

Efficient Protein Optimization via Structure-aware Hamiltonian Dynamics

Jiahao Wang¹, Shuangjia Zheng^{1,*}

¹Global Institute of Future Technology, Shanghai Jiao Tong University, China
{jiahaowang, shuangjia.zheng}@sjtu.edu.cn

Abstract

The ability to engineer optimized protein variants has transformative potential for biotechnology and medicine. Prior sequence-based optimization methods struggle with the high-dimensional complexities due to the epistasis effect and the disregard for structural constraints. To address this, we propose HADES, a Bayesian optimization method utilizing Hamiltonian dynamics to efficiently sample from a structure-aware approximated posterior. Leveraging momentum and uncertainty in the simulated physical movements, HADES enables rapid transition of proposals toward promising areas. A position discretization procedure is introduced to propose discrete protein sequences from such a continuous state system. The posterior surrogate is powered by a two-stage encoder-decoder framework to determine the structure and function relationships between mutant neighbors, consequently learning a smoothed landscape to sample from. Extensive experiments demonstrate that our method outperforms state-of-the-art baselines in in-silico evaluations across most metrics. Remarkably, our approach offers a unique advantage by leveraging the mutual constraints between protein structure and sequence, facilitating the design of protein sequences with similar structures and optimized properties.

Introduction

Designing proteins with improved fitness is a fundamental but challenging task in protein engineering. The search space of protein variants is vastly expanded as the length of the protein sequence grows, which is typically 20^L for a sequence composed of L amino acids for 20 types of amino acids. Optimizing protein fitness is difficult due to the complex epistasis effect, which creates a rugged, multi-peaked landscape (Poelwijk et al. 2011). Traditional directed evolution approaches rely on random variation combined with screening and selection, without incorporating models to understand the sequence-function relationship (Arnold 1998). This process is costly and time-consuming, limiting the exploration to only a small fraction of possible mutations, even with the advent of reasonably high-throughput techniques.

The challenge of identifying novel protein designs for maximal fitness has driven scientists to adopt machine learning approaches, starting with (Fox et al. 2007) and followed

by many others, increasingly utilizing in physics- and machine learning-based exploration beyond traditional experimental methods (Sinai et al. 2020; Frey et al. 2024; Gruver et al. 2024). Recent advancements in protein language models and structural-informed models also offer new opportunities for modeling protein mutational effects and generalizing fitness knowledge across different datasets (Zheng et al. 2023; Notin et al. 2023; Hie et al. 2024).

The primary research challenge in designing an iterative protein engineering strategy is navigating the vast combinatorial space to uncover the sequence-to-function landscape and find the optimal sequences (Romero and Arnold 2009). This daunting task can be tackled by employing a black-box optimization approach, where the inputs are optimized to get better outputs with limited and often expensive black-box oracle calls. However, protein landscapes are notoriously non-smooth, with fitness levels that can drastically change due to a single mutation, and most mutants have poor fitness. As a result, existing machine learning methods often falter, struggling with noisy fitness landscapes that lead to false positives (Kirjner et al. 2023) and failing to venture beyond local optima due to inefficient exploration (Brookes, Park, and Listgarten 2019).

Compared to sequence-based searching strategies, leveraging structural information offers a more natural approach by providing useful constraints for navigating the noisy fitness landscape, given the decisive role of 3D structure in determining protein function (Branden and Tooze 2012). Therefore, when optimizing a protein for a desired function, designing it in structure latent space seems more direct, allowing the use of gradient-based sampling methods in conjunction with carefully engineered potentials (Trippe et al. 2022; Watson et al. 2023). However, a drawback of this approach is that the optimized structure must be converted back to an amino acid sequence for synthesis (Dau-paras et al. 2022), while there is no guarantee that the optimized structure can be realized by an actual sequence. Additionally, current structural models are computationally intensive and constrained by the scarcity of high-quality structural data, posing challenges to optimization and search.

To address these issues, we propose a new learning-based protein engineering framework that combines Bayesian Optimization and **H**AMiltonian dynamics for protein **D**irected **E**volution in a **S**tructure-informed manner (**HADES**, Fig-

*Corresponding Author

Copyright © 2026, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

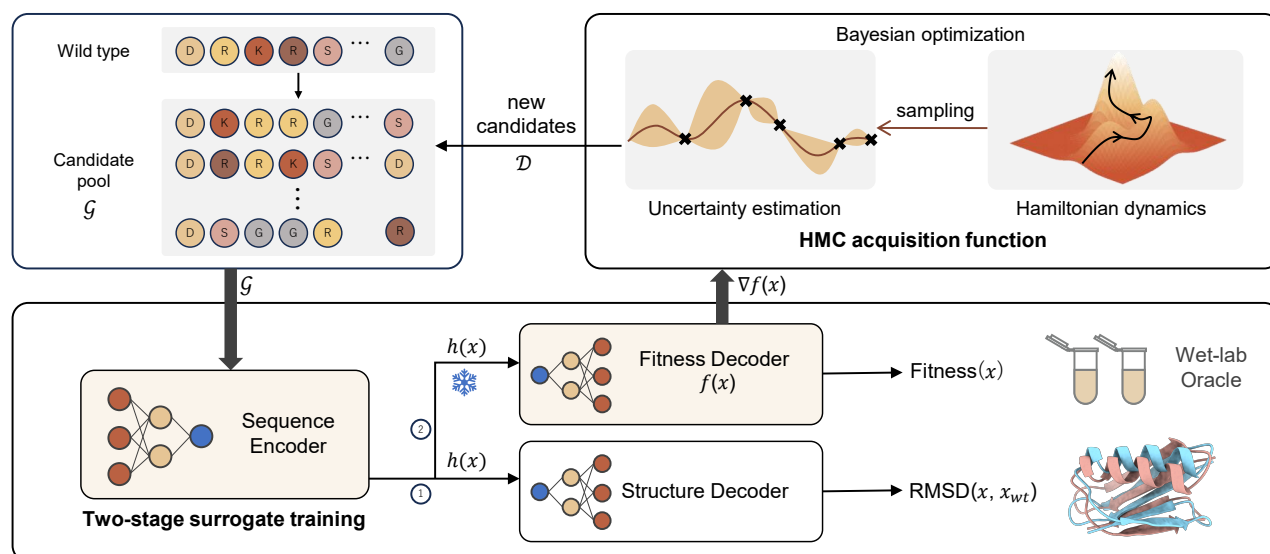


Figure 1: The overall architecture of the proposed HADES framework.

ure 1). Our motivation is based on the mutual constraints between protein sequence and structure. The distribution of conformational perturbations among a series of mutants can serve as prior information to model a smoothed landscape of protein function, potentially enhancing the efficiency of gradient-based algorithms for exploration and sampling. We achieve this by designing a surrogate fitness predictor powered by a two-stage encoder-decoder to learn the structure perturbations distilled from ESMFold (Lin et al. 2023) as the prior. To address the inefficiency in exploring high-dimensional sequence space, we introduce an acquisition function based on Hamiltonian Monte Carlo that proposes distant samples for the Metropolis-Hastings algorithm in MCMC sampling steps by introducing momentum variable in a continuous state space. We further combine the sampling process under Bayesian optimization with estimation of model uncertainty to balance between exploration and exploitation. Our approach offers a significant advantage by leveraging the mutual constraints between protein structure and sequence, facilitating the design of protein sequences with similar structures and enhanced properties.

Related Work

Machine-learning-assisted directed evolution. Machine-learning accelerates the laboratory directed evolution by learning a sequence-to-function mapping and propose promising samples (Yang, Wu, and Arnold 2019). Recent methods are based on paradigm of model-based black-box optimization, where different sequence-function models and exploration algorithms are proposed for better modeling and exploration of the function landscape based on sequence mutations (Hansen and Ostermeier 2001; Brookes, Park, and Listgarten 2019; Jain et al. 2022; Ren et al. 2022; Song and Li 2023; Frey et al. 2024). (Brookes and Listgarten 2018) and (Brookes, Park, and Listgarten 2019) adaptively sample sequences combining unsupervised generative mod-

els and black box predicted models. (Ren et al. 2022) prioritizes the proximal exploration for high-fitness mutants with low mutation counts and design a mutation factorization network to predict low-order mutational effects. (Kirjner et al. 2023) employs a graph-based smoothing to remove noisy gradients in the protein sequence landscape. (Gruver et al. 2024) proposes a diffusion-optimized sampling strategy for controllable categorical diffusion in discrete sequence space. Unlike previous methods, we perform a structure-informed search over continuous sequence space. This allows us to construct a smooth energy surrogate, ensuring strong performance guarantees.

Hamiltonian mechanics and MCMC. Markov Chain Monte Carlo defines a transition kernel to explore target distribution, and Metropolis-Hastings algorithm is a common method to automatically construct appropriate transitions. For high-dimensional spaces, MCMC with Hamiltonian dynamics is an effective way to make large jumps away from the initial point (Neal 2012b; Girolami and Calderhead 2011). A key to its usefulness is that Hamiltonian dynamics preserves volume which can be exactly maintained even when the dynamics is approximated by discretizing time. A special case of implementation which takes a single leapfrog step in HMC is the Langevin Monte Carlo (LMC) algorithm, and the stochastic gradient version, stochastic gradient langevin dynamics, has been widely studied and applied (Welling and Teh 2011; Chen, Fox, and Guestrin 2014; Frey et al. 2024). Compared to other MCMC works, our approach introduces an extra position discretization procedure to propose discrete protein sequence from a continuous state system.

Background

In this study, we address the challenge of machine-learning-guided protein directed evolution within a black-box optimization framework. The process initiates with a single wild

type protein sequence, denoted as x_{wt} , of length L . A protein sequence is represented as a concatenation of one-hot vectors, $x_{wt} = (x_1, x_2, \dots, x_L)$, each vector corresponding to an amino acid. The ground-truth fitness of an unknown protein is assessed through wet-lab evaluations, functioning as the black-box oracle $\mathcal{F}(x)$.

The primary objective of this research is to engineer protein variants exhibiting enhanced fitness scores, particularly under the constraint of limited oracle evaluations. The directed evolution process is conducted iteratively: in each iteration n , where $n \in \{1, 2, \dots, N\}$, a batch of protein variants \mathcal{D}_n , with $|\mathcal{D}_n| = K$, is analyzed by the black-box oracle to determine the true fitness scores $\mathcal{G}_n = \{(x, \mathcal{F}(x)), x \in \mathcal{D}_n\}$. The optimization sequence concludes after N rounds of oracle calls.

Typically, a surrogate fitness predictor $f(x)$ is trained to approximate the black-box oracle, utilizing samples collected through an acquisition function. This function strategically proposes new, promising protein variants for subsequent rounds. Operating under a fixed budget of K oracle queries per round, the acquisition process must effectively balance exploration of the expansive, high-dimensional sequence space with the exploitation of regions potentially close to optima. The surrogate fitness predictor, serving as an approximate posterior, provides fitness predictions that guide the acquisition function in selecting promising variants for further evaluation.

Hamiltonian dynamics and MCMC. Markov Chain Monte Carlo (MCMC) with the Metropolis-Hastings algorithm is a widely-used technique for generating samples from an approximate posterior distribution. For an unknown target distribution $\pi(x)$, proposal distribution is defined by $Q(x'|x)$, and the probability to accept a proposal is $\min(1, \frac{Q(x|x')\pi(x')}{Q(x'|x)\pi(x)})$.

However, in high-dimensional target distributions, the naive proposals in Metropolis-Hastings MCMC (MH-MCMC) often lead to low acceptance rates. Hamiltonian Monte Carlo (HMC), initially developed for molecular simulation and later adapted for use in statistics and neural network models, addresses this issue by leveraging Hamiltonian dynamics to propose new samples (Alder and Wainwright 1959; Neal 2012a).

A Hamiltonian system characterizes the energy of a frictionless particle as the sum of its potential energy $U(q)$ and kinetic energy $K(p)$:

$$H(q, p) = U(q) + K(p)$$

Here, q represents the particle’s position, and p represents its momentum. As the particle moves along a surface of varying height, its momentum allows it to ascend slopes, trading kinetic energy for potential energy.

In practice, Hamiltonian dynamics are approximated via discretization, often using the leapfrog algorithm. HMC has proven to be a highly effective and versatile method for sampling from complex distributions, particularly in high-dimensional spaces, making it a valuable tool for Bayesian inference and other statistical applications (Neal 2012b).

Method

Overall procedures of our method are described in Algorithm 1. We maintain a collection of measured sequence-score pairs \mathcal{G} , and update parameters θ of the surrogate model f using \mathcal{G} at each round. f consists of an ensemble of N models with same architecture and distinct parameters. In each round, a sampling algorithm based on HMC is employed to propose K new variants \mathcal{D}_i for each $f_i \in f$, and select top K variants for black-box evaluation based on the upper confidence bound over \mathcal{D} .

Sampling with Hamiltonian Dynamics

Hamiltonian equation. The Hamiltonian dynamics defines the energy of a particle q by the sum of its potential energy $U(q)$ and kinetic energy $K(p)$, where p is the momentum variable. Here, the q refers to the continuous representation of a protein sequence, initialized with the one-hot representation and then turns into continuous vectors with components constraint between 0 and 1 during the dynamic process. For a sampled q , we take the components as the probabilities of amino acids for each site, thus we can discretize the continuous representation q to the one-hot vector \bar{q} by selecting the amino acids with highest probabilities. The potential energy $U(q)$ serves as the objective function. Since the system seeks to minimize $U(q)$, a lower value of $U(q)$ represents a higher fitness. We define $U(q)$ as the negative log of probability of $f(x)$ and $K(p)$ as the standard kinetic energy:

$$U(q) = -\log P(f(q)), K(p) = \frac{1}{2m} \|p\|^2 \quad (1)$$

where f is the surrogate fitness prediction model and the probability P is defined as the sigmoid function. We set the mass m to be 1 for all components. **Time Step Discretization.** The approximation of Hamiltonian dynamics is achieved by time discretization defined by the leapfrog algorithm. For a given time t , the leapfrog algorithm generate q_{t+1} and p_{t+1} with a half-step update of p followed by a full-step update of q and p , as defined in equation 3 and 5.

$$p_{t+\frac{1}{2}} = p_t - \frac{\epsilon}{2} \nabla U(q_t) \quad (2)$$

$$q_{t+1} = q_t + \epsilon p_{t+\frac{1}{2}} = q_t + \epsilon p_t - \frac{\epsilon^2}{2} \nabla U(q_t) \quad (3)$$

$$p_{t+1} = p_{t+\frac{1}{2}} - \frac{\epsilon}{2} \nabla U(q_{t+1}) \quad (4)$$

$$= p_t - \frac{\epsilon}{2} \nabla U(q_t) - \frac{\epsilon}{2} \nabla U(q_{t+1}) \quad (5)$$

Virtual barriers as position constraints. The gradient update in equation 3 might result in a q_{t+1} with components exceeding the limit of $0 \sim 1$. To handle this constraints, we repeatedly run step 6 and 7 after the leapfrog step of equation 3 until q_{t+1} satisfy the constraint of $0 \leq q_{t+1}(i) \leq 1$. An intuitive illustration is displayed in Figure 2, this procedure establishes two virtual barriers during the movement. Hitting the barrier results in a reversed momentum and new position satisfying dynamics. This “bouncing” movement constrains the location of q within valid zones.

$$p_{t+\frac{1}{2}}(i) = -p_{t+\frac{1}{2}}(i) \quad (6)$$

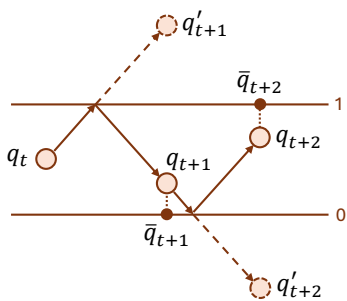


Figure 2: Virtual barrier and position discretization.

$$q_{t+1}(i) = \begin{cases} 2 - q_{t+1}(i), & \text{for } q_{t+1}(i) > 1 \\ -q_{t+1}(i), & \text{for } q_{t+1}(i) < 0 \end{cases} \quad (7)$$

Proposals with Metropolis sampling. Constraints on q provide continuous representation for each amino acid. We take the continuous state as the probability distribution for each site to map back to the one-hot representation. Such approximation inevitably brings error to the discretization. A Metropolis sampling is adopted to reject the samples with large discretization error by discretized representation \bar{q}_{t+1} with probability $\min\{1, \frac{\exp(U(q_t)) \exp(K(p_t))}{\exp(U(\bar{q}_{t+1})) \exp(K(p_{t+1}))}\}$, as described in Algorithm 2.

Candidates selection based on estimated uncertainty. Gaussian Process (GP) is a preferred choice in Bayesian optimization for its natural property to provide uncertainty estimation based on the covariance. In our method, uncertainty is estimated based on deviations of surrogate model ensembles. Specifically, proposed candidates are filtered based on the upper confidence bound defined by the sum of the average value and standard deviation of model predictions. Different from (Ren et al. 2022) which propose samples based on lower confidence bound, samples on upper confidence bound provide refined approximation of high fitness regions with high uncertainty which benefits the Hamiltonian samplers for the exploitation of diverse promising regions.

Two-stage Surrogate Model Learning

The surrogate model aims to approximate the ground-truth fitness landscape within a constrained budget of labeled samples. Given that the model inputs are discrete one-hot vectors, it is essential to model the continuous relationships among mutants derived from discrete amino acid variations to facilitate a gradient-based optimization strategy. In this context, protein structure data is utilized as a prior to model these relationships. The fundamental rationale is to leverage the mutual constraints between protein structure and sequence, facilitating the design of protein sequences with similar structures, thereby avoiding being trapped in local optima induced by epistasis effect. Our surrogate model employs an encoder-decoder architecture, and its training proceeds in two distinct stages. The encoder is utilized in both stages, while two separate decoders are dedicated to learn about protein structure and fitness sequentially.

Sequence encoder. The sequence encoder takes as input a candidate sequence x and applies three self-attention layers

Algorithm 1: Algorithm Overview

- 1: **Input:** candidates set $\mathcal{D} \leftarrow \{x_{wt}\}$
- 2: Initialize θ_i for $f_i \in f$
- 3: Initialize ground-truth protein fitness set: $\mathcal{G} \leftarrow \emptyset$
- 4: $x_{best} = x_{wt}$
- 5: **for** $n = 1$ **to** N **do**
- 6: Query ground truth fitness of x and update \mathcal{G} : $\mathcal{G} \leftarrow \mathcal{G} \cup \{(x, fitness_x), x \in \mathcal{D}\}$
- 7: Update θ_i with \mathcal{G} for $f_i \in f$
- 8: Update x_{best} as the sequence with highest fitness in \mathcal{G}
- 9: **for** f_i **in** f **do**
- 10: $\mathcal{D}_i \leftarrow \{x_{best}\}$
- 11: **while** $|\mathcal{D}_i| < K$ **do**
- 12: $\mathcal{D}_i \leftarrow \mathcal{D}_i \cup \text{HMC}(x_{best}, f_i)$
- 13: **end while**
- 14: $\mathcal{D} \leftarrow \mathcal{D} \cup \mathcal{D}_i$
- 15: **end for**
- 16: $\mathcal{D} \leftarrow \text{Top}_K(\text{UCB}(f, \mathcal{D}))$
- 17: **end for**
- 18: **return** \mathcal{D}

Algorithm 2: HMC(q, f)

- 1: **Input:** seed sequence q_0 , surrogate fitness scorer f
- 2: Initialize $p_0 \in \mathcal{N}(0, \mathbf{I})$
- 3: $\mathcal{D} \leftarrow \emptyset$
- 4: **for** $t = 0$ **to** T **do**
- 5: Update p_{t+1} and q_{t+1} with formula 3 and 5
- 6: Discretize $\bar{q}_{t+1} \leftarrow \text{one-hot}(\text{argmax}(q_{t+1}))$
- 7: Accept \bar{q}_{t+1} with probability: $\min\{1, \frac{\exp(U(q_t)) \exp(K(p_t))}{\exp(U(\bar{q}_{t+1})) \exp(K(p_{t+1}))}\}$
- 8: $\mathcal{D} \leftarrow \mathcal{D} \cup \bar{q}_{t+1}$
- 9: **end for**
- 10: **return** \mathcal{D}

to capture intra- and inter-interactions of amino acid embeddings and produces a latent vector representation, $h(x)$, for each sequence (Vaswani et al. 2017).

Structure decoder. This first decoder is designed to learn the Root Mean Square Deviation (RMSD) score between a sequence candidate and the wild type across mutation sites. We employ ESMFold (Lin et al. 2023) to predict the structures and compute the conformation perturbations through RMSD scores. Notably, while protein structure prediction accounts for the dominant computational overhead, it is highly parallelizable in practice. During the training phase, the encoder captures the latent representations that reflect the structural distance from input protein to the wild type. Our assumption is that understanding this distribution of structural distances provides prior knowledge of how mutations reflect the structural changes, which benefits the gradient calculation and fitness learning.

Fitness decoder. In the second stage, the fitness decoder focuses on learning fitness scores, operating with the frozen encoder parameters obtained from the first stage. This separation ensures that the fitness assessment is based on sta-

Task	Metric	ESM2-zs	BO	CMA-ES	AdaLead	PEX	EvoPlay	HADES-L	HADES
GB1	max fit.	1.00	0.57 ± 0.15	0.69 ± 0.16	0.84 ± 0.15	0.83 ± 0.17	0.91 ± 0.09	0.93 ± 0.14	1.00 ± 0.00
	mean fit.	0.03	0.08 ± 0.01	0.28 ± 0.07	0.49 ± 0.05	0.51 ± 0.06	0.54 ± 0.03	0.51 ± 0.06	0.59 ± 0.02
	fDiv	0.04	0.14 ± 0.02	0.37 ± 0.08	0.64 ± 0.09	0.68 ± 0.14	0.77 ± 0.04	0.70 ± 0.12	0.84 ± 0.03
PhoQ	max fit.	0.32	0.27 ± 0.05	0.47 ± 0.23	0.62 ± 0.21	0.46 ± 0.05	0.69 ± 0.22	0.72 ± 0.24	0.80 ± 0.25
	mean fit.	0.01	0.05 ± 0.01	0.13 ± 0.03	0.19 ± 0.03	0.20 ± 0.01	0.18 ± 0.02	0.20 ± 0.02	0.22 ± 0.02
	fDiv	0.02	0.08 ± 0.01	0.18 ± 0.03	0.26 ± 0.04	0.30 ± 0.01	0.26 ± 0.03	0.28 ± 0.03	0.32 ± 0.02

Table 1: Comparing different models for protein engineering task on GB1 and PhoQ datasets. Cumulative maximum fitness, mean fitness and fDiv scores are presented.

ble, well-defined structural representations derived in the first training stage. Both decoders stack three 1-D convolution layers followed by max pooling and a two multi-layer perceptrons. We did not use complex architectures because these scenarios involve low-resource training, where intricate networks and representations can easily lead to model overfitting.

Experiments

Experimental Setup

Datasets. We evaluate our method on two widely recognized combinatorial datasets, GB1 (Wu et al. 2016) and PhoQ (Podgornaia and Laub 2015) following (Wang et al. 2023). Both datasets feature multi-site saturation mutagenesis, each containing approximately $20^4 = 160,000$ samples, specifically covering 149,361 and 140,517 sequences, respectively. Unlike the majority of existing protein fitness datasets, which predominantly include mutants characterized by only one or two mutations, these two datasets avoid relying on inaccurate oracle models for scoring, providing a rigorous testing ground for machine learning-guided directed evolution methodologies.

Baselines. We compare our method against five baseline approaches: **ESM-zs** is the zero-shot prediction of ESM2 model with 650M parameters (Lin et al. 2023). **BO** refers to Bayesian Optimization with Thompson sampling based on expected improvement. **CMA-ES**, a classical evolutionary search algorithm, estimates the covariance matrix to adaptively update mutation distributions (Hansen and Ostermeier 2001). **AdaLead** adopts a straightforward yet robust approach, employing a hill-climbing style to greedily select the top-predicted candidates (Sinai et al. 2020). **PEX** explores the nearby region around wild type by proximal frontier to trade-off between fitness scores and mutation distances (Ren et al. 2022). **EvoPlay** utilizes a self-play framework within a reinforcement-learning setting, employing Monte Carlo tree search to guide decision-making (Wang et al. 2023). Additionally, we provide the Langevin dynamics version of our methods (HADES-L) for comparison.

Metrics. We evaluate our methods and baseline models by three metrics: cumulative maximum fitness, average fitness and fitness-conditioned diversity score (fDiv). The maximum and average fitness scores are calculated over $N = 10$ query rounds, with $K = 100$ new queries in each round. The fDiv score is calculated as follows:

$$\text{fDiv}(\mathcal{D}) = \frac{\sum_{(x_i, y_i) \in \mathcal{D}} \sum_{(x_j, y_j) \in \mathcal{D} \setminus (x_i, y_i)} d(x_i, x_j) \cdot (\mathcal{F}(x_i) + \mathcal{F}(x_j))}{2 \cdot |\mathcal{D}| \cdot (|\mathcal{D}| - 1)} \quad (8)$$

where \mathcal{D} denotes the top K samples after N query rounds, and $d(x_i, x_j)$ calculates the edit distance of sequence x_i and x_j . The fDiv score follows definition of diversity score from previous studies with a modification to combine the average fitness score of any two different samples, alleviating the dilemma where poor fitness designs report a significantly high diversity score. All results are averaged over 10 runs with different random states.

Settings. For surrogate modeling training, we use Adam optimizer with 0.001 learning rate and mean square error loss, the training stops when the training loss does not decrease for 3 epochs. The hidden vector dimension for all neural network modules is 256, and the kernel size of CNN is 3. For the structure learning module, the RMSD labels define the root mean square deviation on backbone atoms covering C, C α , N after aligning the positions of the four mutation sites. We run 128 parallel samples during sampling, the trajectory length T of Hamiltonian dynamics is 16, and the leapfrog step size ϵ is 0.1 for both tasks. For fair comparison, we evaluate for the best choices of T and ϵ on GB1 task and apply on PhoQ task with the same configuration. Following previous works, the first round of sampling is based on the surrogate learning of K mutants with random mutations from wild-type (Ren et al. 2022; Wang et al. 2023). The baselines are evaluated on a single NVIDIA RTX 4090 GPU with default hyperparameters. We run our method on a single NVIDIA A40 GPU for compatibility with running ESMFold.

Results and Analysis

Overall performance comparison. Table 1 provides a comprehensive evaluation of the performance of our method in comparison to baselines. Bold texts represents best results and the second-best results are marked with underlines. Our method demonstrates superior effectiveness across both tasks on three metrics. Specifically, in the GB1 task, our approach consistently excelled, with all ten experimental runs successfully identifying the optimal protein sequence, providing a full score with zero standard deviation. For the more difficult PhoQ task, our method also significantly outperforms baselines on maximum fitness scores. This is in-

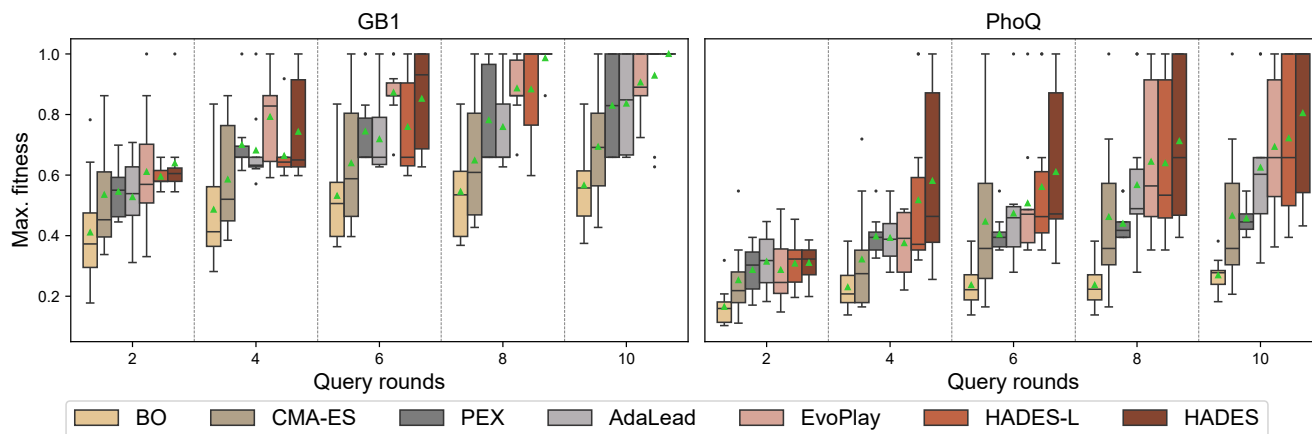


Figure 3: Cumulative maximum fitness scores by experiment rounds on two protein engineering benchmark tasks. Green triangles represent the mean values.

Method	max fit.	mean fit.	fDiv
HADES	1.00±0.00	0.59±0.02	0.84±0.03
w/o HD	0.84±0.15	0.52±0.05	0.73±0.12
w/o structure	0.95±0.10	0.57±0.04	0.84±0.05
w/o ucb	0.93±0.11	0.51±0.04	0.75±0.06
w/o vb.	0.97±0.09	0.56±0.04	0.79±0.05

Table 2: Ablation studies of GB1.

Method	max fit.	mean fit.	fDiv
HADES	0.80±0.25	0.22±0.02	0.32±0.02
w/o HD	0.75±0.25	0.21±0.01	0.32±0.02
w/o structure	0.74±0.27	0.21±0.02	0.31±0.02
w/o ucb	0.76±0.24	0.19±0.02	0.30±0.02
w/o vb.	0.63±0.25	0.20±0.02	0.30±0.03

Table 3: Ablation studies of PhoQ.

dicative of our method’s robustness and its effectiveness in navigating more complex problem spaces. Moreover, the best functional diversity (fDiv) scores obtained in both tasks highlight another critical advantage of our method that it not only discovers high-fitness variants, but also maintains a diverse set of solutions.

To further investigate the optimization curve, we conducted a detailed examination of the optimization efficiency by analyzing the results across various query rounds, as depicted in Figure 3. Our method provides stable performance across different experiment rounds, persistently outperforms baseline methods on both datasets. It is important to note the observation of high variances in the results for the PhoQ task. This variability can be primarily attributed to the inherent sparsity of high-fitness samples within this data set. Nevertheless, our method’s ability to maintain competitive results highlights its advanced capability to navigate and optimize within intricate and sparse fitness landscapes.

Ablation experiments. To verify our proposed framework, we conduct a series of ablation experiments, as summarized in Table 2 and Table 3. We first replace the HMC-based sampling module (HD) with proposals based on random mutations. This change led to a marked performance decline in the GB1 task and a moderate decline in the PhoQ task, verifying the effectiveness of our Hamiltonian acquisition strategy. The removal of the structure decoder module is documented in the third row of results, the relatively modest impact on performance could possibly due to generaliza-

tion error of ESMFold. The fourth row of results involves with omitting the uncertainty estimation process and relying solely on predictions from a single model, this adjustment resulted in uniformly lower scores across all evaluated metrics, confirming the hypothesis that incorporating uncertainty and model ensembles enhances prediction reliability and optimizes performance. The last row of results denotes the modification of removing the virtual barrier mechanism (vb.) and directly applying discretization. This also led to a reduction in the cumulative maximum fitness score, particularly evident in the PhoQ task’s steep landscape. The results demonstrate that the virtual barrier effectively mitigates discretization errors associated with large gradient updates, confirming its utility in enhancing model robustness.

Effect of query size and query round. In practical protein engineering scenarios, the costs associated with wet-lab evaluations can vary significantly depending on the optimization targets. A robust algorithm, therefore, must maintain stable performance across a range of different evaluation costs. To explore how varying the oracle query size affects algorithm performance, we conducted experiments with varying query sizes $K = \{16, 32, 64, 128\}$ under $N = 10$ rounds of experiments, as shown in Figure 4. Our results indicate that all evaluated methods show improvement as the query size increases. Notably, the performance gap between our method and other methods widens with increasing query size. This observation highlights our method’s superior scaling effect, which is particularly pronounced at larger query

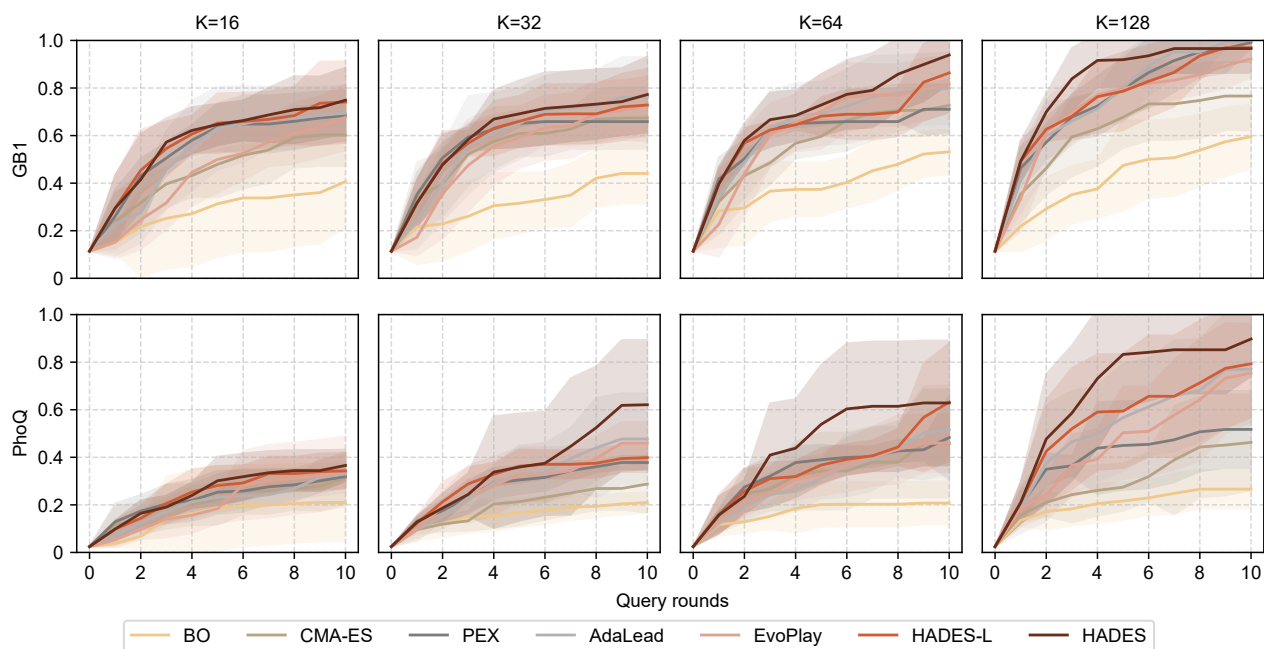


Figure 4: Cumulative maximum fitness scores for $K=\{16, 32, 64, 128\}$ on two protein engineering benchmark tasks. All curves are derived from 10 runs with random network initialization. The shaded regions represent the standard deviation.

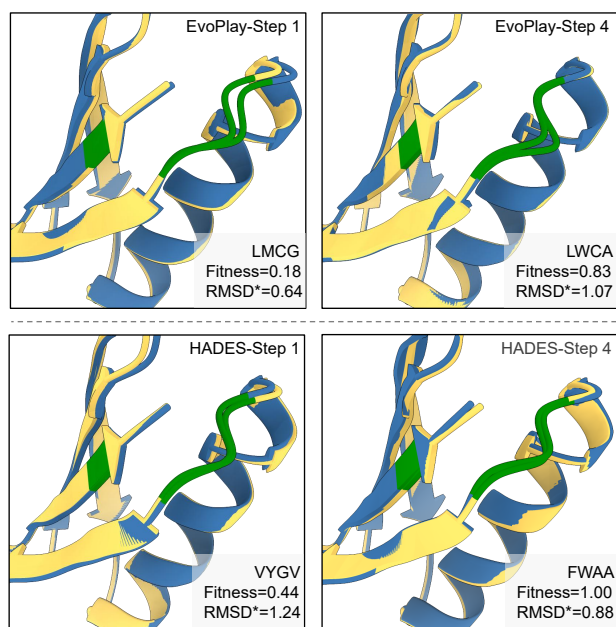


Figure 5: High-fitness structures sampled from EvoPlay and HADES on GB1 task. The RMSD^* score (the lower the better) calculates the root mean square deviation over the 4 mutation sites.

sizes. The increased performance gap observed at larger query sizes indicates that our method not only adapts well to increased information availability but also leverages this information more effectively than baselines.

Analysis of structures. To further investigate the effectiveness of our structure learning module. We visualize several wild-type aligned structures with maximum fitness sampled at different steps of EvoPlay and our method on GB1 task, as displayed in Figure 5. Proteins colored yellow denote the wild-type, and proteins colored blue are the model designs. Green regions represent the four mutation sites. From the structure variances in mutation sites, we observe that samples from our method exhibit minor structural changes in mutation region with improved fitness, proving the effectiveness of structure-informed surrogate module.

Conclusion and Future Work

We propose a protein engineering framework that combines an acquisition function based on Hamiltonian dynamics and a structure-aware surrogate model for protein directed evolution. In-silico experiments verify that our method efficiently discovers diverse protein variants with high fitness levels, achieving improved efficiency with limited wet-lab costs and experiment rounds. Future efforts will focus on enhancing the modeling of protein structures to improve generalization ability and designing proteins that optimize multiple objectives. The code and data are publicly available at <https://github.com/plus-w/HADES>.

Acknowledgments

This study has been supported by the National Natural Science Foundation of China [62041209], Natural Science Foundation of Shanghai [24ZR1440600], the Science and Technology Commission of Shanghai Municipality [24510714300], the Lingang Lab Fund [LGL-8888].

References

- Alder, B. J.; and Wainwright, T. E. 1959. Studies in molecular dynamics. I. General method. *The Journal of Chemical Physics*, 31(2): 459–466.
- Arnold, F. H. 1998. Design by Directed Evolution. *Accounts of Chemical Research*, 31(3): 125–131.
- Branden, C. I.; and Tooze, J. 2012. *Introduction to protein structure*. Garland Science.
- Brookes, D.; Park, H.; and Listgarten, J. 2019. Conditioning by adaptive sampling for robust design. In *International conference on machine learning*, 773–782. PMLR.
- Brookes, D. H.; and Listgarten, J. 2018. Design by adaptive sampling. *arXiv preprint arXiv:1810.03714*.
- Chen, T.; Fox, E.; and Guestrin, C. 2014. Stochastic gradient hamiltonian monte carlo. In *International conference on machine learning*, 1683–1691. PMLR.
- 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. 2022. Robust deep learning-based protein sequence design using ProteinMPNN. *Science*, 378(6615): 49–56.
- Fox, R. J.; Davis, S. C.; Mundorff, E. C.; Newman, L. M.; Gavrilovic, V.; Ma, S. K.; Chung, L. M.; Ching, C.; Tam, S.; Muley, S.; et al. 2007. Improving catalytic function by ProSAR-driven enzyme evolution. *Nature biotechnology*, 25(3): 338–344.
- Frey, N. C.; Berenberg, D.; Zadorozhny, K.; Kleinhenz, J.; Lafrance-Vanasse, J.; Hotzel, I.; Wu, Y.; Ra, S.; Bonneau, R.; Cho, K.; Loukas, A.; Gligorijevic, V.; and Saremi, S. 2024. Protein Discovery with Discrete Walk-Jump Sampling. In *The Twelfth International Conference on Learning Representations*.
- Girolami, M.; and Calderhead, B. 2011. Riemann manifold langevin and hamiltonian monte carlo methods. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 73(2): 123–214.
- Gruver, N.; Stanton, S.; Frey, N.; Rudner, T. G.; Hotzel, I.; Lafrance-Vanasse, J.; Rajpal, A.; Cho, K.; and Wilson, A. G. 2024. Protein design with guided discrete diffusion. *Advances in Neural Information Processing Systems*, 36.
- Hansen, N.; and Ostermeier, A. 2001. Completely derandomized self-adaptation in evolution strategies. *Evolutionary computation*, 9(2): 159–195.
- Hie, B. L.; Shanker, V. R.; Xu, D.; Bruun, T. U.; Weidenbacher, P. A.; Tang, S.; Wu, W.; Pak, J. E.; and Kim, P. S. 2024. Efficient evolution of human antibodies from general protein language models. *Nature Biotechnology*, 42(2): 275–283.
- Jain, M.; Bengio, E.; Hernandez-Garcia, A.; Rector-Brooks, J.; Dossou, B. F.; Ekbote, C. A.; Fu, J.; Zhang, T.; Kilgour, M.; Zhang, D.; et al. 2022. Biological sequence design with gflownets. In *International Conference on Machine Learning*, 9786–9801. PMLR.
- Kirjner, A.; Yim, J.; Samusevich, R.; Bracha, S.; Jaakkola, T. S.; Barzilay, R.; and Fiete, I. R. 2023. Improving protein optimization with smoothed fitness landscapes. In *The Twelfth International Conference on Learning Representations*.
- Lin, Z.; Akin, H.; Rao, R.; Hie, B.; Zhu, Z.; Lu, W.; Smetanin, N.; Verkuil, R.; Kabeli, O.; Shmueli, Y.; dos Santos Costa, A.; Fazel-Zarandi, M.; Sercu, T.; Candido, S.; and Rives, A. 2023. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637): 1123–1130.
- Neal, R. M. 2012a. *Bayesian learning for neural networks*, volume 118. Springer Science & Business Media.
- Neal, R. M. 2012b. MCMC using Hamiltonian dynamics. arXiv:1206.1901.
- Notin, P.; Kollasch, A. W.; Ritter, D.; van Niekerk, L.; Paul, S.; Spinner, H.; Rollins, N.; Shaw, A.; Weitzman, R.; Frazer, J.; et al. 2023. Proteingym: Large-scale benchmarks for protein design and fitness prediction. *bioRxiv*, 2023–12.
- Podgornaia, A. I.; and Laub, M. T. 2015. Pervasive degeneracy and epistasis in a protein-protein interface. *Science*, 347(6222): 673–677.
- Poelwijk, F. J.; Tănase-Nicola, S.; Kiviet, D. J.; and Tans, S. J. 2011. Reciprocal sign epistasis is a necessary condition for multi-peaked fitness landscapes. *Journal of Theoretical Biology*, 272(1): 141–144.
- Ren, Z.; Li, J.; Ding, F.; Zhou, Y.; Ma, J.; and Peng, J. 2022. Proximal Exploration for Model-guided Protein Sequence Design. In *Proceedings of the 39th International Conference on Machine Learning*, 18520–18536. PMLR.
- Romero, P. A.; and Arnold, F. H. 2009. Exploring protein fitness landscapes by directed evolution. *Nature reviews Molecular cell biology*, 10(12): 866–876.
- Sinai, S.; Wang, R.; Whatley, A.; Slocum, S.; Locane, E.; and Kelsic, E. D. 2020. AdaLead: A simple and robust adaptive greedy search algorithm for sequence design. *arXiv preprint arXiv:2010.02141*.
- Song, Z.; and Li, L. 2023. Importance weighted expectation-maximization for protein sequence design. In *International Conference on Machine Learning*, 32349–32364. PMLR.
- Trippe, B. L.; Yim, J.; Tischer, D.; Baker, D.; Broderick, T.; Barzilay, R.; and Jaakkola, T. 2022. Diffusion probabilistic modeling of protein backbones in 3d for the motif-scaffolding problem. *arXiv preprint arXiv:2206.04119*.
- Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; and Polosukhin, I. 2017. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems, NIPS’17*, 6000–6010. Curran Associates Inc.
- Wang, Y.; Tang, H.; Huang, L.; Pan, L.; Yang, L.; Yang, H.; Mu, F.; and Yang, M. 2023. Self-play reinforcement learning

guides protein engineering. *Nature Machine Intelligence*, 5(8): 845–860.

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

Welling, M.; and Teh, Y. W. 2011. Bayesian learning via stochastic gradient Langevin dynamics. In *International Conference on Machine Learning*, 681–688.

Wu, N. C.; Dai, L.; Olson, C. A.; Lloyd-Smith, J. O.; and Sun, R. 2016. Adaptation in protein fitness landscapes is facilitated by indirect paths. *Elife*, 5: e16965.

Yang, K. K.; Wu, Z.; and Arnold, F. H. 2019. Machine-learning-guided directed evolution for protein engineering. *Nature Methods*, 16(8): 687–694.

Zheng, Z.; Deng, Y.; Xue, D.; Zhou, Y.; Ye, F.; and Gu, Q. 2023. Structure-informed language models are protein designers. In *International Conference on Machine Learning*, 42317–42338. PMLR.