nature portfolio

# Peer Review File

# Property guidance for protein sequence generative models with ProteinGuide

Corresponding Author: Dr Jennifer Listgarten

Parts of this Peer Review File have been redacted as indicated as we could not obtain permission to publish the reports from one of the peer reviewers.

**This file contains all reviewer reports in order by version, followed by all author rebuttals in order by version.**

Version 0:

Reviewer comments:

Reviewer #1

(Remarks to the Author)
The manuscript presents "ProteinGuide," a framework designed to enable "plug-and-play" conditioning of pre-trained protein generative models, such as ProteinMPNN and ESM3. The authors argue that this approach avoids the high computational cost of re-training large models. The core idea is to use an auxiliary regression model or classifier to guide the generation process toward user-specified properties like enhanced stability or specific enzyme functions. The authors demonstrate ProteinGuide's utility through three in-silico tasks and validate it with wet-lab experiments, successfully designing adenine base editor (ABE) sequences with high activity.

I have several major concerns regarding the novelty, motivation, and experimental rigor of the study.

Major Concerns
1. Lack of Novelty and Omission of Critical Prior Art
A primary concern is the novelty of the proposed "plug-and-play" framework. The methodology appears to be similar to that presented in "Diffusion Language Models Are Versatile Protein Learners" (Wang et al., ICML 2024), hereafter referred to as DPLM. The DPLM paper clearly introduces the same "plug-and-play" guidance capability for protein language models, employing a nearly identical conceptual and methodological approach (see Fig 1C, page 5, and page 9 of the DPLM paper). In my view, ProteinGuide is based on the same underlying technique without offering a significant or novel technical contribution. The authors of DPLM also demonstrated conditioning on secondary structure. Applying this known technique to a different pre-trained model (ESM3 and ProteinMPNN) or a new property does not, in itself, constitute a major advance.

The author should also explicitly clarify the difference with DPLM and compare it as a key baseline.

2. Weakened Motivation and Missing Baselines

The central motivation for ProteinGuide—that re-training large protein language models (pLMs) is expensive—is questionable in the context of current practices.
• Comparison with LoRA: The authors should compare their method against parameter-efficient fine-tuning (PEFT) techniques, such as Low-Rank Adaptation (LoRA). LoRA allows for efficient fine-tuning of even billion-parameter models with modest computational resources. It is necessary to demonstrate whether the "plug-and-play" paradigm offers comparable or superior performance to LoRA fine-tuning. If ProteinGuide's performance is significantly worse, its practical value to the biological community may be limited, as the efficiency gains would be offset by a loss in performance.
• Context of pLM Scaling: The manuscript's seems to have a similar motivation with the Large Language Model (LLM) community, where fine-tuning models with hundreds of billions of parameters is a major challenge. However, this argument is not very applicable to the pLM (protein language model) field. Several literature suggests that the benefits of scaling pLMs diminish beyond a certain point (e.g., 650M-3B parameters), and there is no community consensus that larger pLMs necessarily yield better generative performance. In fact, fine-tuning even a 10-billion-parameter pLM with LoRA is a tractable task, which is fundamentally different from the challenges faced in the LLM field. This distinction weakens the paper's core premise that avoiding retraining is a paramount challenge for the pLM community. Furthermore, ProteinMPNN is a small

model, making it an insufficient example to back up the claim of avoiding high finetuning costs.

The authors should include a direct performance comparison with LoRA fine-tuning or top-layer finetuning as a crucial baseline.

3. Omission of a Simple yet Powerful Baseline (Post-Hoc Filtering)

The manuscript overlooks a straightforward yet highly relevant baseline: post-hoc filtering. This method would involve generating a large number of sequences (e.g., N=10,000) using the unguided generative model (ESM3 or ProteinMPNN) and then using the same classifier or regression model to rank these sequences. The top-ranked candidates from this filtered set could potentially achieve similar or even better performance than those generated by ProteinGuide. The absence of this simple baseline makes it difficult to assess the true value added by the proposed guidance mechanism.

4. Insufficient Scope of In-Silico Validation
The paper show the effectiveness of ProteinGuide on only three in-silico tasks. Three task evaluation are not sufficient to convincingly prove the general applicability and robustness of the method. For a high-impact journal like Nature Biotechnology (NBT), a more comprehensive validation across a wider range of proteins, properties, and PLM models would be expected. The current workload appears below the typical threshold for NBT.

Minor Concerns
5. Clarity of Presentation
Method Description: The main text currently lacks a sufficient and clear description of the ProteinGuide framework itself. Much of the manuscript is dedicated to demonstrating its applications, but the "how" is not adequately explained for the reader.

Figure 1: The main schematic (Figure 1), which illustrates the equivalence between different generative model training objectives, describes a concept that has been previously well studied in the AI literature. While useful for context, it may not be suitable as the primary figure of the paper. A more impactful main figure would be a detailed diagram illustrating the specific architecture and information flow of the ProteinGuide framework.

Methods Section: The Methods section contains extensive theoretical explanations and formula derivations. While rigor is appreciated, these elements do not appear to be novel contributions in themselves and seem disconnected from the specific context of protein engineering.

Overall Conclusion
In summary, the authors tackle an ‘interesting’ problem of conditioning protein generative models. The wet-lab validation of ABEs is a commendable and interesting result. However, the work in its current form suffers from major concerns regarding its technical novelty, the strength of its motivating premise, the omission of critical baselines.

While the project represents an interesting attempt, the overall contribution does not appear to meet the high bar for novelty and significant impact expected for a publication in Nature Biotechnology. The work does not appear to have led to substantial advances in either the PLM or protein engineering fields. Addressing the major points raised above—particularly through direct and rigorous comparison with existing methods like DPLM, LoRA-based fine-tuning, and post-hoc filtering—would be essential to substantiate the claims of the paper.

(Remarks on code availability)

Reviewer #2

(Remarks to the Author)
In this manuscript by Xiong and colleagues, the authors propose ProteinGuide, a framework to guide protein generation toward specific properties. The main idea is to couple a trained property predictor with a general generative model: the generative model constrains design to the natural protein space, reducing the vast search space, while the predictor biases generation toward proteins with desired properties. The method of performing guidance for diffusion and flow-matching models was presented in the authors’ prior work (ref. 16). Here, they extend the method to a class of masked language models (MLMs, ESM3) and order-agnostic autoregressive models (OA-AR, ProteinMPNN). Their results showed that ProteinGuide can help generate proteins with specific properties compared to unguided model. This method is important in molecular biology and protein engineering, I have some concerns regarding the evaluation and applications.

Major comments:
1. While the results demonstrate that guidance can be applied, it is unclear how effective the method truly is. The method was only compared against the unguided model—which, by design, cannot generate proteins with specific properties. It is important to compare with existing baseline approaches, as the authors acknowledged in “Additional discussion of related work”. A pre-print (https://arxiv.org/abs/2505.15093v1) that the authors cited compared different methods for protein fitness, it’s unclear how different methods work for the applications in this manuscript.

2. Another class of Plug and Play approach is in silico directed evolution (https://arxiv.org/abs/2212.09925). This strategy also integrates a general model with a property predictor and can be applied to protein design tasks presented in the manuscript. What advantages does ProteinGuide offer compared to this approach?

3. For these applications, how would ProteinGuide perform if the property predictors were trained with less data? In real-world scenarios, we often lack sufficient labeled data to train highly accurate property predictors, so understanding the method's robustness to limited training data would be important.

4. For the protein stability design task. ProteinMPNN is capable of zero-shot prediction of mutation effect (https://github.com/OATML-Markslab/ProteinGym/tree/main/proteingym/baselines/protein_mpnn), how do the design results relate to its zero-shot performance?

5. For the enzyme-class guided sequence generation, ProteinGuide outperforms unguided ESM3 only at high mask ratios (≥80% of residues masked). It is unclear why guided and unguided models achieve similar performances at lower mask levels (<80%). 5. Additionally, it is important to report sequence identity of designed proteins compared to the original sequence at different mask levels.

6. In addition, I recommend using independent structure predictors to evaluate pLDDT values, rather than relying on ESM3 itself, because it's unclear whether ESM3 pLDDT is biased toward its own generations. Beyond CLEAN classification, it would be more informative to compare the predicted structures with well-characterized enzymes of the same class, with particular attention to the structure of active-site.

Minor comments:
7. How do the designed ABEs compared to the most active ABEs currently used in the field or designed by others?

8. It's important to discuss how to select guidance strength parameter in equation 29.

(Remarks on code availability)

Author Rebuttal letter:

Overview of changes
We thank the reviewers for their feedback on "Guide your favorite protein sequence generation model", now re-titled as "ProteinGuide: On-the-fly property guidance for protein sequence generative models". Over the past five months, we have expanded our brief communication into a full-length article. The updated version has a new technical contribution and experimental results, along with a clearer exposition of the technical materials and their relationship to prior work.
Note that we now refer to our method as "on-the-fly" conditioning of a pre-trained model, as opposed to "plug-and-play" which has come to mean techniques that use a generative model and predictive model together, even when the method requires training of a generative model. In contrast,"on-the-fly" conditioning does not require any training of a generative model.

Our key new results are:

1. We have added new wet-lab validation showing that the adenine base editor designed with one round of ProteinGuide has activity surpassing variants found after seven rounds of directed evolution, when using the same starting protein sequence.

2. All in silico experiments now include systematic comparison to the most representative alternative approaches used in practice, namely post-hoc filtering and fine-tuning.

3. Our original submission focused on interpolative protein design tasks, wherein one aims to sample novel sequences with properties similar to the training data. However, many protein engineering campaigns seek to engineer sequences with functional properties beyond the range observed in the training data, a so-called extrapolative task. Our new submission focuses on extrapolative tasks, now including three new in silico experiments in this regime. We highlight that ProteinGuide is of particular use in this regime, as opposed to the easier task of interpolative design, where other methods may prove equally useful.

4. We newly demonstrate that ProteinGuide can be used for multi-property conditioning, and in particular to extrapolate beyond the training data when optimizing two properties that are in tension with one another. We refer to such settings as Pareto-extrapolative. We designed the PbrR transcription factor to better bind Pb (II) ions while simultaneously reducing its affinity for the off-target Zn (II) ion.

5. We include a new theoretical result proving that the sampling distributions of discrete flow matching and any-order autoregressive models are equivalent. In doing so, we also provide a

general technique for converting sampling algorithms between the two model classes. This new theoretical result directly enabled the development of a new, more efficient exact guidance algorithm, which we applied to challenging problems such as multi-property conditioning.

Also note that we have moved some of the experiments from the original submission to the Supplementary Information so as to better have the main manuscript expose the three types of problems: interpolative, extrapolative, and multi-property pareto-optimal. Specifically, the Protein-MPNN stability guidance experiment now appears in Figure 2, while the ESM3 enzyme-class and fold-class guidance experiments are presented in Supplementary Figures S6 and S7, respectively. Figures 3 and 4 present new experiments focused on extrapolative and multi-property Pareto-optimal design.
We now address each of the specific reviewer comments, point-by-point:

Responses to Reviewer #1

Reviewer #1
Remarks to the Author

The manuscript presents “ProteinGuide,” a framework designed to enable “plug-and-play” conditioning of pre-trained protein generative models, such as ProteinMPNN and ESM3. The authors argue that this approach avoids the high computational cost of re-training large models. The core idea is to use an auxiliary regression model or classifier to guide the generation process toward user-specified properties like enhanced stability or specific enzyme functions. The authors demonstrate ProteinGuide’s utility through three in-silico tasks and validate it with wet-lab experiments, successfully designing adenine base editor (ABE) sequences with high activity.
I have several major concerns regarding the novelty, motivation, and experimental rigor of the study.

Major Concerns
1. Lack of Novelty and Omission of Critical Prior Art
A primary concern is the novelty of the proposed “plug-and-play” framework. The methodology appears to be similar to that presented in “Diffusion Language Models Are Versatile Protein Learners” (Wang et al., ICML 2024), hereafter referred to as DPLM. The DPLM paper clearly introduces the same “plug-and-play” guidance capability for protein language models, employing a nearly identical conceptual and methodological approach (see Fig 1C, page 5, and page 9 of the DPLM paper). In my view, ProteinGuide is based on the same underlying technique without offering a significant or novel technical contribution. The authors of DPLM also demonstrated conditioning on secondary structure. Applying this known technique to a different pre-trained model (ESM3 and ProteinMPNN) or a new property does not, in itself, constitute a major advance.
The author should also explicitly clarify the difference with DPLM and compare it as a key baseline.

Response.
We appreciate the reviewer pointing out the DPLM paper from Wang et al. [1] as related work. DPLM uses a method called DiGress, originally proposed by Vignac et al. [2] for property conditioning. In our earlier work [3], we discussed the methodological difference between DiGress and our method in detail (e.g. Appendix section E.2 in Nisonoff et al. [3]). Therein, we also performed comprehensive empirical comparisons to DiGress in three experiments across different domains, showing that our method generally outperformed DiGress. Consequently, we do not compare to DiGress again in this work. We have added a more detailed discussion explaining the distinction between ProteinGuide, DiGress, and DPLM (see Supplementary Information section “Detailed discussion on related work”).

2. Weakened Motivation and Missing Baselines
The central motivation for ProteinGuide—that re-training large protein language models (pLMs) is expensive—is questionable in the context of current practices.

• Comparison with LoRA: The authors should compare their method against parameter-efficient fine-tuning (PEFT) techniques, such as Low-Rank Adaptation (LoRA). LoRA allows for efficient fine-tuning of even billion-parameter models

with modest computational resources. It is necessary to demonstrate whether the "plug-and-play" paradigm offers comparable or superior performance to LoRA fine-tuning. If ProteinGuide's performance is significantly worse, its practical value to the biological community may be limited, as the efficiency gains would be offset by a loss in performance.
• Context of pLM Scaling: The manuscript's seems to have a similar motivation with the Large Language Model (LLM) community, where fine-tuning models with hundreds of billions of parameters is a major challenge. However, this argument is not very applicable to the pLM (protein language model) field. Several literature suggests that the benefits of scaling pLMs diminish beyond a certain point (e.g., 650M-3B parameters), and there is no community consensus that larger pLMs necessarily yield better generative performance. In fact, fine-tuning even a 10-billion-parameter pLM with LoRA is a tractable task, which is fundamentally different from the challenges faced in the LLM field. This distinction weakens the paper's core premise that avoiding retraining is a paramount challenge for the pLM community. Furthermore, ProteinMPNN is a small model, making it an insufficient example to back up the claim of avoiding high finetuning costs.

The authors should include a direct performance comparison with LoRA fine-tuning or top-layer finetuning as a crucial baseline.

Response.
We thank the reviewer for pointing this out. We now include LoRA fine-tuning (FT) and compute-matched post-hoc filtering (unguided) baselines for all in silico experiments in the paper, including in a number of new experiments which we will detail below. The experiments and their baseline configurations are summarized in the section "Overview of in silico experiments", with full details described in the Methods.

Regarding the reviewer's second point, we believe compatibility with larger models is only one of the threefold benefits of ProteinGuide over finetuning. First, ProteinGuide is a statistically rigorous conditional sampling method. It combines the information from a trusted pre-trained model and predictive model in a statistically correct manner, and in so doing, avoids fine-tuning's tendency to overfit the training data and "forget" information captured by the pretrained model (e.g., Figure 3, Figure 4, and Supplementary Figure S5). Second, ProteinGuide's on-the-fly conditioning is more flexible in that it can be used at sampling time to combine different predictive models with a baseline generative model, whereas fine-tuning must retrain a generative model with each new predictive+generative model combination. Moreover, because our method performs correct statistical conditioning, the manner in which to use it for conditioning on two properties follows from Bayes' rule. We explore this empirically in the Pareto sampling multi-property experiment (see main text Fig. 4d). Third, ProteinGuide only requires access to pre-trained model outputs

and not their parameters. Therefore, it can be used with closed-source proprietary models that have API-access only (e.g. the large ESM family models and the new ProGen3 models).

3. Omission of a Simple yet Powerful Baseline (Post-Hoc Filtering)
The manuscript overlooks a straightforward yet highly relevant baseline: post-hoc filtering. This method would involve generating a large number of sequences (e.g., N=10,000) using the unguided generative model (ESM3 or ProteinMPNN) and then using the same classifier or regression model to rank these sequences. The top-ranked candidates from this filtered set could potentially achieve similar or even better performance than those generated by ProteinGuide. The absence of this simple baseline makes it difficult to assess the true value added by the proposed guidance mechanism.

Response.
Indeed, we agree, and have now added post-hoc filtering baselines to all in silico experiments.

4. Insufficient Scope of In-Silico Validation
The paper show the effectiveness of ProteinGuide on only three in-silico tasks. Three task evaluation are not sufficient to convincingly prove the general applicability and robustness of the method. For a high-impact journal like Nature Biotechnology (NBT), a more comprehensive validation across a wider range of proteins, properties, and PLM models would be expected. The current workload appears below the typical threshold for NBT.

Response.
Indeed: we have added four new experiments to the paper. We intend for these to address several

of the reviewer's concerns. Each experiment covers a different protein and biological property. We also added an experiment using the sequence only ESM C model as an additional PLM. To test robustness to the availability of fewer labeled training points, we include in silico experiments using between one to two thousand data points, similar to the number used in our wet-lab validation experiments. However, we'd like to point out that others have used the preliminary version of our work finding success with even 200 sequences (e.g., Fig. 5 in Yang et al. [4]). These experiments also target the more challenging regimes of extrapolative design and multi-property design, described in the overview section. Our updated manuscript has a total of seven in silico experiments which are summarized in section "Overview of in silico experiments", with Supplementary Table S1 highlighting the key points.

Minor Concerns
5. Clarity of Presentation
Method Description: The main text currently lacks a sufficient and clear description of the ProteinGuide framework itself. Much of the manuscript is dedicated to demonstrating its applications, but the "how" is not adequately explained for the reader.

Response.
Indeed, the previous draft overlooked an in-depth discussion of the "how" in the main text, owing to our original intent to present this as a Brief Communication. In the present expanded version, we have a more complete introduction and have also included an overview of our method at the beginning of the Results section. We did our best to strike a balance between providing intuition and having some but not too much technical detail for a general audience, leaving the technical material to the Material and Methods, which are also better organized now.

Figure 1: The main schematic (Figure 1), which illustrates the equivalence between different generative model training objectives, describes a concept that has been previously well studied in the AI literature. While useful for context, it may not be suitable as the primary figure of the paper. A more impactful main figure would be a detailed diagram illustrating the specific architecture and information flow of the ProteinGuide framework.

Response.
Finally, thanks for the feedback regarding the content of Figure 1. We have now expanded it, including the practitioner workflow when using ProteinGuide, as well as the original overview of the technical unifying view.

Methods Section: The Methods section contains extensive theoretical explanations and formula derivations. While rigor is appreciated, these elements do not appear to be novel contributions in themselves and seem disconnected from the specific context of protein engineering.

Response.
We thank the reviewer for raising this point, and have now substantially revised both the main text and the Supplementary Information to better clarify i) which theoretical results are novel in this work, and ii) how these results directly address practical challenges in protein engineering. Regarding novelty, while parts of the exposition on the equivalences of training objectives built on and cited prior work on discrete diffusion models (DDMs) and any-order autoregressive models (AO-ARMs), the theoretical results presented herein include several new contributions that were not previously established and that enable a practical guidance framework that can be used for protein engineering. Several new points that are now in the manuscript:
• We now discuss that prior work did not extend equivalences between DDM and AO-ARMs to discrete flow matching models (DFMs).
• We've added a new result, a proof for the equivalence between the sampling distributions for DFMs and AO-ARMs. This proof is new, and separate from the training equivalences shown in our first version of the manuscript.
• Previously, it was thought that exact integration of the DFM sampling process was computationally intractable [3, 5, 6]. However, our proof of the equivalence between the sampling distributions of DFMs and AO-ARMs shows that the continuous time DFM process can be translated into an equivalent time-free sampling process amenable to simple, exact sampling. Our simple, exact sampling algorithm avoids the approximation of numerical integration—the current standard method used when sampling from a DFM

• The aforementioned sampling equivalence and simplification are not purely theoretically interesting: these results directly enabled the development of our exact guidance method, allowing us to tackle more challenging protein design problems (e.g., enzyme design experiment and multi-property experiment) without gradient approximations [2, 3, 7, 8], product-of-marginal approximations [9], or multi-particle Monte-Carlo estimates [10].

• Before our current work (pre-print and the updated version herein), it was not clear that guidance methods for one model type could be applied to the others. It was thought that each model type would need its own conditional sampling framework. For example, our original work [3] did not address how to perform guidance for models other than diffusion and flow matching, thereby limiting its utility to the field of protein engineering and beyond. The new constructive proof we added provides a mathematical "recipe" for converting algorithms for one model type into the other. In so doing, we unlock guidance for a much larger set of models that are actively being developed by the protein engineering community.

• Finally, our earlier conference proceedings paper [3] presented a preliminary version of the core framework presented herein. This earlier work introduced the central idea of on-the-fly guidance for discrete-space diffusion and flow models and has since seen substantial uptake by a number of communities (~80 citations on Google Scholar as of January 2026, initial preprint June 2024). The present manuscript substantially extends this preliminary version with new theoretical results (including the exact DFM sampling equivalence), a unified guidance formulation across model classes. Moreover, the present work is uniquely focused on the use case of protein engineering, including to the point of obtaining laborious wetlab experimental results. Our earlier conference proceedings version is an important part of the development of the present work, with the current journal article providing the completed and consolidated theoretical and experimental results, while focused on protein engineering.

Overall Conclusion
In summary, the authors tackle an 'interesting' problem of conditioning protein generative models. The wet-lab validation of ABEs is a commendable and interesting result. However, the work in its current form suffers from major concerns regarding its technical novelty, the strength of its motivating premise, the omission of critical baselines.
While the project represents an interesting attempt, the overall contribution does not appear to meet the high bar for novelty and significant impact expected for a publication in Nature Biotechnology. The work does not appear to have led to substantial advances in either the PLM or protein engineering fields. Addressing the major points raised above—particularly through direct and rigorous comparison with existing methods like DPLM, LoRA-based fine-tuning, and post-hoc filtering—would be essential to substantiate the claims of the paper.

Response.
Once again, we thank the reviewer for all the valuable feedback. In addressing this feedback, the manuscript has been made substantially stronger.

Responses to Reviewer #2

Reviewer #2
Remarks to the Author

In this manuscript by Xiong and colleagues, the authors propose ProteinGuide, a framework to guide protein generation toward specific properties. The main idea is to couple a trained property predictor with a general generative model: the generative model constrains design to the natural protein space, reducing the vast search space, while the predictor biases generation toward proteins with desired properties. The method of performing guidance for diffusion and flow-matching models was presented in the authors' prior work (ref. 16). Here, they extend the method to a class of masked language models (MLMs, ESM3) and order-agnostic autoregressive models (OA-AR, ProteinMPNN). Their results showed that ProteinGuide can help generate proteins with specific properties compared to unguided model. This method is important in molecular biology and protein engineering, I have some concerns regarding the evaluation and applications.

Major Comments

1. While the results demonstrate that guidance can be applied, it is unclear how effective the method truly is. The method was only compared against the unguided model—which, by design, cannot generate proteins with specific properties. It is important to compare with existing baseline approaches, as the authors acknowledged in “Additional discussion of related work”. A pre-print that the authors cited compared different methods for protein fitness, it’s unclear how different methods work for the applications in this manuscript.

Response.
Indeed, we have now included comparisons to the most common practices of post-hoc filtering and fine-tuning in all of our in silico experiments (see responses to Reviewer #1 comments 2 and 3). As to the preprint by Yang et al. [4] that we cited, they themselves actually benchmarked the preliminary version of our method [3] and found that it outperformed several fine-tuning and other “plug-and-play” guidance methods that are cited in our Related Work section, such as performing guidance on continuous embeddings, and fine-tuning autoregressive models with Direct Preference Optimization (DPO). Accordingly, we do not compare to these methods again. Instead, we focus on demonstrating the efficacy of ProteinGuide on a wider breadth of challenging design scenarios and on a wetlab design campaign. We have also clarified these points in the Introduction and Related Work sections, and Supplementary Information section “Detailed discussion on related work”.

2. Another class of Plug and Play approach is in silico directed evolution. This strategy also integrates a general model with a property predictor and can be applied to protein design tasks presented in the manuscript. What advantages does ProteinGuide offer compared to this approach?

Response.
We thank the reviewer for bringing this in silico directed evolution paper [11] (Plug & Play Directed Evolution, henceforth referred to as PPDE) to our attention. We have now clarified the distinctions between ProteinGuide and PPDE in the manuscript (see Supplementary Information section “Detailed discussion on related work”). To summarize briefly here, the key points are i) PPDE requires taking gradients through the pre-trained model. When these models are large, these gradients cannot be practically computed; ii) PPDE requires repeatedly computing the (unnormalized) likelihood of the generative model, which is either intractable or too expensive for most PLMs, iii) as a Markov Chain Monte-Carlo (MCMC) method, PPDE can be sensitive to its initialization, and will have mixing times that can degrade rapidly with sequence length and rugged fitness landscapes. In contrast, our method does not require gradients (only optionally and only through the predictive model for TAG [3]), does not require likelihood computations, runs in time linear in sequence length, and does not require a wild-type sequence to initialize sampling. However, PPDE is a lovely piece of work that will have its utilties as well, including to use on top of our work. In particular, PPDE could be initialized from sequences sampled by ProteinGuide, although doing so is beyond the scope of our present contributions.

3. For these applications, how would ProteinGuide perform if the property predictors were trained with less data? In real-world scenarios, we often lack sufficient labeled data to train highly accurate property predictors, so understanding the method’s robustness to limited training data would be important.

Response.
We thank the reviewer for their suggestion. In the new version of the paper, we included four new experiments with $\sim 103$ sequences, in line with the experimental budget in our in vivo experiment for ABE engineering. As another reference point, the data used in our multi-property experiment [12] consisted of 1, 099 sequences that came from a single day of high-throughput screening. Finally, others have found useful results with the method from our preprint with as few as 200 samples (e.g., Fig. 5 in Yang et al. [4]).

4. For the protein stability design task. ProteinMPNN is capable of zero-shot prediction of mutation effect, how do the design results relate to its zero-shot performance?

Response.
This is an interesting question. In some sense, this zero-shot capability is captured by the unguided (post-hoc filtered) results. The last step of unguided generation is sampling from the distribution of probabilities that would be used in a zero-shot mutation effect prediction task.

5. For the enzyme-class guided sequence generation, ProteinGuide outperforms unguided ESM3 only at high mask ratios (≥ 80% of residues masked). It is unclear why guided and unguided models achieve similar performances at lower mask levels (< 80%). Additionally, it is important to report sequence identity of designed proteins compared to the original sequence at different mask levels.

Response.
Indeed, good points. We speculate that at lower mask levels, the unconditional model is effectively conditioned on the enzyme class by the non-masked tokens, thereby indexing into the right part of the distribution that the guiding does. This can only work in an interpolative design goal regime, not extrapolative. In contrast, on the new extrapolative problem of say multi-property design, even when we start sampling from about 60% unmasked sequences there remains a sizable performance gap between the unguided samples and ProteinGuide, in favor of ProteinGuide (Figure 4).
Also thank you for pointing out that sequence similarity metrics for this experiment should've been included. We have added tables for the length-normalized Hamming distance to wildtype for both the main enzyme-class guidance experiment (Table S6) and the original partial sequence recovery experiment (Table S7). For the main experiment, we only include sequence identity metrics for ProteinGuide and fine-tuning because the other methods do not produce any sequences in the target enzyme class.

6. In addition, I recommend using independent structure predictors to evaluate pLDDT values, rather than relying on ESM3 itself, because it's unclear whether ESM3 pLDDT is biased toward its own generations. Beyond CLEAN classification, it would be more informative to compare the predicted structures with well-characterized enzymes of the same class, with particular attention to the structure of active-site.

Response.
We thank the reviewer for this recommendation. We now use ESMFold, a separately trained and different model from ESM3, for all of our experiments that generate sequences using ESM3; doing so follows on related work which does the same [13, 14] and is significantly less expensive to run than AlphaFold.

Minor Comments
7. How do the designed ABEs compared to the most active ABEs currently used in the field or designed by others?

Response.
We have now included additional experimental validation addressing this question (Figure 5f). In summary, we find that the variant with the with highest enrichment value from the ProteinGuide library has activity between ABE7.10 and ABE8 (corresponding to seven and eight rounds of directed evolution, respectively). This shows that one round of ProteinGuide can generate proteins with activity as high as using seven rounds of directed evolution [15, 16].

8. It's important to discuss how to select guidance strength parameter in equation 29.

Response.
Indeed, this was an oversight on our part. We have now included details on how to select for the guidance strength parameter in the Methods section "Common setup across in silico experiments"

as well as the Methods sections specific to each experiment. To summarize here: strict adherence to Bayes' rule implies $\gamma = 1$, but others have found that $\gamma > 1$ can be useful in continuous-space diffusion [17, 18]. In each experiment, we start with $\gamma = 1$ and monitor whether the predictor likelihood, $p(y|x_t)$, increases over the generation trajectory. Depending on where the unconditional model started the generation trajectory, and the distribution of $p(y|x_t)$ over the state spaces, $\gamma = 1$ might be not sufficient to increase the guidance signals. In such cases, we increase the guidance strength to either $\gamma = 10$ or $\gamma = 100$. In all experiments, we tested $\gamma \in \{1, 10, 100\}$, and didn't systematically tune it further, although more extensive tuning might improve generation further. We've listed the final value of $\gamma$ for each experiment in the corresponding Methods sections. In practice, users may also tune the guidance strength more systematically based on the generation quality metrics most relevant to their application..

References

[1] Xinyou Wang, Zaixiang Zheng, Fei Ye, Dongyu Xue, Shujian Huang, and Quanquan Gu. Diffusion Language Models Are Versatile Protein Learners. In International Conference on Machine Learning, pages 52309–52333. PMLR, 2024.

[2] Cl´ement Vignac, Igor Krawczuk, Antoine Siraudin, Bohan Wang, Volkan Cevher, and Pascal Frossard. DiGress: Discrete Denoising diffusion for graph generation. In International Conference on Learning Representations, 2023. URL https://openreview.net/forum?id=UaAD-Nu86WX.

[3] Hunter Nisonoff, Junhao Xiong, Stephan Allenspach, and Jennifer Listgarten. Unlocking Guidance for Discrete State-Space Diffusion and Flow Models. In The Thirteenth International Conference on Learning Representations, 2025.

[4] Jason Yang, Wenda Chu, Daniel Khalil, Raul Astudillo, Bruce J. Wittmann, Frances H. Arnold, and Yisong Yue. Steering generative models with experimental data for protein fitness optimization, 2025. URL https://arxiv.org/abs/2505.15093.

[5] Andrew Campbell, Jason Yim, Regina Barzilay, Tom Rainforth, and Tommi S. Jaakkola. Generative Flows on Discrete State-Spaces: Enabling Multimodal Flows with Applications to Protein Co-Design. In International Conference on Machine Learning, 2024.

[6] Marton Havasi, Brian Karrer, Itai Gat, and Ricky TQ Chen. Edit flows: Flow matching with edit operations. arXiv preprint arXiv:2506.09018, 2025.

[7] Will Grathwohl, Kevin Swersky, Milad Hashemi, David Duvenaud, and Chris Maddison. Oops I Took a Gradient: Scalable Sampling for Discrete Distributions. In International Conference on Machine Learning, pages 3831–3841, 2021.

[8] Brandon Trabucco, Aviral Kumar, Xinyang Geng, and Sergey Levine. Conservative objective models for effective offline model-based optimization. In International Conference on Machine Learning, pages 10358–10368. PMLR, 2021.

[9] Xiner Li, Yulai Zhao, Chenyu Wang, Gabriele Scalia, Gokcen Eraslan, Surag Nair, Tommaso Biancalani, Shuiwang Ji, Aviv Regev, Sergey Levine, et al. Derivative-free guidance in continuous and discrete diffusion models with soft value-based decoding. arXiv preprint arXiv:2408.08252, 2024.

[10] Luhuan Wu, Brian Trippe, Christian Naesseth, David Blei, and John P Cunningham. Practical and asymptotically exact conditional sampling in diffusion models. Advances in Neural Information Processing Systems, 36:31372–31403, 2023.

[11] Patrick Emami, Aidan Perreault, Jeffrey Law, David Biagioni, and Peter St John. Plug & play directed evolution of proteins with gradient-based discrete mcmc. Machine Learning: Science and Technology, 4(2):025014, 2023.

[12] Brenda M Wang, Nicole Chiang, Holly M Ekas, Dylan M Brown, Garrett Dildine, Tyler J Lucci, Siyuan Feng, Vanessa Bly, Jean-Fran¸cois Gaillard, Julius B Lucks, et al. Active learning-guided optimization of cell-free biosensors for lead testing in drinking water. bioRxiv, 2025.

[13] Tomas Geffner, Kieran Didi, Zuobai Zhang, Danny Reidenbach, Zhonglin Cao, Jason Yim, Mario Geiger, Christian Dallago, Emine Kucukbenli, Arash Vahdat, and Karsten Kreis. Proteina: Scaling Flow-based Protein Structure Generative Models. In The Thirteenth International Conference on Learning Representations, 2025.

[14] Tianyu Lu, Melissa Liu, Yilin Chen, Jinho Kim, and Po-Ssu Huang. Assessing generative model coverage of protein structures with shapes. Cell Systems, 16(8), 2025. ISSN 2405-4712. doi: 10.1016/j.cels.2025.101347. URL https://doi.org/10.1016/j.cels.2025.101347.

[15] Nicole M Gaudelli, Alexis C Komor, Holly A Rees, Michael S Packer, Ahmed H Badran, David I Bryson, and David R Liu. Programmable base editing of A• T to G• C in genomic DNA without DNA cleavage. Nature, 551(7681):464–471, 2017.

[16] Michelle F Richter, Kevin T Zhao, Elliot Eton, Audrone Lapinaite, Gregory A Newby, B W Thuronyi, Christopher Wilson, Luke W Koblan, Jing Zeng, Daniel E Bauer, et al. Phage-assisted evolution of an adenine base editor with improved Cas domain compatibility and

activity. Nature biotechnology, 38(7):883–891, 2020.

[17] Prafulla Dhariwal and Alexander Nichol. Diffusion Models Beat GANs on Image Synthesis. Advances in Neural Information Processing Systems, 34:8780–8794, 2021.

[18] Jonathan Ho and Tim Salimans. Classifier-Free Diffusion Guidance. In Neural Information Processing Systems - Workshop on Deep Generative Models and Downstream Applications, 2021.

Version 1:

Reviewer comments:

Reviewer #2

(Remarks to the Author)
the authors have addressed all my comments in the revision

(Remarks on code availability)