Steering Protein Generative Models at Test-Time for Guided AAV2 Capsid Design

Ben Viggiano[†1], Wenhui Sophia Lu[†2], Xiaowei Zhang[†3,4], Luis S. Mille-Fragoso[3,4,5], Xiaojing J. Gao[5,8], Euan Ashley[1,6,7], Wing Hung Wong[1,2,5]

[1]*Department of Biomedical Data Science, Stanford University, Stanford, CA, USA*
[2]*Department of Statistics, Stanford University, Stanford, CA, USA*
[3]*Department of Bioengineering, Stanford University, Stanford, CA, USA*
[4] *Sarafan ChEM-H, Stanford University, Stanford, CA, USA*
[5]*Stanford Bio-X, Stanford University, Stanford, CA, USA*
[6]*Department of Medicine, Stanford University, Stanford, CA, USA*
[7]*Center for Undiagnosed Diseases, Stanford University, Stanford, CA, USA*
[8]*Department of Chemical Engineering, Stanford University, Stanford, CA, USA*
[†] *Equal Contribution*

Recent advances in protein generative models have created new opportunities for protein engineering. However, a significant challenge remains in effectively steering these models to generate sequences with specific, desired functionalities, especially when these properties are defined by "black-box" or non-differentiable fitness functions. To address this, we present ProVADA+, a model-agnostic framework that guides pretrained generative models at test-time without costly retraining. Our approach introduces a reinforcement learning-based adaptive masking technique (MADA-DUCB) that significantly accelerates convergence. We demonstrate this framework on the challenging task of designing novel Adeno-Associated Virus 2 (AAV2) capsids. By coupling a ProteinMPNN generative prior with a fine-tuned AAV viability oracle, our method successfully navigates the rugged fitness landscape where unguided random mutagenesis is ineffective—with prior experiments showing as few as 0.3% of variants with six or more mutations are viable. In its final iterations, ProVADA generated a pool of novel candidates with a mean viral selection score of 2.72, consistently scoring highly viable variants while maintaining a diverse range of sequence similarity to the wild-type sequence. Our results show that ProVADA provides a powerful and efficient framework for accelerating the design of proteins with complex, user-defined properties.

Keywords: Protein Engineering, Conditional Adaptation, Generative Guidance, Test-time Steering, Adeno-Associated Virus

1. Introduction

Protein engineering is the process of developing useful or novel proteins by modifying the amino acid sequences to improve existing structural and functional properties or confer new activities. Traditionally, one of the primary techniques available to researchers is directed evolution, an iterative laboratory process in which large libraries of protein variants are generated—typically through random mutagenesis of wild-type sequences—and then screened

using functional assays.[1] While this approach has demonstrated success in many applications,[2] it remains a resource-intensive and time-consuming process, particularly when dealing with intricate sequence-fitness landscapes or when mechanistic insights into protein function are limited.[3]

In the past several years, advances in machine learning have led to the development of powerful new approaches that demonstrate significant potential to accelerate protein engineering methodology. Structure prediction and downstream inverse-folding models have enabled a new way to condition sequence generation that preserves target structural folds in the final protein.[4–6] Large-scale protein language models (pLMs) trained on billions of biological sequences using simple pretraining objectives enable state-of-the-art performance on protein property prediction tasks.[7,8] Together, these tools accelerate candidate prioritization and substantially reduce the need for exhaustive experimental screening.

Researchers are increasingly looking to use a combination of these models to conditionally guide the generative process toward variants with desired functional properties. Most techniques revolve around classifier-guided plug-and-play approaches to steer sequence generation by backpropagating through a differentiable surrogate objective.[9–11] During generation, gradients from this reward bias a pretrained generative model toward specific properties. However, these methods depend on differentiable surrogate models and are thus incompatible with non-differentiable scoring functions. In contrast, fine-tuning and preference-learning approaches, such as classifier-free guidance[12] or reinforcement-learning-based fine-tuning,[13] adapt the model parameters directly to optimize for downstream rewards. While often effective, these techniques typically demand significant computational resources and large labeled datasets.

To address these limitations, we recently introduced ProVADA, a model-agnostic framework that steers any generative model using black-box fitness oracles without requiring costly retraining or gradients.[14] At the core of this framework is Mixture-Adaptation Directed Annealing (MADA), a novel sampling algorithm designed to efficiently navigate high-dimensional sequence landscapes. In this work, we propose several key enhancements to the ProVADA framework (ProVADA+) that substantially improve its biological relevance and algorithmic efficiency. Our primary contribution is an informed and adaptive mutation site selection strategy that employs reinforcement learning (MADA-DUCB) to accelerate convergence by intelligently identifying optimal mutation positions (Figure 1). Additionally, we introduce sequence constraints, including the ability to preserve specific residue positions crucial for protein function, and incorporate a biologically-informed distance metric based on the BLOSUM62 substitution matrix. This metric penalizes biochemically unfavorable mutations while permitting conservative substitutions, thereby generating variants with enhanced biological plausibility.

We demonstrate the power of this enhanced framework by applying it to a challenging protein design task: conditionally engineering Adeno-Associated Virus 2 (AAV2) capsids for improved viability. Our *in silico* results demonstrate that ProVADA+ effectively navigates the rugged fitness landscape to generate a diverse pool of novel candidates with high predicted viral selection scores, substantially outperforming unguided *in silico* selection from random mutagenesis approaches which fail to efficiently identify viable variants.

1.1. *AAV Capsid Design*

AAVs are small, non-pathogenic, single-stranded DNA viruses that rely on co-infection with helper viruses, typically adenoviruses, for replication.[15] Due to their favorable biological properties, engineered AAV capsids have become a leading platform for next-generation *in vivo* gene therapy vector development.[16] For example, the 13 commonly studied AAV serotypes exhibit distinct tissue tropisms, enabling selective transduction of specific cell types. This cell-type specificity is a major objective in vector engineering, as it can significantly reduce off-target effects from delivered therapeutic payloads.[16]

Within the full 60-mer capsid protein complex, sequence and structural variations found in small surfaced exposed *hyper-variable* regions have been shown to have out-sized impact on capsid functional properties.[17,18] For these reasons, AAV genetic modification in vector engineering research has commonly focused on the screening and directed evolution of these regions.[19,20] In this work, we explore utilizing ProVADA+ to engineer novel AAV capsid proteins for *in silico* predicted capsid viability, which characterizes whether a particular capsid is able to form. Moving forward, such predictive sampling could be combined with high-throughput experimental screens to guide selection for desirable functional traits beyond viability, such as transduction efficiency, low immunogenicity, and tissue tropism.

2. Methods

2.1. *ProVADA: Gradient-Free Guidance for Conditional Protein Adaptation at Test-Time*

We begin with a brief overview of the ProVADA framework for gradient-free steering of protein generation at test time. ProVADA comprises two core components: a fitness oracle to provide targeted guidance, and the Mixture-Adaptation Directed Annealing (MADA) sampler to efficiently navigate the intricate topology of sequence-fitness landscapes. To formalize our framework, we first define the key notations and definitions for the problem setup.

Let $\ell \in \mathbb{N}$ be the fixed sequence length, which defines the discrete sequence space $\mathcal{X} = \{1, \ldots, 20\}^\ell$. Our objective is to engineer a given wild-type reference sequence, $x_{wt} \in \mathcal{X}$, to improve its functional properties, which are quantified by a fitness oracle $F : \mathcal{X} \to [0, 1]$, where higher values indicate superior fitness. Additionally, we require a generative model capable of proposing mutations through efficient in-painting of masked sequence regions, thereby producing complete sequences that preserve fundamental properties such as structural integrity. The implicit sampling distribution of this generative model with parameters ϕ—whether autoregressive, diffusion-based, or alternative architectures—is denoted p_ϕ and henceforth referred to as the generative prior. Let $\mathbf{C}$ be an amino acid substitution matrix, where $\mathbf{C}(x_i, x_j)$ represents the score of substituting amino acid x_i with amino acid x_j.

Biochemically-Informed Sequence Divergence Penalty

To maintain proximity to the reference sequence and thus facilitate *conditional engineering* rather than *de novo* generation, we employ a composite fitness objective. This objective penalizes deviations from the wild-type sequence, x_{wt}, using a biologically-informed distance metric

$d(x, x_{wt})$ that respects amino acid properties while penalizing excessive substitutions. This metric is consisted of two terms: First, a *Hamming distance* (d_H), which penalizes the total number of mutations regardless of their biochemical properties. Second, we also include a *substitution cost* (d_C), which employs a penalty based on the BLOSUM62 substitution matrix.[21] Widely utilized in sequence alignment tools such as BLAST,[22] BLOSUM62 provides log-odds scores that reflect the evolutionary likelihood of amino acid substitutions. Positive scores indicate conservative substitutions that frequently occur in nature, while negative scores suggest less favorable, potentially deleterious changes. Under this framework, substitutions with positive BLOSUM62 scores incur no penalty, yielding a more informative and better evolutionarily aligned metric than the Hamming distance alone.

Ensemble-guided Composite Fitness Objective

The composite objective function, $H_\lambda(x)$, balances fitness enhancement with sequence conservation through tunable coefficients ($\lambda = (\lambda_H, \lambda_C)$):

$$d_H(x, x_{wt}) = 1/\ell \sum_{i=1}^{\ell} \mathbb{1}_{\{x_i \neq (x_{wt})_i\}}, \quad d_C(x, x_{wt}) = \sum_{i: x_i \neq (x_{wt})_i} \mathbf{C}(x_i, (x_{wt})_i) \in [0, 1],$$

$$H_\lambda(x) = F(x) - \big(\underbrace{\lambda_H d_H(x, x_{wt})}_{\substack{\text{Hamming} \\ \text{penalty}}} + \underbrace{\lambda_C d_C(x, x_{wt})}_{\substack{\text{Substitution} \\ \text{cost}}} \big),$$

where $\lambda_H, \lambda_C \geq 0$ are tunable regularization parameters that control the strength of the divergence penalties. To this end, we construct an ensemble target distribution proportional to the generative prior exponentially tilted by the tempered composite fitness $\pi_{\phi,H,\tau}(x) \propto p_\phi(x) \exp(H_\lambda(x)/\tau)$, where τ is a temperature parameter that governs how sharply sampling concentrates on high-fitness regions. As τ decreases, the sampler becomes increasingly concentrated on top-scoring sequences, whereas higher τ values encourage broader exploration of the sequence space.

2.2. *Constructing the Fitness Oracle*

ProVADA employs a user-defined black-box fitness oracle to guide its generative process. This oracle, trained on experimentally characterized protein sequences, predicts a numerical score quantifying the desired properties for each input sequence. The implementation of such an oracle is flexible, ranging from simple biophysical calculators to complex machine learning architectures. A particularly effective approach for creating high-performance oracles involves fine-tuning large, pre-trained protein language models (pLMs). Typically, this is achieved by appending a regression head, such as a multi-layer perceptron (MLP), to the pLM's final layer. The resulting model is then trained on task-specific datasets to accurately predict properties of interest from the model's sequence embeddings. In our experiments, we construct an oracle to predict AAV viability based on the ESM-Cambrian model; we provide more details in Section 3.2.

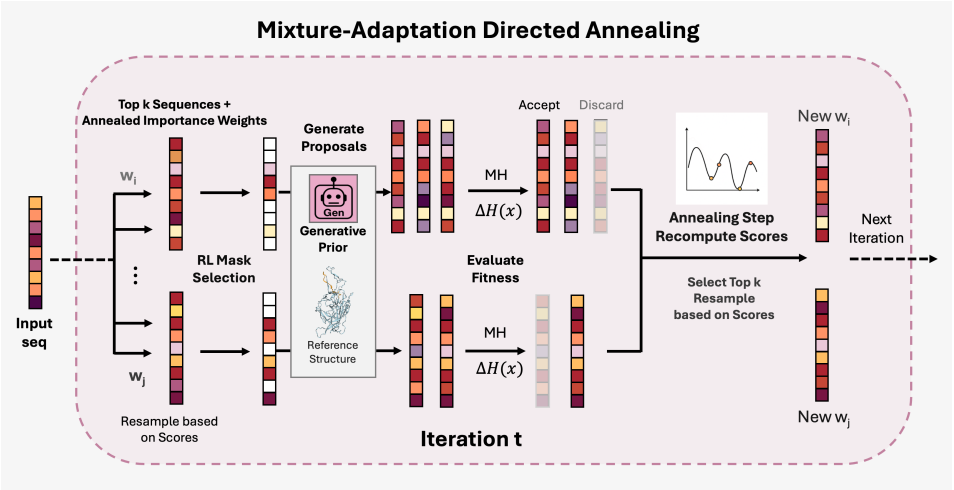

Fig. 1. Schematic overview of the *Mixture-Adaptation Directed Annealing* (MADA) sampling workflow. This iterative process integrates Sequential Monte Carlo population dynamics with adaptive simulated annealing to efficiently explore the protein sequence-fitness landscape.

2.3. *Mixture-Adaptation Directed Annealing with Informed Position Masking (MADA-DUCB)*

To simultaneously navigate the complex fitness landscapes and high-dimensional protein sequence space, ProVADA employs Mixture-Adaptation Directed Annealing (MADA). This novel sampling algorithm integrates Sequential Monte Carlo population dynamics with adaptive simulated annealing, iterating through cycles of *selection*, *mutation*, and *stabilization*. At each iteration, MADA maintains a diverse ensemble of promising candidate sequences, thereby balancing population diversity with focused exploration. Subsequently, offspring are generated through annealed importance sampling and partial rejection control, then refined using Metropolis-Hastings steps with fitness-guided local generative mutation kernels. Throughout the iterations, a gradually decaying temperature schedule transitions the sampler from broad exploration in the early stages to targeted exploitation as the search progresses. This adaptive approach enables MADA to efficiently traverse the vast protein sequence space while concentrating computational efforts on promising regions.

We introduce additional notation required for the MADA sequence manipulation procedure. Let $T \geq 1$ denote the number of iterations for which MADA is executed. Recall that the sequence length is ℓ, and define $I = \{1, 2, \ldots, \ell\}$ as the set of all sequence positions. We denote by $I_{\text{fixed}} \subset I$ the subset of positions that remain fixed throughout the optimization process, with cardinality $|I_{\text{fixed}}| = \ell_{\text{fixed}}$. The corresponding set of all viable positions for mutation is then defined as $I_{\text{mut}} = I \setminus I_{\text{fixed}}$. The masking rate $p_S \in (0, 1]$ controls the proportion of variable positions that are masked at each iteration. Let N denote the population size maintained throughout the sampling process.

Starting from the wild-type sequence x_{wt}, MADA proceeds through the following iterative procedure. At each iteration, we stochastically determine the number of positions M to mask by sampling $M = \max\left\{1, \text{Binomial}\left(|I_{\text{mut}}|, p_S\right)\right\}$, where the maximum operation ensures at least one position is masked. Building upon the work of Lu et al., who select mask locations

uniformly at random from the entire sequence (as the authors do not consider hard-fix constraints), we propose MADA-DUCB, which extends MADA with *informed* and *adaptive* position selection using reinforcement learning.[14] This approach, detailed in Section 2.4, enhances the mutation strategy by introducing targeted position selection. For the current discussion, we assume a set of mask positions $S \subseteq I_{\text{mut}}$ has been identified for mutation.

Mutation and Stabilization

Let the current iteration be $t \in \{1, \ldots, T\}$. Given a pool of N candidate sequences, which contains duplicates, we generate a set of M unique mask positions $S \subseteq I_{\text{mut}}$ for each *unique* candidate. For these identified unique sequences, we employ a generative prior to propose mutations at the masked sites, effectively filling these positions with new amino acids. Notably, this operation leaves the hard-fix positions I_{fixed} and the unmasked positions *unchanged*. Our transition kernel from the sequence before mutation x to the sequence after mutation x' is $q(x' \mid x) = p(S \mid x)p_\phi(x'_S \mid x_{S^c})$, where x_{S^c} represents the unmasked residues. However, not all mutations will necessarily enhance fitness. Thus, we perform a single Metropolis-Hastings step to stabilize the transition from x to x'. The acceptance ratio for this move is given by $a(x, x') = \min\left(1, \frac{p(S|x')}{p(S|x)} \cdot \frac{\exp(H_\lambda(x')/\tau_t)}{\exp(H_\lambda(x)/\tau_t)}\right)$, where we accept the proposed sequence x' if $U \sim \text{Uniform}[0, 1] < a(x, x')$; otherwise, we retain the original sequence x. This process results in an updated pool of N sequences, each having undergone mutation and stabilization.

Selection and Annealed Importance Resampling

After mutation and stabilization, we apply an MH rejuvenation kernel that leaves π_{ϕ,H,τ_t} invariant and then transition the population to $\pi_{\phi,H,\tau_{t+1}}$ via annealed importance resampling. The annealed importance weight for each sequence is given by $w_i^{(t+1)} \propto \exp\left(\left(\frac{1}{\tau_{t+1}} - \frac{1}{\tau_t}\right) H_\lambda(x_i)\right)$.

To focus the particle population on high-potential candidates and amortize the cost of generative prior calls, we employ a two-phase resampling strategy where a small set of K *prototypes* is selected and then used to regenerate the full population. First, we select a subset of K representative sequences, denoted as $\{x_j^*\}_{j=1}^K$, from the original N candidates $\{x_i\}_{i=1}^N$. This selection process uses weighted sampling with replacement, where weights $w_i^{(t+1)}$ reflect each sequence's importance. Then from these K prototypes, we reconstruct a full ensemble of N sequences, $\{\tilde{x}_i^{(t+1)}\}_{i=1}^N$, through uniform random sampling with replacement. The ensemble encapsulates $\leq K$ distinct sequence prototypes primed for the subsequent mutation phase.

Alternatively, instead of stochastic resampling, particles can be selected greedily by their importance weights. Although this top-K selection procedure introduces a small bias through the permanent elimination of low-weight particles, empirically, it often results in accelerated convergence to high-fitness regions.

2.4. *Informed Mutation Strategy via Reinforcement Learning-Based Adaptive Mask Selection*

In this section, we introduce a novel strategy to construct an informed proposal distribution for selecting mutation sites based on reinforcement learning. This adaptive approach enhances the

efficacy of the mutation process within the MADA sampling procedure. We frame the selection of mask positions as a sequential decision problem aimed at maximizing cumulative reward. This framework embodies the classic exploration-exploitation dilemma: balancing the selection of positions that have historically yielded beneficial mutations against exploring positions that may reveal unexpected improvements.

At each mutation step $t_m = 1, \ldots, T_m$, where $T_m = K \times T$, the action involves choosing a mask set $S_{t_m} \subseteq I_{\text{mut}}$ of size M. This is equivalent to simultaneously pulling M arms in a multi-armed bandit setting, constituting a combinatorial bandit problem. Following a mutation based on mask S_{t_m}, proposed by the generative prior $p_\phi(\cdot)$ and subjected to the Metropolis-Hastings step, we observe a reward $r_{t_m} = (H_\lambda(x') - H_\lambda(x))/M$. We employ a key simplifying heuristic by attributing the observed reward equally to each of the M chosen positions within the mask $S_{t_m} = \{S_{t_m,j}\}_{j=1}^{M}$. This approach assesses the impact of individual positions, even though they are selected and mutated in combination. Over many diverse trials, it allows the estimated value of a single mutation site to approximate its current marginal contribution across various contexts. Although this simplification does not capture interaction effects, it provides a computationally tractable approach to an otherwise intractable search space of $\binom{\ell-\ell_{\text{fixed}}}{M}$ possible mask configurations.

For additional computational efficiency, our bandit algorithm is *context-free*, meaning we do not provide the sequence context when making decisions. This makes our masking probability identical for all particles, i.e., $p(S \mid x) = p(S \mid x')$ for all x, x'. Consequently, the acceptance ratio simplifies to $a(x, x') = \min\left(1, \frac{\exp(H_\lambda(x')/\tau_t)}{\exp(H_\lambda(x)/\tau_t)}\right)$. As our particle population evolves through the MADA procedure, this creates a non-stationary reward distribution with the underlying reward for each position constantly shifting as the sequence context changes. To address the non-stationarity inherent in this setting, we employ the discounted Upper-Confidence Bound (D-UCB) algorithm.[23] This approach utilizes a discounted empirical average,

$$N_{j,t_m}(\gamma) = \sum_{o=1}^{t_m-1} \gamma^{t_m-1-o} \mathbb{1}_{\{j \in S_o\}}, \quad \hat{\mu}_{j,t_m} = \frac{\sum_{o=1}^{t_m-1} \gamma^{t_m-1-o} r_o \mathbb{1}_{\{j \in S_o\}}}{N_{j,t_m}(\gamma)},$$

where $\gamma \in (0, 1)$ is the discount factor. The choice of γ creates a bias-variance trade-off: $\gamma \to 1$ yields low bias but slow adaptation to changes, while $\gamma \to 0$ offers fast adaptation but high variance in estimates. In practice, we use $\gamma \approx 0.95$ to balance adaptation speed with estimation stability. The upper-confidence bound is then updated as

$$\text{UCB}_{j,t_m} = \underbrace{\hat{\mu}_{j,t_m}}_{\text{exploitation}} + \alpha \underbrace{\sqrt{\frac{\log \sum_{i=1}^{\ell-\ell_{\text{fixed}}} N_{i,t_m}(\gamma)}{N_{j,t_m}(\gamma)}}}_{\text{exploration}},$$

where $\alpha > 0$ is the exploration constant, balancing exploitation of known high-reward positions with exploration of less certain options. We implement these discounted summations using a recursive update scheme. Subsequently, we select the M positions with the highest upper-confidence bounds to form our mask set for the current iteration.

3. Experiments

3.1. *Dataset*

To evaluate the effectiveness of ProVADA+ in AAV capsid design, we leveraged a large-scale publicly available dataset of experimentally characterized AAV2 variants originally reported by Bryant et al.[3] The dataset comprises 293,574 variants derived through targeted random mutagenesis of a 28-residue hyper-variable region in the VP3 capsid protein (positions 359-387) found to play a critical role in capsid assembly, heparin binding, and immune recognition. Its functional importance has made it a focal point for previous protein engineering efforts.[24]

To quantify capsid viability, each variant in the dataset was evaluated using a high-throughput production assay that measures the efficiency with which it assembles into a capsid and encapsidates a viral genome. This was operationalized by comparing the relative abundance of each variant in two distinct experimental libraries: a plasmid DNA input library representing the initial pool of designed sequences, and a viral library composed of genomes extracted from successfully assembled capsids.

For a given variant i, the viability was summarized by a "viral selection score," defined as the log-ratio between its normalized read count in the viral library ($n_{i,\text{viral}}$) and the corresponding count in the DNA library ($n_{i,\text{DNA}}$), $\text{score}_i = \log\left(\frac{n_{i,\text{viral}}}{n_{i,\text{DNA}}}\right)$. This score serves as a quantitative proxy for packaging fitness: Higher values indicate that a variant is preferentially enriched in the viral output pool relative to its starting frequency, reflecting successful capsid assembly and genome packaging. Notably, this enrichment-based metric captures both structural integrity and functional assembly capacity of the AAV2 capsid, and was validated through replicate experiments showing high reproducibility.[3]

Prior to use, we applied several preprocessing steps to the original dataset to ensure consistency and validity. All variants containing premature stop codons or non-finite viral selection scores were excluded. To resolve conflicting measurements, we removed duplicate sequences that had inconsistent viral selection values across records. In addition, each sequence in the dataset was annotated with its corresponding full-length VP1, VP2, and VP3 isoform sequences. After cleaning, our final dataset comprised 289,736 variants, with a distribution of viral selection scores shown in Figure 2. We include a link to the processed version of this dataset in Section 6.

3.2. *AAV Viability Oracle*

To support generative steering with a black-box oracle, we trained a predictive model to estimate viral selection scores directly from AAV2 hyper-variable region sequences. As our base architecture, we used ESM-Cambrian (ESMC), a 600-million parameter transformer from the ESM family optimized for protein representation learning and sequence-level prediction tasks.[25] A five-layer MLP regression head was added on top of the final layer, which takes the mean-pooled token embeddings to produce a final continuous score prediction. The model was fine-tuned using mean squared error (MSE) loss on the processed dataset, which was randomly split into three parts: training (60%), validation (20%), and test (20%) sets. On the held-out test set, the model achieved strong performance, with an MSE of 1.4262, root

mean squared error (RMSE) of 1.1942, mean absolute error (MAE) of 0.8761, and coefficient of determination (R^2) of 0.8738. We refer to this fine-tuned model as the ESMC oracle, and use it in all subsequent experiments to both guide sequence generation and evaluate generated sequences through *in silico* viability score prediction.

3.3. *Structure-Conditioned AAV2 Capsid Design with ProVADA+*

To generate novel AAV2 capsid variants with high predicted viability, we implemented an AAV2-focused version of ProVADA+ that integrates a structure-based generative model with our black-box viability oracle. This approach allows us to enforce a structural prior on the generated sequences while simultaneously steering the design process toward our desired functional objective.

We used ProteinMPNN—an inverse folding model that designs amino acid sequences conditioned on a fixed protein structure—as our generative prior. Given a backbone with masked residues, ProteinMPNN proposes plausible amino acid substitutions. As a starting point, we first predicted the structure of the wild-type AAV2 VP3 protein using Boltz-2.[5] This predicted structure was then used as a fixed backbone scaffold for ProteinMPNN throughout all subsequent generations. For the fitness oracle ($F(x)$), we utilize the ESMC oracle (Section 3.2) to provide the predicted viral selection score for any given sequence variant. The search was guided by the composite objective ($H_\lambda(x)$) defined in Section 2.1, which balances the predicted viability score with a BLOSUM62-based sequence similarity penalty to maintain proximity to the wild-type sequence. The design process focused on the 28-residue hyper-variable region (positions 359-387). All positions outside this region were held constant, and only positions within the hyper-variable region were masked for redesign. We ran the design trajectories with three different seeds for both the MADA and MADA-DUCB sampling variants.

4. Results

4.1. *Baseline Analysis: In Silico Selection of Random Mutagenesis*

To establish a performance baseline and contextualize the difficulty of the AAV2 capsid design task, we first evaluated a "generate-and-filter" strategy. This approach is analogous to the 'model-selected' approach used in Bryant et al., where a machine learning model is used to screen a large pool of randomly generated sequences,[3] and corresponds to the rejection sampling baseline presented in Lu et al.[14] A workflow without ProVADA+ could similarly use our ESMC oracle to rank random mutants and select the highest-scoring candidates *in silico*. We generated 30,000 random variants at each mutational distance (k) for $k = 1, 3, 5, 7$ from the wild-type AAV2 hyper-variable sequence and used the ESMC oracle to obtain predicted viral selection scores. We then selected the top 2% (600 variants) from each mutational distance. (Note: for $k = 1$, there are only $19 \times 28 = 532$ possible variants, so all were included).

The results, summarized in Figure 3A, demonstrate that this undirected random mutagenesis approach is an inefficient strategy for improving capsid viability. As the number of mutations increases, the distribution of predicted scores shifts progressively toward lower, more deleterious values. Notably, while some variants with three mutations ($k = 3$) achieve high predicted scores, the population exhibits very little sequence diversity, clustering tightly

near the wild-type sequence. This *in silico* finding directly corroborates the experimental results of Bryant et al., who reported that only 10% of randomly chosen variants with 2 to 10 mutations were viable—a figure that plummeted to just 0.3% for variants with six or more mutations.[3] Our baseline analysis similarly underscores the ruggedness of the AAV2 fitness landscape and confirms that a naive, undirected search is highly unlikely to yield improved variants. This motivates the need for more sophisticated, oracle-guided generation strategies like ProVADA+.

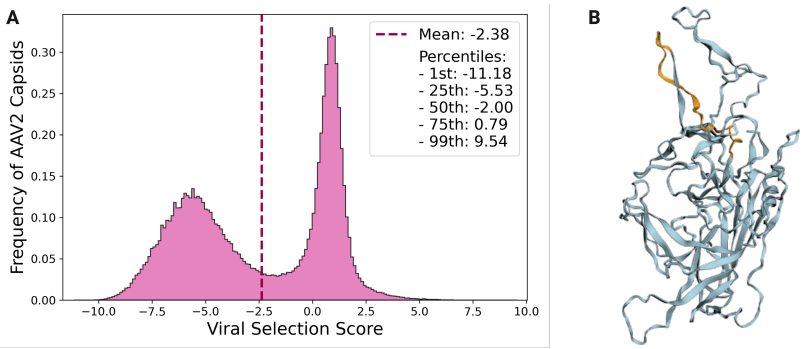

Fig. 2. A) Underlying bimodal distribution of viral selection scores in the processed version of the Bryant et al. dataset. B) Predicted structure of the wild-type VP3 protein containing the 359-387 hyper-variable region, which is highlighted orange in the structure.

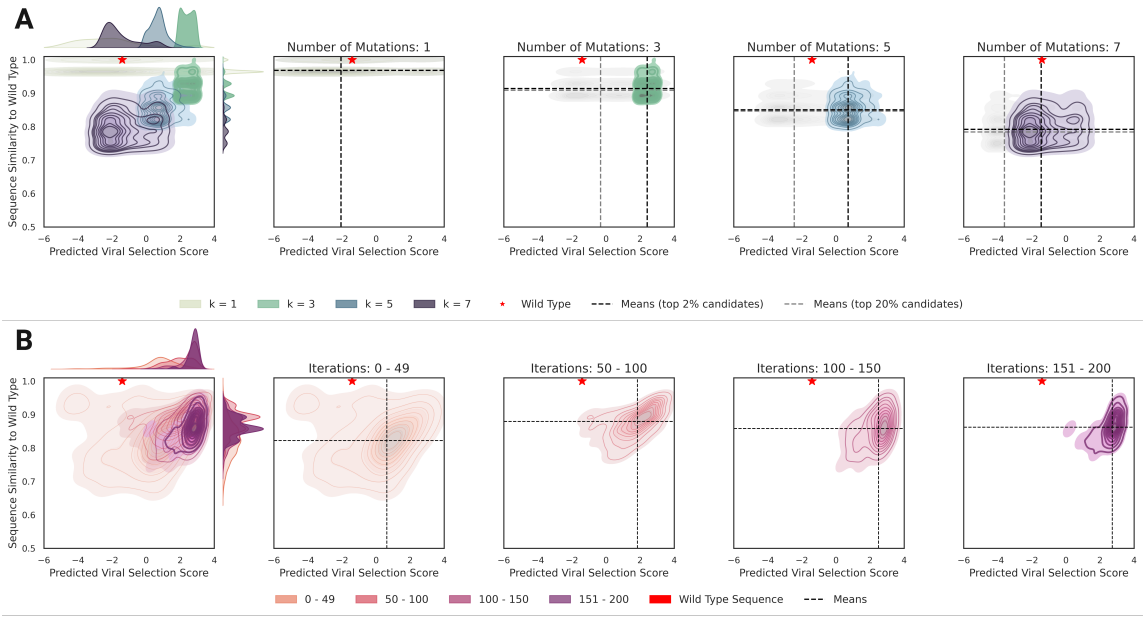

Fig. 3. Joint distributions of predicted viral selection score and similarity to wild-type of the sequences generated through A) *In silico* selected variants from random mutagenesis; B) ProVADA-guided design. Variants generated via ProVADA+ consistently exhibit high viral selection scores while maintaining adequate diversity.

4.2. *MADA and MADA-DUCB Convergence*

To assess the impact of our informed mutation strategy, we compared the convergence performance of the standard Mixture-Adaptation Directed Annealing (MADA) algorithm against our enhanced variant, MADA-DUCB, which employs a reinforcement learning approach for adaptive mask selection. We tracked the average combined fitness objective, $H_\lambda(x)$, for the population of sequences over 200 design iterations. The experiment was conducted using three independent seeds for each method to ensure the robustness of our findings.

The results are summarized in Figure 4. Both sampling methods successfully guide the design process toward sequences with significantly improved fitness scores, demonstrating the overall effectiveness of the ProVADA framework. However, the MADA-DUCB variant exhibits superior sampling efficiency. As shown by the steeper initial slope of its convergence curve, MADA-DUCB achieves higher fitness values much earlier in the trajectory compared to the standard MADA sampler, which relies on uniform random masking. This accelerated convergence highlights the benefit of intelligently targeting mutation sites that are more likely to yield high-reward outcomes. While both methods eventually approach a similar fitness plateau, the adaptive strategy of MADA-DUCB enables a more rapid and efficient exploration of the high-dimensional sequence landscape to identify optimal candidates.

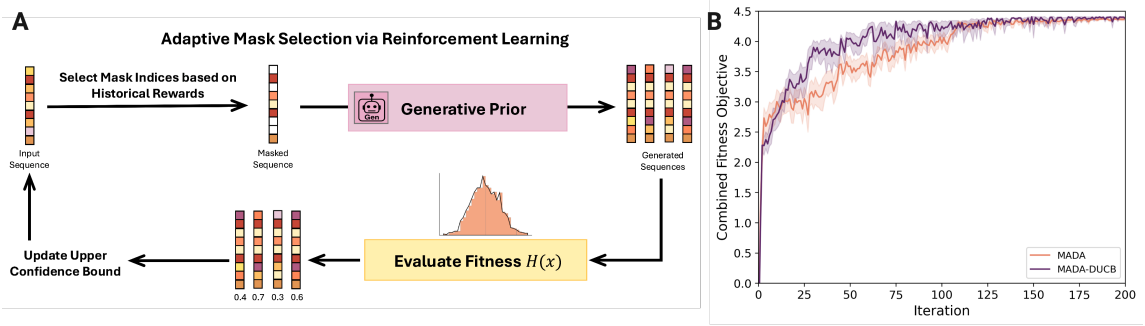

Fig. 4. A) Schematic overview of the informed mutation position selection procedure (D-UCB). B) Comparison of trajectory plots between MADA and MADA-DUCB. The plot shows the mean composite fitness objective $H_\lambda(x)$ with error bands over 200 design iterations. Lines represent the mean across three independent runs for both sampling methods.

4.3. *ProVADA+ Steers Generation Towards High-Viability Sequences*

To better understand the trajectory of the guided design process, we plot the sequence populations at four different stages throughout a MADA-DUCB sampler run. Figure 3B shows the joint distribution of predicted viral selection score and sequence similarity (BLOSUM62-informed Hamming distance divided by the hyper-variable region length) to the wild-type at different stages of the design trajectory. The process begins with a broad exploration phase (Iterations 0-49), where the sampler generates a diverse set of sequences centered around the wild-type reference. In this initial stage, most variants have low predicted viral selection scores, many of which are predicted to be less viable than the wild-type. As the trajectory progresses (Iterations 50-150), the population distribution systematically shifts toward the right, indicat-

ing a steady increase in predicted viability scores. In the final iterations (151-200), the sampler converges on a concentrated population of high-quality sequences. These final candidates exhibit significantly improved predicted viral selection scores, while maintaining an adequately diverse range of sequence similarity (0.8-0.95) to the wild-type.

5. Discussion

In this work, we introduce ProVADA+, an enhanced framework that provides substantial improvements in flexibility, biological relevance, and algorithmic efficiency to ProVADA. First, we incorporate fixed residue constraints to preserve important functional sites and augment the original Hamming distance with a biologically-informed BLOSUM62-based sequence divergence penalty, thereby generating variants with improved biochemical plausibility. Second, we propose an informed and adaptive strategy for selecting which residue positions to mutate. This approach leverages the discounted Upper-Confidence Bound (D-UCB) algorithm to facilitate a more efficient and targeted exploration of the sequence space. Collectively, these methodological advances yield a more powerful and flexible framework for generating optimized protein sequences, with the potential to accelerate the rational design of novel proteins for biotechnological and therapeutic applications. We demonstrate the superiority of ProVADA+ on *in silico* adaptation of AAV capsids for viability over the widely-used targeted random mutagenesis.

This work opens several directions for future investigation. For example, alternative algorithms could be employed to identify high-viability reference sequences that differ substantially from the wild-type AAV sequence, serving as improved initialization points for the MADA algorithm to explore more distant regions of the sequence space. Furthermore, the modularity of the ProVADA+ framework is particularly well-suited for multi-objective optimization through composable guidance. The composite objective function, $H_\lambda(x)$, could be readily extended to incorporate a weighted combination of multiple, independent fitness oracles. Each oracle could be trained on data from a distinct experimental assay, representing a different desired functional property—such as transduction efficiency, tissue tropism, or low immunogenicity. For instance, in AAV engineering, one could simultaneously steer the generative process with the current viability oracle, a second oracle trained on an *in vivo* screen for heart-specific tropism, and a third designed to minimize predicted T-cell epitopes. In future work, we plan to integrate additional stability-related scoring functions and incorporate uncertainty metrics into oracle predictions to prioritize variants for wet laboratory validation to enable more complex designs. This approach would allow researchers to create a bespoke fitness landscape tailored to a complex set of design criteria and thus enable the generation of highly specialized variants that jointly satisfy multiple functional constraints.

6. Code and Data Availability

The implementation of ProVADA+ is publicly available at the following GitHub repository: https://github.com/SUwonglab/ProVADA. In addition, the preprocessed version of the AAV2 capsid viability fitness dataset, originally reported in Bryant et al.,[3] is available at the following Hugging Face link: https://huggingface.co/datasets/bviggiano/aav2_capsid_viability.

References

1. L. Sellés Vidal, M. Isalan, J. T. Heap and R. Ledesma-Amaro, A primer to directed evolution: current methodologies and future directions, *RSC Chem. Biol.* **4**, 271 (April 2023).
2. J. T. Koerber, J.-H. Jang and D. V. Schaffer, DNA shuffling of adeno-associated virus yields functionally diverse viral progeny, *Mol. Ther.* **16**, 1703 (October 2008).
3. D. H. Bryant, A. Bashir, S. Sinai, N. K. Jain, P. J. Ogden, P. F. Riley, G. M. Church, L. J. Colwell and E. D. Kelsic, Deep diversification of an AAV capsid protein by machine learning, *Nat. Biotechnol.* **39**, 691 (June 2021).
4. J. Jumper, R. Evans, A. Pritzel, T. Green, M. Figurnov, O. Ronneberger, K. Tunyasuvunakool, R. Bates, A. Žídek, A. Potapenko, A. Bridgland, C. Meyer, S. A. A. Kohl, A. J. Ballard, A. Cowie, B. Romera-Paredes, S. Nikolov, R. Jain, J. Adler, T. Back, S. Petersen, D. Reiman, E. Clancy, M. Zielinski, M. Steinegger, M. Pacholska, T. Berghammer, S. Bodenstein, D. Silver, O. Vinyals, A. W. Senior, K. Kavukcuoglu, P. Kohli and D. Hassabis, Highly accurate protein structure prediction with AlphaFold, *Nature* **596**, 583 (August 2021).
5. S. Passaro, G. Corso, J. Wohlwend, M. Reveiz, S. Thaler, V. R. Somnath, N. Getz, T. Portnoi, J. Roy, H. Stark, D. Kwabi-Addo, D. Beaini, T. Jaakkola and R. Barzilay, Boltz-2: Towards accurate and efficient binding affinity prediction, *bioRxiv* , p. 2025.06.14.659707 (June 2025).
6. J. Dauparas, I. Anishchenko, N. Bennett, H. Bai, R. J. Ragotte, L. F. Milles, B. I. M. Wicky, A. Courbet, R. J. de Haas, N. Bethel, P. J. Y. Leung, T. F. Huddy, S. Pellock, D. Tischer, F. Chan, B. Koepnick, H. Nguyen, A. Kang, B. Sankaran, A. K. Bera, N. P. King and D. Baker, Robust deep learning-based protein sequence design using ProteinMPNN, *Science* **378**, 49 (October 2022).
7. A. Madani, B. Krause, E. R. Greene, S. Subramanian, B. P. Mohr, J. M. Holton, J. L. Olmos, Jr, C. Xiong, Z. Z. Sun, R. Socher, J. S. Fraser and N. Naik, Large language models generate functional protein sequences across diverse families, *Nat. Biotechnol.* **41**, 1099 (August 2023).
8. T. Hayes, R. Rao, H. Akin, N. J. Sofroniew, D. Oktay, Z. Lin, R. Verkuil, V. Q. Tran, J. Deaton, M. Wiggert, R. Badkundri, I. Shafkat, J. Gong, A. Derry, R. S. Molina, N. Thomas, Y. Khan, C. Mishra, C. Kim, L. J. Bartie, M. Nemeth, P. D. Hsu, T. Sercu, S. Candido and A. Rives, Simulating 500 million years of evolution with a language model, *bioRxiv* , p. 2024.07.01.600583 (July 2024).
9. A. Bansal, H.-M. Chu, A. Schwarzschild, S. Sengupta, M. Goldblum, J. Geiping and T. Goldstein, Universal guidance for diffusion models, *arXiv [cs.CV]* (February 2023).
10. H. Chung, J. Kim, M. T. Mccann, M. L. Klasky and J. C. Ye, Diffusion posterior sampling for general noisy inverse problems, *arXiv [stat.ML]* (September 2022).
11. P. Emami, A. Perreault, J. Law, D. Biagioni and P. C. S. John, Plug & play directed evolution of proteins with gradient-based discrete MCMC, *arXiv [cs.LG]* (December 2022).
12. J. Ho and T. Salimans, Classifier-free diffusion guidance, *arXiv [cs.LG]* (July 2022).
13. N. Blalock, S. Seshadri, A. Babbar, S. A. Fahlberg, A. Kulkarni and P. A. Romero, Functional alignment of protein language models via reinforcement learning, *bioRxiv* , p. 2025.05.02.651993 (May 2025).
14. W. S. Lu, X. Zhang, L. S. Mille-Fragoso, H. Dai, X. J. Gao and W. H. Wong, ProVADA: Generation of subcellular protein variants via ensemble-guided test-time steering, *bioRxiv* , p. 2025.07.11.664238 (July 2025).
15. M. F. Naso, B. Tomkowicz, W. L. Perry, 3rd and W. R. Strohl, Adeno-associated virus (AAV) as a vector for gene therapy, *BioDrugs* **31**, 317 (August 2017).
16. D. Wang, P. W. L. Tai and G. Gao, Adeno-associated virus vector as a platform for gene therapy delivery, *Nat. Rev. Drug Discov.* **18**, 358 (May 2019).
17. M. B. Demircan, L. J. Zinser, A. Michels, M. Guaza-Lasheras, F. John, J. M. Gorol, S. A.

Theuerkauf, D. M. Günther, D. Grimm, F. R. Greten, P. Chlanda, F. B. Thalheimer and C. J. Buchholz, T-cell specific in vivo gene delivery with DART-AAVs targeted to CD8, *Mol. Ther.* **32**, 3470 (October 2024).
18. L. E. Mays, L. Wang, R. Tenney, P. Bell, H.-J. Nam, J. Lin, B. Gurda, K. Van Vliet, K. Mikals, M. Agbandje-McKenna and J. M. Wilson, Mapping the structural determinants responsible for enhanced T cell activation to the immunogenic adeno-associated virus capsid from isolate rhesus 32.33, *J. Virol.* **87**, 9473 (September 2013).
19. K. Börner, E. Kienle, L.-Y. Huang, J. Weinmann, A. Sacher, P. Bayer, C. Stüllein, J. Fakhiri, L. Zimmermann, A. Westhaus, J. Beneke, N. Beil, E. Wiedtke, C. Schmelas, D. Miltner, A. Rau, H. Erfle, H.-G. Kräusslich, M. Müller, M. Agbandje-McKenna and D. Grimm, Pre-arrayed pan-AAV peptide display libraries for rapid single-round screening, *Mol. Ther.* **28**, 1016 (April 2020).
20. M. Tabebordbar, K. A. Lagerborg, A. Stanton, E. M. King, S. Ye, L. Tellez, A. Krunnfusz, S. Tavakoli, J. J. Widrick, K. A. Messemer, E. C. Troiano, B. Moghadaszadeh, B. L. Peacker, K. A. Leacock, N. Horwitz, A. H. Beggs, A. J. Wagers and P. C. Sabeti, Directed evolution of a family of AAV capsid variants enabling potent muscle-directed gene delivery across species, *Cell* **184**, 4919 (September 2021).
21. S. Henikoff and J. G. Henikoff, Amino acid substitution matrices from protein blocks, *Proc. Natl. Acad. Sci. U. S. A.* **89**, 10915 (November 1992).
22. S. F. Altschul, W. Gish, W. Miller, E. W. Myers and D. J. Lipman, Basic local alignment search tool, *J. Mol. Biol.* **215**, 403 (October 1990).
23. A. Garivier and E. Moulines, On upper-confidence bound policies for non-stationary bandit problems, *arXiv preprint arXiv:0805.3415* (2008).
24. Y.-S. Tseng and M. Agbandje-McKenna, Mapping the AAV capsid host antibody response toward the development of second generation gene delivery vectors, *Front. Immunol.* **5**, p. 9 (January 2014).
25. ESM cambrian: Revealing the mysteries of proteins with unsupervised learning Accessed: 2025-5-23.