From the holdings of the Department of Special Collections , Princeton University Library, Princeton, NJ

These documents are either in the public domain, or can only be used for educational and research purposes (Fair use) as per U.S. Copyright law (text below). By accessing this file, all users agree that their use falls in the public domain or within fair use as defined by the copyrightlaw. We do not charge any permission or usage fees for the publication of images of material inour collections, including those provided by Princeton University Library, via our website, catalog, or directly from staff.For more information regarding copyright, credit, and citation please review the information onthe Copyright, Credit and Citation Guidelines page of our website.
https://library.princeton.edu/special-collections/policies/copyright-credit-and-citation-guidelines. If you have any questions, including how to order higher-quality images from the collection, contact Special Collections Public Services via our Ask Us! Form. https://library.princeton.edu/special-collections/ask-us

---

Master's Theses, 2022-2026

Computer Science, 2023

2024

Diversity-Guided Sampling for Protein Sequence Design via Iterative Refinement

Ferragu, Constance

Princeton, NJ : Princeton University

---

http://arks.princeton.edu/ark:/88435/dsp016969z416r

U.S. Copyright law: The copyright law of the United States (Title 17, §108, United States Code) governs the making of photocopies or other reproductions of copyrighted material. Under certain conditions specified in the law, libraries and archives are authorized to furnish a photocopy or other reproduction. One of these specified conditions is that the photocopy or reproduction is not to be used for any purpose other than private study, scholarship, or research. If a user makes a request for, or later uses, a photocopy or reproduction for purposes in excess of fair use that user may be liable for copyright infringement. This institution reserves the right to refuse to accept a copying order if, in its judgment, fulfillment of the order would involve a violation of copyright law. https://www.copyright.gov/title17/title17.pdf

Diversity-Guided Sampling for Protein Sequence Design via Iterative Refinement

Constance Ferragu

Presented to the faculty
of Princeton University
in candidacy for the degree of
Master of Science and Engineering
recommended for acceptance by
the Department of Computer Science

Advisor: Professor Adji Bousso Dieng
Reader: Professor Olga Troyanskaya

May 2024

© Copyright by Constance Ferragu, 2024. All rights reserved.

Abstract

Protein sequence discrete diffusion models have become increasingly valuable for the design of novel proteins, due to their robust generative capabilities and effective bi-directional processing of sequence data. To improve the generation of sequences with desired functions, gradient guidance methods are often used to guide sampling with a discriminative model. However, sampling from these models presents challenges due to the vast and discrete nature of the sequence space. These models tend to prioritize denoising high-likelihood tokens, resulting in similar sequences. Furthermore, gradient guidance methods tend to collapse generation to fewer modes. Protein design pipelines tend to work with fixed-size batches of sequences. Hence, given the high cost of experimental validation, optimizing the sample efficiency of these batches is essential. In this thesis, we propose Vendi Guidance, a guided diffusion sampling algorithm designed to improve the exploration efficiency of sequence space and the diversity of sampled sequence sets. Our method leverages the Vendi Score—a statistical measure of diversity—to select edit positions that will most effectively improve the diversity objective and to guide the model's hidden representations towards diverse denoising steps. We demonstrate that Vendi Guidance can iteratively refine a seed sequence into a more diverse set of sequences while ensuring that the quality of the sequences does not deteriorate.

Acknowledgements

I extend my deepest gratitude to my advisor, Dr. Adji Bousso Dieng, whose ongoing support and guidance have been invaluable during my Master's journey. Academically, it has been an immense privilege to learn from her and engage in numerous insightful discussions. On a personal level, I am particularly grateful for her mentorship, which will allow me to graduate confidently with a clear vision for my career.

I would also like to express my appreciation to Dr. Olga Troyanskaya and Dr. Ines Meraoumia for their invaluable input and meticulous feedback during the final stages of my thesis. Their insights have significantly enriched my work and helped identify possible directions for future research.

I am grateful to the Department of Computer Science and Vertaix for providing me with the resources necessary to undertake my Master's studies. Through this support, I was introduced to the world at the intersection of AI and the natural sciences, a field I have grown very fond of.

I am especially thankful to Cradle, and in particular to Franzi Geiger and Eli Bixby, for their invaluable guidance in defining a compelling research objective. Their expertise and mentorship have been crucial in deepening my understanding of protein design, significantly contributing to my academic growth.

Finally, I wish to express my heartfelt thanks to my parents, siblings, and friends for their support and encouragement throughout the past two years. Merci!

Contents

Abstract . . . iii

Acknowledgements . . . iv

List of Tables . . . vii

List of Figures . . . viii

1 Introduction **1**

1.1 Protein Design . . . 2

1.1.1 Why Design in Sequence Space? . . . 4

1.2 Sequence Space Design . . . 5

1.2.1 Machine Learning Guided Sequence Design . . . 6

1.3 Research Objective . . . 9

2 Background **10**

2.1 Protein Sequence Models . . . 10

2.1.1 Autoregressive Language Models . . . 10

2.1.2 Masked Language Modeling . . . 11

2.1.3 Denoising Diffusion . . . 12

2.2 Generative Guidance . . . 15

2.2.1 Discrete Generative Guidance . . . 15

2.3 Diverse Set Sampling Methods . . . 17

2.4 Formal Objective . . . 18

2.5 Vendi Score 19

3 Method **21**

3.1 Vendi Guidance 21

3.1.1 Salient Edit Positions 22

3.1.2 Gradient Guidance Denoising 24

3.1.3 Vendi Score Kernel 25

4 Results **28**

4.1 Experiment Setup 28

4.1.1 Starting Sequences 28

4.1.2 Experiment Details 29

4.2 Results 33

4.2.1 Diverse Sampled Sets 34

4.3 Ablations 35

5 Conclusion **37**

5.1 Discussion 37

5.2 Limitations 38

5.3 Future Work 39

A **40**

A.1 Parameters 40

List of Tables

A.1 Vendi Guidance hyperparameter settings. 40

List of Figures

1.1 The sequence-structure-function paradigm states that sequence determines structure and structure determines function. 2

2.1 Order agnostic autoregressive diffusion with masked corruption. 14

3.1 Point mutations in the middle of the E4B sequence have a more varied effect on protein fitness, which is also captured by the Vendi Score gradients. (A) Map from [69] which shows the effect of single amino acid changes of 932 variants of the E4B protein, measured by $\log_2 E$ scores. Blue, white, and red boxes indicate variants that were depleted, neutral, or enriched, respectively. Gray boxes represent that no data passed quality filters, and boxed rectangles represent the wild-type (WT) residues. (B) The variance of E scores for each position from [69]. (C) Saliency map of the E4B protein produced by the EvoDiff-Seq model, with positions corresponding to those in (A) and (B). . . 23

3.2 One guided denoising step of Vendi Guidance. 25

3.3 As the diversity measured by the Levenshtein kernel Vendi Score increases, so does the variability in the EvoDiff-Seq cosine Vendi Score. Scatter plot of Vendi Scores of 1000 CATH superfamilies [50] computed with the Levenshtein kernel vs. the EvoDiff-Seq cosine kernel. 26

4.1 Vendi Guidance is able to increase the diversity of sampled sets without deteriorating their fitness. We present the evolution of the (normalized) Vendi Score (*left*) and the fitness score (*right*) over sampling steps of the Markov random walk for sampling sets of $N = 96$ sequences. The four sequences, AAV, E4B, avGFP, and TEM are of increasing lengths from 28 to 286 amino acids, demonstrating that the sequence length does not have a noticeable effect on Vendi Guidance. 31

4.2 Generated protein sequences with Vendi Guidance are more diverse and explore new areas of the local GFP landscape. (A) UMAP of EvoDiff embeddings of natural sequences from a UniRef50 sequence alignment (grey, $N = 986$) and from a Vendi Guidance sampled sequence set (red, $N = 192$). (B) UMAP of EvoDiff embeddings of natural sequences from the same UniRef50 sequence alignment (grey, $N = 986$) and from a Baseline sampled sequence set (red, $N = 192$). 32

4.3 Vendi Guidance proposes novel mutations and a smoother distribution of mutations across sampled sequences. We present heatmaps of the proposed mutations within final sets of $N = 96$ sampled sequences from the Baseline sampling method (*left*) and from Vendi Guidance (*right*) with the avGFP protein (*top*) and the E4B protein (*bottom*). 33

4.4 Choosing salient edit positions has a larger effect on diversity than the Vendi Score gradient guidance. We present ablations of the gradient-guided denoising (*top*) and of edit position selection (*bottom*) on the Vendi Score (*left*) and fitness (*right*) of the avGFP protein. Guidance has a minimal negative impact on sequence fitness while salient edits have a minimal positive impact on sequence fitness. 35

4.5 Choosing salient edit positions can only have a positive effect on diversity when paired with gradient guidance. We present ablations of gradient-guided denoising and edit position selection for the E4B protein. We observe that choosing salient edit positions alone does not improve diversity, but when paired with gradient guidance improves diversity. 36

Chapter 1

Introduction

Protein design is the engineering of novel protein structures with tailored functions, holding significant potential for addressing key challenges in various fields. This includes enhancing drug discovery processes [36], engineering more efficient biochemical pathways for industrial processes [61], and developing bio-based materials with superior qualities [9]. For example, insulin has been optimized to modify how quickly it is absorbed or how long it acts, benefiting patients with diabetes. Designing proteins from scratch is extremely challenging due to the complexity of predicting how amino acid sequences will fold into functional structures. Consequently, refining existing proteins is a more common approach.

The *sequence-structure-function* paradigm, which states that sequence determines structure, and structure determines function [3], underpins protein design methods. A protein is completely defined by its amino acid sequence, from which it is synthesized. However, it is its three-dimensional structure that ultimately dictates its functionality. This dichotomy is apparent in protein design tools, which can typically be categorized into sequence-space search and structure-space search tools.

Although structure space is vastly smaller and theoretically simpler to search, modeling sequence space has several advantages from a machine learning (ML) perspective. In particular, the data availability is non-negligible, with approximately 250 million known sequences

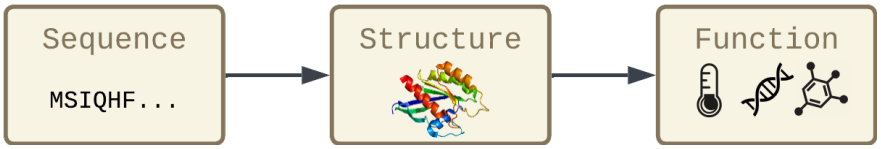

Figure 1.1: The sequence-structure-function paradigm states that sequence determines structure and structure determines function.

[14] compared to only 200,000 known structures [11]. Moreover, structure-based design does not offer straightforward solutions since a protein can only be synthesized from its sequence.

Identifying promising sequences within the vast, discrete sequence space remains a difficult problem. For a sequence of length N, there are 20^N possible sequences[1], amounting to approximately one trillion unique amino acid sequences. However, naturally evolved proteins only represent an infinitesimal fraction of this space. An exhaustive search for a sequence that maximizes an objective in this space is an NP-Hard problem [42]. While certain machine learning protein design methods can sample stable sequences reaching high fitness objectives, these typically sample sequences independently, resulting in sets of similar sequences. Given the high cost of experimental validation and the usual constraint of fixed batch sizes in protein design pipelines, optimizing sample efficiency is essential. The objective becomes sampling fixed-size sets of sequences with high fitness from diverse regions of sequence space that evenly cover various attributes. In this thesis, we introduce a guided sampling method that directly samples N sequences with higher diversity without compromising sequence fitness.

1.1 Protein Design

Traditional Methods Traditional sequence-space tools, such as consensus design [54] and directed evolution [5], rely on the evolutionary similarity of proteins, often measured through amino acid sequence similarity. Both methods leverage the principle that evolutionarily related (or *homologous*) sequences tend to share similar structures and functions. In structure

[1]there are 20 canonical amino acids

space, rational protein design approaches [34] identify key regions and residues crucial for a protein's function of interest to make targeted changes aimed at improving this function. Consequently, to synthesize optimized proteins, these methods must solve the *inverse folding problem* [81], predicting the amino acid sequence that will fold into the target protein structure.

Early Computational Methods Computational methods have significantly enhanced protein design, enabling a more precise and efficient selection of candidates [43]. Non-ML methods use optimization algorithms and simulations to navigate the large, non-convex protein landscape. In sequence space, tools like BLAST [71] and Clustal Omega [64] analyze sequence patterns to identify homologous sequences and leverage evolutionary relationships. In structure space, physics-based tools model the physical interactions at an atomic level to predict a protein's conformation, interactions with other molecules, and various properties [57, 62, 59, 72]. However, these methods are computationally expensive and their accuracy is limited for complex systems.

Machine Learning Methods More recently, deep learning models trained on available protein data have been developed for protein design objectives. Building upon the success of large language models, researchers are now exploring similar approaches in the domain of protein sequences. Much like natural language, sequences of amino acids contain distinct patterns and characteristics that influence their structure and functions. Protein language models have been trained both with masked language modeling (MLM), such as in ProtBERT [20] and ESM [56], and autoregressive modeling (LM), such as in ProtGPT [21]. The former learns strong representation spaces for sequences, while the latter learns a distribution over sequences, $p(s)$. More recently, discrete diffusion models have been developed to combine the representation capabilities of MLMs and the generative power of LMs [1, 24]. Conditioning on structure space are inverse folding models, which learn to predict a sequence s given a structure x—$p(s|x)$—often encoded as a graph neural network [63, 17, 30, 28]. Solving

the reverse problem are structure prediction models, such as AlphaFold [33], which learn to predict the tertiary structure x of a protein given its sequence s, $p(x|s)$ [40, 7, 78]. Finally, working solely in structure space are structure generation models, trained to directly learn the probability $p(x)$. These models range from generative adversarial networks [2] to diffusion models [76, 80, 77], which are often adapted to account for rigid transformation invariances in $SE(3)^N$ space. These various protein models are combined with optimization algorithms and fine-tuned discriminators for guided protein design.

1.1.1 Why Design in Sequence Space?

Designing proteins directly in sequence space offers significant advantages, particularly with the use of deep learning methods, which depend on large and robust datasets for effective training. While the disparity in data availability is non-negligible, the larger concern is likely that protein structural data is biased due to selection bias and technical limitations [53]. Naturally, structures are determined for proteins of interest, such as medically significant proteins. Furthermore, only a handful of structure determination methods exist, which are not adaptable to all proteins, thus limiting the diversity of protein structures that can be effectively studied.

From a non-ML perspective, working directly in sequence space also offers clear advantages. Even with effective methods in structure space, the inverse folding problem must be addressed to derive sequences for synthesis. This introduces additional challenges since there is not always a direct bijection between sequence and structure. There exist structures with no known sequences that can fold into them, and a single sequence may fold into multiple structures – a phenomenon known as *conformational diversity*. Additionally, intrinsically disordered proteins (IDPs) contain regions that lack a fixed three-dimensional structure under certain conditions. Their ability to bind to different molecules and conform to various shapes makes them particularly valuable, however, this makes them difficult to model in structure space. Therefore, the functional protein space is only fully captured within the

sequence space.

1.2 Sequence Space Design

Formal Definition

We define the input sequence space $\mathfrak{S} = \mathcal{V}^t$, where $\mathcal{V}$ is an ordered, discrete vocabulary of v tokens and t is the max sequence length. We let $\mathcal{V}$ include all 20 canonical amino acids, an unknown amino acid token, and a padding token to accommodate for sequences of varying lengths. Protein sequence design is the problem of finding sequences $s \in \mathfrak{S}$ which maximize single or multiple objectives $\{f_j(s)\}_j$. Often, one objective will be protein stability and the others will be functional objectives, such as thermostability or binding affinity. One class of methods uses Bayesian optimization to maximize the black-box objective functions,

$$\max_{S \in \mathfrak{S}^N} \left(\prod_{s \in S} f_1(s), ..., \prod_{s \in S} f_j(s) \right) \tag{1.1}$$

while other methods sample sequences from a learned distribution,

$$s \sim p(s; \theta) = \frac{e^{-f_\theta(s)}}{\sum_{s' \in \mathfrak{S}} e^{-f_\theta(s')}} \tag{1.2}$$

where, $f_\theta(s)$ learns to model a function that combines the objectives, such as,

$$f(s) = \prod_j f_j(s) \tag{1.3}$$

Iterative Design

Most widely used practical approaches to sequence design are iterative processes that can be divided into three steps: (1) typically beginning with a starting sequence or a collection of starting sequences, then, (2) carefully choosing mutations (amino acid changes) to apply to these sequences to create new candidate sequences, and finally, (3) evaluating the effect of

these mutations and carrying this information over to the next round. This iterative process relies on random walks through the sequence space. In its most basic form, this is directed evolution.

This iterative design process is limited by the slow and costly process of experimental validation. Despite recent advancements, high-throughput experimentation methods remain limited, and developing effective screening methods is complex and non-trivial. Therefore, an objective of protein design is to develop pipelines that require fewer experimental iterations. To do so, curated batches must propose promising sequences that explore a diverse range of attributes at every iteration.

Proposing a batch of promising sequences is necessary but not sufficient. It is crucial for the batch to be diverse to ensure a broad search of the sequence space, increasing the likelihood of finding unique and highly functional proteins. This is especially important for the sequence space, which is known to be non-convex, where ascending paths can often become trapped in local maxima, preventing the discovery of better sequences that lie beyond [19]. While choosing larger step sizes in optimization algorithms can help surpass local maxima, this can also lead to uncontrollable results reaching regions of non-functional proteins.

1.2.1 Machine Learning Guided Sequence Design

While sequence space is fundamentally more practical for protein design, it still presents its own challenges. Finding promising sequences within sequence space is difficult due to its vastness and discrete nature. Furthermore, as the requirements for specificity and functionality increase, the subset of functional sequences that meet these criteria sharply diminishes [44].

Protein sequence machine learning models can be leveraged to optimally refine seed sequences by predicting mutational effects on protein function or by directly suggesting novel mutations that maintain protein stability and optimize functionality. These models are paired with optimization algorithms to better navigate the complex protein sequence-

function landscape. However, the discrete nature of this space means that common gradient-based optimization methods are not directly applicable.

Some methods resort to gradient-free methods like Markov Chain Monte Carlo (MCMC) [73, 26, 31]. For example, Gibbs sampling [32] or simulated annealing [73] can be used to sample from MLMs. These methods provide generative capabilities to models that are not directly defined for generation but are strong representation learners. Furthermore, MCMC methods allow for flexible definitions of energy functions (see Equation 1.2) which can incorporate multiple objectives. However, MCMC samplers can require hundreds of thousands of steps to converge [24], a non-negligible limitation.

Improving upon directed evolution are optimization algorithms that involve generating mutations randomly and screening them with sequence-to-function discriminators, formally, predicting the label or class y given a sequence s: $p(y|s)$ [79, 58]. Genetic algorithms or Bayesian optimization, known for their inefficiency in large search spaces, can be enhanced through these discriminators which ensure that the selected sequences at each round are more likely to enhance the fitness objective. This allows for more efficient exploration of mutational effects [48]. These discriminators can classify mutations as beneficial, deleterious, or neutral, or directly predict the objective function $f(\cdot)$. However, these strategies depend on large, annotated datasets of proteins, which are not always available. Furthermore, these datasets are often biased toward several over-represented protein families [45, 12], thus confining exploration to a restricted portion of the landscape. To address this limitation, some methods rely on large pre-trained backbones, improving their ability to learn from limited variant data via transfer learning [10]. Yet, these hill-climbing methods are highly susceptible to local maxima, preventing them from reaching better sequences [13]. While these methods are valuable for reducing the expensive and long experimental screening process, they do not address the challenges of navigating through sequence space.

To better navigate the sequence space, random walks must be properly guided, otherwise, they can become highly inefficient. Addressing this, certain methods use reinforcement

learning to balance the exploration and exploitation of the space [4, 37]. These algorithms start by predominantly exploring uncertain regions of the landscape (exploration) and, as they accumulate data and confidence, gradually focus on identifying sequences that are optimally tuned to the objective function (exploitation). However, these methods can be complex and take a while to converge, especially when the search space is large. Other approaches like AdaLead [65] or PEX [55] develop optimized evolutionary algorithms that adapt how they propose mutations as they progress, rather than proposing mutations at random. Nonetheless, these methods often rapidly degrade sequence stability as the objective function improves, limiting the effective number of edits before reaching poor quality sequences that nevertheless achieve high objective function scores. This limits sample efficiency and ultimately the rate of convergence to optimal solutions.

Further refining the approach are Bayesian optimization methods that use generative models to propose mutations more strategically, in conjunction with discriminative models for conditional sampling [31, 24, 68]. For example, LaMBO [68] uses continuous sequence embeddings from the inner representations of masked language models (MLMs) to navigate the search, circumventing the challenge of working in discrete space. Building on this, diffusion optimized sampling (NOS) [24] improves the approach by incorporating a discrete diffusion model instead of an MLM, thereby improving the generative capabilities of the method. Additionally, NOS further optimizes the process by strategically choosing edit positions that hold the greatest potential for improving the objective function. This allows NOS to optimize the objective function in fewer mutations, which is particularly advantageous when an edit budget is considered, a constraint that is often imposed to control the quality of synthesis. These methods strategically propose mutations, allowing for significant leaps across the protein landscape, which enhances both efficiency and the breadth of exploration.

1.3 Research Objective

Diversity and sample efficiency are very important in protein design pipelines which need to generate fixed-size batches of sequence candidates. We define set diversity by the effective number of equal modes or attributes present in the set. By maximizing the diversity of a set of high-fitness sequences, we can optimize for what can be evaluated during experimental validation. We do not want to assess a set of sequences that all have the same attribute. Furthermore, considering the iterative design process, ensuring random walks are taken in various regions of the sequence space is important. We aim to do so with respect to the presence of various modes, rather than by taking large uncontrollable jumps in the space with large step sizes.

Current sampling methods are almost always designed to generate one sequence with high-likelihood at a time. Sampling techniques that produce a set of sequences, such as beam search, often result in significant overlap between the sequences. Hence, when the task at hand is to sample a set of diverse sequences, many sampling iterations and sub-sampling are needed to obtain a satisfactory set, a cost that is orthogonal to sampling time.

This thesis focuses on directly sampling diverse sets of N high-likelihood sequences from protein discrete diffusion models. We demonstrate that our method can directly generate a batch of N sequences with higher diversity without compromising the fitness of the batch. We observe a smoother distribution of proposed mutations and an increased coverage of the model's sequence space.

Chapter 2

Background

2.1 Protein Sequence Models

Three major types of protein sequence models exist: autoregressive language models, masked language models, and discrete diffusion models. The latter is a generalization of the other two models' objectives.

2.1.1 Autoregressive Language Models

Autoregressive Language Models (LMs) learn to predict the next token or amino acid given a sequence of amino acids, modeling the conditional probability $p(s_i|s_{<i})$, where $s_{<i} = s_1, s_2, ..., s_{i-1}$. LMs factorize the likelihood of a sequence $p(s)$ using the probability chain rule:

$$p(s) = \prod_{i=1}^{|s|} p(s_i|s_{<i}) \tag{2.1}$$

This makes generating samples from these types of models straightforward. LMs learn to generate sequences in a pre-defined left-to-right decoding order, a natural definition for languages that are written from left to right like English. LMs have proved to be highly effective natural language generators, however, they have certain limitations when directly

applied to the protein language. Certain regions of sequences are directly linked to a function that should remain unchanged and thus need to be fixed during generation. This requires fixing arbitrary sub-sequences, which is not possible with the pre-defined decoding order of LMs. More importantly, LMs capture dependencies in one direction only. Considering epistasis, where the effect of one mutation is dependent on the presence of one or more different mutations, bidirectional context is crucial for protein sequence analysis [70].

2.1.2 Masked Language Modeling

Masked language models (MLMs) learn to predict a token within a sequence of amino acids that is masked or hidden, modeling the conditional probability $p(s_i|s_{\setminus i})$, where $s_{\setminus i} = s_1, ..., s_{i-1}, s_{i+1}, ..., s_L$, and accounting for bi-directionality. MLMs are able to better understand long-range interactions within sequences, making them better representation learners than LMs [56] for protein sequences. However, they do not factorize nicely to define a distribution over language,

$$p(s) \neq \prod_{i=1}^{|s|} p(s_i|s_{\setminus i}) \tag{2.2}$$

making them poor generators.

With MLMs, each token prediction can be influenced by any part of the input sequence, not constrained by a sequential order, (i.e. forming a fully connected graph rather than a tree). Methods have been developed to approximate generation from these models. In particular, an iterative refinement method process that starts with a partially or fully masked sequence and decodes masks one at a time in a randomly chosen order [23]. Given a random permutation $\sigma(\cdot)$, we can define this as,

$$p_\sigma(s) \approx \prod_{i \in \{\sigma(1),...,\sigma(|s|)\}} p(s_i|s_{\setminus i}) \tag{2.3}$$

After filling all the masks, the iterative process can continue by introducing new masks and

making adjustments.

2.1.3 Denoising Diffusion

Denoising diffusion models are latent variable generative models defined by a forward and a reverse Markov process. The model seeks to learn a data distribution $q(x_0)$ through reversing a corruption process (the forward process) that maps clean data points x_0 to samples from a prior $\pi(x)$, independent of x_0. In the forward process, training data $x_0 \sim q(x_0)$ is augmented to form a Markov Chain $\{x_t\}_0^T$, where the transition $q(x_t|x_{t-1})$, adds some random noise to the variable. For a T large enough, $p(x_T) \sim \pi(\cdot)$.

The model then learns the reverse denoising process by estimating $p_\theta(x_{t-1}|x_t)$ which aims to model the true distribution $q(x_{t-1}|x_t)$. Hence, by the probability chain rule, we can define the probability of the learned denoising process as,

$$p_\theta(x_0, ..., x_T) = p(x_T) \prod_{i=1}^{T} p_\theta(x_{t-1}|x_t) \tag{2.4}$$

where we choose $\pi(x)$ as the prior $p(x_T)$.

Denoising Discrete Diffusion

Denoising diffusion models were first introduced for continuous variables with Gaussian noise. This process can be adapted to discrete data [6] by defining categorical noise transition matrices $Q_1, ..., Q_T$,

$$[Q_t]_{i,j} = q(x_t = j|x_{t-1} = i) \tag{2.5}$$

which iteratively corrupt via a controlled Markov process. Formally, we define the forward process as,

$$q(x_t|x_{t-1}) = q(x_t|x_{t-1}, x_0) = \text{Cat}(x_t; p = x_{t-1}Q_t) \tag{2.6}$$

$$q(x_t|x_0) = \text{Cat}(x_t; p = x_{t-1}(Q_1Q_2...Q_t)) \tag{2.7}$$

where $\text{Cat}(x; p)$ is a categorical distribution over the one-hot row vector x with probabilities given by the row vector p. Note that Q_t is applied to each token in a sequence independently. These transition matrices can be defined in various ways, such as with discrete uniform noise [67], token embedding distance to transition more frequently between tokens with similar embeddings, or masking tokens independently with some probability β_t [6].

The model learns to directly predict the logits of a categorical distribution $p_\theta(\hat{w}|x_t)$, where $\hat{w} \in \mathcal{V}$ is some token, by minimizing the following denoising objective,

$$L(\theta) = \mathbb{E}_{q(x_0)}\mathbb{E}_{q(x_t|x_0)}[-\log p_\theta(x_0|x_t)] \tag{2.8}$$

and giving the following reverse distribution,

$$p_\theta(x_{t-1}|x_t) \propto \sum_{\hat{w}} q(x_{t-1}|x_t, \hat{w}) p_\theta(\hat{w}|x_t) \tag{2.9}$$

where,

$$q(x_{t-1}|x_t, x_0) = \frac{q(x_t|x_{t-1}, x_0)q(x_{t-1}|x_0)}{q(x_t|x_0)} \tag{2.10}$$

$$= \frac{x_t Q_t^T \odot x_0(Q_1Q_2...Q_{t-1})}{x_0(Q_1Q_2...Q_t)x_t^T} \tag{2.11}$$

The learned reverse transition distribution $p_\theta(x_{t-1}|x_t)$ is applied for $t = [0, ..., T]$ to sample from the model.

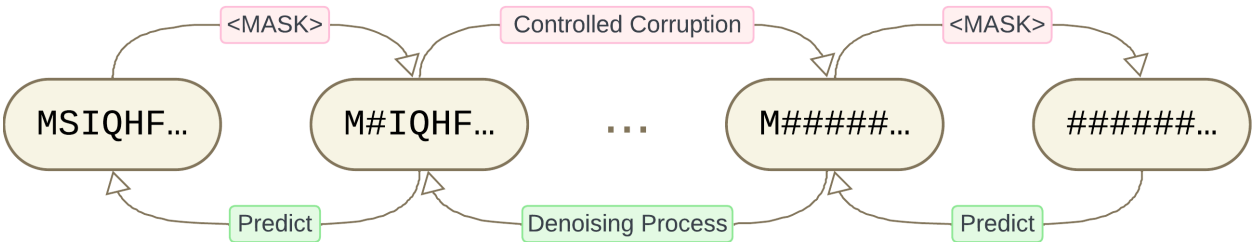

Figure 2.1: Order agnostic autoregressive diffusion with masked corruption.

Absorbing State Diffusion Models Within the scope of protein sequence data, masking in MLMs has shown to be highly effective in learning good representations. Absorbing state diffusion models use masking transition matrices, where the mask token serves as the absorbing state. At each time step t, any dimension can independently transition to the absorbing state with a probability β_t. This approach integrates the bi-directionality of MLMs with the generative capabilities of LMs. However, the number of decoding steps T in this model is not fixed and is often much greater than the sequence length $L << T$. Additionally, this introduces T as a parameter that needs to be optimized.

Order Agnostic Autoregressive Diffusion Order Agnostic Autoregressive Diffusion (OADM) decays a single dimension at every time step rather than setting independent absorbing probabilities. That is, at every time step, one token at a uniformly random position is masked (in the forward process) or denoised (in the reverse process). This allows the diffusion process to terminate when all dimensions are absorbed in $T = L$ steps. It has been shown that OADM corresponds to continuous-time absorbing diffusion models [27]. OADM generalizes both the LM and MLM training tasks while keeping the number of decoding steps fixed to the length of the input sequence [1]. More specifically, the MLM training task corresponds to training a single step of the OADM diffusion process and the LM training task is one specific decoding order of an OADM diffusion process. OADMs have been shown to reach equivalent performances to absorbing state diffusion models in fewer steps [27], and better performance than both MLMs and LMs [1].

2.2 Generative Guidance

Directly generating plausible sequences from generative models is not sufficient for protein design. This process does not allow for any control over the structure or function of the generated sequences, which does not align well with protein design problems that define functional objectives and technical constraints, such as edit budgets. Hence, guided generation is a natural solution.

Guided Generation Problem We want to sample from a distribution $p(s|a)$, where a is some desired controllable objective, such as high binding affinity or any other protein function, and s is a protein sequence. Generative models learn to model $p(s)$, hence if we introduce a discriminative model $p(a|s)$, we can sample from the product of experts,

$$p(s|a) \propto p(s)p(a|s) \tag{2.12}$$

where $p(s|a)$ corresponds to the likelihood of sequence s given attribute a. This can be adapted to the protein design problem, where each objective function $f_i(s)$ can be modeled by a discriminative model $p_i(a_i|s, \theta)$, and $p(s)$ corresponds to the sequence stability objective. Generative guidance methods tend to take gradient update steps with $p(a|s, \theta)$ to push $p(s)$ towards higher likelihood of attribute a [47].

2.2.1 Discrete Generative Guidance

Gradient guidance methods are quite natural within the definition of a continuous reverse diffusion SDE, however, they are not directly applicable to models that predict categorical distributions. A difficulty with categorical data guidance is ensuring the guidance does not directly oppose the distribution. For example, adding penalty scores to output logits goes directly against the predicted likelihood of the model, and thus trades off directly with the modeled distribution. This leads to methods that often see drops in sample log-likelihood in

exchange for higher objectives. For protein design, this would correspond to proteins that are less likely to be stable, but nevertheless have high likelihood of improving the objective, which is undesirable.

One possible adaptation is to use continuous diffusion models which diffuse from sequence embeddings and train a linear head to decode the tokens as a final step [39]. However, this method requires careful manipulations, such as mapping vectors to the nearest word embeddings at intermediate diffusion step. Diverging from diffusion models, Plug-and-Play Language Model (PPLM) [16] is a popular method for gradient guidance of autoregressive language models, which takes update steps on continuous inner representations of the LM with respect to a discriminator model $p_\theta(a|s)$ at each decoding step. DiGress [74], a discrete denoising diffusion model for generating graphs with categorical node and edge attributes, applies gradient guidance to one-hot token embeddings of a categorical distribution. However, Digress is directly applied to the output logits of the distribution using a continuous relaxation of the one-hot encoded input. This method still goes against the model's distribution, making it more difficult to balance the objective $p(a|s)$ and fitness $p(s)$ of the samples.

Diffusion Optimized Sampling Gruver et al. [24] propose Diffusion Optimized Sampling (NOS), a gradient-guided method for discrete diffusion models, which is closely related to PPLMs. Given any discrete diffusion network that predicts the logits of a distribution $p_\theta(\hat{w}|s)$ for sequence s, there exist multiple continuous hidden states within the model. We note that the output logits $\hat{w}$ given any sequence s are the same given any hidden state h_l of s. That is $p_\theta(\hat{w}|s) = p_\theta(\hat{w}|h_l)$ for all hidden layers l. NOS makes local updates to these hidden continuous states. NOS requires that the diffusion model and discriminative model share hidden states up to depth d and that the discriminative model be well-defined for corrupted inputs. At each diffusion step, NOS uses Langevin dynamics with temperature $\tau > 0$, with the following update step,

$$h'_t \leftarrow h'_t - \eta \nabla_{h'_t} [\lambda KL(p_\theta(\hat{w}|h_t)||p_\theta(\hat{w}|h'_t)) - v_\theta(h'_t)] + \sqrt{2\eta\tau}\epsilon \tag{2.13}$$

where $\epsilon \sim \mathcal{N}(0, I)$, η is the step size and λ is the regularization strength. The KL term ensures the resulting distribution does not diverge too far from the original distribution. At every diffusion step, this method shifts the inner representation in the direction of the sum of two gradients: towards higher likelihood of the diffusion model and towards higher likelihood of the objective function.

2.3 Diverse Set Sampling Methods

A lot of focus has been placed on developing stable guided sampling methods, however these methods almost always generate samples independently. While these methods are promising, they have also been shown to collapse the generation to a single or few modes, hurting the diversity of samples [18]. This limitation paired with the importance of being able to sample fixed batches of sequences with optimal sample efficiency for protein design, motivates the need to develop diverse set sampling methods.

In the continuous setting, several enhanced sampling methods for molecular simulations, like metadynamics and coupled replica methods, have been proposed to sample diverse sets with low-energy (high fitness). Metadynamics is a sequential sampling technique where the diversity of the $k-1$ previous samples is taken as a prior when generating the k-th sample to ensure the diversity of the set [8]. Coupled replica methods directly sample n dependent particles of a system with different sampling processes (replicas) in parallel [52, 29]. Rather than sampling one particle at a time, they sample a set of particles $(x_1, ..., x_n)$ targeting,

$$\tilde{p}(x_1, ..., x_n) = \phi(x_1, ..., x_n) \prod_{i=1}^{n} p(x_i) \tag{2.14}$$

where the potential $\log \phi(t)$ is a permutation invariant set function and introduces a depen-

dency between the replicas. Similarly, Particle Guidance [15] proposes a continuous diffusion sampling method that enforces a diversity constraint with a joint-particle time-evolving potential $\phi(\cdot)$. This method modifies the reverse diffusion process by adding the gradients of the potential, similar to classifier-guidance. Metadynamics and coupled replica methods typically use MCMC sampling methods which lack the finite-time sampling advantage of diffusion models, offered by Particle Guidance. However, this latter method is not directly applicable to discrete distributions.

Several diverse set decoding methods for models with categorical distribution outputs have been developed, such as Diverse Beam Search [75] or Determinantal Beam Search [46]. The former decodes k groups or sub-beams simultaneously, where typical beam search is used within each group while diversity is enforced between groups. The latter reformulates beam search as a diverse subset optimization problem through a series of subdeterminant maximization problems. These methods were primarily developed to overcome the poor diversity of Beam Search. However, they are not well-defined for sampling methods that may require guidance from an objective function, which is often the case in protein design.

In the protein design setting, DyNA PPO [4] uses a diversity-promoting reward function within its Proximal Policy Optimization (PPO) framework. Specifically, for a sequence s, it applies a penalty based on the weighted number of sequences previously proposed that are within a maximum distance of s. This reward system directly penalizes sequences that are too similar to those already proposed. However, this approach defines diversity solely on sequence similarity, which may not fully capture functional or structural diversity.

2.4 Formal Objective

In this work, we redefine the objective function to be a set function that accounts for set diversity. Given some objective function $f_\theta(s)$ defined for any sequence s, and some diversity score $DIV(S)$ defined for any set of sequences $S = \{s_i\}_{i=1}^N$, we redefine the objective function

as:

$$\tilde{f}_\theta(S) = \left[DIV(S)^\gamma \left(\prod_{s \in S} f_\theta(s) \right)^{1-\gamma} \right] \tag{2.15}$$

We want to solve the following optimization problem,

$$\max_{S \in \mathfrak{S}^N} \tilde{f}_\theta(S) \tag{2.16}$$

The parameter $\gamma \in [0, 1]$ is defined to balance the diversity and fitness of the set of N decoded sequences. When $\gamma = 0$, the problem becomes sampling N independent sequences without replacement for max fitness. When $\gamma = 1$, the problem becomes sampling a set with maximal diversity, without considering sequence fitness. In this work, we choose the Vendi Score [22] as the diversity function.

2.5 Vendi Score

The Vendi Score [22] is a statistical metric that quantitatively assesses the diversity within a collection of items. For a collection of samples $\{s_1, ..., s_N\}$, the Vendi Score is defined as the Shannon entropy of the eigenvalues of the normalized pairwise similarity matrix,

$$VS(\{s_1, ..., s_N\}) = \exp\left(-\sum_{i=1}^{N} \lambda_i \log \lambda_i \right) \tag{2.17}$$

where $\lambda_1, ..., \lambda_N$ are the eigenvalues of K/n, and K is the pairwise similarity matrix defined by a positive semi-definite similarity function $k : S \times S \to \mathbb{R}$, such that $k(s, s) = 1, \forall s$.

Given a collection of N items, the Vendi Score takes a maximal value of N when all items in the collection are uniquely distinct from each other and takes a value of 1 when all the items in the collection are identical. The Vendi Score corresponds to the effective rank of the kernel matrix K, which can be interpreted as the effective number of dissimilar elements in the collection being evaluated for diversity. In the context of machine learning, an easy way

to define the positive semi-definite similarity function for the Vendi Score is to use a kernel similarity metric between the embedded samples $\{e_1, ..., e_N\}$, where e_i is the embedding of s_i.

Chapter 3

Method

We present Vendi Guidance, a gradient-guided sampling method that decodes a diverse set of optimized protein sequences. In Vendi Guidance, an initial set of seed sequences is iteratively refined, guided by the gradients of the Vendi Score. Vendi Guidance aims to encourage the sequences to cover diverse regions of the protein sequence space, thereby reaching a more diverse final set of sequences while maintaining high sequence fitness.

3.1 Vendi Guidance

Vendi Guidance is an iterative refinement sampling method from an OADM that consists of iterations over two steps: (1) choosing B edit positions to mask and (2) denoising the partially masked sequence. To guide this random walk, Vendi Guidance proposes two solutions: one for choosing edit positions that have the most potential to increase diversity, and another that uses gradient guidance to update the logits to push the model towards diverse denoising steps.

Diffusion Model

We choose to use a discrete diffusion model with absorbing state transitions due to the strong representation learning abilities of masking objectives and the generative power of diffusion

models. Specifically, we choose to use an OADM rather than an absorbing state diffusion model due to its improved performance [1] and fixed number of sampling steps from any point in the diffusion process. This allows us to mask certain regions of the sequence and infill with part of a denoising process. We note that this method is still well-defined for absorbing state diffusion models.

3.1.1 Salient Edit Positions

Rather than randomly selecting edit positions, we aim to choose edit positions that have the most potential to improve the diversity among sampled sequences. We use the idea of saliency maps, which has previously been used to explain the decisions of classifiers and to choose edit positions in protein optimization [24] or language model decoding [25].

Given a set of sequences $S = \{s_i\}_{i \in \{1,\dots,N\}}$ that are being sampled,

$$S = \begin{cases} s_1 = s_{1,1} \cdots s_{1,L} \\ \vdots \\ s_N = s_{N,1} \dots s_{N,L} \end{cases} \tag{3.1}$$

we compute the Vendi Score of this set, $\mathrm{VS}(S)$, and compute the gradients back to the first embedding layer with respect to each position j in each sequence i,

$$\nabla_{i,j} \mathrm{VS}(S) \tag{3.2}$$

We can then compute a distribution over each position in each sequence,

$$\mathbb{P}[\text{edit } s_{i,j}] = \frac{v_{i,j}}{\sum_{a,b} v_{a,b}} \tag{3.3}$$

where,

$$v_{i,j} = \left(\sum_k \left| \left(\nabla_{i,j} \mathrm{VS}(S) \right)_k \right| \right)^{1/\tau} \tag{3.4}$$

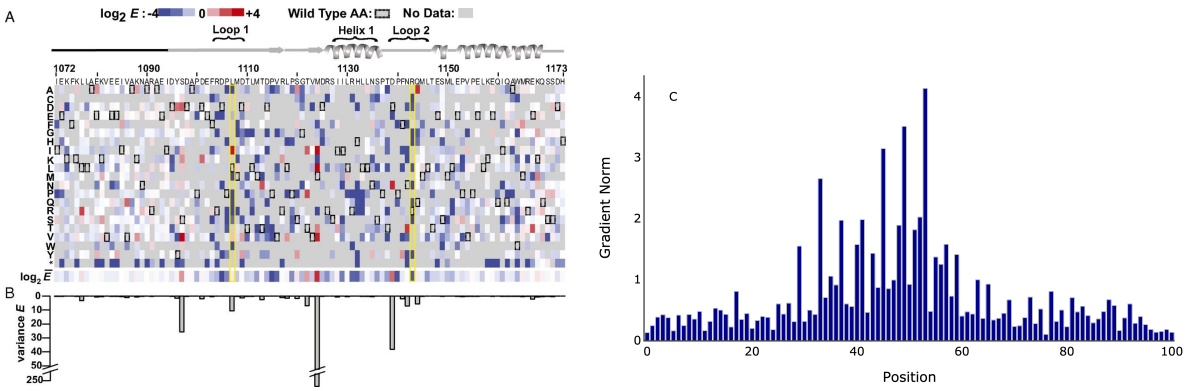

Figure 3.1: Point mutations in the middle of the E4B sequence have a more varied effect on protein fitness, which is also captured by the Vendi Score gradients. (A) Map from [69] which shows the effect of single amino acid changes of 932 variants of the E4B protein, measured by $\log_2 E$ scores. Blue, white, and red boxes indicate variants that were depleted, neutral, or enriched, respectively. Gray boxes represent that no data passed quality filters, and boxed rectangles represent the wild-type (WT) residues. (B) The variance of E scores for each position from [69]. (C) Saliency map of the E4B protein produced by the EvoDiff-Seq model, with positions corresponding to those in (A) and (B).

and $\tau \geq 0$ is the softmax temperature. To conserve certain parts of the starting sequence, we can set the saliency to 0 before computing and sampling from the edit position distribution. We draw (without replacement) B edit positions from the edit position distribution to mask.

Reliability of Saliency Maps A thorough investigation of the reliability of saliency maps is beyond the scope of this work. However, the results and investigation of saliency maps for this method consistently point in the right direction. In Figure 3.1, we illustrate the E4B protein saliency map and compare it to a map showing the effects of point mutations on the same protein from [69]. The E4B protein is known for its role in the cellular trash degradation process, which we further introduce in our Results section. This map shows that point mutations in the middle of the sequence have a more varied effect on protein fitness, a variation that the Vendi Score gradients also successfully capture.

3.1.2 Gradient Guidance Denoising

In the first step of iteration i, B edit positions are masked within the set of samples. The following step involves denoising these sequences. Vendi Guidance adapts NOS to denoise the partially masked set of N sequences simultaneously, guided by a set function that couples the sequences together. Instead of independently generating N sequences from the distribution $p(s)$, we aim to sample from the following joint distribution:

$$\propto \mathrm{VS}(S)^{\gamma} \left(\prod_{s \in S} p(s) \right)^{1-\gamma} \tag{3.5}$$

where $|S| = N$ and $\gamma \in [0, 1]$. We note that this can be adapted to any permutation invariant set function F by replacing VS with F.

Following the idea of NOS or PPLM, Vendi Guidance makes local update steps to the hidden continuous states of the OADM model. At denoising step t for sequence s_i, Vendi Guidance denoises from a modified reverse transition distribution $p_\theta(\hat{w}|h^{i\prime}_t)$, with the following update step to the hidden state h^i_t,

$$h^{i\prime}_t \leftarrow h^{i\prime}_t - \eta \nabla_{h^{i\prime}_t} [\gamma \mathrm{KL}(p_\theta || p'_\theta) - \mathrm{VS}(h^1_t, ..., h^{i\prime}_t, ..., h^N_t)] + \epsilon \sqrt{2\eta\tau} \tag{3.6}$$

where $p_\theta = p_\theta(\cdot|h^i_t)$ and $p'_\theta = p_\theta(\cdot|h^{i\prime}_t)$, $\epsilon \sim \mathcal{N}(0, I)$, η is the step size, γ is the regularization strength, and $\tau > 0$ is the temperature. We note that during the denoising process, when an infill has been made, the inner representation used to compute the VS term is updated. The KL divergence term ensures that the original transition distribution does not change too much, balancing the trade-off between sequence likelihood from the generative model and the diversity objective. These update steps shift the inner representation towards higher likelihood of diversity and higher likelihood of the diffusion model (higher fitness). Vendi Guidance is closely related to NOS and PPLM, however, it is defined in a space where N sequences are denoised dependently, rather than independently. Vendi Guidance extends

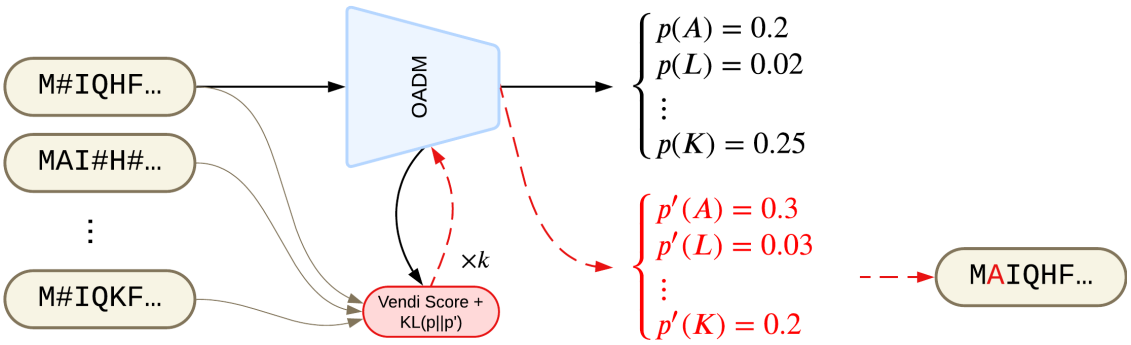

Figure 3.2: One guided denoising step of Vendi Guidance.

NOS by integrating it in an iterative refinement method which creates N diverse random walks in sequence space, allowing for a diverse exploration of sequence space.

We note that this method can easily be adapted to include multiple objective functions, such as thermostability or binding affinity, by simply adding a value function $v_\theta(h)$, defined from an inner representation h, to the loss,

$$h^{i\prime}_t \leftarrow h^{i\prime}_t - \eta\nabla_{h^{i\prime}_t}[\gamma KL(p_\theta||p'_\theta) - VS(h^1_t, ..., h^{i\prime}_t, ..., h^N_t) - v_\theta(h^{i\prime}_t)] + \sqrt{2\eta\tau}\epsilon \tag{3.7}$$

This could then also be used within a Bayesian optimization algorithm such as LaMBO-2 [24], which is defined to work with NOS gradient guidance.

3.1.3 Vendi Score Kernel

The Vendi Score is dependent on a user-specified kernel function, which should be defined accordingly to capture item similarity. Protein similarity is often computed through observing sequence patterns with sequence alignments or experimentally comparing functionality and structure [35]. For the purpose of protein design, the kernel should ideally be defined to capture protein diversity holistically, capturing sequence, function, and structural similarities. Given that protein language models are able to learn a feature space that contains information about sequence and function as captured in GO annotations [41], we can define

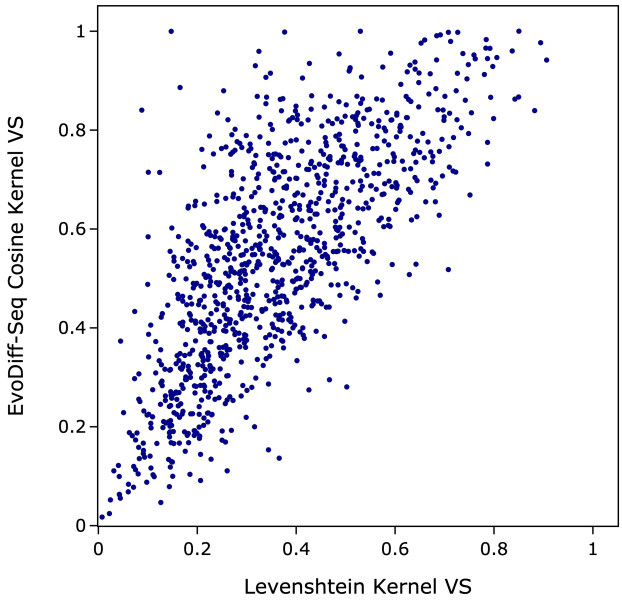

Figure 3.3: As the diversity measured by the Levenshtein kernel Vendi Score increases, so does the variability in the EvoDiff-Seq cosine Vendi Score. Scatter plot of Vendi Scores of 1000 CATH superfamilies [50] computed with the Levenshtein kernel vs. the EvoDiff-Seq cosine kernel.

a kernel as the cosine similarity between model embeddings,

$$k(s_i, s_j) = \frac{e(s_i)^T e(s_j)}{\|e(s_i)\| \, \|e(s_j)\|} \tag{3.8}$$

where $e(s_i)$ is the OADM model embedding of sequence s_i, which we compute by average pooling the token embeddings.

A Holistic Similarity Score

To justify our kernel choice, we compare this presumed more holistic score to a sequence similarity kernel. The Levenshtein edit distance is the minimum number of edits required to change one sequence into another [38], and has played an important role in sequence alignments [66]. We define the *Levenshtein kernel*, a sequence similarity kernel, as,

$$\text{LevSim}(s_i, s_j) = \frac{\max\{|s_i|, |s_j|\} - \text{Lev}(s_i, s_j)}{\max\{|s_i|, |s_j|\}} \in [0, 1] \tag{3.9}$$

where $\text{Lev}(s_i, s_j)$ is the Levenshtein edit distance between sequences s_i and s_j. We note that this kernel is well defined since when $s_i = s_j$, we have $\text{LevSim}(s_i, s_j) = 1$ and when s_i and s_j are most different, we have $\text{LevSim}(s_i, s_j) = 0$. We define the *EvoDiff-Seq cosine kernel* as the cosine similarity between EvoDiff-Seq embeddings.

We randomly sampled 1000 CATH super-families [50], collections of sequences that show evidence of a relationship based on at least two criteria: sequence, structure, or function. We then compared their Levenshtein Vendi Scores with their EvoDiff-Seq cosine Vendi Scores. As shown in Figure 3.3, as the diversity measured by the Levenshtein kernel Vendi Score increases, so does the variability in the EvoDiff-Seq cosine Vendi Score. This indicates that sequence similarity alone does not capture all forms of diversity between proteins, but the inner representations of EvoDiff-Seq can. This supports our decision to use a kernel directly defined in the model's feature space. Furthermore, this allows us to have a differentiable Vendi Score, which is easily applicable to diffusion optimized sampling.

Chapter 4

Results

4.1 Experiment Setup

We evaluate Vendi Guidance on four common protein design starting sequences of various lengths. First, we show the evolution of Vendi Score and sequence fitness over sampling steps to demonstrate that Vendi Guidance can increase set diversity while keeping the fitness stable. Next, we perform ablations on choosing salient edit positions and the denoising guidance to better understand the two components of Vendi Guidance. Lastly, we evaluate the effect of Vendi Guidance on output sequence sets.

4.1.1 Starting Sequences

To study the versatility of our method, we choose four different starting sequences with lengths varying from 28 to 286 amino acids.

Green Fluorescent Protein (avGFP) avGFP is a common protein that can exhibit bright green fluorescence when exposed to light in the blue to ultraviolet range [60]. This protein is often used as a biosensor to detect gene expressions and protein locations. We use the wild-type sequence of length 239 as our starting sequence.

Adeno-associated Virus (AAV) We selected a 28-amino acid segment (positions 561 to 588) of the VP1 protein from the Adeno-associated Virus for analysis due to its short sequence length. This protein has extensively been studied for its potential for DNA payload packaging in gene therapy applications [49].

Ubiquitination Factor Ube4b (E4B) The E4B protein is known for its important role in the trash degradation process within a cell by interacting with ubiquitin [69]. We use the same starting sequence of length 102 as described in [69] for our analysis.

TEM-1 β-lactamase (TEM) The TEM protein is known for its role in antibiotic resistance, with its optimization problem focused on improving thermostability. For our analysis, we use the wild-type sequence of length 286 as our starting sequence.

4.1.2 Experiment Details

Baseline In all our experiments, we refer to the *Baseline* as the process of iteratively denoising steps with edit positions chosen uniformly and with no gradient guidance.

Model We use the pre-trained EvoDiff-Seq model, an OADM trained on 42M sequences from UniRef50 [1], a subset of UniProt which clusters sequences that have at least 50% sequence identity to each other. The model architecture is a dilated convolutional neural network. A smaller version of the model with 38M parameters exists, however in this work we use the 640M parameter model unless otherwise stated.

Sequence Fitness In all our experiments, we define sequence fitness with the pseudo log-likelihood (PLL) of EvoDiff-Seq,

$$\log pll(s) = \sum_{i=1}^{|s|} \log p(s_i | s_{<i}) \tag{4.1}$$

where $s_{<i}$ is the sequence s masked from position i to position $|s|$. We choose to interpret this fitness score as protein stability, a standard approach in the field. When reporting the fitness of a set of sequences, we report the sum of the PLL of all sequences within this set. We note that the sequence fitness score could easily be combined or replaced with a discriminative model for a specific functional objective, such as thermostability.

Vendi q The Vendi Score is a diversity metric that extends the Hill number of order $q = 1$, and has been adapted to other Hill numbers by relating the eigenvalues of the normalized similarity matrix to item prevalence [51]. This is defined as,

$$VS_q(s_1, ..., s_N) = \exp\left(\frac{1}{1-q}\log \sum_{i \in \text{supp}(\lambda(s_1,...,s_N,k))} \lambda_i^q(s_1, ..., s_N, k)\right) \tag{4.2}$$

where $\lambda(s_1, ..., s_N, k)$ denotes the set of eigenvalues of the normalized similarity matrix induced by kernel k and $\text{supp}(\lambda(s_1, ..., s_N, k))$ denotes the indices for the nonzero eigenvalues. The q in VS_q becomes a parameter in Vendi Guidance. In our grid search we evaluate for $q \in \{0.1, 0.5, 1.0, 5.0, \infty\}$. We find that a $q = 0.5$, and otherwise $q = 1$, almost always works best. Vendi scores with orders smaller than 1 have been shown to be more sensitive to rare modes and are better adapted for mode imbalance. This could indicate model imbalances in certain local areas of sequence space captured by EvoDiff-Seq.

Edit Budget B We evaluate different numbers of edit positions in multiples of N, the size of the sampled set. In particular we try $N \times (0.01)L$ and $N \times (0.05)L$, where L is the length of the starting sequence. We note that since the edit positions are chosen based on saliency, the number of edit positions per sequence in the set varies. We limit the number of edit positions per sequence to 20% of sequence length. We note that this limit could easily be adapted to account for edit budget constraints.

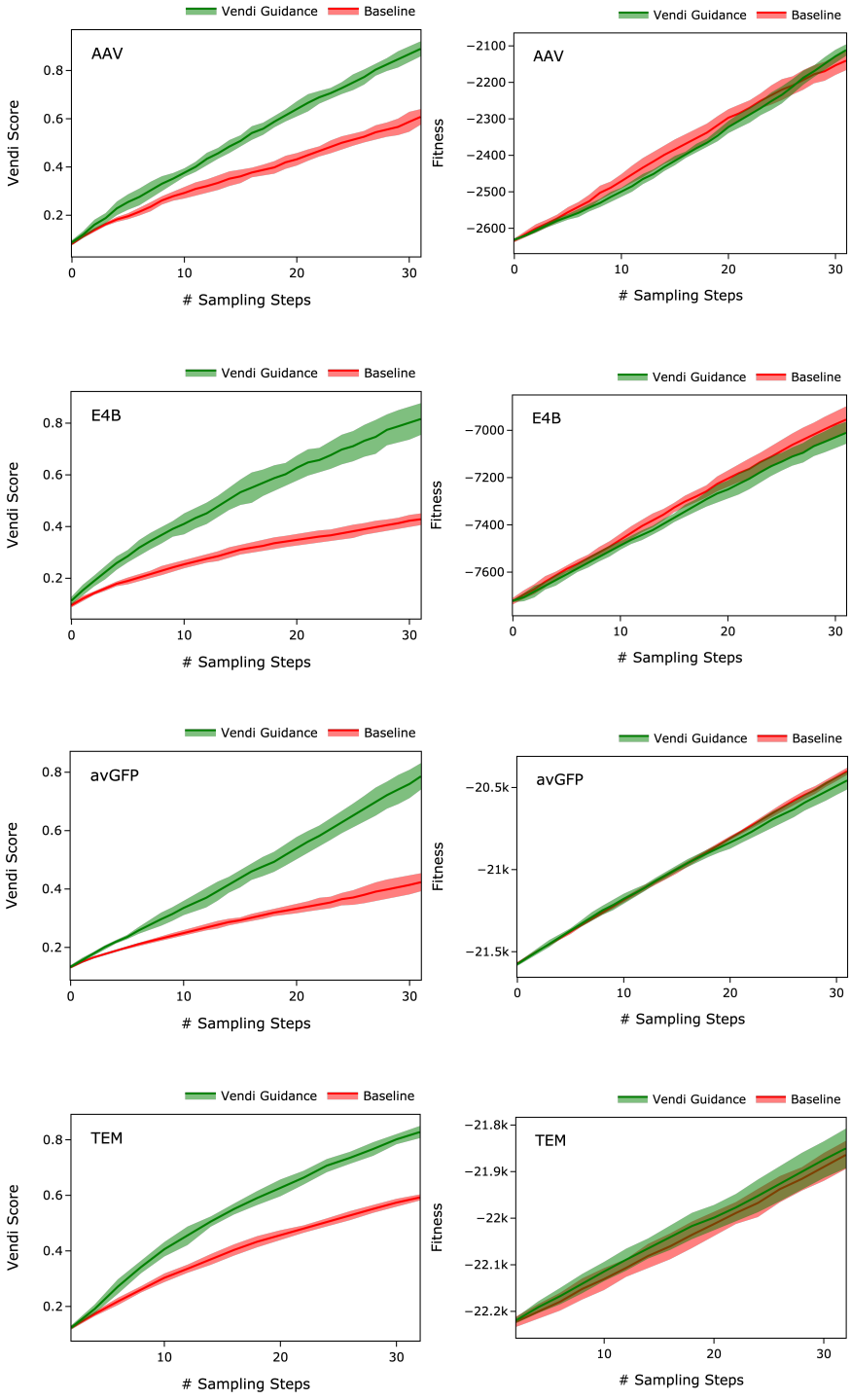

Figure 4.1: Vendi Guidance is able to increase the diversity of sampled sets without deteriorating their fitness. We present the evolution of the (normalized) Vendi Score (*left*) and the fitness score (*right*) over sampling steps of the Markov random walk for sampling sets of $N = 96$ sequences. The four sequences, AAV, E4B, avGFP, and TEM are of increasing lengths from 28 to 286 amino acids, demonstrating that the sequence length does not have a noticeable effect on Vendi Guidance.

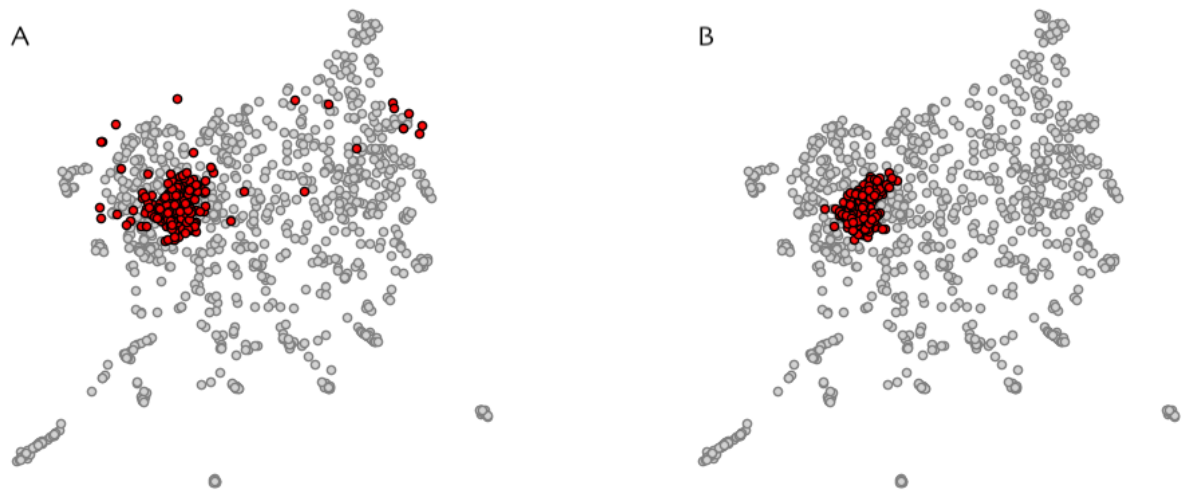

Figure 4.2: Generated protein sequences with Vendi Guidance are more diverse and explore new areas of the local GFP landscape. (A) UMAP of EvoDiff embeddings of natural sequences from a UniRef50 sequence alignment (grey, $N = 986$) and from a Vendi Guidance sampled sequence set (red, $N = 192$). (B) UMAP of EvoDiff embeddings of natural sequences from the same UniRef50 sequence alignment (grey, $N = 986$) and from a Baseline sampled sequence set (red, $N = 192$).

Sample Batch Size All the results we present are for sampling sets of $N = 96$ sequences, a common well plate size, unless otherwise stated. We also analyze our results for $N = 48$ and $N = 192$ and do not observe any noticeable effect from varying the sample batch size. We assume that $N = 192$ is large enough for the scope of this problem.

Number of Sampling Steps For all experiments we choose to sample for 32 steps since it is sufficient to demonstrate the effectiveness of Vendi Guidance and keeps the number of total edits to the starting sequences within a reasonable range. In a true protein design setting, the number of sampling steps would be chosen with respect to the edit budget, if there is one. Since the number of edits is not necessarily evenly distributed over all sequences in the sampled set, this parameter could be adaptively chosen during the sampling process as edits are made. For example, when a maximal number of edits are made in a subset of the sequences, the sampling stops.

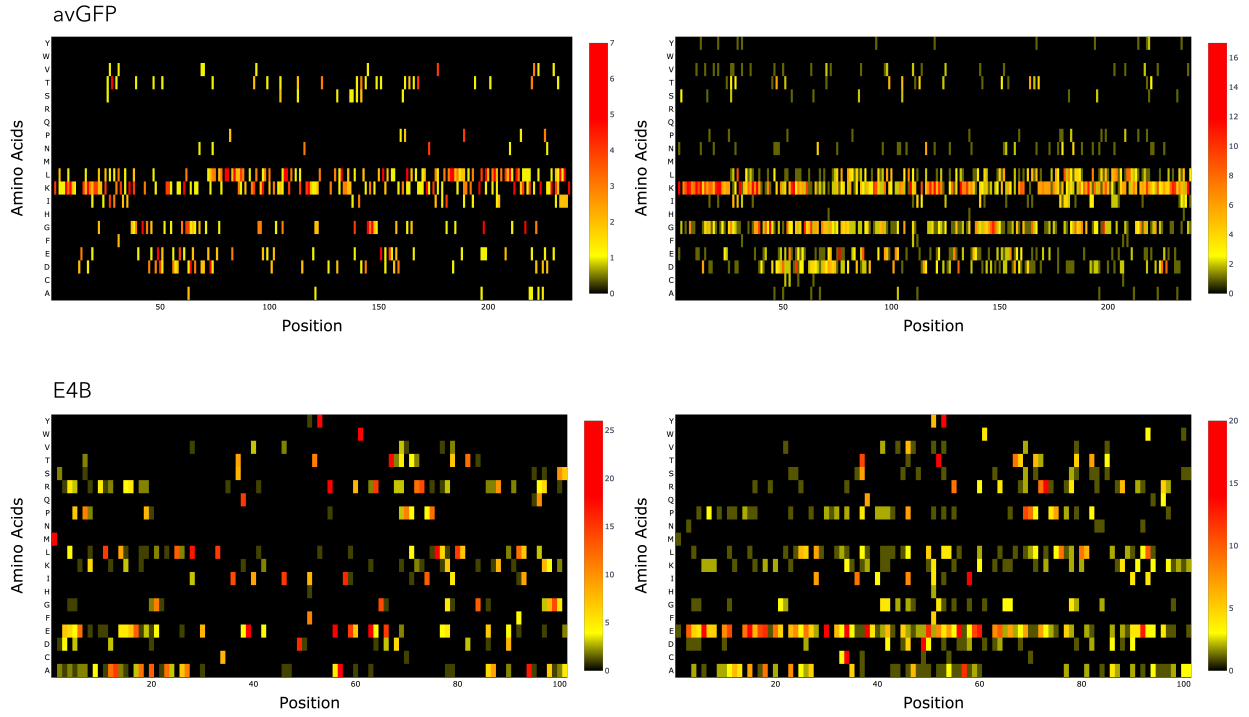

Figure 4.3: Vendi Guidance proposes novel mutations and a smoother distribution of mutations across sampled sequences. We present heatmaps of the proposed mutations within final sets of $N = 96$ sampled sequences from the Baseline sampling method (*left*) and from Vendi Guidance (*right*) with the avGFP protein (*top*) and the E4B protein (*bottom*).

4.2 Results

We present our main results, the evolution of Vendi Score and sequence fitness over sampling steps, in Figure 4.1. We show these results for all four proteins, each of varying lengths. We can observe that Vendi Guidance increases the diversity of the sampled set as refinements are made, while the fitness scores do not deviate much from the Baseline. This confirms that Vendi Guidance is able to generate a set of more diverse sequences without having to trade off fitness, a common dilemma in score-guided sampling methods. We also demonstrate that the length of the sequence does not affect the performance of Vendi Guidance.

4.2.1 Diverse Sampled Sets

We observe the impact of Vendi Guidance on the final sampled sets by analyzing the distribution of proposed mutations and visualizing the embeddings in 2D via uniform manifold approximation and projection (UMAP).

Mutation Distribution As shown in Figure 4.3, Vendi Guidance proposes new mutations and presents a generally smoother distribution of mutations across sampled sequences. Naturally, we do not observe a uniform distribution of mutations from the Vendi Guidance sampled sequences, confirming that the model's distribution is still valued in the guided denoising process. These results suggests that Vendi Guidance effectively pushes the sampling process towards new areas of the sequence space, possibly finding novel, high-fitness mutations. If the sequence space has unbalanced modes, I.I.D. sampled sequences would typically favor larger modes. However, guidance with the Vendi Score shifts the sampling to find different less dominant modes.

UMAP To evaluate sequence coverage, we compared EvoDiff-Seq embeddings from the sampled sequence sets of size $N = 192$ with embeddings from a UniRef50 sequence alignment of size 986. This effectively evaluates sequence coverage within the local area of the sequence landscape we are interested in. The embeddings were visualized in 2D using UMAP, fitted to the sequence alignment data with `n_neighbors=50`. This `n_neighbors` hyperparameter was chosen to emphasize more local similarities over global ones. As observed in Figure 4.2, sampled sequences from the Baseline method cluster within the same dense region of the local sequence landscape, confirming the common pitfall of low-N I.I.D. sampling, which tends to generate similar items. In comparison, Vendi Guidance generates more sequences from different areas of the sequence space. We note that certain sequences cover regions not represented in the sequence alignment, indicating that Vendi Guidance is potentially able to discover more novel sequences.

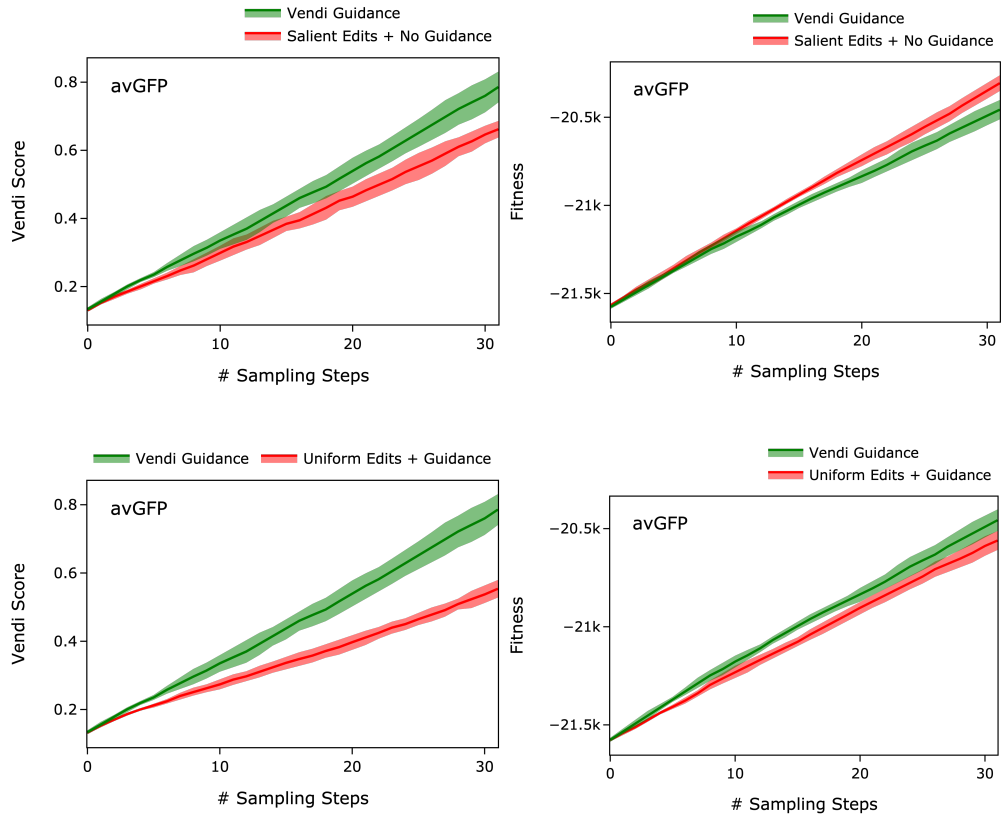

Figure 4.4: Choosing salient edit positions has a larger effect on diversity than the Vendi Score gradient guidance. We present ablations of the gradient-guided denoising (*top*) and of edit position selection (*bottom*) on the Vendi Score (*left*) and fitness (*right*) of the avGFP protein. Guidance has a minimal negative impact on sequence fitness while salient edits have a minimal positive impact on sequence fitness.

4.3 Ablations

We independently study the effect of edit position selection and gradient-guided denoising, which we present in Figure 4.4 with the avGFP protein and in Figure 4.5 with the E4B protein. To isolate the effect of choosing salient edit positions, we choose a step size of 0. This effectively does not perform any update step to the inner representations and the model distributions are left intact. To isolate the effect of the gradient-guided denoising, we uniformly choose edit positions. With the avGFP protein, we see that the effect of the saliency edits alone has a much greater impact on the diversity of the sampled set than the gradient-guided denoising alone. Choosing edit positions alone does not go against the model's distribution in any way, making this method alone quite advantageous. This

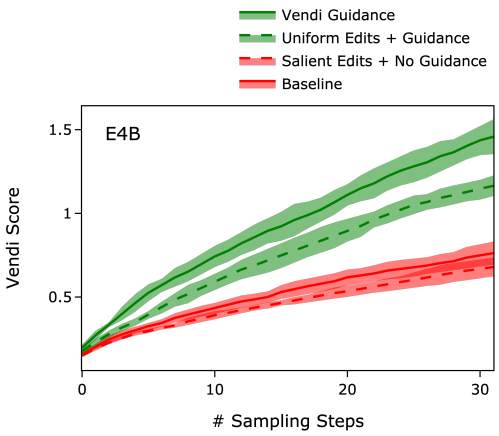

Figure 4.5: Choosing salient edit positions can only have a positive effect on diversity when paired with gradient guidance. We present ablations of gradient-guided denoising and edit position selection for the E4B protein. We observe that choosing salient edit positions alone does not improve diversity, but when paired with gradient guidance improves diversity.

eliminates the common pitfall of degrading sample fitness with score-guided methods. This is confirmed when observing fitness scores. We see that the salient edits alone slightly improve the fitness, whereas the gradient guidance alone slightly worsens the fitness. The latter is a direct consequence of guidance methods with categorical distributions since most changes will directly oppose the model log-likelihoods. In Figure 4.5, we also observe that the salient edit positions only have a positive impact when paired with the gradient guidance. This suggests that the model is able to capture some degree of diversity which is not directly found in the model log-likelihood. However, the guidance method is able to push the model's inner representations towards this diversity. These results indicate that the difficulty of discovering more diverse sequences varies based on the starting sequence. This variability is expected due to the complex nature of the protein landscape.

Chapter 5

Conclusion

5.1 Discussion

In this thesis, we highlight the importance of developing set sampling methods that incorporate set diversity, particularly in the context of protein design. Given the high cost of experimental validation, one objective of protein design is to develop methods that optimize sample efficiency. Independent sequence sampling methods often require sampling a large number of sequences to achieve a diverse set. This challenge is amplified by gradient guidance methods, which tend to restrict sampling to certain prevalent modes of the data distribution.

To address these limitations, we introduce Vendi Guidance, a diversity-guided sampling method for discrete diffusion models. To the best of our knowledge, Vendi Guidance is the first method that samples sequences dependently, using a set function constraint within a discrete denoising process. This approach has many potential applications for sampling methods that aim to couple samples together. In our work, we use the Vendi Score as our set function, although this method is adaptable to any differentiable set function defined on the inner representations of the generative model. Furthermore, we develop our method in line with recent advancements in sampling methods for protein design, making it versatile

and easily integrated into existing frameworks.

Our results demonstrate that Vendi Guidance can directly sample more diverse sets of a fixed size N without compromising the sequences' fitness. We observe that sequences sampled with Vendi Guidance propose novel mutations and cover different regions of the local sequence space of interest. This capability is crucial for protein design, which depends on efficiently exploring diverse regions of the sequence space within fixed batch sizes.

5.2 Limitations

Parameters Vendi Guidance has multiple sensitive parameters that need to be optimally calibrated to obtain desired results. These parameters are dependent on certain constraints of the protein design problem, such as edit budget, invariant positions, and the defined objective functions. An important parameter in this setup is the temperature of the softmax function applied to the saliency maps. When the temperature is set too low, the method will repeatedly select the same edit positions, thus blocking the sampling process. Conversely, setting the temperature too high will default to a uniform distribution, undermining the effectiveness of the guidance. Furthermore, optimizing the gradient guidance parameters is essential to find an optimal balance between the diversity and fitness of the protein sequences. Fortunately, a simple grid search is usually sufficient to finetune these parameters.

Saliency Maps Saliency maps are often known to be unreliable due to their sensitivity to model parameters and training conditions. Typically, these maps are generated based on gradient analysis, but this approach can be noisy. As a result, newer methods have been developed to mitigate this noise. Given the effect of choosing salient edit positions in this work, an interesting analysis could be made to compare various saliency map methods. However, certain methods are known to be more computationally expensive, which could become a disadvantage in this iterative refinement process.

5.3 Future Work

There are many directions for further exploration. In this work, we provide a proof of concept by benchmarking against sampling from EvoDiff-Seq, a model known for its generative capabilities for protein design. However, there are many more concrete protein design problems this method could be evaluated on. For example, one could evaluate Vendi Guidance within the LaMBO-2 optimization process and include empirical validations to evaluate the effect of improving sample efficiency over rounds.

Another direction of future improvement is with respect to the denoising process. Currently, the diffusion process employs a masking or absorbing state, which does not reveal any particular relationships between tokens. The primary role of the mask token is to distinguish between corrupted and original tokens. Empirical research indicates that certain amino acids are more or less similar in terms of their impact on structure or functionality. An interesting direction for future research could involve incorporating amino acid similarity, either by integrating this within the definition of the diversity score or within the denoising step itself.

Appendix A

A.1 Parameters

Sample Batch Size N	$48, 96, 192$
Edit Budget N	$N \times (0.01)L, N \times (0.05)L$
Sampling Steps	32
Saliency Temp	$0.5, 0.7, 1.0, 1.5$
Vendi q	$0.1, 0.5, 1.0, 5.0, \infty$
Step Size	$0.1, 0.5, 1.0$
Stability Coef γ	$0.001, 0.01, 0.1, 1.0$
Number of Guidance Steps K	$5, 10$

Table A.1: Vendi Guidance hyperparameter settings.

Bibliography

[1] S. Alamdari, N. Thakkar, R. van den Berg, A. Lu, N. Fusi, A. Amini, and K. Yang. Protein generation with evolutionary diffusion: Sequence is all you need. biorxiv 2023. *Google Scholar*.

[2] N. Anand, R. Eguchi, and P.-S. Huang. Fully differentiable full-atom protein backbone generation. 2019.

[3] C. B. Anfinsen. Principles that govern the folding of protein chains. *Science*, 181(4096):223–230, 1973.

[4] C. Angermueller, D. Dohan, D. Belanger, R. Deshpande, K. Murphy, and L. Colwell. Model-based reinforcement learning for biological sequence design. In *International conference on learning representations*, 2019.

[5] F. H. Arnold. Design by directed evolution. *Accounts of chemical research*, 31(3):125–131, 1998.

[6] J. Austin, D. D. Johnson, J. Ho, D. Tarlow, and R. Van Den Berg. Structured denoising diffusion models in discrete state-spaces. *Advances in Neural Information Processing Systems*, 34:17981–17993, 2021.

[7] M. Baek et al. Efficient and accurate prediction of protein structures and interactions using rosettafold. *Acta Crystallogr. A Found. Adv.*, 78:a235, 2022.

[8] A. Barducci, G. Bussi, and M. Parrinello. Well-tempered metadynamics: a smoothly converging and tunable free-energy method. *Physical review letters*, 100(2):020603, 2008.

[9] L. Bedian, A. M. Villalba-Rodríguez, G. Hernández-Vargas, R. Parra-Saldivar, and H. M. Iqbal. Bio-based materials with novel characteristics for tissue engineering applications–a review. *International journal of biological macromolecules*, 98:837–846, 2017.

[10] S. Biswas, G. Khimulya, E. C. Alley, K. M. Esvelt, and G. M. Church. Low-n protein engineering with data-efficient deep learning. *Nature methods*, 18(4):389–396, 2021.

[11] S. K. Burley, H. M. Berman, G. J. Kleywegt, J. L. Markley, H. Nakamura, and S. Velankar. Protein data bank (pdb): the single global macromolecular structure archive. *Protein crystallography: methods and protocols*, pages 627–641, 2017.

[12] O. Caldararu, R. Mehra, T. L. Blundell, and K. P. Kepp. Systematic investigation of the data set dependency of protein stability predictors. *Journal of Chemical Information and Modeling*, 60(10):4772–4784, 2020.

[13] E. Castro, A. Godavarthi, J. Rubinfien, K. Givechian, D. Bhaskar, and S. Krishnaswamy. Transformer-based protein generation with regularized latent space optimization. *Nature Machine Intelligence*, 4(10):840–851, 2022.

[14] U. Consortium. The universal protein resource (uniprot). *Nucleic acids research*, 36(suppl_1):D190–D195, 2007.

[15] G. Corso, Y. Xu, V. De Bortoli, R. Barzilay, and T. Jaakkola. Particle guidance: non-iid diverse sampling with diffusion models. *arXiv preprint arXiv:2310.13102*, 2023.

[16] S. Dathathri, A. Madotto, J. Lan, J. Hung, E. Frank, P. Molino, J. Yosinski, and R. Liu. Plug and play language models: A simple approach to controlled text generation. *arXiv preprint arXiv:1912.02164*, 2019.

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

[18] P. Dhariwal and A. Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.

[19] S. F. Elena, I. Fragata, S. Matuszewski, et al. The utility of fitness landscapes and big data for predicting evolution. *Heredity*, 2018.

[20] A. Elnaggar, M. Heinzinger, C. Dallago, G. Rehawi, Y. Wang, L. Jones, T. Gibbs, T. Feher, C. Angerer, M. Steinegger, et al. Prottrans: Toward understanding the language of life through self-supervised learning. *IEEE transactions on pattern analysis and machine intelligence*, 44(10):7112–7127, 2021.

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

[22] D. Friedman and A. B. Dieng. The vendi score: A diversity evaluation metric for machine learning. *Transactions on Machine Learning Research*, 2023.

[23] M. Ghazvininejad, O. Levy, Y. Liu, and L. Zettlemoyer. Mask-predict: Parallel decoding of conditional masked language models. *arXiv preprint arXiv:1904.09324*, 2019.

[24] N. Gruver, S. Stanton, N. Frey, T. G. Rudner, I. Hotzel, J. Lafrance-Vanasse, A. Rajpal, K. Cho, and A. G. Wilson. Protein design with guided discrete diffusion. *Advances in Neural Information Processing Systems*, 36, 2024.

[25] Z. He, B. P. Majumder, and J. McAuley. Detect and perturb: Neutral rewriting of biased and sensitive text via gradient-based decoding. *arXiv preprint arXiv:2109.11708*, 2021.

[26] B. Hie, S. Candido, Z. Lin, O. Kabeli, R. Rao, N. Smetanin, T. Sercu, and A. Rives. A high-level programming language for generative protein design. *bioRxiv*, pages 2022–12, 2022.

[27] E. Hoogeboom, A. A. Gritsenko, J. Bastings, B. Poole, R. v. d. Berg, and T. Salimans. Autoregressive diffusion models. *arXiv preprint arXiv:2110.02037*, 2021.

[28] C. Hsu, R. Verkuil, J. Liu, Z. Lin, B. Hie, T. Sercu, A. Lerer, and A. Rives. Learning inverse folding from millions of predicted structures. In *International conference on machine learning*, pages 8946–8970. PMLR, 2022.

[29] G. Hummer and J. Köfinger. Bayesian ensemble refinement by replica simulations and reweighting. *The Journal of chemical physics*, 143(24), 2015.

[30] J. Ingraham, V. Garg, R. Barzilay, and T. Jaakkola. Generative models for graph-based protein design. *Advances in neural information processing systems*, 32, 2019.

[31] J. H. Jensen. A graph-based genetic algorithm and generative model/monte carlo tree search for the exploration of chemical space. *Chemical science*, 10(12):3567–3572, 2019.

[32] S. R. Johnson, S. Monaco, K. Massie, and Z. Syed. Generating novel protein sequences using gibbs sampling of masked language models. *bioRxiv*, pages 2021–01, 2021.

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

[34] G. Klebe. Recent developments in structure-based drug design. *Journal of molecular medicine*, 78:269–281, 2000.

[35] N. Krasnogor and D. A. Pelta. Measuring the similarity of protein structures by means of the universal similarity metric. *Bioinformatics*, 20(7):1015–1021, 2004.

[36] H. D. Lagassé, A. Alexaki, V. L. Simhadri, N. H. Katagiri, W. Jankowski, Z. E. Sauna, and C. Kimchi-Sarfaty. Recent advances in (therapeutic protein) drug development. *F1000Research*, 6, 2017.

[37] M. Lee, L. F. Vecchietti, H. Jung, H. Ro, M. Cha, and H. M. Kim. Protein sequence design in a latent space via model-based reinforcement learning. 2022.

[38] V. I. Levenshtein et al. Binary codes capable of correcting deletions, insertions, and reversals. In *Soviet physics doklady*, volume 10, pages 707–710. Soviet Union, 1966.

[39] X. Li, J. Thickstun, I. Gulrajani, P. S. Liang, and T. B. Hashimoto. Diffusion-lm improves controllable text generation. *Advances in Neural Information Processing Systems*, 35:4328–4343, 2022.

[40] Z. Lin, H. Akin, R. Rao, B. Hie, Z. Zhu, W. Lu, N. Smetanin, R. Verkuil, O. Kabeli, Y. Shmueli, et al. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637):1123–1130, 2023.

[41] M. Littmann, M. Heinzinger, C. Dallago, T. Olenyi, and B. Rost. Embeddings from deep learning transfer go annotations beyond homology. *Scientific reports*, 11(1):1160, 2021.

[42] W. Mandecki. The game of chess and searches in protein sequence space. *Trends in Biotechnology*, 16(5):200–202, 1998.

[43] E. Marcos and D.-A. Silva. Essentials of de novo protein design: Methods and applications. *Wiley Interdisciplinary Reviews: Computational Molecular Science*, 8(6):e1374, 2018.

[44] J. Maynard Smith. Natural selection and the concept of a protein space. *Nature*, 225(5232):563–564, 1970.

[45] S. Mazurenko, Z. Prokop, and J. Damborsky. Machine learning in enzyme engineering. *ACS Catalysis*, 10(2):1210–1223, 2019.

[46] C. Meister, M. Forster, and R. Cotterell. Determinantal beam search. *arXiv preprint arXiv:2106.07400*, 2021.

[47] A. Nguyen, J. Clune, Y. Bengio, A. Dosovitskiy, and J. Yosinski. Plug & play generative networks: Conditional iterative generation of images in latent space. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4467–4477, 2017.

[48] A. Nigam, P. Friederich, M. Krenn, and A. Aspuru-Guzik. Augmenting genetic algorithms with deep neural networks for exploring the chemical space. *arXiv preprint arXiv:1909.11655*, 2019.

[49] P. J. Ogden, E. D. Kelsic, S. Sinai, and G. M. Church. Comprehensive aav capsid fitness landscape reveals a viral gene and enables machine-guided design. *Science*, 366(6469):1139–1143, 2019.

[50] C. A. Orengo, A. D. Michie, S. Jones, D. T. Jones, M. B. Swindells, and J. M. Thornton. Cath–a hierarchic classification of protein domain structures. *Structure*, 5(8):1093–1109, 1997.

[51] A. Pasarkar and A. B. Dieng. Cousins of the vendi score: A family of similarity-based diversity metrics for science and machine learning. *arXiv preprint arXiv:2310.12952*, 2023.

[52] A. P. Pasarkar, G. M. Bencomo, S. Olsson, and A. B. Dieng. Vendi sampling for molecular simulations: Diversity as a force for faster convergence and better exploration. *The Journal of chemical physics*, 159(14), 2023.

[53] K. Peng, Z. Obradovic, and S. Vucetic. Exploring bias in the protein data bank using contrast classifiers. In *Biocomputing 2004*, pages 435–446. World Scientific, 2003.

[54] B. T. Porebski and A. M. Buckle. Consensus protein design. *Protein Engineering, Design and Selection*, 29(7):245–251, 2016.

[55] Z. Ren, J. Li, F. Ding, Y. Zhou, J. Ma, and J. Peng. Proximal exploration for model-guided protein sequence design. In *International Conference on Machine Learning*, pages 18520–18536. PMLR, 2022.

[56] A. Rives, J. Meier, T. Sercu, S. Goyal, Z. Lin, J. Liu, D. Guo, M. Ott, C. L. Zitnick, J. Ma, et al. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *Proceedings of the National Academy of Sciences*, 118(15):e2016239118, 2021.

[57] C. A. Rohl, C. E. Strauss, K. M. Misura, and D. Baker. Protein structure prediction using rosetta. In *Methods in enzymology*, volume 383, pages 66–93. Elsevier, 2004.

[58] Y. Saito, M. Oikawa, H. Nakazawa, T. Niide, T. Kameda, K. Tsuda, and M. Umetsu. Machine-learning-guided mutagenesis for directed evolution of fluorescent proteins. *ACS synthetic biology*, 7(9):2014–2022, 2018.

[59] R. Salomon-Ferrer, D. A. Case, and R. C. Walker. An overview of the amber biomolecular simulation package. *Wiley Interdisciplinary Reviews: Computational Molecular Science*, 3(2):198–210, 2013.

[60] K. S. Sarkisyan, D. A. Bolotin, M. V. Meer, D. R. Usmanova, A. S. Mishin, G. V. Sharonov, D. N. Ivankov, N. G. Bozhanova, M. S. Baranov, O. Soylemez, et al. Local fitness landscape of the green fluorescent protein. *Nature*, 533(7603):397–401, 2016.

[61] A. Schmid, J. Dordick, B. Hauer, A. Kiener, M. Wubbolts, and B. Witholt. Industrial biocatalysis today and tomorrow. *nature*, 409(6817):258–268, 2001.

[62] J. Schymkowitz, J. Borg, F. Stricher, R. Nys, F. Rousseau, and L. Serrano. The foldx web server: an online force field. *Nucleic acids research*, 33(suppl_2):W382–W388, 2005.

[63] V. R. Shanker, T. U. Bruun, B. L. Hie, and P. S. Kim. Inverse folding of protein complexes with a structure-informed language model enables unsupervised antibody evolution. *bioRxiv*, pages 2023–12, 2023.

[64] F. Sievers and D. G. Higgins. Clustal omega. *Current protocols in bioinformatics*, 48(1):3–13, 2014.

[65] S. Sinai, R. Wang, A. Whatley, S. Slocum, E. Locane, and E. D. Kelsic. Adalead: A simple and robust adaptive greedy search algorithm for sequence design. *arXiv preprint arXiv:2010.02141*, 2020.

[66] T. F. Smith and M. S. Waterman. Comparison of biosequences. *Advances in applied mathematics*, 2(4):482–489, 1981.

[67] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pages 2256–2265. PMLR, 2015.

[68] S. Stanton, W. Maddox, N. Gruver, P. Maffettone, E. Delaney, P. Greenside, and A. G. Wilson. Accelerating bayesian optimization for biological sequence design with denoising autoencoders. In *International Conference on Machine Learning*, pages 20459–20478. PMLR, 2022.

[69] L. M. Starita, J. N. Pruneda, R. S. Lo, D. M. Fowler, H. J. Kim, J. B. Hiatt, J. Shendure, P. S. Brzovic, S. Fields, and R. E. Klevit. Activity-enhancing mutations in an e3 ubiquitin ligase identified by high-throughput mutagenesis. *Proceedings of the National Academy of Sciences*, 110(14):E1263–E1272, 2013.

[70] T. N. Starr and J. W. Thornton. Epistasis in protein evolution. *Protein science*, 25(7):1204–1218, 2016.

[71] T. A. Tatusova and T. L. Madden. Blast 2 sequences, a new tool for comparing protein and nucleotide sequences. *FEMS microbiology letters*, 174(2):247–250, 1999.

[72] D. Van Der Spoel, E. Lindahl, B. Hess, G. Groenhof, A. E. Mark, and H. J. Berendsen. Gromacs: fast, flexible, and free. *Journal of computational chemistry*, 26(16):1701–1718, 2005.

[73] R. Verkuil, O. Kabeli, Y. Du, B. I. Wicky, L. F. Milles, J. Dauparas, D. Baker, S. Ovchinnikov, T. Sercu, and A. Rives. Language models generalize beyond natural proteins. *bioRxiv*, pages 2022–12, 2022.

[74] C. Vignac, I. Krawczuk, A. Siraudin, B. Wang, V. Cevher, and P. Frossard. Digress: Discrete denoising diffusion for graph generation. *arXiv preprint arXiv:2209.14734*, 2022.

[75] A. K. Vijayakumar, M. Cogswell, R. R. Selvaraju, Q. Sun, S. Lee, D. Crandall, and D. Batra. Diverse beam search: Decoding diverse solutions from neural sequence models. *arXiv preprint arXiv:1610.02424*, 2016.

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

[77] K. E. Wu, K. K. Yang, R. van den Berg, S. Alamdari, J. Y. Zou, A. X. Lu, and A. P. Amini. Protein structure generation via folding diffusion. *Nature Communications*, 15(1):1059, 2024.

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

[79] Z. Wu, S. J. Kan, R. D. Lewis, B. J. Wittmann, and F. H. Arnold. Machine learning-assisted directed protein evolution with combinatorial libraries. *Proceedings of the National Academy of Sciences*, 116(18):8852–8858, 2019.

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

[81] K. Yue and K. A. Dill. Inverse protein folding problem: designing polymer sequences. *Proceedings of the National Academy of Sciences*, 89(9):4163–4167, 1992.