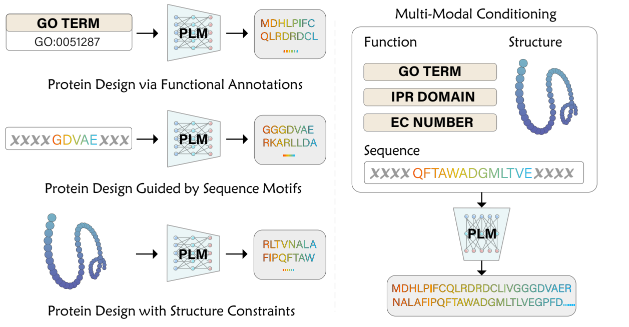
CFP-Gen: Combinatorial Functional Protein Generation via Diffusion Language Models

Junbo Yin [1 2 3] **Chao Zha** [1 2 3] **Wenjia He** [1 2 3] **Chencheng Xu** [1 2 3] **Xin Gao** [1 2 3]

Abstract

Existing PLMs generate protein sequences based on a single-condition constraint from a specific modality, struggling to simultaneously satisfy multiple constraints across different modalities. In this work, we introduce **CFP-Gen**, a novel diffusion language model for **C**ombinatorial **F**unctional **P**rotein **GEN**eration. CFP-Gen facilitates the *de novo* protein design by integrating multimodal conditions with functional, sequence, and structural constraints. Specifically, an Annotation-Guided Feature Modulation (AGFM) module is introduced to dynamically adjust the protein feature distribution based on composable functional annotations, *e.g.*, GO terms, IPR domains and EC numbers. Meanwhile, the Residue-Controlled Functional Encoding (RCFE) module captures residue-wise interaction to ensure more precise control. Additionally, off-the-shelf 3D structure encoders can be seamlessly integrated to impose geometric constraints. We demonstrate that CFP-GEN enables high-throughput generation of novel proteins with functionality comparable to natural proteins, while achieving a high success rate in designing multifunctional proteins.



(a) PLMs with Single-Condition Constraint (b)PLMs with Multi-Condition Constraint

Figure 1. **Motivation of CFP-Gen.** (a) Previous PLMs typically generate proteins based on *single-modality* conditioning, considering only individual functional constraints. (b) In contrast, CFP-Gen incorporates multiple conditions from *diverse modalities*—function, sequence and structure—to impose comprehensive functional constraints, thereby leading to optimized proteins.

1. Introduction

De novo protein design (Watson et al., 2023; Krishna et al., 2024; Dauparas et al., 2022; Wang et al., 2024; Lisanza et al., 2024) has emerged as a powerful strategy for numerous biotechnological applications, including drug development, enzyme engineering, and the creation of novel therapeutic proteins. The growing capabilities of large-scale protein language models (PLMs) offer tremendous potential for discovering novel proteins that do not exist in nature or would require millions of years of evolution to emerge (Hayes et al., 2025a; Madani et al., 2023; Alamdari et al., 2023).

Most current PLMs are primarily designed for unconditional protein generation (Lin et al., 2023; Ren et al., 2024; Qu et al., 2024), limiting their efficiency in addressing complex real-world challenges. In contrast, controllable protein generation (Listov et al., 2024; Ferruz & Höcker, 2022; Madani et al., 2023), which tailors proteins to meet specific biological functions by explicitly defined conditions, holds significant promise for advancing practical applications.

Biologically meaningful proteins often simultaneously satisfy multiple functional constraints across diverse modalities, *e.g.*, functional annotation, sequence or structure. As shown in Fig. 1, previous PLMs typically generate protein candidates based on a single-condition input from a specific modality, and thus have to rely on iterative filtering or multi-step optimization to meet constraints across multiple modalities (Goverde et al., 2024). Moreover, this pipeline often struggles to satisfy all desired functionalities and becomes impractical when available data is limited. Therefore,

Code and data available at https://github.com/yinjunbo/cfpgen.

[1] Computer Science Program, Computer, Electrical and Mathematical Sciences and Engineering Division, King Abdullah University of Science and Technology (KAUST), Thuwal 23955-6900, Kingdom of Saudi Arabia [2] Center of Excellence for Smart Health (KCSH), KAUST [3] Center of Excellence for Generative AI, KAUST. Correspondence to: Xin Gao <xin.gao@kaust.edu.sa>.

Proceedings of the 42^{nd} International Conference on Machine Learning, Vancouver, Canada. PMLR 267, 2025. Copyright 2025 by the author(s).

developing an advanced protein generative model capable of simultaneously handling multiple function constraints within a unified model would significantly streamline the multi-objective optimization in protein engineering.

In this work, we introduce **CFP-Gen**, a novel large-scale diffusion language model specifically developed for **C**ombinatorial **F**unctional **P**rotein **GEN**eration. CFP-Gen iteratively denoises the input protein sequence while simultaneously accounting for various functional conditions, including functional annotations, desired sequence motifs and 3D atomic coordinates of reference proteins. During inference, composable annotation tags (*e.g.*, GO terms, IPR domains and EC numbers) are encoded as one-hot embeddings and injected to the model through an Annotation-Guided Feature Modulation (AGFM) module. AGFM dynamically modulates the normalized feature distributions of the noised protein sequence by leveraging all available annotation tags. Compared to classifier-guided diffusion, this approach enables joint training to ensure strict alignment between functions and sequences, while also allowing for the flexible combinations of various functional annotations. Additionally, certain amino acid residues (*i.e.*, sequence motif) are critical in real-world applications, as they often determine the desired functional properties. To cater to this need, a Residue-Controlled Functional Encoding (RCFE) module is introduced to explicitly encode these functional domains. RCFE employs an ESM-like transformer-based controller to effectively capture epistasis and evolutionary relationships among residues. As a result, this approach has the potential to generate novel protein sequences with optimized functional sites. Moreover, CFP-Gen supports direct optimization of user-provided proteins by encoding their 3D backbone atomic coordinates as conditional features using off-the-shelf structure encoders. By jointly incorporating functional constraints across diverse modalities, our approach ensures that the inverse folding process preserves structural accuracy while simultaneously satisfying functional requirements. This provides a significant advantage in tasks that demand multi-objective optimization.

Our contributions can be summarized as follows:

- A novel generative model, CFP-Gen, is developed for *de novo* design of functional proteins. CFP-Gen addresses multiple functional conditions across various modalities simultaneously, offering more promising starting points compared to previous PLMs limited by single-condition inputs.
- An Annotation-Guided Feature Modulation (AGFM) is introduced to enable highly-controllable protein generation by incorporating commonly used functional annotations in a composable and flexible manner.
- A Residue-Controlled Functional Encoder (RCFE) is proposed to encode functional domains at the residue level. It captures epistasis and interactions within sequences, enabling the generation of novel yet functional proteins.

We thoroughly evaluate CFP-Gen across various tasks, including functional sequence generation, functional protein inverse folding, and multi-objective protein design. CFP-Gen achieves exceptional function performance, *e.g.*, improving ESM3 by 30% in $\boldsymbol{F}_1$-score, as demonstrated by leading function predictors. Additionally, it improves Amino Acid Recovery (AAR) of DPLM by 9% in inverse folding. Notably, CFP-Gen demonstrates a remarkable success rate in designing multi-functional proteins (*e.g.*, enzymes exhibiting multiple catalytic activities). We anticipate CFP-Gen to become a valuable and practical computational tool for addressing biomedical and biotechnological challenges.

2. Related Work

PLMs have become indispensable tools in protein science, widely applied to both discriminative and generative tasks. ESM2 (Lin et al., 2023) is a pioneering approach that utilizes scalable language models to uncover patterns in protein sequences across evolutionary space. By optimizing a masked modeling objective, ESM2 learns rich sequence representations, significantly improving performance in various discriminative tasks, including structure prediction (Baek et al., 2021) and functional classification (Kulmanov et al., 2018). Meanwhile, ProtGPT2 (Ferruz et al., 2022) performs unconditional sequence generation, akin to natural language sentence generation. Moving toward controllable sequence generation, ProGen (Nijkamp et al., 2023) augments the language model with protein family tags, enabling the generation of sequences within specific functional constraints. ProteoGAN (Kucera et al., 2022) incorporates GO terms for sequence generation, providing detailed *in silico* validations of its output sequences. Later, EvoDiff (Alamdari et al., 2023) introduces a discrete diffusion model instead of the masked language modeling objective. This enables conditional protein generation, such as sequence inpainting and MSA-guided generation. Further, ZymCTRL (Munsamy et al., 2022) specializes in generating enzymes by conditioning on EC numbers, successfully designing sequences with catalytic activity. Most recently, DPLM (Wang et al., 2024) is proposed as a new foundation model that not only supports representation learning for discriminative tasks but also enables conditional generation guided by secondary structure classifications and sequence or structure motifs.

All of the above PLMs are fundamentally limited to single-modality conditioning, *i.e.*, generating sequences based on one functional constraint at a time. This reduces the flexibility and success rate of generated proteins when multiple functional constraints are required. Although the recently published ESM3 (Hayes et al., 2025b) supports multimodal

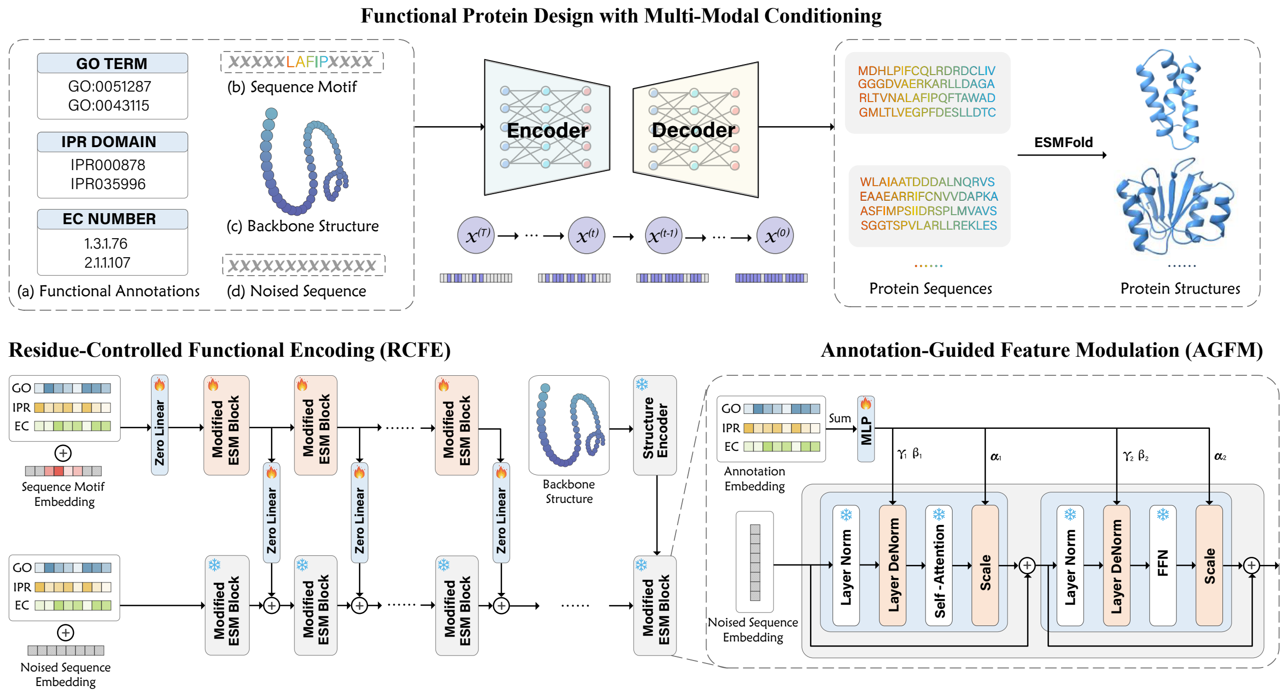


Figure 2. **Pipeline of CFP-Gen model.** Functional conditions from diverse modalities, combined with the noised sequence, are iteratively processed by the model to generate desired proteins. Within each modified ESM block, AGFM adaptively adjusts the noised sequence embedding based on combinations of various functional annotations. Furthermore, sequence motifs and backbone atomic coordinates are embedded by RCFE and a structure encoder, respectively, providing precise and flexible guidance for the generation process.

inputs, it defines function using a limited vocabulary, restricting its broader applicability. In contrast, our model incorporates more diverse functions. This ensures more precise control and leads to functionally superior proteins.

3. Methodology

We begin by introducing the essential concepts of CFP-Gen in §3.1. Next, we present the AGFM module in §3.2. Following that, we detail the RCFE module in §3.3, which is designed to encode desired sequence motifs. Finally, we demonstrate how CFP-Gen smoothly incorporates functional conditions across diverse modalities in §3.4.

3.1. Preliminaries of CFP-Gen

Diffusion models are widely recognized for their capability in *de novo* protein design. CFP-Gen is built upon the leading diffusion protein language model DPLM (Wang et al., 2024) to leverage its well-pretrained parameters attained from evolutionary-scale datasets. Concretely, discrete diffusion mechanism (Austin et al., 2021) is employed to model the protein sequence distribution at the amino acid category level. Let $\mathbf{x} \sim q(\mathbf{x})$ denote a protein sequence of length L, represented as $\mathbf{x} = (x_1, x_2, \ldots, x_L)$, where each $x_i \in \{0,1\}^{|V|}$ is a one-hot vector indicating an amino acid category from the set V of 20 standard types. The categorical distribution $\text{Cat}(\mathbf{x}; \mathbf{p})$ models the sequence $\mathbf{x}$, with $\mathbf{p} = (\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_L)$ as the collection of probability vectors. Each $\mathbf{p}_i = (p_{i,1}, p_{i,2}, \ldots, p_{i,|V|})$ specifies the categorical distribution of the i-th residue in the sequence, *i.e.*, $p_{i,v}$ represents the probability of selecting amino acid category v, always ensuring that $\sum_{v \in V} p_{i,v} = 1, \forall i \in \{1, \ldots, L\}$.

Forward Process with Discrete Diffusion. We apply discrete diffusion to gradually corrupt the raw sequence $\mathbf{x}^{(0)}$ over $t \in \{1, \ldots, T\}$ time steps by transitioning each amino acid token towards a stationary noise distribution. The stationary noise distribution is parameterized by a fixed probability vector $\mathbf{q}_{\text{noise}}$ and can be expressed as:$q_{\text{noise}}(\mathbf{x}^{(t)}) = \text{Cat}(\mathbf{x}^{(t)}; \mathbf{p} = \mathbf{q}_{\text{noise}})$. Following DPLM, $q_{\text{noise}}(\mathbf{x}^{(t)})$ satisfies: $q_{\text{noise}}(\mathbf{x}^{(t)}) = 1$ if $\mathbf{x}^{(t)} = [X]$ and 0 otherwise, where $[X]$ refers to the absorbing state (*e.g.*, `<mask>`). Once a token transitions into this absorbing state, it will remain unchanged in all subsequent diffusion steps. This ensures that the forward process eventually leads all amino acid tokens to `<mask>`, unifying the principles behind masked language models (Kenton & Toutanova, 2019) and autoregressive language models (Ferruz et al., 2022). Mathematically, the forward transition process is given by:

$$q(\mathbf{x}^{(t)} \mid \mathbf{x}^{(t-1)}) = \text{Cat}(\mathbf{x}^{(t)}; \mathbf{p} = \mathbf{x}^{(t-1)} \mathbf{Q}_t), \quad (1)$$

where $\mathbf{Q}_t$ is the transition matrix at step t. Each row of $\mathbf{Q}_t$ is a probability vector defined as: $\mathbf{Q}_t = \beta_t I + (1-\beta_t)\mathbf{q}_{\text{noise}}$, where I is the identity matrix and $\beta_t \in [0,1]$ is the noise schedule. Due to the Markov property, the overall transition from $\mathbf{x}^{(0)}$ to $\mathbf{x}^{(t)}$ can be expressed as follows:

$$q(\mathbf{x}^{(t)} \mid \mathbf{x}^{(0)}) = \text{Cat}(\mathbf{x}^{(t)}; \mathbf{p} = \alpha_t \mathbf{x}^{(0)} + (1-\alpha_t)\mathbf{q}_{\text{noise}}), \tag{2}$$

where $\alpha_t = \prod_{i=1}^{t} \beta_i$ represents the accumulated effect of the noise schedule over t steps. As $t \to T$, $\alpha_t \to 0$, ensuring that the sequence $\mathbf{x}^{(t)}$ converges to the stationary noise $\mathbf{q}_{\text{noise}}$ at time step T.

Reverse Denoising with Composable Conditions. To enable the generation of protein sequences with desired functionalities, we incorporate multimodal conditions into the discrete diffusion framework during the reverse process, as illustrated in Fig. 2. These include 0D annotation tags c_{anno}, 1D sequence motifs c_{seq}, and 3D structures c_{str}, encoded by f_{AGFM}, f_{RCFE}, and f_{GVPT}, respectively. The details of these networks will be introduced in the following sections, where GVPT indicates the GVP-Transformer (Hsu et al., 2022) for encoding 3D backbone atomic coordinates.

The reverse process reconstructs the sequence by iteratively denoising $\mathbf{x}^{(t)}$ back to $\mathbf{x}^{(0)}$, using the predicted $\hat{\mathbf{x}}^{(0)}$ at each step, which is derived from the KL divergence, *i.e.*, $D_{\text{KL}}\left[q(\mathbf{x}^{(t-1)} \mid \mathbf{x}^{(t)}, \mathbf{x}^{(0)}) \parallel p_\theta(\mathbf{x}^{(t-1)} \mid \mathbf{x}^{(t)})\right]$. Thus, the reverse step is expressed as follows:

$$p_\theta(\mathbf{x}^{(t-1)} \mid \mathbf{x}^{(t)}) = \sum_{\hat{\mathbf{x}}^{(0)}} q(\mathbf{x}^{(t-1)} \mid \mathbf{x}^{(t)}, \hat{\mathbf{x}}^{(0)}) p_\theta(\hat{\mathbf{x}}^{(0)} \mid \mathbf{x}^{(t)}, c), \tag{3}$$

where $c \in \{c_{\text{anno}}, c_{\text{seq}}, c_{\text{str}}\}$ represents any combinations of conditions, depending on their availability. To predict $\hat{\mathbf{x}}^{(0)}$, the model incorporates c throughout the network:

$$p_\theta(\hat{\mathbf{x}}^{(0)} \mid \mathbf{x}^{(t)}, c) = \text{Softmax}(W h(\mathbf{x}^{(t)}, c)),$$
$$h(\mathbf{x}^{(t)}, c) = \mathcal{A}\Big(f_{\text{RCFE}}\big(f_{\text{AGFM}}(\mathbf{x}^{(t)}, c_{\text{anno}}), c_{\text{seq}}\big), f_{\text{GVPT}}(c_{\text{str}})\Big), \tag{4}$$

where W is the final output layer, $\mathcal{A}$ is a cross-attention layer and $h(\mathbf{x}^{(t)}, c)$ represents the core of CFP-Gen, sequentially integrating modules such as f_{AGFM}, f_{RCFE} and f_{GVPT}.

Optimization of CFP-Gen. The training objective is to optimize the predicted $\hat{\mathbf{x}}^{(0)}$ against the raw sequence $\mathbf{x}^{(0)}$ using a weighted cross-entropy loss:

$$\mathcal{L} = \mathbb{E}_{q(\mathbf{x}^{(0)})}\left[\lambda^{(t)} \sum_{1 \le i \le L} b_i(t) \cdot \log p_\theta(\mathbf{x}_i^{(0)} | \mathbf{x}^{(t)}, \gamma(c))\right]. \tag{5}$$

Here, $\lambda^{(t)}$ adjusts the influence of each diffusion time step t, $b_i(t)$ determines the contribution of each position i, and $\gamma(c)$ controls the strength of the condition c. In the following sections, we demonstrate how the inclusion of c facilitates the prediction of highly functional proteins and how its composable nature empowers CFP-Gen to achieve multi-objective protein design.

3.2. Annotation-Guided Conditioning by AGFM

Functional annotations are both highly informative and representative, as they are rigorously curated by biological experts to capture essential properties of proteins. In this work, we consider three commonly used annotations, *i.e.*, GO terms (gen, 2021), IPR numbers (Hunter et al., 2009), and EC numbers (Schomburg et al., 2004), to guide the generation process. Typically, each type of annotation characterizes the protein's profile from a different perspective: GO captures the molecular functions, IPR defines the functional domains, and EC describes the related catalytic processes. Unlike ESM3 (Hayes et al., 2025b) relying only on a restricted vocabulary mapping to represent IPR annotations, we adopt a more tactful approach to combine diverse functions. While three types of annotations are presented as examples, our paradigm is extensible to other forms of annotations, such as Pfam (Bateman et al., 2000).

Existing PLMs often encode a single type of annotation. For instance, ZymCTRL (Munsamy et al., 2022) designs enzymes conditioned solely on EC numbers, while ProteoGAN (Kucera et al., 2022) operates only with GO terms. Although ProGen (Madani et al., 2023) supports multiple annotation tags, it requires extensive fine-tuning with sufficient homologous sequences to ensure performance, which limits its application when training samples are scarce. By contrast, CFP-Gen is more flexible and versatile by addressing multiple annotations simultaneously, without the need of further fine-tuning. By leveraging the complementary characteristics of these high-quality annotations, we also enable a more comprehensive description of a protein.

As shown in Fig. 2, an Annotation-Guided Feature Modulation (AGFM) module is integrated into each modified ESM block (Lin et al., 2023), where each block has been well pretrained by DPLM. CFP-Gen is achieved by training only the annotation embedding layers and a single MLP layer. For each type of annotation (*e.g.*, GO, IPR, EC), we maintain a dedicated embedding layer. Each annotation is mapped to a same-dimensional vector representation through its respective embedding layer. This flexible design ensures that annotations from different sources can be directly summed together. The process is formally represented as: $\mathbf{x}_{\text{anno}} = f_{\text{AGFM}}(\mathbf{x}, \mathbf{c}_{\text{GO}}, \mathbf{c}_{\text{IPR}}, \mathbf{c}_{\text{EC}})$, where $\mathbf{x}, \mathbf{x}_{\text{anno}} \in \mathbb{R}^{L \times D}$ denotes the input feature and the modulated feature output by AGFM within one block, based on the denoised sequence $\mathbf{x}^{(t)}$ at time step t. Concretely, the sum of embeddings is passed through an MLP layer to regress three parameters, *i.e.*, $\boldsymbol{\gamma}, \boldsymbol{\beta}, \boldsymbol{\alpha} = \mathcal{F}(\mathbf{c}_{\text{GO}} + \mathbf{c}_{\text{IPR}} + \mathbf{c}_{\text{EC}})$, where $\boldsymbol{\gamma}$, $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$ are the scaling, shifting and gating factors, respectively. The

MLP layer $\mathcal{F}$ follows a specialized initialization strategy: the weights predicting $\boldsymbol{\gamma}$ and $\boldsymbol{\beta}$ are initialized to zeros, ensuring that conditional information is gradually injected into $\mathbf{x}^{(t)}$, while the weights predicting $\boldsymbol{\alpha}$ are initialized to ones, allowing for effective gating from the beginning. The scale $\boldsymbol{\gamma}$ and shift $\boldsymbol{\beta}$ parameters modulate the distribution of the noised feature $\mathbf{x}$ before both the self-attention (SA) and feed-forward network (FFN) layers within each ESM block, akin to a feature de-normalization process: $\mathbf{x}_{\text{out}} = \boldsymbol{\gamma} \odot \mathbf{x} + \boldsymbol{\beta}$. Additionally, the gating factor $\boldsymbol{\alpha}$ operates on the feature distribution $\mathbf{x}'_{\text{out}}$ output by the SA and FFN layers through the following formulation: $\hat{\mathbf{x}}_{\text{out}} = \boldsymbol{\alpha} \odot \mathbf{x}'_{\text{out}} + \mathbf{x}'_{\text{out}}$.

Through the collaboration of diverse annotations, AGFM effectively adjusts the intermediate representations during diffusion, resulting in higher-quality proteins compared to previous approaches relying on single-condition control.

3.3. Sequence-Guided Conditioning with RCFE

While AGFM has provided effective control over protein generation with annotations, certain applications require finer-grained guidance, *e.g.*, biological experts often focus on functional sequence fragments (referred to as sequence motifs) to ensure that the generated proteins retain desired functionality associated with these motifs. Thus, we propose the Residue-Controlled Functional Encoder (RCFE) with an additional network branch to handle this sequence-level condition. The core of RCFE lies in inferring the complete sequence while preserving or even optimizing the specified sequence motifs, akin to the sequence inpainting task introduced by DPLM (Wang et al., 2024) and EvoDiff (Alamdari et al., 2023). However, unlike these methods, which rely on fixed amino acids throughout generation, RCFE dynamically updates the specified motifs during inference. This dynamic adjustment enables the model to emulate evolutionary processes, offering the potential to discover improved sequence motifs with enhanced functional properties.

Specifically, f_{RCFE} in Eq.4 consists of two branches of modified ESM blocks, along with zero-initialized linear layers to jointly handle sequence-level and annotation-level information. As shown in Fig.2, the bottom gray blocks represent the main branch, which processes the noised sequence and annotation tags, while the upper orange blocks represent the second branch, composed of trainable copies of the main branch, for tackling sequence-level conditioning. Inspired by ControlNet (Zhang et al., 2023), RCFE adapts this dual-branch design to transformer-based ESM blocks, making it well-suited for protein sequence generation.

Let $\mathcal{E}_{\text{esm}}(\cdot, \Theta_{\text{esm}})$ denote a modified ESM block in the main branch, which has been enhanced by AGFM to integrate the functional annotations. The parameters Θ_{esm} in the main branch blocks remain frozen during training RCFE, while Θ_{seq} in another branch blocks using a trainable copy of Θ_{esm} need to be updated, denoted as $\mathcal{E}_{\text{seq}}(\cdot, \Theta_{\text{seq}})$. This design preserves the powerful representation capabilities of the main branch that has been fully trained, while enabling the other branch to dynamically encode sequence motifs. Formally, we first pad the sequence motif with `<mask>` and project it into the same latent space as the denoised feature $\mathbf{x} \in \mathbb{R}^{L \times D}$ within one block, which is denoted as $\mathbf{c}_{\text{seq}} \in \mathbb{R}^{L \times D}$. In this way, the workflow of RCFE to handle the sequence-level condition can be represented as follows:

$$\mathbf{x}_{\text{seq}} = \mathcal{E}_{\text{esm}}(\mathbf{x}; \Theta_{\text{esm}}) + \mathcal{F}_{\text{out}}(\mathcal{E}_{\text{seq}}(\mathbf{x} + \mathcal{F}_{\text{in}}(\mathbf{c}_{\text{seq}}; \Theta_{\text{in}}); \Theta_{\text{seq}}); \Theta_{\text{out}}), \tag{6}$$

where $\mathcal{F}_{\text{in}}(\cdot; \Theta_{\text{in}})$ denotes a zero-initialized linear layer applied to $\mathbf{c}_{\text{seq}}$. Note that this layer is only utilized in the first block of the conditional branch. Similarly, $\mathcal{F}_{\text{out}}(\cdot; \Theta_{\text{out}})$ is another zero-initialized linear layer. Zero-initialization enables the gradual incorporation of meaningful information from the sequence motif condition while ensuring the stability of the main blocks. Finally, $\mathbf{x}_{\text{seq}}$ represents the updated feature from a main block, enriched with conditional information at both the annotation and sequence motif levels. It is worth mentioning that, in image generation models, typically only the encoder part of a U-Net is used as the trainable branch. Similarly, we leverage only the first half blocks of the ESM2 model to encode sequence motifs, ensuring an efficient yet expressive representation.

Consequently, the sequence-level condition serves as a strong complement to the annotation-level condition, leading to enhanced controllability over the generated sequences compared to using single condition modality alone.

3.4. Multimodal Conditioning within CFP-GEN

The fundamental principle in protein science is that function, sequence, and structure are inherently interdependent. This intricate relationship reveals the importance of jointly modeling these modalities to achieve accurate protein design.

To this end, we consider a practical scenario where biological experts have obtained the backbone structures of interest (*e.g.*, from RFDiffusion (Watson et al., 2023)) and aim to determine the functional sequence that folds into this structure, a problem known as the inverse folding task. In the traditional paradigm, the goal is to generate a sequence that correctly folds into a given backbone structure. However, when additional functional constraints c_{anno} are required, this task can be extended to *functional protein inverse folding* or the *inverse function task* - to generate a sequence $\mathbf{x}^*$ that maximizes sequence recovery while simultaneously optimizing the functionality recovery matching c_{anno}:

$$\mathbf{x}^* = \arg\max_{\mathbf{x}} p(\mathbf{x} \mid c_{\text{anno}}, c_{\text{seq}}, c_{\text{str}}) + S_{\text{func}}(\mathbf{x}, c_{\text{anno}}), \tag{7}$$

where S_{func} is a scoring function that evaluates how well the generated sequence $\mathbf{x}$ aligns with the desired functional properties. More discussion on S_{func} can be found in §4.2.

Table 1. **Function evaluation of the generated sequences using diverse function predictors**, *e.g.*, DeepGO-SE (Kulmanov et al., 2024) for GO terms, InterProScan (Blum et al., 2021) for IPR domains, and CLEAN (Yu et al., 2023) for EC numbers. The best result in each column is highlighted in **bold**, while the second-best is underlined (excluding the positive control).

Models	Supported Condtions	MRR↑	MMD↓	MMD-G↓	mic. F_1↑	mac. F_1↑	AUPR↑	AUC↑
Evaluating the Protein Functionality via predicted GO Terms.								
Positive Control	-	0.939	0.000	0.000	0.543	0.522	0.402	0.775
Negative Control	-	0.017	0.215	0.125	0.205	0.020	0.025	0.501
DPLM (Wang et al., 2024)	Sequence Motif, Structure	0.134	0.189	0.109	0.332	0.189	0.109	0.581
ProteoGAN (Kucera et al., 2022)	GO Term	0.277	0.095	0.055	0.376	0.093	0.121	0.510
ProGen2 (Nijkamp et al., 2023)	GO Term, Sequence Motif	0.545	0.109	0.064	0.414	0.355	0.240	0.663
CFP-Gen *(w/ GO)*	GO, IPR, EC, Seq., Struc.	0.601	0.112	0.060	0.429	0.370	0.245	0.674
CFP-Gen *(w/ GO and IPR)*	GO, IPR, EC, Seq., Struc.	0.779	0.066	0.039	0.496	0.458	0.339	0.732
CFP-Gen *(w/ Motif)*	GO, IPR, EC, Seq., Struc.	0.839	0.046	0.028	0.504	0.492	0.370	0.762
CFP-Gen *(w/ GO, IPR and Motif)*	GO, IPR, EC, Seq., Struc.	**0.870**	**0.036**	**0.022**	**0.532**	**0.550**	**0.435**	**0.795**
Evaluating the Protein Functionality via predicted IPR Domains.								
Positive Control	-	1.000	0.000	0.000	1.000	1.000	1.000	1.000
Negative Control	-	0.016	0.224	0.131	0.003	0.000	0.004	0.500
DPLM (Wang et al., 2024)	Sequence Motif, Structure	0.053	0.192	0.112	0.548	0.422	0.369	0.695
ESM3 (Hayes et al., 2025b)	IPR Domain, Seq., Struc.	0.148	0.101	0.060	0.690	0.565	0.502	0.762
ProGen2 (Nijkamp et al., 2023)	GO Term, Sequence Motif	0.281	0.112	0.067	0.712	0.584	0.536	0.772
CFP-Gen *(w/ IPR)*	GO, IPR, EC, Seq., Struc.	0.332	0.094	0.056	0.882	0.826	0.782	0.899
CFP-Gen *(w/ GO and IPR)*	GO, IPR, EC, Seq., Struc.	0.386	0.078	0.047	0.909	0.858	0.824	0.922
CFP-Gen *(w/ Motif)*	GO, IPR, EC, Seq., Struc.	0.583	0.056	0.035	0.937	0.927	0.891	0.970
CFP-Gen *(w/ GO, IPR and Motif)*	GO, IPR, EC, Seq., Struc.	**0.654**	**0.046**	**0.030**	**0.983**	**0.975**	**0.965**	**0.992**
Evaluating the Protein Functionality via predicted EC Numbers.								
Positive Control	-	1.000	0.000	0.000	0.943	0.927	0.895	0.952
Negative Control	-	0.012	0.222	0.129	0.036	0.028	0.016	0.510
DPLM (Wang et al., 2024)	Sequence Motif, Structure	0.211	0.189	0.109	0.241	0.186	0.127	0.572
ProGen2 (Nijkamp et al., 2023)	GO Term, Sequence Motif	0.530	0.127	0.075	0.519	0.387	0.318	0.661
ZymCTRL (Munsamy et al., 2022)	EC Number	0.562	0.044	0.026	0.901	0.774	0.743	0.876
CFP-Gen *(w/ EC)*	GO, IPR, EC, Seq., Struc.	0.567	0.141	0.082	0.722	0.673	0.559	0.786
CFP-Gen *(w/ EC, GO and IPR)*	GO, IPR, EC, Seq., Struc.	0.774	0.103	0.060	0.780	0.743	0.644	0.831
CFP-Gen *(w/ EC and Motif)*	GO, IPR, EC, Seq., Struc.	0.898	0.049	0.029	0.937	0.920	0.883	0.946
CFP-Gen *(w/ EC, GO, IPR and Motif)*	GO, IPR, EC, Seq., Struc.	**0.924**	**0.041**	**0.024**	**0.942**	**0.925**	**0.892**	**0.951**

To achieve the objective in Eq. 7, we employ a structure encoder f_{GVPT} (*i.e.*, GVP-Transformer) to embed the backbone atom coordinates of the protein: $\mathbf{c}_{\text{str}} = f_{\text{GVPT}}(c_{\text{str}})$. Subsequently, $\mathbf{c}_{\text{str}}$ is injected into the final ESM block of the main branch through a cross-attention layer, following the strategy of DPLM. In our implementation, we found that the pretrained cross-attention layer from DPLM could be directly utilized without the need of further training, suggesting that more functional adapters can be incorporated. The denoised features, having already incorporated both $\mathbf{c}_{\text{anno}}$ and $\mathbf{c}_{\text{seq}}$, further interact with $\mathbf{c}_{\text{str}}$ to enable a more precise and functionally coherent protein design.

Our functional protein inverse folding reduces the search space from a vast sequence space to sequences that are both structurally viable and functionally relevant. Compared to traditional inverse folding, which rely only on structural constraints, our approach incorporates functional information as an additional constraint. This leads to higher sequence recovery rates. We demonstrate significant improvements for both the zero-shot and supervised fine-tuning (SFT) versions of CFP-Gen with structural adapter in §4.3.

4. Experiments

4.1. Experimental Setup

Datasets. To collect high-quality data for training CFP-Gen, we employ expert-curated functional annotations from SwissProt (UniProtKB) (Consortium, 2019), InterPro (Hunter et al., 2009), and CARE (Yang et al., 2024) databases. Two datasets were constructed for general protein design and enzyme design, respectively. For the *general protein dataset*, we include 103,939 protein sequences covering 375 GO terms and 1,154 IPR domains. For the *enzyme design dataset*, we intersected SwissProt with the CARE dataset, yielding 139,551 enzyme sequences annotated with 661 EC numbers (4-level EC annotations). Additionally, the PDB (Berman et al., 2000) and AFDB (Jumper et al., 2021) databases were exploited to provide backbone atomic coordinates, ensuring structural constraints are incorporated into the dataset. Additional details regarding the datasets can be found in the Appendix §A.

Implementation Details. Our model is built upon the pretrained DPLM-650M and trained in two progressive stages.

Table 2. **Evaluation of functional protein inverse folding.** CFP-GEN achieves significantly higher AAR and improved functionality compared to other inverse folding methods.

Models	AAR	MRR	$F_{\max}$	scTM	pLDDT
Positive Control	100.00	0.939	0.574	0.910	86.12
ProteinMPNN (Dauparas et al., 2022)	45.28	0.133	0.470	**0.902**	85.04
ESM-IF (Hsu et al., 2022)	57.39	0.495	0.551	0.895	83.49
LM-DESIGN (Zheng et al., 2023)	63.81	0.730	0.549	0.898	**85.46**
DPLM (Wang et al., 2024)	66.94	0.721	0.552	0.883	85.33
CFP-GEN (Zero-Shot, *w/ GO and IPR*)	72.05	0.866	0.571	0.887	83.28
CFP-GEN (SFT, *w/ GO and IPR*)	**76.39**	**0.882**	**0.581**	0.889	83.53

First, we train the AGFM module to enable effective integration of functional annotations. The pretrained parameters from the first stage are then used in the next stage, where we train the copy ESM blocks in RCFE, allowing the model to condition on sequence motifs. The optimization schedule is the same as unconditional DPLM. For the structural conditioning, we directly use the GVP-Transformer and structure adapter from DPLM, which are pretrained on the CATH (Sillitoe et al., 2021) database, without additional fine-tuning for CFP-GEN. During inference, users can specify any functional constraints and their combinations as conditioning signals. Further model implementation details can be found in the Appendix §D.

4.2. Benchmarking Protein Functional Performance

In this section, we evaluate the functional properties of the generated protein sequences. To achieve this, we split two validation datasets from the general protein dataset and enzyme dataset introduced in §A. We evaluate GO and IPR functions on the *general protein dataset* and EC function on the *enzyme design dataset*. The functional annotations of each natural sequence are used as conditioning prompts for the PLMs, generating one sequence per prompt with varying length to strictly evaluate nearest-neighbor performance. We assess the similarity between the generated and the real sequences using various metrics, such as Maximum Mean Discrepancy (MMD), including its Gaussian kernel variant (MMD-G), and Mean Reciprocal Rank (MRR). Then, various function predictors (*i.e.*, $\mathcal{S}_{\text{func}}$ in Eq. 7) are applied to assign functional labels and evaluate whether the generated sequences exhibit functional consistency. Specifically, we use DeepGO-SE (Kulmanov et al., 2024) for GO label prediction, InterProScan (Blum et al., 2021) for homology-based annotation, and CLEAN (Yu et al., 2023) for catalytic function prediction. Since these predicted labels correspond to a multi-label classification problem, we adopt commonly used function prediction metrics (Kim et al., 2023) to quantitatively evaluate the alignment between predicted functions and the prompt functions (*i.e.*, ground-truths), such as micro $\boldsymbol{F}_1$-score, macro $\boldsymbol{F}_1$-score, macro AUPR and macro AUC. Details about the evaluation metrics are in Appendix §B.

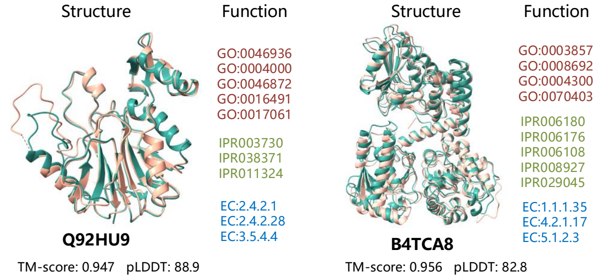


Figure 3. **Examples of multi-catalytic enzymes.** CFP-GEN generates high-quality proteins (*i.e.*, TM-score above 90) with multimodal conditions. The ground-truth structures from the AFDB database are in green, while the generated structures are in red.

We conduct performance comparison against leading PLMs under varying condition guidance. Notably, most of these PLMs support only single-condition input, whereas CFP-GEN allows combinations of input conditions across different modalities. Detailed implementation for sequence generation using each approach can be found in the Appendix §E. Furthermore, we provide positive and negative controls to establish the upper and lower bounds of these evaluation metrics. The positive control consists of the real sequences from UniProtKB, while the negative control is the unconditionally generated sequences by DPLM.

4.2.1. EVALUATION OF GO TERMS

As shown in Tab. 1, the sequences in the negative control exhibit little functional consistency with the prompt GO terms (macro $\boldsymbol{F}_1$-score = 0.02), proving the necessity of conditional generation. While both DPLM and ProteoGAN generate some functional proteins, ProGen2 achieves improved function when conditioned on the first 30 residues. In contrast, CFP-GEN demonstrates superior performance across various conditioning levels. Firstly, the generated sequences closely match the distribution of real sequences, as evidenced by low MMD. Even when conditioned only on GO terms, the model attains satisfactory functional consistency. Further integrating IPR information enhances performance, yielding a micro $\boldsymbol{F}_1$-score of 0.496 and a macro $\boldsymbol{F}_1$-score of 0.458. To further assess the impact of sequence motifs, we extract sequence fragments from IPR annotations, selecting those whose descriptions best align with the corresponding GO functions (resulting in 10–30 residues). Notably, even when using only these residues as conditioning inputs, the model generates functional sequences, reinforcing the strong correlation between sequence and function. By combining sequence motifs and functional annotations, the designed sequences outperform real sequences (positive control), achieving a macro $\boldsymbol{F}_1$-score of 0.550 *vs.* 0.522, macro AUPR of 0.435 *vs.* 0.402, and AUC of 0.795 *vs.* 0.775. These results highlight the effectiveness

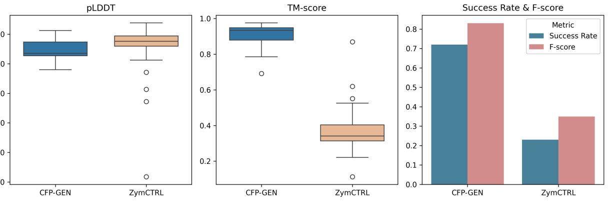


Figure 4. **Evaluation of multi-catalytic enzyme design.** Our generated proteins exhibit high designability, meanwhile achieving high success rate and functionality as validated by CLEAN.

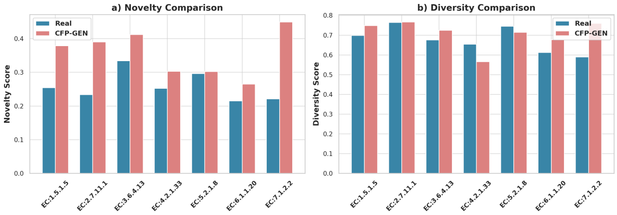


Figure 5. **Comparison of sequence novelty and diversity** between real and our designed proteins across 7 typical EC numbers from different enzyme families.

of CFP-GEN in designing functionally enriched proteins.

4.2.2. EVALUATION OF IPR DOMAINS

The evaluation results based on IPR domains provide further evidence of the superior performance of CFP-GEN. Since running InterProScan on large-scale sequences is computationally expensive, we perform the evaluation on a uniformly downsampled subset, using one-tenth of the GO validation set. We also implement the recently published ESM3, which supports IPR domain site input. Since ESM3 is a masked prediction model, we provide an additional 30 residue from the real sequences as prompts. Similar residue prompts are also given to DPLM and ProGen2 for sequence inpainting. ProGen2 demonstrates improved functionality. However, CFP-GEN demonstrates strong performance in terms of MRR and micro/macro $\boldsymbol{F}_1$-score without requiring any sequence-level prompts. When further incorporating sequence motifs, it achieves functionality comparable to that of natural proteins, *i.e.*, an macro $\boldsymbol{F}_1$-score of 0.975.

4.2.3. EVALUATION OF EC NUMBERS

In this section, we mainly compare CFP-GEN with ZymCTRL. Please note that some EC numbers are not included in ZymCTRL and we thus evaluate it on a subset, where ZymCTRL achieves slightly higher results. Additionally, we provide sequence segments derived from IPR annotations as input prompts to DPLM and ProGen2 to conduct the sequence inpainting task for further comparison. As expected, the natural sequences in the positive control exhibit strong enzymatic activity, validating CLEAN as a reliable enzyme function predictor. CFP-GEN, when conditioned only on EC numbers, generates sequences with high catalytic activity. When further incorporating GO and IPR annotations, we observe significant improvements in both distribution-level metrics (*i.e.*, improving 21% in MRR) and macro $\boldsymbol{F}_1$-score (*i.e.*, improving 7%), highlighting the importance of integrating comprehensive functional annotations. Finally, combining both functional labels and sequence motifs achieves the highest functional scores, approaching that of natural ones (*i.e.*, 0.925 *vs.* 0.924 in macro $\boldsymbol{F}_1$-score). These results further prove the effectiveness of CFP-GEN.

4.3. Functional Protein Inverse Folding

In Tab. 2, we evaluate the inverse folding task on the general protein dataset, where backbone structure information, combined with the GO and IPR annotations are provided as input. Several state-of-the-art methods, including ProteinMPNN, ESM-IF, LM-DESIGN and DPLM trained on CATH are used for comparison. Notably, most of these methods—including CFP-GEN (Zero-Shot)—employ encoder networks pretrained only on UniProtKB. In particular, CFP-GEN (Zero-Shot) does not involve any fine-tuning of DPLM's structural adapter. For comparison, we also provide a supervised fine-tuned variant, CFP-GEN (SFT), in which the structural adapter is trained on our general protein dataset using backbone structure inputs. The generated sequences are then folded using ESMFold (Lin et al., 2023) to assess their designability. Overall, most methods achieve high $\boldsymbol{F}_{\max}$-score (used in DeepGO-SE (Kulmanov et al., 2024)), revealing the strong coupling between structure and function. Moreover, LM-DESIGN and DPLM demonstrate superior AAR and MRR performance. CFP-GEN (SFT), guided by functional annotations, achieves significantly higher AAR (+9.45%) and MRR (+16.10%) compared to the baseline DPLM while maintaining comparable designability. This highlights the importance of incorporating functional annotations to guide the design process.

4.4. Multi-objective Protein Design

We further demonstrate that using combinations of multimodal conditions enables the design of multi-functional proteins. Specifically, we consider six multi-catalytic enzymes as design targets, each annotated at least three EC numbers. For each enzyme, we construct three homologous sequences as validation set. Some generated proteins are visualized in Fig. 3 to enhance interpretability. More introductions of these enzymes can be found in the Appendix §F. We fine-tune CFP-GEN on the enzyme design dataset, where the homologous enzymes with same annotations are held out from the training set. We then compare our model against ZymCTRL by generating three sequences per enzyme, and the results are illustrated in Fig. 4. In particular, we compute the average success rate across all generated proteins, where a design is considered successful if all assigned EC numbers are present in the predicted labels, while also meeting the structural criteria of TM-score and pLDDT above 70. According to success rate and macro $\boldsymbol{F}_1$-score, CFP-GEN significantly outperforms ZymCTRL, highlighting the advantages of multimodal conditioning.

4.5. Analysis of Novelty and Diversity

A crucial aspect of *de novo* protein design is the ability to generate novel and diverse proteins that do not exist in nature. To examine whether our model generates truly new proteins, we select 7 diverse EC numbers from different top-level categories: Oxidoreductases, Transferases, Hydrolases, Lyases, Isomerases, Ligases, and Translocases. For each class, CFP-GEN generates 30 sequences conditioned only on the EC number. We then compare with 30 real proteins from the enzyme validation set with the corresponding EC number. Novelty is computed by measuring how different (*i.e.*, sequence identity after alignment) each generated sequence is from its most similar real protein in the training set, while diversity is computed by capturing how different the generated sequences are from the overall training set. To ensure both metrics are interpretable in the same direction, we subtract the scores from 1 (*i.e.*, higher is better). The results are shown in Fig. 5. We observe that CFP-GEN consistently achieves higher sequence novelty across all 7 EC numbers, demonstrating its strong potential for *de novo* protein design beyond simply replicating known sequences. Moreover, the generated sequences exhibit high intra-class diversity in 5 out of the 7 EC categories. These results suggest that the model has learned a more generalized representation, rather than overfitting to training examples. Additional analysis of CFP-GEN is presented in Appendix §C.

5. Conclusion and Future Work

This work introduces a novel PLM, CFP-GEN, for functional protein design with a multimodal conditioning mechanism. The core of CFP-GEN comprises three modules, each addressing functional annotations, sequence motifs, and backbone structures, respectively. Specifically, an Annotation-Guided Feature Modulation (AGFM) module is designed to flexibly and smoothly incorporate composable functional tags, such as GO terms, IPR domains, and EC numbers. A Residue-Controlled Functional Encoding (RCFE) module is introduced to encode crucial sequence motifs, facilitating scenarios where functional annotations are unavailable. Finally, a structure adapter is integrated into CFP-GEN without requiring additional training, ensuring coherence between function and structure. We evaluate CFP-GEN on three tasks: functional sequence generation, functional inverse folding, and multi-functional protein design, where it consistently outperforms existing approaches.

Although CFP-GEN represents an early attempt to jointly integrate multiple modalities for conditional protein generation, it has several limitations. First, it currently supports only a subset of GO, IPR, and EC labels, limiting its generalizability to broader functional categories. Second, beyond commonly used functional annotations, richer conditional inputs—such as physicochemical properties (*e.g.*, hydrophobicity, charge, polarity)—are also essential to meet the requirements of bio-manufacturing applications. Moreover, sequence–structure co-design should be incorporated to enable end-to-end protein design. Future work will focus on scaling up the training datasets, enriching the scope of functional annotations, and further advancing unified models for more scalable and function-aware protein generation.

Impact Statement

This work introduces a multimodal conditional protein design framework, enabling precise control over generated proteins by integrating diverse functional constraints. This approach has broad potential applications in enzyme engineering, drug discovery, and synthetic biology. We also provide a comprehensive benchmark of existing controllable PLMs, offering valuable insights for researchers utilizing these computational tools.

From an ethical standpoint, our model is trained entirely on publicly available datasets (*e.g.*, UniProtKB), ensuring reproducibility and transparency. However, potential biases may arise from unequal representation across protein families and functions, as public datasets often reflect human-curated biases. Addressing these imbalances requires continuous expansion of training data and careful validation on diverse proteins. Biosecurity risks must also be carefully considered. The ability of generative models to design novel proteins raises dual-use concerns, particularly in the context of biomedical applications. The safe use of biological generative models requires risk assessment, regulatory compliance and safeguards to prevent misuse while ensuring ethical and beneficial applications.

Acknowledgements

This work was supported by the King Abdullah University of Science and Technology (KAUST) Office of Research Administration (ORA) under Award No REI/1/5234-01-01, REI/1/5414-01-01, REI/1/5289-01-01, REI/1/5404-01-01, REI/1/5992-01-01, URF/1/4663-01-01, Center of Excellence for Smart Health (KCSH), under award number 5932, and Center of Excellence for Generative AI, under award number 5940. For computer time, this research used Ibex managed by the Supercomputing Core Laboratory at KAUST in Thuwal, Saudi Arabia.

References

The gene ontology resource: enriching a gold mine. *Nucleic Acids Research*, 49(D1):D325–D334, 2021.

Alamdari, S., Thakkar, N., van den Berg, R., Tenenholtz, N., Strome, B., Moses, A., Lu, A. X., Fusi, N., Amini, A. P., and Yang, K. K. Protein generation with evolutionary diffusion: sequence is all you need. *BioRxiv*, pp. 2023–09, 2023.

Austin, J., Johnson, D. D., Ho, J., Tarlow, D., and Van Den Berg, R. Structured denoising diffusion models in discrete state-spaces. In *NeurIPS*, 2021.

Baek, M., DiMaio, F., Anishchenko, I., Dauparas, J., Ovchinnikov, S., Lee, G. R., Wang, J., Cong, Q., Kinch, L. N., Schaeffer, R. D., et al. Accurate prediction of protein structures and interactions using a three-track neural network. *Science*, 373(6557):871–876, 2021.

Bateman, A., Birney, E., Durbin, R., Eddy, S. R., Howe, K. L., and Sonnhammer, E. L. The pfam protein families database. *Nucleic Acids Research*, 28(1):263–266, 2000.

Berman, H. M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T. N., Weissig, H., Shindyalov, I. N., and Bourne, P. E. The protein data bank. *Nucleic Acids Research*, 28(1): 235–242, 2000.

Blum, M., Chang, H.-Y., Chuguransky, S., Grego, T., Kandasaamy, S., Mitchell, A., Nuka, G., Paysan-Lafosse, T., Qureshi, M., Raj, S., et al. The interpro protein families and domains database: 20 years on. *Nucleic Acids Research*, 49(D1):D344–D354, 2021.

Consortium, U. Uniprot: a worldwide hub of protein knowledge. *Nucleic Acids Research*, 47(D1):D506–D515, 2019.

Dauparas, J., Anishchenko, I., Bennett, N., Bai, H., Ragotte, R. J., Milles, L. F., Wicky, B. I., Courbet, A., de Haas, R. J., Bethel, N., et al. Robust deep learning–based protein sequence design using proteinmpnn. *Science*, 378 (6615):49–56, 2022.

Ferruz, N. and Höcker, B. Controllable protein design with language models. *Nature Machine Intelligence*, 4(6): 521–532, 2022.

Ferruz, N., Schmidt, S., and Höcker, B. Protgpt2 is a deep unsupervised language model for protein design. *Nature Communications*, 13(1):4348, 2022.

Goverde, C. A., Pacesa, M., Goldbach, N., Dornfeld, L. J., Balbi, P. E., Georgeon, S., Rosset, S., Kapoor, S., Choudhury, J., Dauparas, J., et al. Computational design of soluble and functional membrane protein analogues. *Nature*, pp. 1–10, 2024.

Hayes, T., Rao, R., Akin, H., Sofroniew, N. J., Oktay, D., Lin, Z., Verkuil, R., Tran, V. Q., Deaton, J., Wiggert, M., et al. Simulating 500 million years of evolution with a language model. *Science*, pp. eads0018, 2025a.

Hayes, T., Rao, R., Akin, H., Sofroniew, N. J., Oktay, D., Lin, Z., Verkuil, R., Tran, V. Q., Deaton, J., Wiggert, M., et al. Simulating 500 million years of evolution with a language model. *Science*, pp. eads0018, 2025b.

Hsu, C., Verkuil, R., Liu, J., Lin, Z., Hie, B., Sercu, T., Lerer, A., and Rives, A. Learning inverse folding from millions of predicted structures. In *ICML*, 2022.

Hunter, S., Apweiler, R., Attwood, T. K., Bairoch, A., Bateman, A., Binns, D., Bork, P., Das, U., Daugherty, L., Duquenne, L., et al. Interpro: the integrative protein signature database. *Nucleic Acids Research*, 37(suppl_1): D211–D215, 2009.

Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., Tunyasuvunakool, K., Bates, R., Žídek, A., Potapenko, A., et al. Highly accurate protein structure prediction with alphafold. *Nature*, 596(7873):583–589, 2021.

Kenton, J. D. M.-W. C. and Toutanova, L. K. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, 2019.

Kim, G. B., Kim, J. Y., Lee, J. A., Norsigian, C. J., Palsson, B. O., and Lee, S. Y. Functional annotation of enzyme-encoding genes using deep learning with transformer layers. *Nature Communications*, 14(1):7370, 2023.

Krishna, R., Wang, J., Ahern, W., Sturmfels, P., Venkatesh, P., Kalvet, I., Lee, G. R., Morey-Burrows, F. S., Anishchenko, I., Humphreys, I. R., et al. Generalized biomolecular modeling and design with rosettafold all-atom. *Science*, 384(6693):eadl2528, 2024.

Kucera, T., Togninalli, M., and Meng-Papaxanthos, L. Conditional generative modeling for de novo protein design with hierarchical functions. *Bioinformatics*, 38(13):3454–3461, 2022.

Kulmanov, M., Khan, M. A., and Hoehndorf, R. Deepgo: predicting protein functions from sequence and interactions using a deep ontology-aware classifier. *Bioinformatics*, 34(4):660–668, 2018.

Kulmanov, M., Guzmán-Vega, F. J., Duek Roggli, P., Lane, L., Arold, S. T., and Hoehndorf, R. Protein function prediction as approximate semantic entailment. *Nature Machine Intelligence*, 6(2):220–228, 2024.

Lin, Z., Akin, H., Rao, R., Hie, B., Zhu, Z., Lu, W., Smetanin, N., Verkuil, R., Kabeli, O., Shmueli, Y., et al.

Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637): 1123–1130, 2023.

Lisanza, S. L., Gershon, J. M., Tipps, S. W., Sims, J. N., Arnoldt, L., Hendel, S. J., Simma, M. K., Liu, G., Yase, M., Wu, H., et al. Multistate and functional protein design using rosettafold sequence space diffusion. *Nature Biotechnology*, pp. 1–11, 2024.

Listov, D., Goverde, C. A., Correia, B. E., and Fleishman, S. J. Opportunities and challenges in design and optimization of protein function. *Nature Reviews Molecular Cell Biology*, pp. 1–15, 2024.

Madani, A., Krause, B., Greene, E. R., Subramanian, S., Mohr, B. P., Holton, J. M., Olmos, J. L., Xiong, C., Sun, Z. Z., Socher, R., et al. Large language models generate functional protein sequences across diverse families. *Nature Biotechnology*, 41(8):1099–1106, 2023.

Munsamy, G., Lindner, S., Lorenz, P., and Ferruz, N. Zymctrl: a conditional language model for the controllable generation of artificial enzymes. In *NeurIPS Workshop*, 2022.

Nijkamp, E., Ruffolo, J. A., Weinstein, E. N., Naik, N., and Madani, A. Progen2: exploring the boundaries of protein language models. *Cell systems*, 14(11):968–978, 2023.

Qu, W., Guan, J., Ma, R., Zhai, K., Wu, W., and Wang, H. P (all-atom) is unlocking new path for protein design. *bioRxiv*, pp. 2024–08, 2024.

Ren, M., Zhu, T., and Zhang, H. Carbonnovo: Joint design of protein structure and sequence using a unified energy-based model. In *ICML*, 2024.

Schomburg, I., Chang, A., Ebeling, C., Gremse, M., Heldt, C., Huhn, G., and Schomburg, D. Brenda, the enzyme database: updates and major new developments. *Nucleic Acids Research*, 32(suppl_1):D431–D433, 2004.

Sillitoe, I., Bordin, N., Dawson, N., Waman, V. P., Ashford, P., Scholes, H. M., Pang, C. S., Woodridge, L., Rauer, C., Sen, N., et al. Cath: increased structural coverage of functional space. *Nucleic Acids Research*, 49(D1): D266–D273, 2021.

Wang, X., Zheng, Z., Ye, F., Xue, D., Huang, S., and Gu, Q. Diffusion language models are versatile protein learners. In *ICML*, 2024.

Watson, J. L., Juergens, D., Bennett, N. R., Trippe, B. L., Yim, J., Eisenach, H. E., Ahern, W., Borst, A. J., Ragotte, R. J., Milles, L. F., et al. De novo design of protein structure and function with rfdiffusion. *Nature*, 620(7976): 1089–1100, 2023.

Yang, J., Mora, A., Liu, S., Wittmann, B., Anandkumar, A., Arnold, F., and Yue, Y. Care: a benchmark suite for the classification and retrieval of enzymes. In *NeurIPS*, 2024.

Yu, T., Cui, H., Li, J. C., Luo, Y., Jiang, G., and Zhao, H. Enzyme function prediction using contrastive learning. *Science*, 379(6639):1358–1363, 2023.

Zhang, L., Rao, A., and Agrawala, M. Adding conditional control to text-to-image diffusion models. In *ICCV*, 2023.

Zheng, Z., Deng, Y., Xue, D., Zhou, Y., Ye, F., and Gu, Q. Structure-informed language models are protein designers. In *ICML*, 2023.

Supplementary Material of CFP-Gen

A. Dataset Curation

Existing PLMs for protein generation often focus on either unconditional sequence generation or single-condition controllable generation. For instance, ProteoGAN (Kucera et al., 2022) utilizes GO terms for functional control, ESM3 (Hayes et al., 2025b) leverages IPR domain information, and ZymCTRL (Munsamy et al., 2022) conditions on EC numbers for enzyme design. However, relying on single-condition constraints makes these datasets unsuitable for training a multimodal, multi-condition controllable PLM like CFP-Gen, necessitating the curation of new datasets for functional protein design.

A protein is usually defined by three different modalities, *i.e.*, function, sequence, and structure. These three aspects are generally considered to be highly interdependent and mutually constraining, as a protein's sequence determines its structural fold, while both sequence and structure influence its functional properties. Given this intrinsic coupling, we annotate each protein from all three perspectives to ensure a comprehensive representation. Specifically, we construct two specialized datasets to facilitate functional protein design: (1) A general protein design dataset with GO terms, IPR domains, functional sequence segments (motifs) and protein structures. (2) An enzyme design dataset specifically curated for generating catalytically active proteins by further adding EC number annotations.

A.1. The General Protein Dataset.

Gene Ontology (GO) terms provide a structured and comprehensive vocabulary for annotating proteins with respect to their molecular functions, biological processes, and cellular components, thereby offering a proper framework for understanding protein roles in various biological contexts. In this work, we focus primarily on molecular function annotations, which offer a more direct and interpretable characterization of each protein's activity. In addition to GO annotations, we incorporate InterPro (IPR) numbers, which categorize proteins based on conserved domains and families, facilitating the identification of functional and structural relationships across diverse protein sequences. Combining GO and IPR labels allows for a deeper understanding of proteins from both global and local perspectives and thus facilitating the protein generation process.

To build the general protein dataset, we collect protein sequences from the manually curated Swiss-Prot subset of UniProtKB (Consortium, 2019), ensuring high annotation reliability. To maintain balanced class distributions, we implement a filtering strategy that retains only GO and IPR terms with at least 100 annotated protein sequences, as suggested by ProteoGAN, to mitigate the impact of long-tail categories. Although this inevitably excludes some sequences from the training set, it has demonstrated the model's capability to generalize across commonly used functional labels. Besides, our method primarily focuses on integrating different modalities as conditioning signals, enabling more comprehensive protein function modeling. We leave the addressing of long-tail data challenges in large-scale diffusion models to future iterations.

The final dataset contains 103,939 protein sequences covering 375 GO terms and 1,154 IPR domains. The distribution of GO and IPR annotations in our dataset is illustrated in Fig. 6. As shown in the GO annotation distribution, 30.6% of GO categories contain 100–200 protein sequences, while only 6.4% have more than 1,000 sequences. Similarly, in the IPR annotation distribution, 35.4% of IPR categories consist of 100–200 sequences, whereas only 2.0% contain more than 1,000 sequences. Despite the remaining long-tail distribution in the dataset, we observe that the model generalizes well across different categories, as indicated by the macro F_1 score in Tab. 1.

Additionally, for each IPR number, we extract motif information from the annotated domain boundary positions. During both training and inference, we select the IPR domain most relevant to the given protein's GO annotations (if multiple domains are present, we use their intersection) to define the functional sequence segments. These segments (motifs) are then used as sequence-level conditioning inputs to enhance the model's ability to capture functionally relevant patterns. Furthermore, for each retained protein sequence, we retrieve its corresponding structural information from PDB and AlphaFold DB (AFDB) databases. Specifically, we extract the backbone atomic coordinates of 'N', 'CA', 'C', and 'O' and used them as structure condition. Our analysis reveals that the backbone structure plays a crucial role in determining protein function, proving the importance of incorporating structural features into PLMs.

To evaluate GO function, we construct a validation set by selecting 30 sequences per GO/IPR label, resulting in a subset of 8,309 sequences. The training dataset is then formed by holding out these sequences to ensure an unbiased evaluation. For IPR function assignment, we further perform a 10-fold uniform downsampling of the GO validation set, yielding 831 sequences, to alleviate the computational overhead associated with InterProScan.

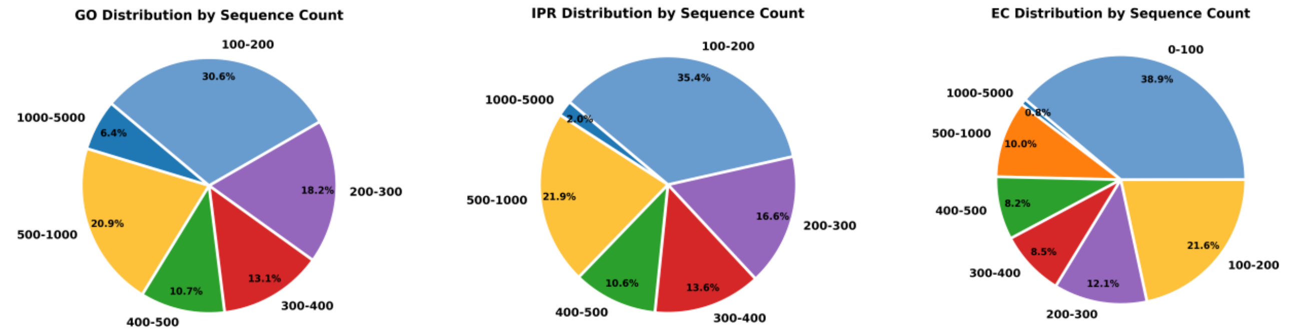


Figure 6. **Categorical distribution of functional annotations in the two datasets.** Most functional labels contain fewer than 500 sequences, highlighting the long-tail distribution issue. However, CFP-GEN exhibits strong generalization across these functions.

A.2. The Enzyme Design Dataset.

Enzymes, as a critically important class of proteins with wide applications in biotechnology and industrial catalysis, requiring a specialized dataset for high-quality enzyme design. EC numbers (Enzyme Commission numbers) classify enzymes based on their catalytic activities. Linking protein sequences to specific biochemical reactions facilitates a better understanding of enzyme function. To construct a robust enzyme dataset, we leverage CARE (Yang et al., 2024), a newly curated enzyme dataset comprising 185,995 sequences, each annotated with a level-4 EC number for precise functional clustering. To ensure compatibility with GO and IPR annotations, we extract the subset of CARE that overlaps with Swiss-Prot, yielding a dataset of 139,551 sequences spanning 661 EC labels. The distribution of EC categories is illustrated in Fig. 6.

In line with our approach for the General Protein Dataset, we also integrate sequence motifs from IPR annotations and structural information from PDB/AFDB databases. To enable rigorous evaluation, we construct a validation set by sampling 30 sequences per EC label, resulting in a high-quality evaluation set of 16,187 sequences, while the remaining data is allocated to the training set. Besides, multi-catalytic enzymes in Fig. 4 are also held out from the training set to ensure a fair evaluation. As demonstrated in Tab. 1, our model achieves catalytic function comparable to natural enzymes on this large-scale validation set, as assessed by the widely used CLEAN tool.

B. Evaluation Metrics

We conduct a comprehensive evaluation of the generated proteins across multiple levels to assess their quality rigorously:

(1) Sequence-level evaluation: We measure the similarity between the generated sequences and natural protein sequences. Additionally, we analyze the inter-group distances by clustering sequences based on their functional labels, ensuring that functionally related sequences exhibit appropriate distribution.

(2) Function-level evaluation: We assess the functional consistency between the functions of generated sequences (*i.e.*, estimated by SOTA function predictors) and their input prompts (*i.e.*, derived from natural sequences). This is evaluated through multi-label classification metrics, ensuring that the predicted functions align with their prompt functions.

(3) Structure-level evaluation: We evaluate the structural consistency of the generated sequences by predicting their 3D structures using ESMFold and comparing them with their corresponding PDB/AFDB structures. This ensures that the generated proteins not only maintain sequence-level fidelity but also exhibit high designability.

B.1. Sequence Evaluation Metrics

- **Maximum Mean Discrepancy (MMD)**: Maximum Mean Discrepancy (MMD) measures the distribution difference between generated protein sequences P and their corresponding natural sequences S. Due to its efficiency and high expressive capacity, we adopt normalized Spectrum Mapping to obtain sequence distribution, as suggested by ProteoGAN. This method computes k-mer occurrences within a sequence, providing a sufficiently rich representation to effectively distinguish protein properties. Given a kernel function $k(\cdot,\cdot)$, MMD is computed as:

$$\mathrm{MMD}^2(S,P)=\mathbb{E}_{s,s'\sim S}[k(s,s')]+\mathbb{E}_{p,p'\sim P}[k(p,p')]-2\mathbb{E}_{s\sim S,p\sim P}[k(s,p)] \tag{8}$$

where $s, s' \in S$ are real sequences, and $p, p' \in P$ are generated sequences. A lower MMD value indicates better alignment between the generated and real distributions. For a linear kernel, the MMD computation simplifies to a direct Euclidean distance between the mean embeddings of the two distributions:

$$\text{MMD}_{\text{linear}}(S, P) = \|\mu_S - \mu_P\|, \quad \text{where,} \quad \mu_S = \frac{1}{m}\sum_{i=1}^{m} s_i, \quad \mu_P = \frac{1}{n}\sum_{j=1}^{n} p_j \tag{9}$$

represent the mean embeddings of the real and generated sequences, respectively, and $\|\cdot\|$ denotes the Euclidean norm.

- **Gaussian Kernel MMD (MMD-G)**: For an RBF (Gaussian) kernel, the MMD computation incorporates the radial basis function (RBF) to capture higher-order distributional differences:

$$\text{MMD}^2_{\text{Gaussian}}(S, P) = \frac{1}{m^2}\sum_{i,j} k(s_i, s_j) + \frac{1}{n^2}\sum_{i,j} k(p_i, p_j) - \frac{2}{mn}\sum_{i,j} k(s_i, p_j) \tag{10}$$

where the Gaussian kernel function is defined as:

$$k(x, y) = \exp\left(-\gamma\|x - y\|^2\right) \tag{11}$$

with γ typically determined via a median heuristic approach to adaptively select an appropriate kernel bandwidth, *i.e.*,

$$\sigma = \text{median}(|x_i - x_j|_{i,j}), \quad \gamma = \frac{1}{2\sigma^2} \tag{12}$$

where σ is the median of all pairwise Euclidean distances between embeddings from both groups.

By considering both the linear kernel MMD, which captures global differences, and the Gaussian kernel MMD, which captures higher-order distributional variations, we ensure a robust evaluation of the similarity between real and generated protein distributions.

- **Mean Reciprocal Rank (MRR)**: Mean Reciprocal Rank (MRR) evaluates how well the generated sequences P match the real sequences S in a function-aware manner based on linear MMD distance.

We define $S = \{S_c\}_{c=1}^{C}$ as a set of real protein sequences grouped by their function label c, where C is the total number of function labels (*i.e.*, GO/IPR/EC annotations). Similarly, let $P = \{P_c\}_{c=1}^{C}$ be the corresponding generated protein sequences for each functional group. The goal is to assess whether the generated sequences for each function closely match the real ones. The MRR metric is defined as:

$$\text{MRR}(S, P) = \frac{1}{C}\sum_{c=1}^{C} \frac{1}{\text{rank}_S(\text{MMD}(S_c, P_c))} \tag{13}$$

where $\text{rank}_S(\text{MMD}(S_c, P_c))$ represents the ranking position of the MMD score between the real sequences S_c and generated sequences P_c. Specifically, for each function label pair (c, c'), we compute $\text{MMD}(S_c, P_{c'})$, measuring the distributional difference between the real sequences of function c and the generated sequences of function c'. The ranking of $\text{MMD}(S_c, P_c)$ is then determined by sorting these MMD values across all function pairs.

MRR ensures that the evaluation considers not only how well each function's generated sequences match their corresponding real sequences but also their relative distinction from other functional groups. A higher MRR indicates better functional alignment between the real and generated distributions. When $\text{MRR}(S, P) = 1$, it signifies that for each functional class, the generated sequences are the closest in distribution to their corresponding real sequences, demonstrating high-quality protein generation.

B.2. Function Evaluation Metrics

- **Macro $\boldsymbol{F}_1$-score**: Macro $\boldsymbol{F}_1$-score computes the per-class $\boldsymbol{F}_1$-scores, treating all classes equally, which is denoted as:

$$\text{macro } \boldsymbol{F}_1 = \frac{1}{C}\sum_{i=1}^{C} \frac{2 \cdot \text{Precision}_i \cdot \text{Recall}_i}{\text{Precision}_i + \text{Recall}_i} \tag{14}$$

where C is the number of classes, and Precision_i, Recall_i are the precision and recall for class i. This metric is particularly useful for evaluating performance across imbalanced datasets, as it ensures that smaller classes contribute equally to the final score.

- **Micro $\boldsymbol{F}_1$-score**: In contrast, micro $\boldsymbol{F}_1$-score aggregates the counts of true positives, false positives, and false negatives across all classes before computing a single $\boldsymbol{F}_1$-score:

$$\text{micro } \boldsymbol{F}_1 = \frac{2 \cdot \text{micro Precision} \cdot \text{micro Recall}}{\text{micro Precision} + \text{micro Recall}} \tag{15}$$

  where micro precision and micro recall are computed as:

$$\text{micro Precision} = \frac{\sum_{i=1}^{C} TP_i}{\sum_{i=1}^{C} (TP_i + FP_i)}, \quad \text{micro Recall} = \frac{\sum_{i=1}^{C} TP_i}{\sum_{i=1}^{C} (TP_i + FN_i)} \tag{16}$$

  where TP_i, FP_i, and FN_i represent true positives, false positives, and false negatives for class i, respectively. Unlike macro $\boldsymbol{F}_1$-score, micro $\boldsymbol{F}_1$-score gives more weight to high-frequency classes.

- **Macro AUPR (Area Under the Precision-Recall Curve)**: Macro AUPR computes the area under the precision-recall curve for each class and averages them:

$$\text{macro AUPR} = \frac{1}{C} \sum_{i=1}^{C} \text{AUPR}_i \tag{17}$$

  where AUPR_i is the precision-recall area for class i. This metric is particularly valuable in scenarios with highly imbalanced datasets, where precision-recall trade-offs are more informative than ROC-based metrics.

- **Macro AUC (Area Under the Receiver Operating Characteristic Curve)**: Macro AUC measures classification performance by averaging the AUC values across all classes:

$$\text{macro AUC} = \frac{1}{C} \sum_{i=1}^{C} \text{AUC}_i \tag{18}$$

  where AUC_i is the ROC-AUC for class i. Higher values indicate better model discrimination between positive and negative instances.

- **$\boldsymbol{F}_{\max}$-score**: In Tab. 1, we evaluate function scores using standard multi-label metrics, with confidence thresholds of 0.1 for GO and 0.01 for EC. To account for the hierarchical structure of GO functions, ancestor GO terms are also incorporated in evaluation. Additionally, in Tab. 2, to align with DeepGO-SE, we also consider a $\boldsymbol{F}_{\max}$-score computed at the sequence level. Given a confidence threshold τ (ranging from 0.1 to 1.0 in increments of 0.01), we define: - TP_τ: The number of correctly predicted GO terms present in the ground truth set of each sequence. - FP_τ: The number of predicted GO terms that do not exist in the ground truth of each sequence. - FN_τ: The number of ground truth GO terms that were not predicted. Using these, Precision_τ and Recall_τ at threshold τ are calculated as:

$$\text{Precision}_\tau = \frac{TP_\tau}{TP_\tau + FP_\tau}, \quad \text{Recall}_\tau = \frac{TP_\tau}{TP_\tau + FN_\tau} \tag{19}$$

  The $\boldsymbol{F}_{\max}$-score is then obtained by maximizing the $\boldsymbol{F}_1$-score over a range of confidence thresholds:

$$\boldsymbol{F}_{\max} = \max_{\tau} \left\{ \frac{2 \cdot \text{Precision}_\tau \cdot \text{Recall}_\tau}{\text{Precision}_\tau + \text{Recall}_\tau} \right\} \tag{20}$$

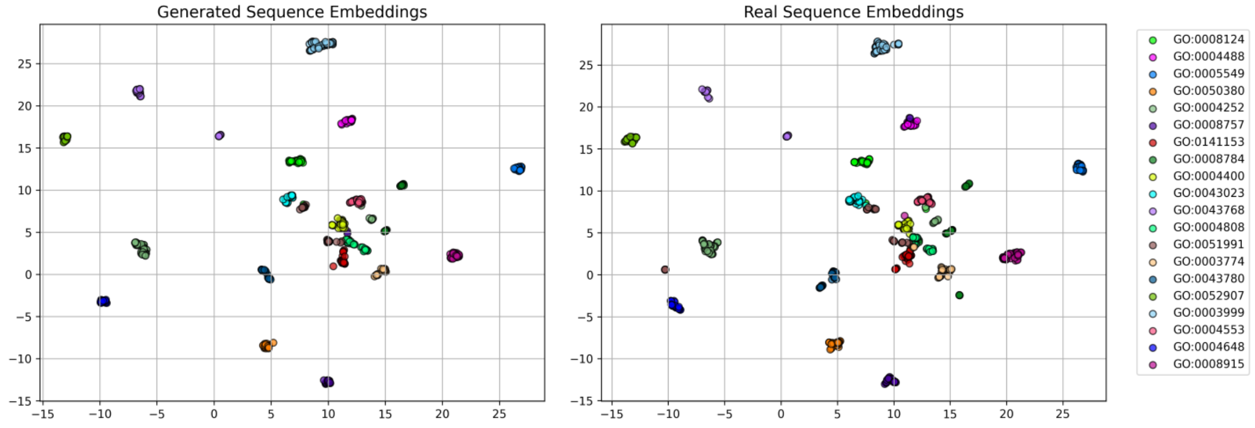
  Unlike macro $\boldsymbol{F}_1$-score, which averages per-class performance, $\boldsymbol{F}_{\max}$ focuses on per-sequence predictions.

B.3. Structure Evaluation Metrics

- **Amino Acid Recovery (AAR)**: AAR measures how well a generated protein sequence $\mathbf{x}'$ matches a reference sequence $\mathbf{x}$, given the protein structures as model inputs. Let $\mathbf{x}' = (x'_1, x'_2, \ldots, x'_L)$ and $\mathbf{x} = (x_1, x_2, \ldots, x_L)$ denote the generated sequence and the reference sequence of length L. AAR is then defined as:

$$\text{AAR} = \frac{1}{L} \sum_{i=1}^{L} \delta(x'_i, x_i), \quad \delta(x'_i, x_i) = \begin{cases} 1, & \text{if } x'_i = x_i \\ 0, & \text{otherwise} \end{cases} \tag{21}$$

  The function $\delta(x'_i, x_i)$ is the indicator function. A higher AAR value indicates greater sequence fidelity to the reference, meaning the generated sequence closely resembles the original target sequence at the residue level.



(a) Embeddings of generated sequences based on different GO functions. **(b) Embeddings of real sequences based on different GO functions.**

Figure 7. **UMAP visualization of sequence embeddings conditioned on GO functions.** The generated sequences from CFP-GEN exhibit a highly similar distribution to natural sequences, demonstrating the strong expressiveness and functional consistency of our approach.

- **TM-score (scTM)**: scTM globally evaluates the structural similarity between a generated protein sequence $\mathbf{x}'$ and a reference protein sequence $\mathbf{x}$, considering the backbone structure c_{str} as a conditioning factor. Let $\mathbf{F}(x)$ and $\mathbf{F}(x')$ be the 3D structures predicted for sequences $\mathbf{x}$ and $\mathbf{x}'$ respectively (using ESMFold in our case), then scTM is defined as:

$$\text{scTM} = \text{TM-score}(\mathbf{F}(x'), \mathbf{F}(x) \mid c_{\text{str}}) \tag{22}$$

  Higher scTM scores indicate better structural consistency under backbone structure guidance.

- **Predicted Local Distance Difference Test (pLDDT)**: pLDDT quantifies the confidence of a predicted protein structure by assessing the local agreement of residue positions. Given a predicted structure $\mathbf{F}(x')$ with L residues, the pLDDT score is computed as:

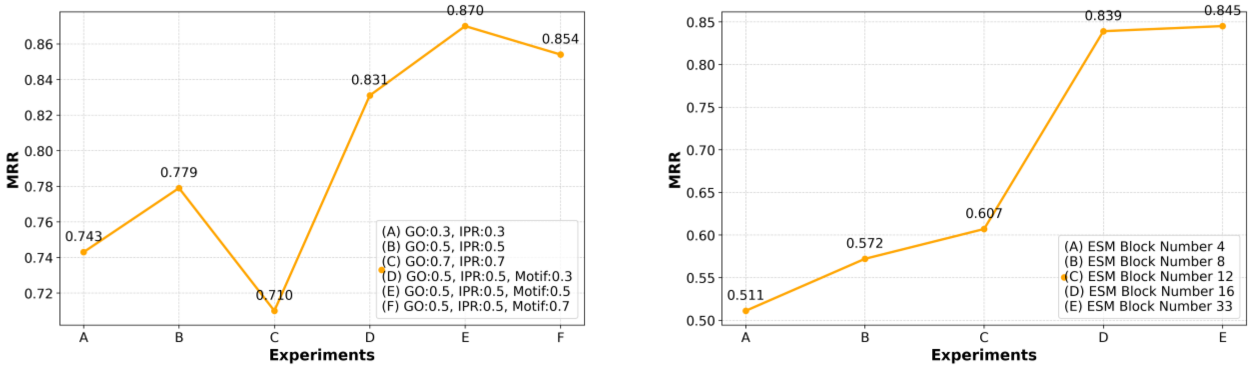
$$\text{pLDDT} = \frac{1}{L}\sum_{i=1}^{L} \text{pLDDT}_i \tag{23}$$

  where pLDDT_i is the per-residue confidence score of $x_i \in \{x'_1, x'_2, \ldots, x'_L\}$ derived from the folding model. A higher pLDDT score indicates greater reliability in the predicted structure.

C. Extensive Analysis of CFP-GEN

C.1. Analysis of Generated Protein Embeddings

To evaluate the alignment between generated and real protein sequences, we compare their embedding distributions across 20 randomly selected GO terms, with 30 sequences per GO term for both real and generated data. The embeddings are obtained using Spectrum Mapping, a highly expressive sequence representation method, and visualized via UMAP for dimensionality reduction. As shown in Fig. 7, the distributions of generated sequences closely align with those of real sequences, with each functional category maintaining a consistent spatial position across both plots. The inter-cluster distances remain well-preserved, indicating that proteins with different functions occupy distinct embedding spaces, while the intra-class compactness suggests that functional specificity is effectively captured. Additionally, the model generalizes well across diverse functional spaces, preserving structured and biologically meaningful separations. Minor overlaps in some central regions suggest that certain sequences may exhibit multiple GO functions, reflecting the inherent complexity of protein function annotation. The overall results demonstrate that CFP-GEN leverages Spectrum Mapping embeddings to effectively preserve the functional landscape of protein sequences, ensuring both functional coherence and class separability in the learned representation space.



(a) Model performance with varying control intensity of each functional condition.

(b) Model performance with varying numbers of copy ESM blocks in the RCFE module.

Figure 8. **Hyperparameter Analysis of CFP-GEN.** We examine the the impact of control intensity and the number of copy ESM blocks.

C.2. Mitigation of Mode Collapse.

One common issue with PLMs is mode collapse in the generated sequences. To evaluate whether CFP-GEN alleviates this problem, we analyzed the frequency of repeated n-gram patterns ($n = 2, 3, 4, 5, 6$) in sequences generated under GO-conditioned prompts, as reported in Tab.1. Sequences generated by DPLM are used as a reference, and real protein sequences from the validation set serve as a positive control.

As shown in Tab. 3, CFP-GEN produces a similar number of 2-grams to real proteins, while significantly reducing the number of longer repetitive n-grams, especially 4-gram to 6-gram patterns. Notably, the more functional conditions (*e.g.*, GO, IPR, Motif) are provided, the fewer repetitive patterns appear in the output, indicating better sequence quality and reduced mode collapse. These results provide strong evidence that CFP-GEN effectively alleviates the mode collapse issue observed in mainstream PLMs.

Table 3. Comparison of n-gram counts across different methods.

Method	**2-gram**	**3-gram**	**4-gram**	**5-gram**	**6-gram**
Positive Control	404	164	0	0	0
DPLM	315	462	104	46	26
CFP-GEN (*w/ GO*)	363	351	15	9	8
CFP-GEN (*w/ GO and IPR*)	365	332	9	5	4
CFP-GEN (*w/ GO, IPR and Motif*)	377	336	4	1	1

D. Hyperparameter Details

As introduced in the main paper, unless otherwise specified, most of the learning strategies and hyperparameters of the diffusion model remain consistent with those of DPLM. The batch size is set to 1 million tokens, and training is conducted on 8 NVIDIA A100 GPUs for around 72 hours of each stage. The AdamW optimizer is employed with a maximum learning rate of 0.00004. During inference, we allow the model to perform 100 sampling steps, following the DPLM conditional generation, with sequence length varying from 200 to 400. The total model size of CFP-GEN is 1.48B parameters, excluding the GVP-transformer structure encoder.

During training, we adjust $\gamma(c)$ in Eq. 5 by applying dropout to each conditioning input. The probability of randomly dropping out each condition and its corresponding impact on model performance is shown in Fig. 8 (a). MRR is used to present the performance as we found this metric is more robust and expressive. The results indicate that setting the dropout probability to 0.5 for each condition achieves the best performance. Additionally, we investigate the optimal number of ESM block copies in the RCFE module. We compare the model performance conditioning on only the sequence motifs. As shown in Fig. 8 (b), setting the number of blocks to 16 achieves the best trade-off between performance and model complexity.

E. Implementation of Existing PLMs

For ProGen2, we provide the model with GO terms along with the first 30 residues of the real sequence as prompts. When evaluating IPR/EC functions, only the first residues are used, as ProGen2 does not support EC/IPR annotations.

For ProteoGAN, we directly input the GO terms of the real sequence. However, since ProteoGAN only supports 50 predefined GO terms, for GO terms not included in ProteoGAN's vocabulary, we attempt to map them to their closest ancestor terms that are supported. If no suitable ancestor is found, the sequence is ignored.

For discrete diffusion models such as DPLM, we provide the model with functional motifs (30 residues) and task it with performing sequence inpainting to reconstruct the missing residues.

For ESM3, we input both the IPR domain descriptions and their start-end positions, along with 30 residues to initialize the sequence generation process.

For ZymCTRL, only a single EC number per sequence is provided, as the model does not support multi-label inputs. If an EC number is not supported by ZymCTRL, we exclude the sequence from evaluation to ensure a fair comparison.

F. Introduction of Multi-catalytic Enzymes

- $5'/3'$-Nucleotidase SurE (**EC Numbers: 3.1.3.5, 3.1.3.6, 3.6.1.11**): This nucleotidase has broad substrate specificity, dephosphorylating various ribo- and deoxyribonucleoside $5'$-monophosphates and ribonucleoside $3'$-monophosphates, with the highest affinity for $3'$-AMP. It also hydrolyzes polyphosphate (exopolyphosphatase activity), preferentially targeting short-chain-length substrates (P20-25). This enzyme is potentially involved in the regulation of dNTP and NTP pools, as well as the turnover of $3'$-mononucleotides, which are produced by various intracellular RNases (T1, T2, and F) during RNA degradation.

- Fatty Acid Oxidation Complex Subunit Alpha (**EC Numbers: 1.1.1.35, 4.2.1.17, 5.1.2.3, 5.3.3.8**): This enzyme plays a crucial role in the fatty acid β-oxidation cycle, enabling both aerobic and anaerobic degradation of long-chain fatty acids. It catalyzes the conversion of enoyl-CoA to 3-oxoacyl-CoA via L-3-hydroxyacyl-CoA and can also process D-3-hydroxyacyl-CoA and cis-3-enoyl-CoA as substrates. This pathway is fundamental for cellular energy production and lipid metabolism.

- Geranylgeranyl Pyrophosphate Synthase (**EC Numbers: 2.5.1.1, 2.5.1.10, 2.5.1.29**): This enzyme catalyzes the synthesis of geranylgeranyl pyrophosphate (GGPP) from farnesyl pyrophosphate and isopentenyl pyrophosphate, initiating the janthitremane biosynthesis pathway. It participates in multiple steps involving prenylation, oxidation, and cyclization, ultimately leading to the formation of complex indole diterpenes such as paspaline and shearinine A. These compounds exhibit diverse bioactivities.

- Siroheme Synthase (**EC Numbers: 1.3.1.76, 2.1.1.107, 4.99.1.4**): This multifunctional enzyme catalyzes a series of key reactions in the siroheme biosynthesis pathway. It first facilitates the S-adenosylmethionine (SAM)-dependent methylation of uroporphyrinogen III at C-2 and C-7, forming precorrin-2 via precorrin-1. It then catalyzes the NAD-dependent ring dehydrogenation of precorrin-2 to produce sirohydrochlorin. Finally, it promotes the ferrochelation of sirohydrochlorin, leading to the formation of siroheme, an essential cofactor in sulfite and nitrite reductases.

- Putative Fatty Acid Oxidation Complex Trifunctional Enzyme (**EC Numbers: 1.1.1.35, 4.2.1.17, 5.3.3.8**): This enzyme exhibits multiple catalytic activities involved in fatty acid metabolism, including 3-hydroxyacyl-CoA dehydrogenase activity, delta(3)-delta(2)-enoyl-CoA isomerase activity, and enoyl-CoA hydratase activity. It also possesses NAD^+-binding capacity and plays a role in fatty acid catabolic processes. This enzyme is essential in the β-oxidation pathway, facilitating the conversion of fatty acids into energy.

- Purine Nucleoside Phosphorylase RC0672 (**EC Numbers: 2.4.2.1, 2.4.2.28, 3.5.4.4**): This purine nucleoside enzyme catalyzes the phosphorolysis of adenosine and inosine nucleosides, producing D-ribose 1-phosphate and the respective free bases, adenine and hypoxanthine. It also catalyzes the phosphorolysis of S-methyl-5'-thioadenosine into adenine and S-methyl-5-thio-α-D-ribose 1-phosphate. Additionally, it exhibits adenosine deaminase activity, contributing to purine metabolism.

G. Visualization of Generated Proteins

In Fig. 9, we present additional visualizations of generated proteins produced by CFP-GEN under multimodal conditions, specifically GO terms and backbone atomic coordinates. The structures are predicted using ESMFold, applied to the generated sequences. The reference PDB/AFDB structures, from which the backbone constraints are derived, are shown in green, while the structures of the designed proteins are shown in red. The real functions of the reference proteins are labeled in red, whereas the predicted functions of the generated proteins, inferred using DeepGO-SE, are labeled in gray.

Several key observations can be drawn from these results. First, most generated proteins exhibit high structural fidelity, as indicated by TM-scores and pLDDT values exceeding 90, demonstrating that CFP-GEN can effectively design structurally stable proteins. Second, the majority of prompted functions are successfully recovered. Since DeepGO-SE also predicts ancestor GO terms, the predicted functions often include more terms than the input prompt. Here, we display only those with a confidence score greater than 0.3, further confirming CFP-GEN's ability to design functionally relevant proteins. Third, some failure cases were also identified. For instance, UniProt ID Q9FYJ2 contains long, flexible loop regions extending far from the core fold. These regions are typically associated with intrinsically disordered regions (IDRs), which lack a single well-defined conformation. As a result, the predicted structure exhibits high deviations in loop regions compared to the ground truth structure. Similarly, UniProt ID Q9PS08 corresponds to an extremely short protein (around 30 residues), which poses significant challenges for both generative and folding models, as such short sequences are underrepresented in large protein datasets. Consequently, these proteins exhibit low TM-scores, highlighting the inherent difficulties in designing ultra-short protein sequences. These findings demonstrate the strengths and limitations of CFP-GEN in functional protein generation, emphasizing its effectiveness in generating proteins with both high-fidelity structures and functional relevance while also highlighting areas for future improvements to enhance model generalization.

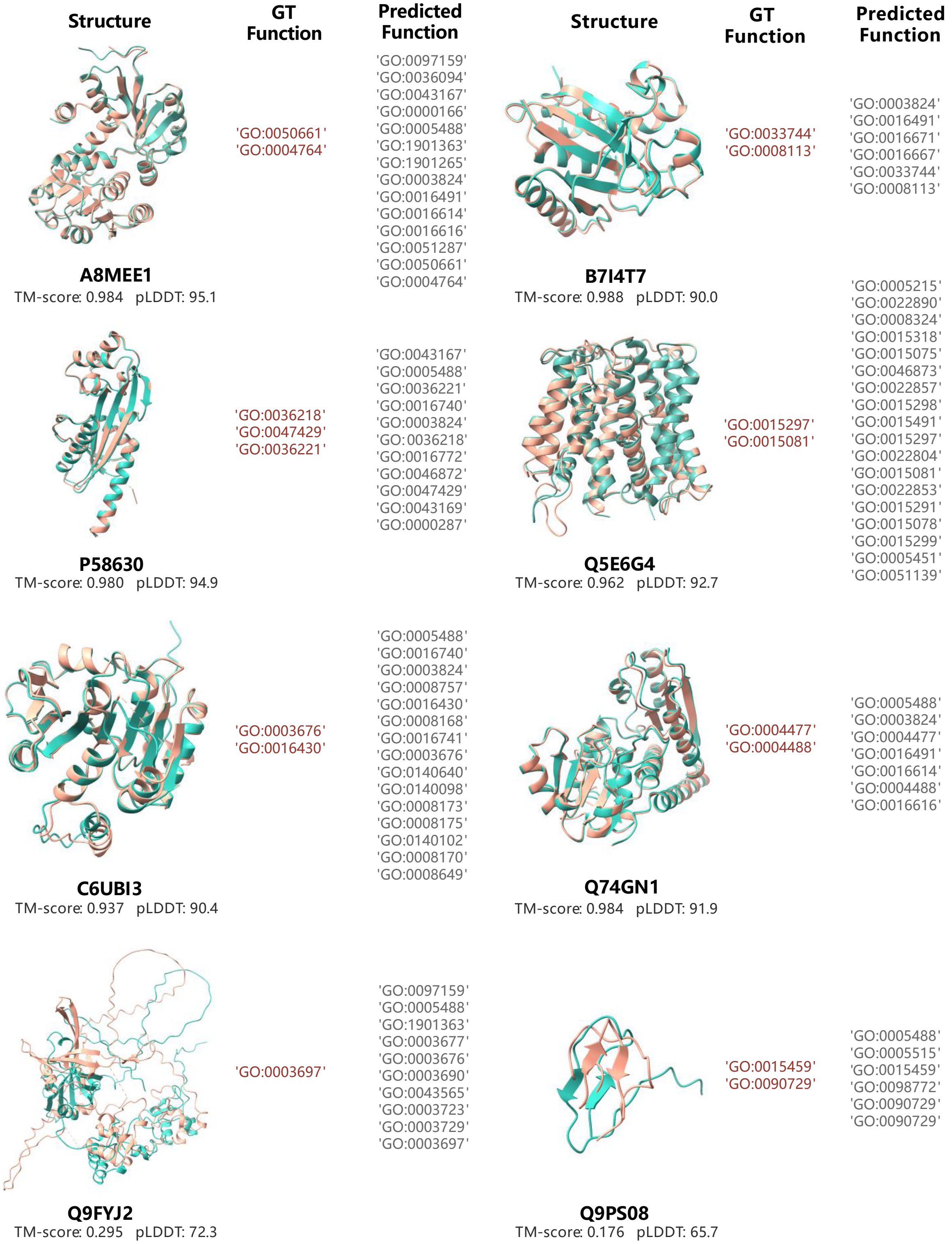


Figure 9. **Examples of generated functional proteins.** We compare the generated proteins and their predicted functions with corresponding PDB/AFDB structures and functions. These results demonstrate the effectiveness and functional consistency of CFP-GEN.