Generating Plausible Sequences of Protein Groups with Latent Optimal Transport Flow Matching

Zitai Kong[1], Yinlong Xu[1], Xiaohong Jiang[1], Jian Wu[3,4,5,†], and Hongxia Xu[2,†]
[1]College of Computer Science and Technology, Zhejiang University, Hangzhou, China
[2]Liangzhu Laboratory and Innovation Institute for Artificial Intelligence in Medicine, Zhejiang University, Hangzhou, China
[3]The Second Affiliated Hospital, Zhejiang University School of Medicine, Hangzhou, China
[4]State Key Laboratory of Transvascular Implantation Devices and TIDRI, Hangzhou, China
[5]Zhejiang Key Laboratory of Medical Imaging Artificial Intelligence, Hangzhou, China
{kongzitai, 22321336, jiangxh, wujian2000, Einstein}@zju.edu.cn

***Abstract*—The *de novo* design of protein sequences with targeted functionalities is a significant task in bioengineering. Deep generative methods such as autoregressive models and diffusion models have promoted the rapid discovery of novel protein sequences. However, these algorithms mainly suffer from expensive computing costs, low inference efficiency, and underused global protein information on both sequences and space. To overcome these issues, we introduce *PLFM*, an optimal transport conditional flow matching-based protein sequence designer operating on latent embeddings of protein language models. Additionally, we develop a joint design pipeline for the design scene of multichain proteins. We apply PLFM to various protein design tasks, including representative protein types: general peptides, general proteins, antimicrobial peptides, and antibodies. Taking advantage of advanced flow matching and pLM embeddings, PLFM surpasses task-specific methods in these targeted applications, highlighting its significant potential and broad applicability in computational protein design and analysis.**

***Index Terms*—Flow matching, Optimal transport, Generative AI, Protein design, Sequence design.**

I. Introduction

Proteins constitute a highly structured yet diverse class of biological sequences whose functional behaviors emerge from complex interactions across multiple scales. Designing proteins that retain such multifunctionality has therefore become a central objective in modern protein engineering [1]. Generative modeling now provides a new computational paradigm for navigating protein sequence spaces, enabling data-driven exploration beyond the limits of manual design [2], [3].

Early progress in this area was largely inspired by advances in natural language processing, where protein sequences were modeled as symbolic sequences amenable to autoregressive (AR) generation. This sequential and directional generation process imposes strong conditional dependencies on previously generated residues, which may constrain the model's ability to capture the global organization of protein sequences [4]. This observation has motivated growing interest in non-autoregressive alternatives, particularly diffusion-based models, which aim to model sequences through iterative refinement [5], [6].

†: Corresponding authors.

Despite their conceptual appeal, diffusion models often face challenges such as long generation times, high computational costs, and sensitivity to hyperparameters [7], [8]. Even with recent improvements in score-based formulations, the restricted family of sampling paths can still limit the quality and diversity of generated sequences [9]. These challenges suggest that a more direct and globally consistent formulation of probability transport may be beneficial for protein sequence generation.

As a novel generative paradigm, flow matching (FM) [10] provides such a formulation by learning a deterministic vector field that maps a simple prior distribution to the data distribution without explicitly simulating stochastic trajectories. Compared to diffusion models, FM requires substantially fewer integration steps and provides a more flexible description of probability paths. However, extending FM to protein sequence generation is nontrivial, as symbolic protein sequences do not naturally align with continuous flow formulations.

To bridge this gap, we draw inspiration from latent-variable generative modeling and consider embedding discrete protein sequences into a continuous latent space suitable for flow-based learning [11]. Protein language models (pLMs) pretrained on evolutionary-scale protein corpora offer a natural foundation for such representations [12], [13]. By operating in the semantically meaningful latent spaces induced by pLMs, we enable FM to model protein sequences more effectively while retaining biologically relevant information, including residue-level properties and higher-order structural cues [14].

In this paper, we introduce **PLFM** (**P**rotein **L**atent **F**low **M**atching), an optimal-transport conditional flow-matching model for *de novo* protein sequence generation based on pLM embeddings. We also comprehensively evaluate the model in various practical protein design scenarios. Our main contributions are as follows:

- We introduce PLFM, which explores fast protein sequence design via optimal-transport flow matching in the latent space of pLMs.
- We experimentally demonstrate the advantages of flow matching over traditional diffusion models and show the improvements brought by latent-space design based on large-scale pLMs.

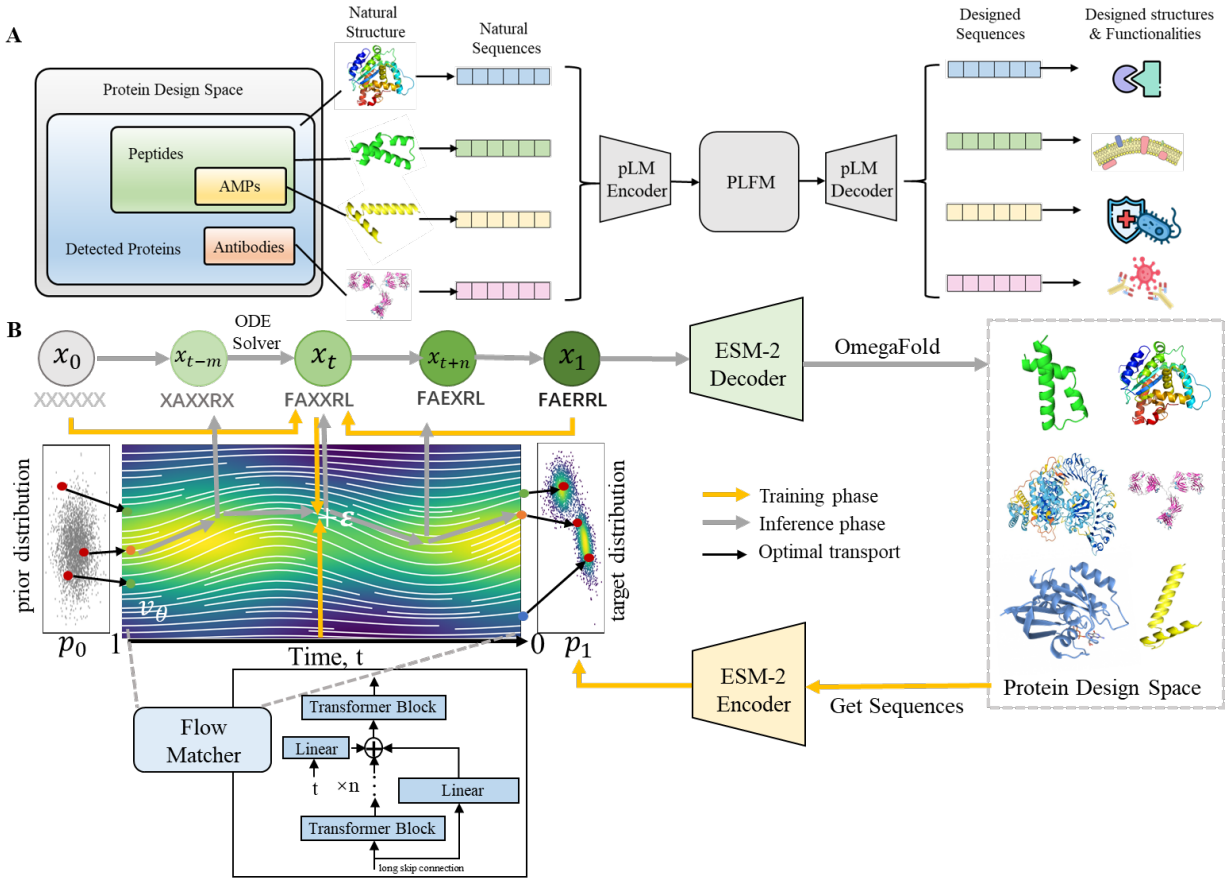

Fig. 1. Overview of PLFM for protein sequence design. (A) Left: Relationships of different protein groups. Right: PLFM is trained on various natural protein sequence subsets from the protein design space encoded by pLMs, generating new sequences with potentially desired structures and functions; (B) the visualization illustrates the mathematical principles and model architecture of PLFM, including the training and inference phases.

- We develop a multichain joint design pipeline and showcase PLFM's capability to generate structurally plausible, reliable, and natural protein sequences across various design tasks, including general peptides, general proteins, antimicrobial peptides, and antibodies, consistently outperforming existing methods.

II. Background

A. Problem formulation

A protein sequence x of length L is a string of amino acids, $x = [a_1, a_2, \ldots, a_L] \in \mathcal{V}^L$, where $\mathcal{V}$ denotes the amino acid vocabulary. *De novo* protein sequence design aims to generate novel protein sequences from scratch according to the learned protein distribution, i.e., $x \sim p(x)$.

B. Preliminaries

1) Conditional Flow Matching: Inspired by score-based generative models [15], we define the mapping between samples from the source/prior distribution $x_0 \sim p_0$ and the target distribution $x_1 \sim p_1$ as the ordinary differential equation

$$dx = u(x,t)dt \tag{1}$$

where $x \sim p_t(x)$ and $u : \mathbb{R}^d \times [0,1] \to \mathbb{R}^d$ is a vector field parameterized by a neural network v_θ. The model is then trained with

$$\mathcal{L}_{\text{FM}} = \mathbb{E}_{t,x} \| v_\theta(x,t) - u(x,t) \|_2^2. \tag{2}$$

In the basic formulation of **Flow Matching** [10], directly obtaining $p_t(x)$ can be difficult in practice. In this case, we introduce a Gaussian approximation conditioned on a random variable z:

$$p_t(x) = \mathbb{E}_{z \sim q(z)} p_t(x|z) = \mathcal{N}(x|\mu_t(z), \sigma_t^2(z)). \tag{3}$$

The vector field at time t, denoted by u_t, can be analytically defined as

$$u_t(x|z) = \frac{\sigma_t'(z)}{\sigma_t(z)}(x - \mu_t(z)) + \mu_t'(z). \tag{4}$$

The objective function of **Conditional Flow Matching (CFM)** can then be defined as

$$\mathcal{L}_{\text{CFM}} = \mathbb{E}_{t, z \sim q(z), x \sim p_t(x|z)} \| v_\theta(x,t) - u_t(x|z) \|_2^2. \tag{5}$$

2) Optimal Transport CFM: *Optimal transport* seeks the minimum displacement cost between two distributions under a given ground cost. **Optimal Transport Conditional Flow**

Matching (OT-CFM) [16] uses the 2-Wasserstein distance to measure this displacement:

$$\mathcal{W}(p_0, p_1)_2^2 = \inf_{x \in \Pi} \int_{\chi^2} c(x, y)^2 \pi(dx, dy) \tag{6}$$

where c is the ground cost, and Π is the set of all joint probability measures on χ^2 whose marginals are p_0 and p_1.

By sampling z using the 2-Wasserstein optimal mapping,

$$q(z) = \mathcal{W}(x_0, x_1), \tag{7}$$

flow matching can achieve improved transport paths through optimal transport.

III. Methods

As depicted in Figure 1, PLFM comprises three key components: the pLM encoder E_ϕ, the flow matcher v_θ, and the pLM decoder D_ψ. During training, the encoder maps discrete protein sequences into a latent space, while the flow matcher learns the vector field. During inference, the flow matcher uses an ODE solver to generate protein embeddings from Gaussian noise, and the decoder translates these embeddings back into protein sequences. This section describes the adapted FM algorithm, the model architecture of each component, the joint design pipeline for multichain proteins, and the implementation details.

A. Latent Space Design

To adapt continuous FM models to the inherently discrete nature of protein sequences, we adopt an encoder–decoder framework. Instead of training a VAE-like model from scratch, we adapt the embedding space of pLMs to incorporate protein semantic information obtained from pretraining on large-scale protein sequence databases.

1) Encoder: For the encoder, we use the pretrained transformer-based ESM-2 [12], one of the strongest pLMs, as E_ϕ. The encoder transforms the amino acid sequence $x = [a_1, a_2, \ldots, a_L] \in \mathcal{V}^L$ into a continuous embedding $E_\phi(x) = [e_1, e_2, \ldots, e_L] \in \mathbb{R}^{L \times D}$ with latent dimension D. To standardize the sequence length L, we append ESM-2 padding tokens to shorter proteins. In addition, we apply dimension-wise z-score normalization to better align the embeddings with a standard Gaussian distribution. To study the effect of encoder and decoder capacity, we use two ESM-2 variants with different parameter scales (8M and 35M).

2) Decoder: Correspondingly, we use the decoder D_ψ to reconstruct the protein sequence from $E_\phi(x)$. We adopt the ESM-2 LMHead architecture, which consists of two linear layers, an activation layer, and a layer normalization module. To tailor the pretrained language model to specific tasks and improve reconstruction accuracy, we fine-tune the decoder independently using the following reconstruction objective:

$$\mathcal{L}_{\text{reconstruct}} = \text{CrossEntropyLoss}(x, D_\psi(E_\phi(x))). \tag{8}$$

Before decoding, $E_\phi(x)$ is denormalized, and padding tokens are removed from the reconstructed sequences. During flow-matcher training, the parameters of both the encoder and decoder are frozen. This design keeps training stable, controllable, and flexible.

B. Flow Matching in Latent Space

1) Training with OT-CFM: Given an encoded protein latent $z_1 \sim E_\phi(p_1)$, we aim to generate it from standard Gaussian noise $z_0 \sim \mathcal{N}(0, I)$ by learning a vector field over the time interval $t \in [0, 1]$.

To improve the determinism of the probability path computation, we first compute the optimal transport in Equation (7), i.e., $z \sim q(z) = \mathcal{W}(z_0, z_1)$. To simplify the computation, we define the Gaussian approximation path $p_t(x|z)$ in Equation (3) using the mean of the linear interpolation between z_0 and z_1, with $t \sim \text{Uniform}(0, 1)$ and a constant variance [17]:

$$\mu_t(z) = t z_1 + (1 - t) z_0, \tag{9}$$

$$\sigma_t^2(z) = \sigma^2 I. \tag{10}$$

Then, the vector field defined in Equation (4) reduces to a straight line associated with the OT-paired sample $z = (z_0, z_1)$:

$$u_t(z_t|z) = z_1 - z_0. \tag{11}$$

We train the flow matcher v_θ with the loss

$$\mathcal{L}_{\text{OT-CFM}} = \mathbb{E}_{t, z \sim q(z), z_t \sim p_t(z_t|z)} \|v_\theta(z_t, t) - (z_1 - z_0)\|_2^2. \tag{12}$$

The training details of PLFM are given in Algorithm 1.

Algorithm 1 PLFM Training

Require: data distribution p_1, flow network v_θ, encoder E_ϕ, number of training iterations N
for $k = 1$ to N **do**
  Sample protein sequences $x_1 \sim p_1$
  Sample $t \sim \text{Uniform}(0, 1)$
  Encode the padded sequences into latent targets:

$$z_1 \leftarrow E_\phi(\text{padding}(x_1))$$

  Sample latent priors and perturbation noise:

$$z_0 \sim \mathcal{N}(0, I), \qquad \epsilon \sim \mathcal{N}(0, \sigma^2 I)$$

  Compute the exact OT coupling between z_0 and z_1 under squared Euclidean cost, and pair $(z_0, z_1) \leftarrow \mathcal{W}(z_0, z_1)$
  Construct the noisy interpolation:

$$z_t \leftarrow (1 - t) z_0 + t z_1 + \epsilon$$

  Compute the OT-CFM loss:

$$\mathcal{L}(\theta) \leftarrow \|v_\theta(z_t, t) - (z_1 - z_0)\|_2^2$$

  Update θ with $\nabla_\theta \mathcal{L}(\theta)$
end for
Output: trained flow network v_θ

2) Generation process: During inference, we sample a standard Gaussian latent and iteratively solve the learned ODE, after which the generated latent is decoded back into a protein sequence. In addition, protein sequence length is implicitly encoded in the latent representation through trailing padding tokens. This may introduce ambiguity and lead to an inaccurate sequence-length distribution. To address this issue, we sample the sequence length from the empirical distribution observed in the training dataset and apply attention masks to ensure a clear delineation of sequence length. The inference details of PLFM are given in Algorithm 2.

Algorithm 2 PLFM Inference

1: **Require:** trained flow network v_θ, decoder D_ψ, number of ODE steps N, empirical length distribution P_L

2: Sample initial latent state:

$$z_0 \sim \mathcal{N}(0, I)$$

3: Generate the terminal latent by solving

$$\frac{dz(t)}{dt} = v_\theta(z(t), t), \qquad t \in [0, 1],$$

with a fixed-step ODE solver using N steps:

$$z_N \leftarrow \mathrm{ODESolve}(v_\theta, z_0, 0, 1, N)$$

4: Sample sequence length $L \sim P_L$

5: Decode and remove padding:

$$x \leftarrow \mathrm{unpadding}(D_\psi(z_N, L))$$

6: **Output:** generated protein sequence x

3) Model architecture: The flow matcher is implemented as a 12-layer Transformer with 16 attention heads. We use the GELU activation function, an intermediate layer size of 3072, an attention dropout rate of 0.1, and a hidden size of 320 for ESM-2 8M or 480 for ESM-2 35M. Time information is sinusoidally embedded and added to each layer through a linear projection. Inspired by the U-ViT architecture [18] and prior diffusion-based studies [19], we incorporate long skip connections into the Transformer to improve performance and facilitate more effective learning dynamics.

C. Multichain Joint Design Pipeline

For multichain proteins such as antibodies, significant interdependence often exists among individual chains. Designing each chain independently may compromise the structural integrity of the overall protein. Therefore, we develop a joint design pipeline for multichain proteins. Taking antibodies as an example, as shown in Figure 2, we fine-tune two separate ESM-2 models to encode and decode the heavy and light chains, and concatenate their embeddings so that they can be processed jointly by PLFM. During inference, the generated embeddings are split and independently converted back into protein sequences.

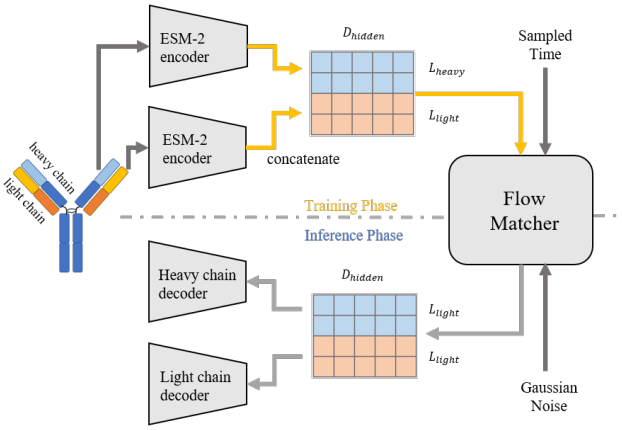

Fig. 2. Joint Design of Antibodies. Heavy chains and light chains are utilized to finetune ESM-2 decoders respectively. The embeddings are concatenated before fed into PLFM. D_{hidden} is the ESM-2 hidden dimension; L_{heavy} and L_{light} are the maximum lengths of the heavy and light chains.

D. Implementation Details

The flow matcher was trained with a learning rate of 0.0001, a batch size of 32, and the AdamW optimizer with beta values of $(0.9, 0.98)$ and a weight decay of 0.01. All experiments converged within 1,000,000 steps, taking approximately 42 hours. The ESM-2 decoder was trained with a learning rate of 0.00005, a batch size of 128, and the AdamW optimizer with beta values of $(0.9, 0.98)$ and a weight decay of 0.01. Decoder training converged within 5,000 steps, taking approximately 1 hour. All training was conducted on a single NVIDIA GeForce RTX 3090 GPU. For OT-CFM training, we used exact optimal transport to construct the coupling between prior and target samples within each mini-batch. Each protein latent embedding was first flattened, and the ground cost was defined as the squared Euclidean distance. During inference, we solved the learned ODE in latent space using a 4th-order Runge–Kutta (RK4) solver with 25 integration steps.

IV. Experiments

In this section, we first evaluate the model on general peptide and long-chain protein generation. Building on the strong results in these general settings, we then apply PLFM to the design of a practical functional protein class, namely antimicrobial peptides. Finally, we evaluate the model on the more challenging antibody design task, which involves multiple sequences and more complex structures. Through these experiments, we demonstrate PLFM's ability to design general and functional proteins with varying lengths and multiple chains, highlighting its potential and broad applicability in computational protein sequence design.

A. General Peptide and Protein Design

This section covers general protein design settings, including short-chain proteins, i.e., peptides, and long-chain proteins. In these settings, we evaluate PLFM and analyze the contribution of each major design choice. The results demonstrate its versatility and strong performance, laying a

TABLE I
PERFORMANCE COMPARISON BETWEEN PLFM AND BASELINE MODELS ON UNIPROT PEPTIDES AND SWISSPROT DATASETS.

	Model	pLDDT (↑)	ESM-2 pppl (↓)	scPerplexity (↓)	TM-score (↑)	FPD (↓)	MMD (↓)	OT (↓)	NFE (↓)
UniProt Peptides	Dataset	71.35	12.58	12.31	0.57	0.23	0.000	1.74	nan
	Random sequences	62.89	21.71	20.40	0.44	3.66	0.084	5.75	nan
	ProteinGAN	67.84	15.14	13.89	0.54	1.51	0.034	2.78	nan
	EvoDiff-OADM	64.99	14.26	13.61	0.50	0.60	0.014	2.29	1000
	EvoDiff-OADM w/ Transformer	63.78	17.76	14.75	0.52	1.24	0.051	2.72	1000
	DiMA	70.95	13.72	14.12	0.54	0.87	0.021	2.57	100
	Dirichlet FM	67.19	14.13	13.59	0.53	0.50	0.014	2.10	50
	PLFM ESM-2 8M	**71.09**	**12.94**	13.05	**0.56**	0.49	0.011	2.06	**25**
	PLFM ESM-2 35M	70.49	13.07	**13.03**	**0.56**	**0.47**	**0.010**	**2.02**	**25**
SwissProt	Dataset	79.36	9.88	5.21	0.80	0.27	0.002	1.37	nan
	Random sequences	26.31	21.17	14.84	0.33	3.07	0.205	3.88	nan
	proteinGAN	32.77	16.12	12.36	0.24	3.08	0.176	4.02	nan
	EvoDiff-OADM	39.36	15.97	10.76	0.2	2.17	0.112	3.50	1000
	DiMA	77.56	11.46	6.54	0.42	1.58	0.104	2.25	100
	Dirichlet FM	53.33	12.88	9.35	0.37	1.38	0.073	2.56	50
	PLFM	**79.65**	**8.77**	**4.24**	**0.47**	**1.13**	**0.059**	**2.35**	**25**

solid foundation for the subsequent application of PLFM to functional protein design.

1) Data: For general peptides, we collected peptides with lengths ranging from 2 to 50 from UniProt [20]. After removing sequences containing uncommon amino acids (B, U, J, O, and X) and lowercase characters, we obtained 2.6M peptides. For general long proteins, we follow the setting of [19], which uses SwissProt [21], a high-quality, manually annotated subset of UniProt. After filtering out sequences shorter than 128, truncating sequences longer than 254, and removing sequences with uncommon amino acids, we obtained a total of 470k proteins. The data were split into training and validation sets with a ratio of 9:1.

2) Baseline: Following and slightly modifying the benchmark setting of [19], we include the generative adversarial network (GAN) method ProteinGAN [22], the discrete diffusion method EvoDiff-OADM with both ByteNet-CNN and Transformer backbones [23], the continuous diffusion method DiMA [19], and the discrete FM method Dirichlet Flow Matching [24]. All baseline models are configured with the same or similar numbers of parameters and use a standard character-level tokenization scheme.

3) Metrics: We follow the benchmark setting of [19]. To evaluate sequence quality, we use ESM-2 Perplexity and ESM-IF scPerplexity. To evaluate structural foldability and quality, we use OmegaFold [25] pLDDT and TM-score against SwissProt. To reflect the ability to capture the training data distribution, we use Fréchet ProtT5 Distance (FPD), Maximum Mean Discrepancy (MMD), and 1-Wasserstein optimal transport (OT). For generation speed, we use the Number of Function Evaluations (NFE).

4) Results: The results are presented in Table I. We summarize the main findings as follows:

a) PLFM generates high-quality sequences with strong distributional fidelity: We evaluate the quality of the generated sequences using ESM-2 Perplexity and ESM-IF scPerplexity. ESM-2 Perplexity reflects how well a sequence aligns with the patterns learned by ESM-2 from its training data, while ESM-IF scPerplexity estimates sequence quality and reliability at the structural level. PLFM achieves the best perplexity results, indicating that it can generate protein sequences with high reliability and strong alignment with natural protein patterns. FPD, MMD, and OT measure the distributional similarity between generated sequences and the training data. PLFM outperforms all baselines on these metrics, indicating that it is a strong learner of the underlying data distribution. Notably, when the model achieves better distributional metrics, its perplexity may degrade slightly. This may reflect a trade-off between modeling fidelity and residue-level diversity.

b) PLFM generates sequences with plausible and natural predicted structures: The predicted local distance difference test (pLDDT) score is a widely used measure of the structural plausibility of protein sequences. Typically, a protein with a pLDDT score greater than 70 is considered to have high structural confidence. Although pLDDT predictions do not guarantee true foldability, they provide a strong indicator of potential folding quality. We then compared each predicted PDB structure against the Foldseek built-in AlphaFold/SwissProt reference database and reported the nearest-neighbor TM-score. PLFM achieves the highest pLDDT and TM-score values, which are comparable to or even better than those of the dataset in the general protein design task. This demonstrates that PLFM generates sequences with high foldability and structural naturalness.

5) Flow Matching Enables Fast and Effective Generation: One of the most significant advantages of FM is its generation speed. Unlike diffusion models, which require hundreds or even thousands of steps, Dirichlet FM achieves comparable results with only 50 steps, making it twice as efficient as DiMA. By leveraging a nearly linear probability path, PLFM achieves excellent results with only 25 steps, yielding a four-fold speedup over DiMA. Moreover, by learning an optimal probability path, FM achieves strong generation quality while maintaining high sampling efficiency. This is most directly reflected in the distributional metrics. Notably, both Dirichlet FM and PLFM outperform the diffusion models EvoDiff

and DiMA, which operate in discrete and continuous spaces, respectively. While DiMA achieves strong scores on the initial quality metrics, it performs less favorably on distributional metrics than both flow-based methods. This consistency underscores the advantages of FM methods in accurately capturing the data distribution through optimal probability paths.

6) Latent-Space Design Improves FM Performance: As a method operating in the continuous latent space induced by ESM-2, PLFM outperforms Dirichlet FM, which operates directly on discrete sequences. Another latent-space method, DiMA, also achieves better scores on foldability, reliability, and novelty than the discrete methods EvoDiff and Dirichlet FM. This advantage likely reflects the benefits of operating in a more compact and semantically rich latent space.

Regarding the choice of encoder and decoder, [26] suggests that their quality limits the upper bound of latent-space methods. We compare two pLMs with different capacities: ESM-2 8M and ESM-2 35M. The differences between the two models on metrics reflecting protein quality and novelty are smaller than we initially expected. However, on distribution-related metrics, the stronger ESM-2 35M model yields better results. Specifically, the reconstruction FPD is 0.03592 for ESM-2 8M and 0.01487 for ESM-2 35M. Although ESM-2 35M demonstrates better reconstruction ability, both models are sufficient for PLFM to learn high-quality protein sequences. Therefore, while the encoder and decoder can enhance the model's ability to learn from data, once they reach a sufficient level of performance, the overall results are more constrained by the capacity of the flow matcher.

TABLE II
PERFORMANCE ON AMP DESIGN OF PLFM AND BASELINE METHODS ON THE AMP DATASET. †: BENCHMARKED RESULTS ARE QUOTED FROM [27].

Model	Perplexity	Entropy	JS-6	P_{amp}	P_{mic}
Train/Real AMPs†	16.23 ± 3.80	3.23 ± 0.40	-	-	-
AMPGAN†	17.70 ± 4.08	2.90 ± 0.61	0.016	0.54	0.32
PepCVAE†	19.82 ± 3.83	3.12 ± 0.36	0.020	0.41	0.20
HydrAMP†	17.27 ± 6.02	2.82 ± 0.48	0.014	0.77	0.49
AMP-Diffusion†	12.84 ± 4.53	3.17 ± 0.56	0.028	0.81	0.50
PLFM	**12.43 ± 4.16**	**3.19 ± 0.55**	**0.080**	**0.84**	**0.63**

B. Antimicrobial Peptide Design

Antimicrobial peptides (AMPs) are a representative class of short functional proteins with antimicrobial activity and play a crucial role in the innate immune response of mammals against invasive bacterial infections. Developing novel and robust AMPs is an important goal in biochemical research, and generative models can help accelerate this process. Our experiments demonstrate that PLFM can generate high-quality, novel, and diverse AMPs, highlighting its potential for designing proteins with other desired properties in practical applications.

1) Data: Following the benchmark of AMP-Diffusion [27], we collected annotated AMPs from the dbAMP [28], AMP Scanner [29], and DRAMP [30] databases. After filtering out sequences longer than 40 residues, augmenting the data with HydrAMP [31], and removing duplicate sequences, we obtained 195k AMPs. The data were split into training and validation sets with a ratio of 9:1.

2) Baseline: We follow the AMP-Diffusion benchmark and include the conditional VAE methods HydrAMP [31] and PepCVAE [32], the GAN method AMPGAN [33], and the diffusion-based method AMP-Diffusion.

3) Metrics: We also follow the metric setting of AMP-Diffusion. To evaluate sequence quality, we use ESM-2 perplexity. To evaluate diversity and similarity, we use Shannon entropy and 6-mer Jaccard similarity (JS-6). Finally, we use the HydrAMP external classifier to compute the proportion of generated peptides classified as AMPs, denoted by P_{amp}, and the proportion predicted to be active against *E. coli*, denoted by P_{mic}.

4) Results: Table II compares the performance of PLFM with that of other relevant models. PLFM outperforms other AMP-specific methods. Its lowest ESM-2 perplexity indicates that it can generate highly reliable peptides. Shannon entropy measures sequence uncertainty or randomness and reflects information content and complexity. PLFM achieves the highest entropy, indicating that it can generate the most diverse peptide sequences. Meanwhile, PLFM obtains a substantially higher 6-mer JS score, indicating its ability to learn complex sequence motifs relevant to AMP design.

For the antimicrobial property classification test, we use a threshold of 0.8 for P_{amp}, which indicates a high likelihood of antimicrobial activity. Since the MIC property is a subset of antimicrobial activity and the number of peptides with experimentally measured MIC values used to train the HydrAMP classifier is limited [31], a threshold of 0.5 for P_{mic} is sufficient to indicate potential activity against *E. coli*. PLFM generates the highest proportion of peptides exceeding both thresholds, meaning that it can generate peptides that are not only classified as AMPs but are also predicted to be effective against specific pathogenic bacteria.

C. Antibody Design

Antibodies are large Y-shaped proteins composed of heavy and light chains that perform essential immune functions. The design of novel antibodies has long been an important focus in bioengineering and medicine. However, due to their complexity, antibody design requires the simultaneous generation of both heavy and light chains. Leveraging our multichain joint design pipeline, PLFM can operate on this more challenging task. Through the antibody design experiments, we show that PLFM is capable of designing large and complex proteins while jointly modeling multiple related sequences. This further expands the practical applicability of PLFM. We follow the benchmark of L-WJS [14].

1) Data: Following the benchmark of [14], we collected 1.4M paired antibodies (containing heavy–light chain pairs) from the Observed Antibody Space (OAS) database [34]. Sequences were clustered at 95% sequence identity with 80%

coverage using MMseqs2 [35]. Sequences with heavy chains longer than 149 residues or light chains longer than 148 residues were filtered out. The dataset was split into training and validation sets with a ratio of 9:1.

2) Baseline: Following the benchmark of [14], we include the language models GPT3.5 [36], IgLM [37], and ESM-2 [38]; the energy-based method DEEN [39]; and the diffusion-based methods SeqVDM [40], dWJS [41], and L-WJS [14].

3) Metrics: To evaluate the overall performance of PLFM on antibody design, we use the four metrics defined in the benchmark of [14]. To measure how well the model captures the training data distribution, we use the Wasserstein distance $W_{property}$ between sampled sequences and the reference distribution over 15 biological properties. To evaluate novelty, we use the proportion of unique sequences (Uniqueness). To evaluate sequence diversity, we use the average edit distance between sampled sequences and the reference distribution, denoted by E_{dist}, and the average pairwise edit distance within the generated set, denoted by IntDiv.

4) Results: Table III shows the performance of PLFM in comparison with other relevant models. Notably, ESM-2 achieves the highest E_{dist} and IntDiv. However, as noted in dWJS [41], ESM-2 does not generate antibody-like sequences, making these high diversity scores less meaningful. PLFM achieves the lowest $W_{property}$, indicating strong ability to learn the biophysical property distribution of paired OAS antibodies. Most models also achieve high uniqueness. PLFM attains the second- or third-best values on IntDiv and E_{dist}, which remain competitive. This may reflect a trade-off between distributional fidelity, novelty, and diversity.

TABLE III
PERFORMANCE ON ANTIBODY DESIGN OF PLFM AND BASELINE METHODS ON THE OAS DATASET. THE **BEST**, SECOND BEST AND *third best* RESULTS ARE MARKED. * MEANS NOT DESIRABLE RESULTS WITH HIGH DIVERSITY WITHOUT MATCHING THE REFERENCE DISTRIBUTION. †: BENCHMARKED RESULTS ARE QUOTED FROM [14] AND [41].

Model	$W_{\text{property}}\downarrow$	Uniqueness ↑	$E_{\text{dist}}(\hat{z},x)\uparrow$	IntDiv ↑
dWJS †	0.065	*0.97*	**62.7**	**65.1**
L-WJS †	0.053	**1.0**	56.6	54.1
SeqVDM †	*0.062*	**1.0**	60.0	*57.4*
DEEN †	0.087	0.99	50.9	42.7
GPT3.5 †	0.140	0.66	55.4	46.1
IgLM †	0.087	**1.0**	48.6	34.6
ESM2 †	0.150	**1.0**	70.99*	77.56*
PLFM	**0.045**	**1.0**	*58.6*	58.98

V. RELATED WORK

Recent *de novo* protein sequence generation methods span multiple paradigms, including autoregressive, VAE-, GAN-, and diffusion-based models [13], [19], [22], [23]. While these methods have shown promise, most are developed for specific protein families or application settings, such as peptides [27], antibodies [14], [41], or enzymes [42], rather than a unified framework across diverse protein domains.

Latent feature space learning introduces benefits in complex scenarios [43] and a related line of work leverages protein language models as representation backbones. Large-scale pLMs provide semantically meaningful embeddings that support downstream protein analysis and structure-related tasks [12], [38]. This makes latent-space generation an appealing direction for protein design, since continuous generative models can operate more naturally on learned embeddings than on discrete amino acid tokens. Recent latent diffusion approaches have demonstrated the effectiveness of this strategy for protein generation [19].

In parallel, flow matching has emerged as an efficient alternative to diffusion for continuous generative modeling [10], [16]. By learning vector fields along prescribed probability paths, flow-based methods can reduce sampling cost and improve generation efficiency. Although flow matching has started to be explored in biological molecule designs, including nucleic acid sequences [24] and protein structures [44]–[46], its use for protein sequence generation in pLM latent spaces remains limited.

Our work connects these directions by introducing an OT-based flow matching model in the latent space of protein language models, and by evaluating it across multiple protein design settings.

VI. CONCLUSIONS

In this work, we introduce PLFM, a continuous optimal transport conditional flow-matching-based model for *de novo* protein sequence generation. We demonstrate the advantages of flow matching and pLM-based latent representations for protein generation, and further apply our method to various protein design tasks, including general peptides, general proteins, antimicrobial peptides, and antibodies. PLFM exhibits competitive performance, demonstrating strong capability in general protein design, functional protein design, and complex joint protein design. In future work, scaling to larger datasets, incorporating conditional generation strategies, and integrating additional modalities such as protein structure data may further extend the applications of PLFM to a broader range of protein design problems.

ACKNOWLEDGMENT

This research was partially supported by National Key Research and Development Program of China under Grant No. 2025YFC2423903, National Natural Science Foundation of China under Grant No. T2541004, "Pioneer" and "Leading Goose" R&D Program of Zhejiang under Grant No. 2024SSYS0026, No. 2025C02120 and No. 2024C03048.

REFERENCES

[1] P.-S. Huang, S. E. Boyken, and D. Baker, "The coming of age of de novo protein design," *Nature*, vol. 537, no. 7620, pp. 320–327, 2016.
[2] H. Khakzad, I. Igashov, A. Schneuing, C. Goverde, M. Bronstein, and B. Correia, "A new age in protein design empowered by deep learning," *Cell Systems*, vol. 14, no. 11, pp. 925–939, 2023.
[3] Y. Zhu, Z. Kong, J. Wu, W. Liu, Y. Han, M. Yin, H. Xu, C.-Y. Hsieh, and T. Hou, "Generative ai for controllable protein sequence design: A survey," *arXiv preprint arXiv:2402.10516*, 2024.

[4] K. Liu, S. Shen, and H. Chen, "From sentences to sequences: Rethinking languages in biological system," *arXiv preprint arXiv:2507.00953*, 2025.

[5] Y. Li, K. Zhou, W. X. Zhao, and J.-R. Wen, "Diffusion models for non-autoregressive text generation: A survey," *arXiv preprint arXiv:2303.06574*, 2023.

[6] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in neural information processing systems*, vol. 33, pp. 6840–6851, 2020.

[7] Q. Dao, H. Phung, B. Nguyen, and A. Tran, "Flow matching in latent space," *arXiv preprint arXiv:2307.08698*, 2023.

[8] V. Hu, D. Wu, Y. Asano, P. Mettes, B. Fernando, B. Ommer, and C. Snoek, "Flow matching for conditional text generation in a few sampling steps," in *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 2: Short Papers)*, 2024, pp. 380–392.

[9] F. Eijkelboom, G. Bartosh, C. A. Naesseth, M. Welling, and J.-W. van de Meent, "Variational flow matching for graph generation," *arXiv preprint arXiv:2406.04843*, 2024.

[10] Y. Lipman, R. T. Chen, H. Ben-Hamu, M. Nickel, and M. Le, "Flow matching for generative modeling," *arXiv preprint arXiv:2210.02747*, 2022.

[11] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 10 684–10 695.

[12] Z. Lin, H. Akin, R. Rao, B. Hie, Z. Zhu, W. Lu, A. dos Santos Costa, M. Fazel-Zarandi, T. Sercu, S. Candido *et al.*, "Language models of protein sequences at the scale of evolution enable accurate structure prediction," *BioRxiv*, vol. 2022, p. 500902, 2022.

[13] N. Ferruz, S. Schmidt, and B. Höcker, "Protgpt2 is a deep unsupervised language model for protein design," *Nature communications*, vol. 13, no. 1, p. 4348, 2022.

[14] S. P. Mahajan, N. C. Frey, D. Berenberg, J. Kleinhenz, R. Bonneau, V. Gligorijevic, A. Watkins, and S. Saremi, "Exploiting language models for protein discovery with latent walk-jump sampling," in *Machine Learning for Structural Biology Workshop, NeurIPS*, 2023.

[15] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole, "Score-based generative modeling through stochastic differential equations," *arXiv preprint arXiv:2011.13456*, 2020.

[16] A. Tong, N. Malkin, G. Huguet, Y. Zhang, J. Rector-Brooks, K. Fatras, G. Wolf, and Y. Bengio, "Conditional flow matching: Simulation-free dynamic optimal transport," *arXiv preprint arXiv:2302.00482*, vol. 2, no. 3, 2023.

[17] X. Liu, C. Gong, and Q. Liu, "Flow straight and fast: Learning to generate and transfer data with rectified flow," *arXiv preprint arXiv:2209.03003*, 2022.

[18] F. Bao, S. Nie, K. Xue, Y. Cao, C. Li, H. Su, and J. Zhu, "All are worth words: A vit backbone for diffusion models," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 22 669–22 679.

[19] V. Meshchaninov, P. Strashnov, A. Shevtsov, F. Nikolaev, N. Ivanisenko, O. Kardymon, and D. Vetrov, "Diffusion on language model embeddings for protein sequence generation," *arXiv preprint arXiv:2403.03726*, 2024.

[20] U. Consortium, "Uniprot: a worldwide hub of protein knowledge," *Nucleic acids research*, vol. 47, no. D1, pp. D506–D515, 2019.

[21] B. Boeckmann, A. Bairoch, R. Apweiler, M.-C. Blatter, A. Estreicher, E. Gasteiger, M. J. Martin, K. Michoud, C. O'Donovan, I. Phan *et al.*, "The swiss-prot protein knowledgebase and its supplement trembl in 2003," *Nucleic acids research*, vol. 31, no. 1, pp. 365–370, 2003.

[22] D. Repecka, V. Jauniskis, L. Karpus, E. Rembeza, I. Rokaitis, J. Zrimec, S. Poviloniene, A. Laurynenas, S. Viknander, W. Abuajwa *et al.*, "Expanding functional protein sequence spaces using generative adversarial networks," *Nature Machine Intelligence*, vol. 3, no. 4, pp. 324–333, 2021.

[23] S. Alamdari, N. Thakkar, R. van den Berg, A. X. Lu, N. Fusi, A. P. Amini, and K. K. Yang, "Protein generation with evolutionary diffusion: sequence is all you need," *bioRxiv*, pp. 2023–09, 2023.

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

[25] R. Wu, F. Ding, R. Wang, R. Shen, X. Zhang, S. Luo, C. Su, Z. Wu, Q. Xie, B. Berger *et al.*, "High-resolution de novo structure prediction from primary sequence," *BioRxiv*, pp. 2022–07, 2022.

[26] P. Esser, S. Kulal, A. Blattmann, R. Entezari, J. Müller, H. Saini, Y. Levi, D. Lorenz, A. Sauer, F. Boesel *et al.*, "Scaling rectified flow transformers for high-resolution image synthesis," in *Forty-first International Conference on Machine Learning*, 2024.

[27] T. Chen, P. Vure, R. Pulugurta, and P. Chatterjee, "Amp-diffusion: Integrating latent diffusion with protein language models for antimicrobial peptide generation," *bioRxiv*, pp. 2024–03, 2024.

[28] J.-H. Jhong, Y.-H. Chi, W.-C. Li, T.-H. Lin, K.-Y. Huang, and T.-Y. Lee, "dbamp: an integrated resource for exploring antimicrobial peptides with functional activities and physicochemical properties on transcriptome and proteome data," *Nucleic acids research*, vol. 47, no. D1, pp. D285–D297, 2019.

[29] D. Veltri, U. Kamath, and A. Shehu, "Deep learning improves antimicrobial peptide recognition," *Bioinformatics*, vol. 34, no. 16, pp. 2740–2747, 2018.

[30] X. Kang, F. Dong, C. Shi, S. Liu, J. Sun, J. Chen, H. Li, H. Xu, X. Lao, and H. Zheng, "Dramp 2.0, an updated data repository of antimicrobial peptides," *Scientific data*, vol. 6, no. 1, p. 148, 2019.

[31] P. Szymczak, M. Możejko, T. Grzegorzek, R. Jurczak, M. Bauer, D. Neubauer, K. Sikora, M. Michalski, J. Sroka, P. Setny *et al.*, "Discovering highly potent antimicrobial peptides with deep generative model hydramp," *nature communications*, vol. 14, no. 1, p. 1453, 2023.

[32] P. Das, K. Wadhawan, O. Chang, T. Sercu, C. dos Santos, M. Riemer, V. Chenthamarakshan, I. Padhi, and A. Mojsilovic, "Pepcvae: Semi-supervised targeted design of antimicrobial peptide molecules," *arXiv preprint arXiv:1810.07743*, 2018.

[33] C. M. Van Oort, J. B. Ferrell, J. M. Remington, S. Wshah, and J. Li, "Ampgan v2: machine learning-guided design of antimicrobial peptides," *Journal of chemical information and modeling*, vol. 61, no. 5, pp. 2198–2207, 2021.

[34] T. H. Olsen, F. Boyles, and C. M. Deane, "Observed antibody space: A diverse database of cleaned, annotated, and translated unpaired and paired antibody sequences," *Protein Science*, vol. 31, no. 1, pp. 141–146, 2022.

[35] M. Steinegger and J. Söding, "Mmseqs2 enables sensitive protein sequence searching for the analysis of massive data sets," *Nature biotechnology*, vol. 35, no. 11, pp. 1026–1028, 2017.

[36] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray *et al.*, "Training language models to follow instructions with human feedback," *Advances in neural information processing systems*, vol. 35, pp. 27 730–27 744, 2022.

[37] R. W. Shuai, J. A. Ruffolo, and J. J. Gray, "Generative language modeling for antibody design," *BioRxiv*, pp. 2021–12, 2021.

[38] Z. Lin, H. Akin, R. Rao, B. Hie, Z. Zhu, W. Lu, N. Smetanin, R. Verkuil, O. Kabeli, Y. Shmueli *et al.*, "Evolutionary-scale prediction of atomic-level protein structure with a language model," *Science*, vol. 379, no. 6637, pp. 1123–1130, 2023.

[39] S. Saremi, A. Mehrjou, B. Schölkopf, and A. Hyvärinen, "Deep energy estimator networks," *arXiv preprint arXiv:1805.08306*, 2018.

[40] D. Kingma, T. Salimans, B. Poole, and J. Ho, "Variational diffusion models," *Advances in neural information processing systems*, vol. 34, pp. 21 696–21 707, 2021.

[41] N. C. Frey, D. Berenberg, K. Zadorozhny, J. Kleinhenz, J. Lafrance-Vanasse, I. Hotzel, Y. Wu, S. Ra, R. Bonneau, K. Cho *et al.*, "Protein discovery with discrete walk-jump sampling," *arXiv preprint arXiv:2306.12360*, 2023.

[42] A. H.-W. Yeh, C. Norn, Y. Kipnis, D. Tischer, S. J. Pellock, D. Evans, P. Ma, G. R. Lee, J. Z. Zhang, I. Anishchenko *et al.*, "De novo design of luciferases using deep learning," *Nature*, vol. 614, no. 7949, pp. 774–780, 2023.

[43] S. Gao, Y. Fu, K. Liu, W. Gao, H. Xu, J. Wu, and Y. Han, "Collaborative knowledge amalgamation: Preserving discriminability and transferability in unsupervised learning," *Information Sciences*, vol. 669, p. 120564, 2024.

[44] J. Yim, B. L. Trippe, V. De Bortoli, E. Mathieu, A. Doucet, R. Barzilay, and T. Jaakkola, "Se (3) diffusion model with application to protein backbone generation," *arXiv preprint arXiv:2302.02277*, 2023.

[45] J. Yim, A. Campbell, E. Mathieu, A. Y. Foong, M. Gastegger, J. Jiménez-Luna, S. Lewis, V. G. Satorras, B. S. Veeling, F. Noé *et al.*, "Improved motif-scaffolding with se (3) flow matching," *ArXiv*, 2024.

[46] J. Li, C. Cheng, Z. Wu, R. Guo, S. Luo, Z. Ren, J. Peng, and J. Ma, "Full-atom peptide design based on multi-modal flow matching," *arXiv preprint arXiv:2406.00735*, 2024.