperature sweep shows that PepSAGE can trade accuracy for diversity without geometric collapse; the full curve, novelty, AAR, bond-length deviation, and paired-bootstrap table are provided in the supplementary material. The Discrete Token control is worse on every metric in Table 1, supporting continuous pre-quantization local-state generation over a categorical FSQ target. Figure 2 shows representative generated complexes.

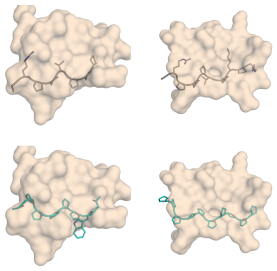

Figure 2: Representative native (top) and generated (bottom) peptides for 1CKB and 2DRK.

5.2 Local Side-Chain Accuracy

We next isolate side-chain conformation, which directly probes whether the local-geometry interface improves residue-level atomic placement (Table 2). PepSAGE is best on all three reported metrics. For backbone-aligned side-chain RMSD, it improves over the strongest baseline, PepFlow, by 0.762 Å (5.050 vs. 5.812). For rotameric recovery, the margin is larger: joint χ_1 accuracy reaches 0.438 compared with 0.211 for the strongest baseline, PepGLAD, and joint $\chi_1 \wedge \chi_2$ reaches 0.305 compared with 0.072. These gains indicate that PepSAGE improves not only global coordinate fit but also the local side-chain conformations that are most sensitive to all-atom geometry.

The joint-χ metrics are computed only on residues whose generated amino acid matches the native residue. They therefore evaluate conformation conditional on sequence recovery, rather than amino-acid prediction itself. This protocol also differs from the all-residue ablation setting in Table 3 (e.g. $\chi_1 = 0.438$ here vs. 0.383 there), so the two tables should be read as complementary rather than directly interchangeable. Deeper χ_3/χ_4 accuracies, where PepSAGE still trails PepGLAD, are reported in the supplementary material.

Table 2: Side-chain conformation accuracy on sequence-matched residues. BB-SC denotes backbone-aligned side-chain RMSD.

Metric	PepFlow	PepGLAD	PepBFN	PepSAGE
BB-SC RMSD ↓	5.812	6.112	7.341	**5.050**
Joint χ_1 ↑	0.171	0.211	0.108	**0.438**
Joint $\chi_{1:2}$ ↑	0.027	0.072	0.063	**0.305**

5.3 Ablation of the Local-Geometry Interface

Table 3 tests whether the local-geometry variable is merely an auxiliary feature or a necessary part of the generated residue state. Removing it almost eliminates side-chain recovery: setting $z = 0$ reduces joint $\chi_{1:2}$ from 0.213 to 0.010 and worsens BB-aligned side-chain RMSD from 5.050 to 5.634 Å, showing that identity and backbone frame alone are insufficient for residue-internal conformation.

The post-quantization control further asks whether the benefit comes from any compact structural code, or specifically from generating the continuous pre-quantization state. Coarsening the generated state after FSQ projection weakens rotamer recovery (χ_1: 0.383 to 0.294; $\chi_{1:2}$: 0.213 to 0.105) and worsens BB-SC RMSD from 5.050 to 5.206 Å. These results support the central representational choice of PepSAGE: generate the continuous local-geometry state and decode it through the shared geometry-constrained pathway, rather than generating only residue identity and backbone frame or relying on a coarsened post-quantization code. A DDPM variant using the same interface gives comparable performance in the supplementary material.

Table 3: Ablation of the continuous local-geometry interface.

Variant	Change	$\chi_1\uparrow$	$\chi_{1:2}\uparrow$	BB-SC$\downarrow$
Main	Continuous $\mathbb{R}^6$	0.383	0.213	5.050
Post-quant	Coarsen state	0.294	0.105	5.206
$z = 0$	Remove state	0.096	0.010	5.634

5.4 Effect of Geometry-Constrained Reconstruction

We further examine how the final reconstruction step affects coordinate accuracy and local chemical validity. Table 4 compares the intermediate atom coordinates produced by the trainable coordinate head with the final structure obtained after torsion extraction and geometry-constrained reconstruction. The intermediate coordinates are slightly closer to the native structure in RMSD, but they contain substantially worse local geometry. The final reconstruction increases all-atom RMSD by $0.334\,\text{Å}$ and side-chain RMSD by $0.711\,\text{Å}$, while reducing bond-length deviation from 1.166 to 0.139, bond-angle deviation from $15.79°$ to $10.46°$, and clash score from 0.091 to 0.0125.

This result clarifies the source of PepSAGE's geometric validity. The model does not rely on unconstrained atom coordinates alone. Instead, it generates a continuous local-geometry state, maps it to atom-level information, and then projects the result through a geometry-constrained reconstruction pathway. This coupling preserves the learned local conformational signal while converting it into chemically more regular all-atom structures.

Table 4: Intermediate coordinate-head output versus the final geometry-constrained reconstruction.

Metric	Intermed.	Final	Δ
All-atom RMSD $\downarrow$	**5.726**	6.060	$+0.334$
Side-chain RMSD $\downarrow$	**6.190**	6.900	$+0.711$
Bond length dev. $\downarrow$	1.166	**0.139**	-1.027
Bond angle dev. $\downarrow$	15.790	**10.459**	-5.331
Clash $\downarrow$	0.091	**0.013**	-0.078

6 Conclusion

PepSAGE frames all-atom peptide design as conditional generation of hierarchical residue states, where residue-internal geometry is represented by a compact continuous pre-quantization variable. This formulation places the generation target between a discrete token, which can discard fine conformational variation, and a free latent, which carries no prescribed path to physical realizability; a fixed quantized, geometry-constrained reconstruction then converts the generated states into chemically regular all-atom structures. Evaluated on 120 leakage-controlled targets with retrained baselines, PepSAGE achieves the best backbone, all-atom, and side-chain RMSD, as well as the lowest clash and highest rotamer recovery. Controlled ablations further show that the continuous local-geometry channel is central to side-chain recovery, and that the final reconstruction step substantially improves geometric validity. While this work demonstrates the promise of continuous local geometry for all-atom peptide generation, several directions remain open. For instance, stronger joint modeling of pocket contacts, sequence recovery, and deeper side-chain torsions may further improve binding-specific accuracy and sample diversity. Overall, PepSAGE establishes a compact and principled geometry-state generation framework for all-atom peptide design, offering a useful foundation for future peptide engineering.

References

Avishek Joey Bose, Tara Akhound-Sadegh, Kilian Fatras, Guillaume Huguet, Jarrid Rector-Brooks, Cheng-Hao Liu, Andreea C. Nica, Maksym Korablyov, Michael M. Bronstein, and Alexander

Tong. SE(3)-stochastic flow matching for protein backbone generation. In *International Conference on Learning Representations*, 2024.

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

Ricky T. Q. Chen and Yaron Lipman. Riemannian flow matching on general geometries, 2023.

Tianlai Chen, Madeleine Dumas, Rio Watson, Sophia Vincoff, Christina Peng, Lin Zhao, Lauren Hong, Sarah Pertsemlidis, Mayumi Shaepers-Cheu, Tian Zi Wang, Divya Srijay, Connor Monticello, Pranay Vure, Rishab Pulugurta, Kseniia Kholina, Shrey Goel, Matthew P. DeLisa, Ray Truant, Hector C. Aguilar, and Pranam Chatterjee. Pepmlm: Target sequence-conditioned generation of therapeutic peptide binders via span masked language modeling, 2023.

Alexander E. Chu, Jinho Kim, Lucy Cheng, Gina El Nesr, Minkai Xu, Richard W. Shuai, and Po-Ssu Huang. An all-atom protein generative model, 2024.

David J. Craik, David P. Fairlie, Spiros Liras, and David Price. The future of peptide-based drugs. *Chemical Biology & Drug Design*, 81(1):136–147, 2013.

Justas Dauparas, Ivan Anishchenko, Nathaniel Bennett, Hua Bai, Robert J. Ragotte, Lukas F. Milles, Basile I. M. 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.

Keld Fosgerau and Torsten Hoffmann. Peptide therapeutics: Current status and future directions. *Drug Discovery Today*, 20(1):122–128, 2015.

Zhangyang Gao, Cheng Tan, Jue Wang, Yufei Huang, Lirong Wu, and Stan Z. Li. Foldtoken: Learning protein language via vector quantization and beyond. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 219–227, 2025. doi: 10.1609/aaai.v39i1.31998.

Tomas Geffner, Kieran Didi, Zhonglin Cao, Danny Reidenbach, Zuobai Zhang, Christian Dallago, Emine Kucukbenli, Karsten Kreis, and Arash Vahdat. La-proteina: Atomistic protein generation via partially latent flow matching, 2025.

Nabin Giri, Steven Farrell, and Kristofer E. Bouchard. Yeti: A compact protein structure tokenizer for reconstruction and multi-modal generation, 2026.

Alex Graves, Rupesh Kumar Srivastava, Timothy Atkinson, and Faustino Gomez. Bayesian flow networks, 2023.

Antoine Henninot, John C. Collins, and John M. Nuss. The current state of peptide drug discovery: Back to the future? *Journal of Medicinal Chemistry*, 61(4):1382–1414, 2018.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, pp. 6840–6851, 2020.

Emiel Hoogeboom, Victor Garcia Satorras, Clement Vignac, and Max Welling. Equivariant diffusion for molecule generation in 3d. In *Proceedings of the International Conference on Machine Learning*, pp. 8867–8887, 2022.

Bowen Jing, Stephan Eismann, Patricia Suriana, Raphael J. L. Townshend, and Ron O. Dror. Learning from protein structure with geometric vector perceptrons, 2020.

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

Georgii G. Krivov, Maxim V. Shapovalov, and Roland L. Jr. Dunbrack. Improved prediction of protein side-chain conformations with SCWRL4. *Proteins: Structure, Function, and Bioinformatics*, 77(4):778–795, 2009.

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. In *Proceedings of the International Conference on Machine Learning*, 2024.

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. In *International Conference on Learning Representations*, 2025.

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. In *Proceedings of the International Conference on Machine Learning*, 2024.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling, 2022.

Andrew Liu, Axel Elaldi, Nathan Russell, and Olivia Viessmann. Bio2token: All-atom tokenization of any biomolecular structure with mamba, 2024.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow, 2022.

Mikołaj Misiura, Raghav Shroff, Ross Thyer, and Anatoly B. Kolomeisky. DLPacker: Deep learning for prediction of amino acid side chain conformations in proteins. *Proteins: Structure, Function, and Bioinformatics*, 90(6):1278–1290, 2022.

Markus Muttenthaler, Glenn F. King, David J. Adams, and Paul F. Alewood. Trends in peptide drug discovery. *Nature Reviews Drug Discovery*, 20:309–325, 2021.

Hao Qian, Shikui Tu, and Lei Xu. Full-atom peptide design via riemannian–euclidean bayesian flow networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2026.

Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations, 2020.

Aaron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural discrete representation learning. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Fanhao Wang, Yuzhe Wang, Laiyi Feng, Changsheng Zhang, and Luhua Lai. Target-specific de novo peptide binder design with diffpepbuilder. *Journal of Chemical Information and Modeling*, 64(24), 2024. doi: 10.1021/acs.jcim.4c00975.

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. De novo design of protein structure and function with RFdiffusion. *Nature*, 620(7976):1089–1100, 2023.

Hong Wei, Wenkai Wang, Zhenling Peng, and Jianyi Yang. Q-BioLiP: A comprehensive resource for quaternary structure-based protein–ligand interactions. *Genomics, Proteomics & Bioinformatics*, 22(1):qzae001, 2024.

Zhi Wen, Jiyang He, Hui Tao, and Sheng-You Huang. PepBDB: A comprehensive structural database of biological peptide–protein interactions. *Bioinformatics*, 35(1):175–177, 2019.

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. In *Proceedings of the International Conference on Machine Learning*, 2023.

Xinyu Yuan, Zichen Wang, Marcus Collins, and Huzefa Rangwala. Protein structure tokenization: Benchmarking and new recipe, 2025.

Yang Zhang and Jeffrey Skolnick. TM-align: A protein structure alignment algorithm based on the TM-score. *Nucleic Acids Research*, 33(7):2302–2309, 2005.

Yangtian Zhang, Zuobai Zhang, Bozitao Zhong, Sanchit Misra, and Jian Tang. DiffPack: A torsional diffusion model for autoregressive protein side-chain packing. In *Advances in Neural Information Processing Systems*, 2023.

SUPPLEMENTARY MATERIAL

A ALGORITHMIC COMPONENTS AND PSEUDOCODE

This section describes the algorithmic modules that define PepSAGE, rather than the outer training or evaluation workflow. The model contains four main components: a residue-local continuous geometry interface, a pocket-conditioned hierarchical denoiser, Bayesian-flow state transitions for the generated variables, and an all-atom reconstruction module.

We denote a protein–peptide complex by $\mathcal{C} = \{\mathcal{P}, \mathcal{G}\}$, where $\mathcal{P}$ is the receptor pocket and $\mathcal{G}$ is the bound peptide. At all-atom resolution, they are represented as

$$\begin{aligned} \mathcal{P} &= \{(a_j^P, \boldsymbol{X}_j^P)\}_{j=1}^{N_P}, \\ \mathcal{G} &= \{(a_i, \boldsymbol{X}_i)\}_{i=1}^{N}, \end{aligned} \tag{S1}$$

where a denotes amino-acid identity and $\boldsymbol{X}$ contains the corresponding heavy-atom coordinates.

A.1 LOCAL-GEOMETRY STATE CONSTRUCTION

For each residue i, global atom coordinates are first expressed in a backbone-derived local frame,

$$\widetilde{\boldsymbol{x}}_{ia} = \boldsymbol{R}_i^\top (\boldsymbol{x}_{ia} - \boldsymbol{t}_i), \qquad a \in \mathcal{A}_i. \tag{S2}$$

The coordinate-only local interface maps the valid atom sequence to atom-level pre-quantization states and pools them into one residue-level state:

$$\begin{aligned} \boldsymbol{h}_{1:T} &= E_\phi(\widetilde{\boldsymbol{x}}_{1:T}), \\ u_a &= W_{\text{in}} \boldsymbol{h}_a, \\ \boldsymbol{z}_i^\star &= \frac{1}{|\mathcal{A}_i|} \sum_{a \in \mathcal{A}_i} u_a \in \mathbb{R}^6. \end{aligned} \tag{S3}$$

Algorithm S1: Local-geometry state construction

Input: valid atom14 coordinates $\{\boldsymbol{x}_{ia}\}_{a \in \mathcal{A}_i}$, residue frame $(\boldsymbol{t}_i, \boldsymbol{R}_i)$, frozen encoder E_ϕ, projection W_{in}.
Output: continuous pre-quantization state $\boldsymbol{z}_i^\star$.

1. Transform each valid atom to the residue-local frame: $\widetilde{\boldsymbol{x}}_{ia} \leftarrow \boldsymbol{R}_i^\top (\boldsymbol{x}_{ia} - \boldsymbol{t}_i)$.
2. Serialize valid atoms in canonical atom14 order; invalid slots are omitted and no residue-type or atom-type labels are supplied.
3. Compute contextual atom features $\boldsymbol{h}_{1:T} \leftarrow E_\phi(\widetilde{\boldsymbol{x}}_{1:T})$.
4. Project each valid atom feature: $u_a \leftarrow W_{\text{in}} \boldsymbol{h}_a \in \mathbb{R}^6$.
5. Return the residue-level state $\boldsymbol{z}_i^\star \leftarrow |\mathcal{A}_i|^{-1} \sum_{a \in \mathcal{A}_i} u_a$.

A.2 POCKET-CONDITIONED HIERARCHICAL DENOISER

At a BFN time t, each peptide residue is represented by the current noisy or intermediate hierarchical state $\mathcal{S}_i^t = (\boldsymbol{t}_i^t, \boldsymbol{R}_i^t, \boldsymbol{z}_i^t, \pi_i^t)$, where π_i^t denotes the amino-acid distribution. Pocket residues provide fixed context. The leakage-controlled node and pair embeddings are combined with time features and updated by an IPA–Transformer trunk:

$$\begin{aligned} \boldsymbol{n}^0 &= \text{Embed}_{\text{node}}(\mathcal{S}^t, \mathcal{P}, t), \\ \boldsymbol{p}^0 &= \text{Embed}_{\text{pair}}(\mathcal{S}^t, \mathcal{P}), \\ (\boldsymbol{h}^L, \boldsymbol{p}^L) &= \Phi_\theta(\boldsymbol{n}^0, \boldsymbol{p}^0). \end{aligned} \tag{S4}$$

Clean endpoints are predicted by dedicated heads:

$$(\widehat{\boldsymbol{t}}_i, \widehat{\boldsymbol{R}}_i, \widehat{\boldsymbol{z}}_i, \ell_i) = H_\theta(\boldsymbol{h}_i^L), \tag{S5}$$

where ℓ_i are amino-acid logits.

Algorithm S2: PepSAGE hierarchical denoiser

Input: fixed pocket context $\mathcal{P}$, current residue states $\{\mathcal{S}_i^t\}$, time t.
Output: clean endpoint predictions $(\widehat{\boldsymbol{t}}, \widehat{\boldsymbol{R}}, \widehat{\boldsymbol{z}}, \ell)$ and residue embeddings.

1. Build leakage-controlled node features: pocket features remain visible, while peptide slots use the current generated state and mask embeddings.
2. Build pair features from allowed sequence, chain, distance, and dihedral encodings; target-dependent peptide geometry is not exposed.
3. Add the time embedding to generated peptide residues.
4. Apply six IPA–Transformer blocks, updating residue embeddings, pair embeddings, and peptide frames while keeping receptor context fixed.
5. Apply output heads to generated residues: translation head for $\widehat{\boldsymbol{t}}$, rotation head for $\widehat{\boldsymbol{R}}$, local-state head for $\widehat{\boldsymbol{z}}$, and sequence head for logits ℓ.

A.3 Bayesian-Flow State Transition

Translations and local-geometry states use Gaussian BFN paths. For a clean endpoint prediction $\widehat{x} \in \{\widehat{\boldsymbol{t}}, \widehat{\boldsymbol{z}}\}$, the next input-distribution parameters are updated by the Gaussian Bayesian-flow transition with schedule σ_1. Rotations use an analogous Matrix–Fisher transition on $\mathrm{SO}(3)$, and residue identities use the categorical update implied by the sequence logits.

Algorithm S3: One-step hierarchical Bayesian-flow transition

Input: current states $(\boldsymbol{t}^t, \boldsymbol{R}^t, \boldsymbol{z}^t, \pi^t)$, endpoint predictions $(\widehat{\boldsymbol{t}}, \widehat{\boldsymbol{R}}, \widehat{\boldsymbol{z}}, \ell)$, next time t'.
Output: updated states $(\boldsymbol{t}^{t'}, \boldsymbol{R}^{t'}, \boldsymbol{z}^{t'}, \pi^{t'})$.

1. Update translation distribution parameters with the Gaussian BFN rule using endpoint $\widehat{\boldsymbol{t}}$ and $\sigma_{1,\mathrm{coord}}$.
2. Update local-geometry distribution parameters with the Gaussian BFN rule using endpoint $\widehat{\boldsymbol{z}}$ and $\sigma_{1,z}$.
3. Update rotational state on $\mathrm{SO}(3)$ using the Matrix–Fisher path centered at $\widehat{\boldsymbol{R}}$.
4. Update the categorical amino-acid distribution using logits ℓ.
5. Return the updated hierarchical state; pocket residue states are copied unchanged.

A.4 All-Atom Geometry Reconstruction

The generated continuous local state is decoded through the frozen FSQ projection and a trainable atom-coordinate head:

$$\widehat{q}_i = \mathrm{round}(\mathrm{bound}(\widehat{\boldsymbol{z}}_i)) \oslash \boldsymbol{m}, \qquad c_i = W_{\mathrm{out}} \widehat{q}_i, \tag{S6}$$

$$\begin{aligned} \widehat{\boldsymbol{X}}_i^{\mathrm{local,MLP}} &= D_\psi([\boldsymbol{h}_i^L; c_i]), \\ \widehat{\boldsymbol{x}}_{ia}^{\mathrm{global,MLP}} &= \widehat{\boldsymbol{R}}_i \widehat{\boldsymbol{x}}_{ia}^{\mathrm{local,MLP}} + \widehat{\boldsymbol{t}}_i. \end{aligned} \tag{S7}$$

Torsions extracted from the coordinate-head output are then combined with rigid-group templates and reference bond geometry to obtain the final atom14 structure.

Algorithm S4: Hierarchical state to all-atom structure

Input: generated state $(\widehat{\boldsymbol{t}}_i, \widehat{\boldsymbol{R}}_i, \widehat{\boldsymbol{z}}_i, \widehat{a}_i)$, residue embedding $\boldsymbol{h}_i^L$, frozen W_{out}, atom-coordinate head D_ψ.
Output: final atom14 coordinates $\widehat{\boldsymbol{X}}_i^{\mathrm{rec}}$.

1. Project the continuous local state through fixed FSQ: $\widehat{q}_i \leftarrow \mathrm{round}(\mathrm{bound}(\widehat{\boldsymbol{z}}_i)) \oslash \boldsymbol{m}$.

2. Map the FSQ code to contextual local-geometry features: $c_i \leftarrow W_{\text{out}}\widehat{q}_i$.
3. Decode local atom coordinates: $\widehat{\boldsymbol{X}}_i^{\text{local,MLP}} \leftarrow D_\psi([\boldsymbol{h}_i^L; c_i])$.
4. Transform local coordinates to global coordinates using $(\widehat{\boldsymbol{t}}_i, \widehat{\boldsymbol{R}}_i)$.
5. Select the valid atom slots from generated identity $\widehat{a}_i$.
6. Extract side-chain torsions from the coordinate-head output and rebuild atom14 coordinates using fixed rigid-group templates and reference geometry.

B LEAKAGE-CONTROLLED TARGET FILTERING

All evaluation tables in the main paper use the same leakage-controlled target set. The original common evaluation pool contains 187 unique target complexes, indexed by their chain-resolved `PDB_chain` identifiers. Before inspecting any generated structures or metric values, we remove a target if its `PDB_chain` identifier appears in either the PepSAGE training set or the PepFlow training set. This union filter removes 67 targets and leaves 120 targets for all reported comparisons.

The filtering is applied at the target level, not at the sample level. For generated samples whose identifiers contain a batch suffix, the suffix is first removed to recover the underlying `PDB_chain` identifier; all candidate samples from a filtered target are then excluded together. The same retained target list is used for PepSAGE, PepFlow, PepGLAD, PepBFN, the Discrete Token control, temperature sweeps, ablations, and paired statistical tests.

Table S1: Leakage-controlled filtering of the evaluation targets.

Evaluation set	Number of targets
Original common target pool	187
Removed by train–test overlap union	67
Final leakage-controlled target set	120

The excluded identifiers are stored in pep_sage/data_analyze/leak_union.txt. All leakage-affected tables are regenerated by pep_sage/data_analyze/recompute_clean_tables.py, which applies the same target filter and writes the clean-subset CSV files used for the paper. The command-line option `--leak-union` specifies the excluded-ID file; when omitted, the script defaults to the file above if it is present. The script also fixes the paired-bootstrap seed to 1234 by default and uses 10,000 bootstrap resamples unless otherwise specified.

C TRANSFER ACROSS GENERATIVE PROCESSES

To test whether the continuous local-geometry interface is tightly coupled to the BFN sampler, we trained a DDPM variant that keeps the same hierarchical state, frozen local-geometry interface, fixed FSQ projection, atom-coordinate head, and geometry-constrained reconstruction. Table S2 compares this variant with the BFN model used in the main paper.

Table S2: Transfer analysis using the same local-geometry interface with BFN and DDPM generative processes.

Metric	BFN	DDPM
Backbone RMSD ↓	**3.360**	3.397
All-atom RMSD ↓	6.060	**5.972**
Side-chain RMSD ↓	6.900	**6.868**
Bond angle dev. ↓	10.46	**9.58**
Rotamer recovery ↑	0.163	**0.178**

The two variants obtain comparable core RMSD, and the DDPM variant slightly improves several local-geometry metrics. This result supports the view that the continuous pre-quantization local-geometry state and reconstruction interface can be used beyond the BFN setting tested in the main

model. We do not interpret this as evidence of full generator-agnostic behavior, since only two generative processes are tested here.

D Additional Ablation and Temperature Details

The ablation in the main paper should be interpreted conservatively. Replacing the continuous pre-quantization state with a coarsened post-quantization representation weakens joint $\chi_{1:2}$ recovery (0.105), showing that local geometric information is lost after quantization. Removing the state entirely is more damaging. Together, these controls support the combination of a consistent continuous local-geometry state and geometry-constrained reconstruction, rather than relying on a coarsened structural code or on residue identity and backbone frame alone.

For the temperature sweep, Table S3 gives the full numerical curve. Lower temperature produces the most accurate samples: at $T = 0.5$, backbone RMSD is 3.343 Å and diversity is 0.244. Increasing temperature raises diversity and novelty, reaching diversity 0.546 and novelty 0.974 at $T = 2.0$, but also increases backbone RMSD to 4.249 Å. The geometric metrics degrade gradually rather than collapsing: the clash score remains within 0.0125–0.0171, and bond-angle deviation stays near the default operating point even at $T = 2.0$ ($10.47°$).

Table S3: Full temperature sweep on 120 leakage-controlled targets. BB, AA, and SC denote backbone, all-atom, and side-chain RMSD, respectively. Bond angle is reported in degrees.

T	BB↓	AA↓	SC↓	Contact↑	Div.↑	Novel↑
0.5	3.343	6.005	6.758	0.854	0.244	0.762
0.8	3.346	6.013	6.809	0.844	0.331	0.775
1.0	3.360	6.060	6.900	0.840	0.397	0.810
1.5	3.741	6.352	7.267	0.811	0.519	0.944
2.0	4.249	6.751	7.585	0.733	0.546	0.974

T	AAR↑	Bond len.↓	Angle↓	Clash↓	Rotamer↑
0.5	0.214	0.112	9.043	0.0131	0.155
0.8	0.217	0.125	9.734	0.0127	0.161
1.0	0.207	0.139	10.459	0.0125	0.163
1.5	0.194	0.152	11.016	0.0131	0.156
2.0	0.192	0.144	10.466	0.0171	0.153

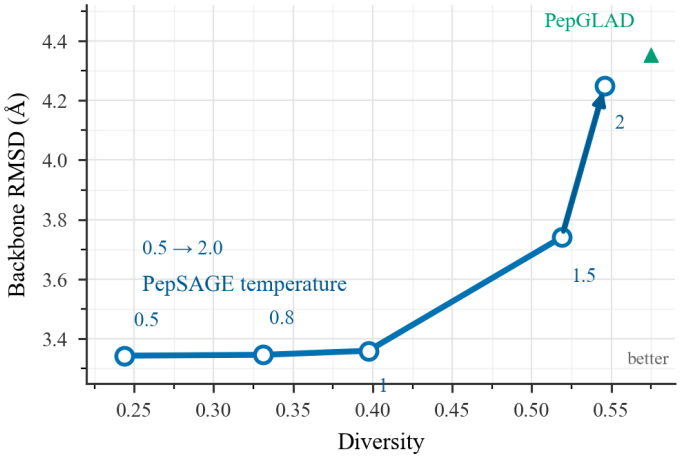

Figure S1: Temperature sweep of the empirical quality–diversity frontier.

Table S4: Seed robustness over three PepSAGE runs on the 120 leakage-controlled targets. Per-seed values are shown where stored in the experiment record; otherwise we report the run-level mean and standard deviation.

Metric	Mean ± std	1234	4321	8989
BB RMSD↓	3.387 ± 0.080	3.360	3.476	3.324
AA RMSD↓	5.841 ± 0.220	6.060	5.842	5.621
SC RMSD↓	6.651 ± 0.221	6.900	6.574	6.480
Contact↑	0.846 ± 0.005	–	–	–
Div.↑	0.373 ± 0.023	–	–	–
Novel↑	0.794 ± 0.030	–	–	–

E Evaluation Metric Definitions

All reported metrics are computed after matching generated and native peptide residues by sequence position and using the heavy-atom mask implied by the corresponding residue identity. Cα backbone RMSD is computed after optimal superposition of matched Cα atoms. All-atom RMSD and side-chain RMSD are computed over matched heavy atoms after a Kabsch alignment fitted to all matched heavy atoms. For local side-chain analysis, BB-SC RMSD instead aligns the backbone atoms and then evaluates side-chain heavy atoms.

Contact recovery measures pocket consistency. Receptor residues whose Cα atoms lie within 10 Å of the native peptide define the native contact set; the score is the fraction of these contacts recovered by the generated complex under the same cutoff. Rotamer recovery is computed over valid side-chain torsions. A χ angle is counted as recovered when its circular error is at most 20°, and joint $\chi_{1:k}$ accuracy requires all valid torsions through depth k to satisfy this threshold.

Geometric validity is measured by residue-internal heavy-atom bond-length and bond-angle deviations from the native structure, together with a clash score. The clash score is the fraction of nonbonded heavy-atom pairs closer than 1.6 Å. Sample coverage is summarized by diversity, novelty, and amino-acid recovery (AAR). Diversity is the within-target pairwise average of $(1-\mathrm{TMscore})(1-\mathrm{sequence\ similarity})$ among generated samples. A generated sample is counted as novel when both its native-normalized TM-score and native sequence similarity are at most 0.5. AAR is the sequence-similarity ratio between the generated peptide and the native peptide.

F Full Paired Bootstrap Results

Table S5 reports the full paired target-level bootstrap analysis for the overall metrics used in the main paper. Differences are defined as PepSAGE minus the baseline, so negative values favor PepSAGE for lower-is-better metrics and positive values favor PepSAGE for higher-is-better metrics. Bold entries indicate that the 95% confidence interval excludes zero and that the direction favors PepSAGE.

Table S5: Full paired target-level bootstrap results on the 120 leakage-controlled targets. Each cell reports the mean paired difference and 95% confidence interval, with differences defined as PepSAGE minus the baseline. Disc. denotes the Discrete Token representation control.

Metric	vs PepFlow	vs PepGLAD	vs PepBFN	vs Disc.
Backbone RMSD ↓	**-0.563 [-0.742, -0.382]**	**-0.997 [-1.19, -0.821]**	**-1.96 [-2.41, -1.52]**	**-2.13 [-2.56, -1.68]**
All-atom RMSD ↓	**-1.32 [-1.60, -1.05]**	**-1.58 [-1.85, -1.32]**	**-1.62 [-2.03, -1.22]**	**-2.09 [-2.51, -1.68]**
Side-chain RMSD ↓	**-1.22 [-1.48, -0.955]**	**-1.55 [-1.81, -1.30]**	**-2.61 [-2.98, -2.24]**	**-1.96 [-2.34, -1.57]**
Contact ↑	**+0.061 [+0.039, +0.085]**	-0.055 [-0.073, -0.034]	**+0.131 [+0.106, +0.157]**	**+0.159 [+0.137, +0.182]**
Diversity ↑	-0.137 [-0.148, -0.126]	-0.177 [-0.191, -0.164]	-0.072 [-0.085, -0.060]	**+0.069 [+0.056, +0.081]**
Novelty ↑	-0.107 [-0.142, -0.074]	-0.157 [-0.198, -0.117]	-0.176 [-0.222, -0.133]	-0.176 [-0.222, -0.133]
AAR ↑	-0.0087 [-0.023, +0.0064]	-0.017 [-0.031, -0.0029]	**+0.062 [+0.043, +0.083]**	**+0.040 [+0.021, +0.060]**
Bond length dev. ↓	+0.025 [+0.022, +0.029]	**-0.095 [-0.109, -0.083]**	**-0.034 [-0.043, -0.025]**	**-1.03 [-1.04, -1.03]**
Bond angle dev. ↓	+1.40 [+1.21, +1.60]	**-27.55 [-28.22, -26.93]**	**-1.54 [-2.05, -1.05]**	**-3.10 [-3.48, -2.73]**
Clash ↓	**-0.0016 [-0.0018, -0.0015]**	**-0.010 [-0.013, -0.0083]**	**-0.0027 [-0.0029, -0.0025]**	**-0.067 [-0.073, -0.060]**
Rotamer recovery ↑	**+0.041 [+0.034, +0.049]**	**+0.026 [+0.017, +0.034]**	**+0.058 [+0.046, +0.069]**	**+0.041 [+0.028, +0.054]**

G Additional Discussion and Limitations

PepSAGE is evaluated in this work for pocket-conditioned peptide generation, and its conclusions should be interpreted within that setting. The current experiments do not establish performance for other biomolecular design tasks, such as unconditional peptide generation, protein binder generation beyond short peptides, or joint receptor–ligand flexibility.

The low-dimensional local-geometry state is an empirical design choice. We use a compact continuous state because it provides a uniform interface across residue types and works well in the reported ablations, but we do not claim that this dimensionality is theoretically optimal. In addition, the local interface is learned by a bidirectional encoder over local structural context; the resulting state should therefore be viewed as a reconstruction-learned local geometry interface rather than a strictly residue-independent descriptor.

Several trade-offs remain. PepSAGE improves RMSD, clash, and rotamer recovery, but its diversity is lower than PepFlow and PepGLAD under the default sampling temperature. Temperature sampling partially moves the model along an empirical quality–diversity frontier, but it does not remove the trade-off. For side-chain conformation, PepSAGE improves χ_1 and $\chi_{1:2}$ recovery while still trailing PepGLAD on deeper χ_3/χ_4 accuracies. The geometry-constrained reconstruction stage also introduces a modest coordinate RMSD trade-off while improving local bond geometry and clashes.

Finally, the transfer analysis in Table S2 is limited to the generative processes tested in this study. Although the same local-geometry interface can be used with both BFN and DDPM variants, this does not by itself establish full generator-agnostic behavior. Broader validation across additional generative families and larger peptide-design benchmarks is left for future work.

H Local-Geometry Interface Configuration

The continuous local-geometry interface is trained separately and then frozen for PepSAGE generation. Table S6 lists its architecture, quantization, and reconstruction objective. The interface receives only local-frame heavy-atom coordinates: residue-type, atom-slot, and atom-class inputs are all disabled, so the six-dimensional state is learned purely from local geometry. Its nonzero training objectives are local- and global-coordinate reconstruction MSE together with intra-residue distance consistency; bond, angle, and clash terms are set to zero for the reported checkpoint.

The compact reconstruction objective used to train the interface is

$$\mathcal{L}_{\text{interface}} = \mathcal{L}_{\text{local-coord}} + \mathcal{L}_{\text{global-coord}}. \tag{S8}$$

Valid atoms are serialized in canonical atom14 order with invalid slots omitted; this sequence order acts only as a preprocessing convention, not as an additional learned atom-label input. Bidirectionality is implemented without doubling the Mamba state-space parameters. In each encoder block, the same Mamba mixer is applied to the canonical atom sequence and to its reversed sequence; the reversed output is flipped back and summed with the forward output before the residual update. Thus, each atom feature can use context from both directions while sharing one set of mixer weights. The interface compression factor is one, so each valid atom retains one projected pre-quantization state before residue-level mean pooling.

I Bayesian Flow Details

This section gives the weighting and sampling details abbreviated in the main method section. Translations and local-geometry states use Gaussian BFN paths (Graves et al., 2023). For a clean endpoint $x \in \mathbb{R}^d$, the time-t input state is centered at

$$\begin{aligned} \gamma(t) &= 1 - \sigma_1^{2t}, \\ \mu_t &= \gamma(t)x + \sqrt{\gamma(t)(1-\gamma(t))}\,\epsilon, \quad \epsilon \sim \mathcal{N}(0, I). \end{aligned} \tag{S9}$$

The network predicts the clean endpoint $\widehat{x}$ and is trained with the weighted endpoint objective

$$\mathcal{L}_{\text{cont}}(x; \sigma_1) = \mathbb{E}_{k,r}\left[w_k(\sigma_1)\, \|\widehat{x} - x\|_2^2 \right], \tag{S10}$$

Table S6: Frozen local-geometry interface: architecture, quantization, and reconstruction objective. The interface uses coordinate reconstruction and intra-residue distance consistency, with no bond, angle, or clash penalties; all inputs are local-frame coordinates with no residue- or atom-type labels.

Component	Setting
Encoder	Bidirectional Mamba
Encoder layers	4
Model dimension	128
Reconstruction decoder layers	6
Normalization	RMSNorm
Residue-type / atom-type inputs	Disabled
Input feature	Local-frame coordinate
Compression factor	1 (one state per atom)
Quantizer	FSQ
FSQ levels	[4, 4, 4, 4, 4, 4]
Local-state dimension	6
Residue-level pooling	Mean over valid atoms
Local-coordinate MSE weight	1.0
Global-coordinate MSE weight	1.0
Intra-residue distance weight	0.5
Bond / angle / clash weights	0.0

averaged over discrete BFN steps k and generated residues r. We use

$$\begin{aligned} \mathcal{L}_{\text{trans}} &= \mathcal{L}_{\text{cont}}(\boldsymbol{t}; \sigma_{1,\text{coord}}), \\ \mathcal{L}_{z} &= \mathcal{L}_{\text{cont}}(\boldsymbol{z}; \sigma_{1,z}), \end{aligned} \tag{S11}$$

with $\sigma_{1,\text{coord}} = 0.03$, $\sigma_{1,z} = 0.10$, and $K_{\text{BFN}} = 1{,}000$.

For Gaussian BFN paths, training samples a discrete time index k uniformly from $\{1, \ldots, K_{\text{BFN}}\}$. The continuous endpoint loss uses

$$w_k(\sigma_1) = \frac{K_{\text{BFN}} \left(1 - \sigma_1^{2/K_{\text{BFN}}}\right)}{2\sigma_1^{2k/K_{\text{BFN}}}}, \tag{S12}$$

with $K_{\text{BFN}} = 1{,}000$, $\sigma_{1,\text{coord}} = 0.03$, and $\sigma_{1,z} = 0.10$ for translations and local-geometry states, respectively.

For rotational states, we use an isotropic Matrix–Fisher perturbation on $\mathrm{SO}(3)$ (Qian et al., 2026). Given a clean frame $\boldsymbol{R}$, its time-t state is

$$\boldsymbol{R}_t = \boldsymbol{R} Q_t, \qquad Q_t \sim \mathcal{M}\left(\lambda(t)^2 I_3\right), \tag{S13}$$

where $\mathcal{M}(F)$ denotes a Matrix–Fisher distribution with density proportional to $\exp(\mathrm{tr}(F^\top Q))$. The network predicts the clean frame $\widehat{\boldsymbol{R}}$ and minimizes a weighted trace loss,

$$\mathcal{L}_{\text{rot}} = K_{\text{BFN}}\, \mathbb{E}_{k,r}\left[w_R(k) \left(3 - \mathrm{tr}(\boldsymbol{R}^\top \widehat{\boldsymbol{R}})\right)\right], \tag{S14}$$

where the trace distance is minimized when predicted and target frames coincide. The Matrix–Fisher concentration follows

$$\lambda(t) = \lambda_1 \frac{\exp(2t) - 1}{\exp(2) - 1}, \qquad \lambda_1 = 10. \tag{S15}$$

The weighted trace loss uses

$$w_R(t) = \lambda(t) \left(1 - \frac{1}{2\lambda(t) + 1}\right). \tag{S16}$$

The expectation is averaged over generated residues. Rotation noise is sampled by rejection sampling at lower concentration and by a local Lie-algebra approximation at higher concentration.

J Context Embedding and Trunk Details

For leakage-controlled context construction, pocket residue identities, atomic coordinates, and backbone dihedrals remain visible. Peptide identities are replaced by an unknown token, and peptide local

coordinates and backbone-dihedral features are zeroed. Pair features that depend on the unknown peptide structure are hidden.

Node features combine amino-acid embeddings, type-indexed local atom coordinates, and backbone-dihedral encodings. Edge features combine amino-acid pair embeddings, same-chain sequence offsets, Gaussian encodings of atom-pair distances, and pairwise backbone-dihedral encodings. Cross-chain sequence offsets and target-dependent pocket–peptide geometric features are not exposed.

At each BFN step, the leakage-free context embedding is combined with the current local state and a time embedding:

$$h_i^{(0)} = \mathrm{MLP}\left[h_i^{\mathrm{ctx}};\, E_z(\boldsymbol{z}_{t,i});\, E_t(t)\right], \tag{S17}$$

where $E_z : \mathbb{R}^6 \to \mathbb{R}^{128}$.

The IPA–Transformer trunk contains six blocks. Each block contains eight-head invariant point attention (IPA), a two-layer four-head residue Transformer, a node transition, and a rigid-frame update; edge features are updated after all but the final block. IPA combines scalar query–key similarity, learned pair bias, and distances between query and key points transformed by the current residue frames.

K Local-Geometry Reconstruction Details

The generated continuous state is projected through the frozen FSQ interface using

$$\widehat{q}_i = \mathrm{round}(\mathrm{bound}(\widehat{\boldsymbol{z}}_i)) \oslash \boldsymbol{m}, \qquad c_i = W_{\mathrm{out}}\widehat{q}_i, \tag{S18}$$

where $\oslash$ denotes elementwise division and $\boldsymbol{m} = \lfloor \boldsymbol{L}/2 \rfloor = (2, \ldots, 2)$ for FSQ levels $\boldsymbol{L} = (4, \ldots, 4)$. The projection W_{out} is frozen, so continuous generation and scalar projection play distinct roles.

The atom-coordinate head outputs at most 15 heavy-atom slots. The predicted amino-acid identity selects the valid residue-specific atom subset. Torsions are extracted from the first 14 MLP atom slots and used with rigid-group templates and reference bond geometry to rebuild the standard atom14 representation. The optional fifteenth atom slot is retained directly from the MLP output.

L Sampling and Evaluation Details

Inference processes one target complex at a time. The input complex is replicated 64 times to generate 64 candidate peptides for each target. During sampling, the receptor pocket provides fixed structural context, whereas translations, rotations, amino-acid distributions, and local-geometry states are updated only at masked peptide positions.

At the endpoint, amino-acid identities are selected by the argmax of the final categorical logits. The generated continuous local states are decoded into atom coordinates and then passed through geometry-constrained reconstruction. Heavy-atom masks are recomputed from the decoded residue identities. When serializing each candidate, generated coordinates replace only the masked peptide residues, receptor coordinates are copied from the input complex, and the complete pocket–peptide complex is saved as a PDB file.

M Architecture and Training Hyperparameters

Table S7 provides the full architecture and optimization settings abbreviated in the main text.

We use

$$(\lambda_t, \lambda_R, \lambda_z, \lambda_a, \lambda_s) = (4.0, 0.03, 0.5, 0.5, 1.0). \tag{S19}$$

Table S7: Architecture and training hyperparameters.

Component	Setting
Node / IPA single dim	128
Pair dim	64
IPA blocks	6
IPA heads	8
Query/key points per head	8
Value points per head	12
Transformer layers per block	2
Transformer heads	4
Local-state dimension	6
FSQ levels	$[4, 4, 4, 4, 4, 4]$
Atom-coordinate head	256–256 MLP
Max heavy-atom slots	15
Batch size	16
Optimizer	Adam
Learning rate	5×10^{-4}
Adam betas	$(0.95, 0.999)$
Weight decay	0
Scheduler	Cosine, 20k steps
Minimum learning rate	10^{-6}
EMA decay	0.999
Gradient clipping	50
Coordinate normalization	Divide by 5.0
Maximum epochs	200
Reported checkpoint budget	180 epochs
Random seed	1234