Improving Inverse Folding models at Protein Stability Prediction without additional Training or Data

Oliver Dutton*
Peptone Ltd.
oliver@peptone.io

Sandro Bottaro
Peptone Ltd.
sandro@peptone.io

Istvan Redl
Peptone Ltd.
istvan@peptone.io

Michele Invernizzi
Peptone Ltd.
michele@peptone.io

Albert Chung
Peptone Ltd.
albert@peptone.io

Carlo Fisicaro
Peptone Ltd.
carlo@peptone.io

Stefano Ruschetta
Peptone Ltd.
stefano@peptone.io

Fabio Airoldi
Peptone Ltd.
fabio@peptone.io

Louie Henderson
Peptone Ltd.
louie@peptone.io

Benjamin M J Owens
Peptone Ltd.
ben@peptone.io

Patrik Foerch
Peptone Ltd.
patrik@peptone.io

Kamil Tamiola
Peptone Ltd.
kamil@peptone.io

Abstract

Deep learning protein sequence models have shown outstanding performance at de novo protein design and variant effect prediction. We substantially improve performance without further training or use of additional experimental data by introducing a second term derived from the models themselves which align outputs for the task of stability prediction. On a task to predict variants which increase protein stability the absolute success probabilities of PROTEINMPNN and ESMIF are improved by 11% and 5% respectively. We term these models PROTEINMPNN-DDG and ESMIF-DDG.

1 Introduction

Models trained to predict native protein sequence based off of structural and partial sequence context show remarkable versatility. They generalise both to the macro scale, designing whole proteins, and the micro scale, predicting beneficial single point mutations without explicit training [1, 2]. Previous work has improved performance of ProteinMPNN [3] at designing soluble analogues of membrane proteins by retraining with a modified training set, while AlphaMissense [4] improved AlphaFold2's accuracy at distinguishing disease related mutations by fine-tuning with unlabelled data. The present work continues in this direction but without the usage of retraining or fine-tuning, instead predictions are made with two different inputs and the difference in outputs are utilised.

Unsupervised deep learning models leveraging sequence and/or structural information have demonstrated zero-shot prediction of the change in protein properties upon mutation, including expression, activity and stability [2, 5, 6, 7, 8]. Models leveraging only sequence data have been shown to

*The Connolly Works, 41-43 Chalton Street, London, NW1 1JD

Preprint. Under review.

outperform those incorporating structural data on the majority of properties except for stability [7]. The effect of point mutations on stability was best predicted by inverse folding models which are trained to recover the native sequence of a protein using its backbone structure and partial sequence context. The differences between the likelihood of the native and a mutant amino acid predicted by these models correlate to the relative stability of the two sequences.

By considering toy cases we find that optimal predictions from an inverse folding model are non-optimal for the prediction of physical quantities such as protein stability. We introduce an additional term derived from the model itself to improve the prediction of relative stability upon mutation without experimental data or further training. We demonstrate this modification improves accuracy in predicting the effect of point mutations for two popular models, PROTEINMPNN and ESMIF [1, 2], across three benchmark datasets. We term the resulting improved, unsupervised models PROTEINMPNN-DDG and ESMIF-DDG.

In addition, we implement a novel tied decoding scheme which limits the increased computation cost of PROTEINMPNN-DDG relative to common usage of PROTEINMPNN.

2 Specialisation of model for mutation stability prediction

2.1 Leveraging maximum sequence context

We aim to predict the effect of a mutation from the wildtype amino acid, X, to a mutant, Y, at position i, denoted $f_{i,X\rightarrow Y}$, using a subset of the sequence, s, and positions of the backbone atoms, x.

Following common usage of large language models [6, 9, 2] we mask the identity of the amino acid at position i and predict using all other possible sequence and structural context (Equation 1). In the baseline models we score mutations as

$$f^{\text{Baseline}}_{i,X\rightarrow Y} = \log \frac{p(Y|s_{\mathbb{U}\setminus i}, x_{\mathbb{U}})}{p(X|s_{\mathbb{U}\setminus i}, x_{\mathbb{U}})} \tag{1}$$

where $\mathbb{U}$ denotes the set of all residues and $\mathbb{U} \setminus i$ is the set of all residues with residue i masked. The log-odds ratio are calculated using either PROTEINMPNN or ESMIF.

In its naïve implementation PROTEINMPNN only uses a subset of the full sequence context for predicting most tokens as residue identities are decoded autoregressively with a random order. Here, we generate decoding orders such that residue i is decoded last and find this procedure improves accuracy over the classic usage, improving sequence recovery on the PROTEINMPNN validation set by nearly 4% (Table 1). A single structure for each of the 1,464 clusters in the PROTEINMPNN validation set was taken and predicted, resulting in predictions for 375,445 residues on which sequence recovery metrics were computed in Table 1.

As the predictor ESMIF is autoregressive from N to C terminus only sequence context from residues before position i can be given so we use $s_{\{0,1,\ldots,i-1\}}$ in place of $s_{\mathbb{U}\setminus i}$ for ESMIF. With retraining this limitation could be removed, but we do not pursue it in this work.

Table 1: Improved sequence recovery metrics from tailored usage of PROTEINMPNN
*Exponentiated mean cross-entropy computed over 20 natural amino acids

MODEL	SEQUENCE RECOVERY	ECE*
PROTEINMPNN	51.2%	4.76
+ DECODE LAST	**55.0%**	**4.22**

2.2 Utilising logit perturbation between different inputs

In the limiting case where no structural or sequence context are given an optimal model would return a distribution mirroring the amino acid frequencies in its training set. This would lead to log-odds ratios suggesting mutations to more abundant amino acids such as Tryptophan to Leucine are more

likely to increase stability than Leucine to Tryptophan. However, the natural abundance of amino acids in proteins has been shown to be linked to a trade-off between the metabolic costs associated with each amino acid and the complexity of the proteome that composition generates [10] rather than due to an intrinsic ability to stabilise a proteins structure. Motivated by that observation, we shift the log-odds ratios such that mutations are predicted to have no effect when no context is given. This decision does not leverage any experimental stability data.

In an extended limiting case where the backbone atoms of a single residue are given, a model can also utilise discrepancies in the relative geometry of the backbone atoms. This backbone residue internal geometry gives information about the amino acid identity but not stability as protein stability arises from the interactions between amino acids. Hence we recommend corrections to nullify this contribution to the models predictions.

We therefore introduce an additional term consisting of the log-odds ratio as in Equation (1) but where only backbone atoms of the single residue being predicted are given as context (Equation (2)).

$$f^{ddG}_{i,X\to Y} = \log \frac{p(Y|s_{\mathbb{U}\setminus i}, x_{\mathbb{U}})}{p(X|s_{\mathbb{U}\setminus i}, x_{\mathbb{U}})} - \log \frac{p(Y|x_i)}{p(X|x_i)} \tag{2}$$

We note Equation (2) resembles the standard procedure in classical free energy calculations in which the free energy change upon mutation is estimated both in folded and unfolded states [11]. In this analogy the unfolded state, the second term, is modelled as a single residue.

We first verified PROTEINMPNN and ESMIF generalise to single residue inputs. We calculated averaged predictions for each of the 380 possible $X \to Y$ mutations as $\delta_{X\to Y}$

$$\delta_{X\to Y} = \frac{1}{M} \sum_{s_j = X} \log \frac{p(Y|x_j)}{p(X|x_j)} \tag{3}$$

where the sum runs over M structures in a structural database.

A single structure was taken for each of the 23,349 clusters in the training set of PROTEINMPNN and residues for which the backbone atom positions are unknown were removed. This resulted in 5,615,050 residue geometries with at least 77,000 geometries for each distinct amino acid. The values for PROTEINMPNN are shown in Figure 2a.

We found the log-odds ratios calculated from the amino acid frequencies in the training set of PROTEINMPNN correlate well with averaged log-odds ratios $\delta^{\text{PROTEINMPNN}}_{X\to Y}$ (spearman correlation coefficient 0.72).

As log-odds ratios calculated from the amino acid frequencies are naturally antisymmetric the only possible origin of deviation from antisymmetry, $|\delta_{X\to Y}+\delta_{Y\to X}|$, is structural context derived from the backbone atoms N, Cα, C and O geometry. Fitting with our expectations the largest size of deviation from antisymmetry for PROTEINMPNN were observed for Glycine, Proline, Valine and Isoleucine (Figure 2b). Looking at the backbone N-Cα-C opening angle we observe most amino acids display a very similar distribution so cannot be distinguished, while Glycine, Proline, Valine and Isoleucine are quite distinct (Figure 3). Glycine due to its lack of sidechain and Proline due to its ring incorporating Cα and N display increased N-Cα-C opening angle. Valine and Isoleucine are the only amino acids with tertiary substituted Cβ, this causes steric crowding around Cα, leading to a smaller N-Cα-C opening angle.

The clear relationship between amino acid frequencies in the training set and log-odds ratios in $\delta_{X\to Y}$ in addition to pronounced deviation from antisymmetry, $|\delta_{X\to Y} + \delta_{Y\to X}|$, for amino acids with distinct geometries demonstrated PROTEINMPNN generalised well to single residue inputs.

When single residue inputs are given to ESMIF it infers them as the first residue in the sequence. This leads ESMIF to more strongly reflect amino acid frequencies of the first position in its training set than overall. Over 98.5% of the training set is made up of AlphaFold2 structures predicted for UniRef50 sequences [2]. These UniRef50 sequences all begin with methionine, being coded at the mRNA level by the canonical eukaryotic start codon AUG. As a result, we observe predictions with a single residue context show high probabilities of methionine. PROTEINMPNN does not use absolute positional information and does not appear to suffer from this behaviour. We observed that the non-methionine $\delta_{X\to Y}$ for ESMIF correlated well to those from PROTEINMPNN (pearson correlation coefficient

0.83), and we therefore utilised PROTEINMPNN predictions to derive a correction for the methionine bias of ESMIF. We fitted one coefficient, representing the background over-prediction of ESMIF for methionine, shifting the related parameters for $M \rightarrow X$ and $X \rightarrow M$ by adding or subtracting the coefficient and maximising the pearson correlation coefficient between the PROTEINMPNN and ESMIF $\delta_{X \rightarrow Y}$. The pearson correlation coefficient between $\delta_{X \rightarrow Y}^{\text{ProteinMPNN}}$ and $\delta_{X \rightarrow Y}^{\text{ESMIF}}$ improved from 0.52 to 0.85 after training the single methionine coefficient, which affects only 38 of the 380 values in $\delta_{X \rightarrow Y}$. We applied this correction to all ESMIF predictions, following equation (4). For completeness, results with mutations involving methionine removed are available in the SI (Table 5).

2.3 Time complexity reduction to enable proteome-scale prediction with PROTEINMPNN-DDG

PROTEINMPNN implements a causal decoding scheme of protein sequence identity using any order. This means the first residue to be decoded has purely structural context with no sequence information while the final one to be decoded is given the sequence information on all other residues. By running the model separately for each residue with an appropriate decoding order we can predict sequence probabilities for every residue with full context, increasing native sequence recovery (Table 1).

Naïvely decoding every residue with full context requires N passes of PROTEINMPNN, where N is the number of residues in the protein. We analysed the model to find any shared work between passes. The model involves three stages:

1. Computation of the nearest neighbors for each residue, these define edges in the sparse graph

2. Encoder of structural features

3. Decoder using both sequence and structural features

The decoding order of sequence information is only relevant for the final stage, so the first two stages only need to be computed once and reused with different decoding orders. We implemented a version of PROTEINMPNN with shared work and find this displays linear slowdown of approximately $\frac{N}{5}$ relative to a single pass rather than the full N fold on an NVIDIA A100 40GB GPU. Such a large slowdown relative to a single pass of the network would hinder mutation stability prediction at scale.

We made further improvements inspecting our aim for full sequence context in Equation (1). The only requirement on the decoding order for predicting residue i is that it is decoded last, there are no constraints placed on the order up until then. Acceptable orders to predict residues 0 and 1 can share the same order up until the last two, e.g. in an 8 residue system the orders $\underline{23456710}$ and $\underline{23456701}$ can be used with the partial decoding of 234567 shared. This principal was extended to maximise the amount of shared partial decoding which reduces the number of token decodes from $\mathcal{O}(N^2)$ when random orders are used to $\mathcal{O}(N \log N)$ (Figures 4 and 5). This tied decoding greatly reduced the additional cost associated with using full sequence context relative to a single pass of PROTEINMPNN from over 200-fold for a 1024 residue protein to under 4-fold (Figure 1). This enabled identification of mutations affecting protein stability at the proteome-scale. Saturation mutagenesis predictions were made for all 23,391 AlphaFold2 predicted structures of the human proteome (UP000005640_9606_HUMAN) in 30 minutes on a single V100 16GB GPU.

PROTEINMPNN-DDG represents a million fold speedup relative to established predictors such as Rosetta and FoldX on similar hardware, and a four hundred fold speedup relative to Rapid Stability Prediction (RaSP) which is trained to approximate Rosetta predictions at a lower computational cost (Table 2). Throughput for PROTEINMPNN-DDG was computed as an average over computing the whole UP000005640_9606_HUMAN proteome (including compilation and pdb loading times) on a single NVIDIA V100 16 GB GPU machine. Throughputs for RaSP, Rosetta and FoldX taken from highest throughput results in [12] where RaSP runtime is calculated on single NVIDIA V100 16 GB GPU machine while Rosetta and FoldX computations were parallelized and run on a server using 64 2.6 GHz AMD Opteron 6380 CPU cores with three $\Delta\Delta G$ computations per mutation.

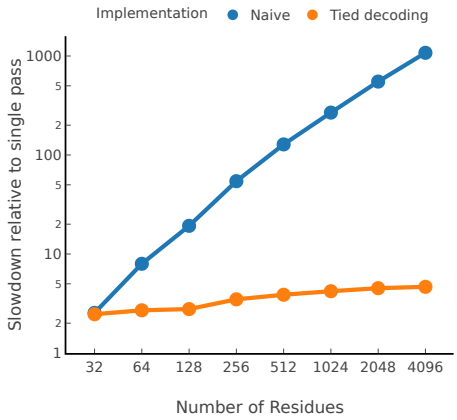


Figure 1: Slowdown relative to a single pass of PROTEINMPNN of naïve and tied decoding order schemes

Table 2: Throughput of PROTEINMPNN-DDG and other methods

MODEL	THROUGHPUT (RESIDUES/SECOND)
PROTEINMPNN-DDG	**9800**
RASP	2.5
ROSETTA	0.0052
FOLDX	0.0025

3 Experiments

3.1 Datasets

We consider three protein stability datasets used in previous studies:

1. Tsuboyama [13] produced a saturation mutagenesis dataset measuring resistance to proteolytic digest which has been shown to correlate well with thermal stability. We follow the subset selected in [12] where only single amino acid substitutions with high quality $\Delta\Delta G$ values for natural proteins were available, comprising of 164,524 point mutations across 164 proteins.
2. S2648 dataset contains 2,648 point mutations spread across 131 proteins, with values curated from the ProTherm database. S2648 has been used to train $\Delta\Delta G$ predictors in previous studies [14, 15, 16].
3. S669 [17] contains 669 variants of 94 proteins with less than 25% sequence identity to those in S2648 such that it can function as a test dataset for models trained on S2648.

3.2 Metrics

A common task in protein engineering is to identify protein variants with increased thermodynamic stability. We therefore focus on differentiating sequences with increased stability ($\Delta\Delta G < 0$), mirroring real world usage as closely as possible, subject to constraints imposed by the size and sparsity of available datasets.

Often only a manageable handful ($\sim 10^1$) single mutations are experimentally measured which corresponds to less than 1% of the possible options for a >50 residue protein. This selection can be made e.g. by identifying the top-scoring mutations from all possible options. We define a metric, 'Success@10', where the 10 single point mutations with highest predictions for each protein are selected and the proportion which display higher stability than the wildtype protein is calculated. As

the Tsuboyama dataset contains data for nearly all possible point mutations for 164 proteins, the Success@10 metric is averaged over 1,640 datapoints so is statistically meaningful. In line with previous works, for S2648 and S669 datasets where a low number of datapoints are available per protein we compute the area under the receiver operating characteristic curve (auROC) and area under precision recall curve (auPRC) for differentating mutations which experimentally show increased stability relative to the wildtype protein. Note that auROC and auPRC do not take into account that predictions at the top end of the score distribution are more relevant to real world usage. Metrics such as Normalized Discounted Cumulative Gains, NDCG [7, 18, 19] have been proposed to increase the weighting of high scoring mutations, however the selection of discount rate in NDCG is required.

Thanks to the large number of mutations per protein approximating saturation mutagenesis in the Tsuboyama dataset we can compute statistically meaningful metrics on a per protein basis then average over all proteins as is common practice for deep mutation scanning datasets [20, 7]. Due to the low number of mutations for each protein in S2648 and S699 metrics must be computed over the aggregate of all wildtype proteins present.

3.3 Benchmark models

We compared the performance of ProteinMPNN-ddG and ESMif-ddG with publicly available methods that enable large-scale predictions at an affordable computational cost.

Rapid Stability Prediction (RaSP) [12] is a deep learning predictor trained to reproduce the predictions of Rosetta, a well-established energy-function based method [21, 22].

DDGun [23] is a mutant effect predictor based off of a linear combination of features shown to correlate to protein stability, weighted by their correlation on a mutant stability dataset. Here we consider both the sequence based model, as well as the structure and sequence based model DDGun3D. It leverages the multiple sequence alignment (MSA) for the protein being predicted which we generated using hhblits [24] based on the June 2020 edition of UniRef30 where tabulated predictions were not found.

ACDC-NN [25] is a deep neural network trained on S2648 demonstrating strong performance on a related dataset, outperforming energy-based methods such as Rosetta and FoldX as well as DDGun [26]. We here consider both the sequence based model (ACDC-NN-Seq), as well as the structure and sequence based model ACDC-NN-Struct. This model leverages an MSA which was generated according to instructions on the associated GitHub page (accessed 04/09/2023) where tabulated predictions were not found.

RaSP, DDGun and ACDC-NN have been explicitly trained to reproduce experimental stability data, and they are therefore considered here as supervised models. No experimental protein stability data was used to derive ProteinMPNN and ESMif, and the same applies for ProteinMPNN-ddG and ESMif-ddG which we introduce here. The resulting models are therefore unsupervised.

As this work is focused on methods to improve unsupervised inverse folding models at mutation effect predictions tasks, in particular stability, we only benchmark a small subset of supervised models available in literature. RaSP, DDGun and ACDC-NN were chosen as they are predictors with public code with strong results in literature which could be scaled to predict $> 100,000$ point mutations within a reasonable computational budget.

4 Results

Improvements were observed for both ProteinMPNN-ddG and ESMif-ddG across all datasets and metrics (Table 3). We find in the challenging task of selecting 10 single point mutations with highest predictions for each protein and measuring the proportion which display higher stability than the wildtype protein, 'Success@10', the alterations described in this work improve absolute success rates by 11% ($66 \rightarrow 77\%$) and 5% ($64 \rightarrow 69\%$) for ProteinMPNN and ESMif, respectively, to our -ddG variants. This represents a significant improvement for real world usage with no additional model training or data usage and maintains compute efficiency of up to 10,000 residues per second on a NVIDIA V100 16 GB GPU.

We note the approach of leveraging difference in model outputs from different inputs can be extended beyond stability prediction. We have observed the change in logit differences between predictions for

Table 3: Accuracy of predictions for various models and datasets

	Tsuboyama, S164524			S2648		S669	
Model	Success@10	auROC	auPRC	auROC	auPRC	auROC	auPRC
Unsupervised							
ProteinMPNN	66%	0.78	0.49	0.75	0.44	0.64	0.36
ProteinMPNN-ddG (ours)	**77%**	**0.82**	**0.56**	**0.80**	**0.52**	**0.71**	**0.44**
ESMif	64%	0.77	0.47	0.69	0.41	0.63	0.39
ESMif-ddG (ours)	69%	0.79	0.50	0.72	0.45	0.67	0.41
Supervised							
RaSP	51%	0.73	0.41	0.76	0.43	0.70	0.43
ACDC-NN-Seq	51%	0.70	0.38	0.75	0.43	**0.71**	0.41
ACDC-NN-Struct	42%	0.71	0.38	0.76	0.40	**0.71**	**0.44**
DDGun	50%	0.69	0.37	0.70	0.39	0.69	0.36
DDGun3D	41%	0.70	0.36	0.73	0.38	0.70	0.40

a protein with and without its binding partner can accurately differentiate mutations which strengthen binding. We leave a thorough benchmark of this to later works.

References

[1] J. Dauparas, I. Anishchenko, N. Bennett, H. Bai, R. J. Ragotte, L. F. Milles, B. I. M. Wicky, A. Courbet, R. J. de Haas, N. Bethel, P. J. Y. Leung, T. F. Huddy, S. Pellock, D. Tischer, F. Chan, B. Koepnick, H. Nguyen, A. Kang, B. Sankaran, A. K. Bera, N. P. King, and D. Baker, "Robust deep learning–based protein sequence design using proteinmpnn," *Science*, vol. 378, no. 6615, pp. 49–56, 2022.

[2] 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 *Proceedings of the 39th International Conference on Machine Learning* (K. Chaudhuri, S. Jegelka, L. Song, C. Szepesvari, G. Niu, and S. Sabato, eds.), vol. 162 of *Proceedings of Machine Learning Research*, pp. 8946–8970, PMLR, 17–23 Jul 2022.

[3] B. I. M. Wicky, L. F. Milles, A. Courbet, R. J. Ragotte, J. Dauparas, E. Kinfu, S. Tipps, R. D. Kibler, M. Baek, F. DiMaio, X. Li, L. Carter, A. Kang, H. Nguyen, A. K. Bera, and D. Baker, "Hallucinating symmetric protein assemblies," *Science*, vol. 378, no. 6615, pp. 56–61, 2022.

[4] J. Cheng, G. Novati, J. Pan, C. Bycroft, A. Žemgulytė, T. Applebaum, A. Pritzel, L. H. Wong, M. Zielinski, T. Sargeant, R. G. Schneider, A. W. Senior, J. Jumper, D. Hassabis, P. Kohli, and Žiga Avsec, "Accurate proteome-wide missense variant effect prediction with alphamissense," *Science*, vol. 381, no. 6664, p. eadg7492, 2023.

[5] Z. Lin, H. Akin, R. Rao, B. Hie, Z. Zhu, W. Lu, A. d. S. Costa, M. Fazel-Zarandi, T. Sercu, S. Candido, and A. Rives, "Language models of protein sequences at the scale of evolution enable accurate structure prediction," *bioRxiv*, p. 2022.07.20.500902, 2022.

[6] J. Meier, R. Rao, R. Verkuil, J. Liu, T. Sercu, and A. Rives, "Language models enable zero-shot prediction of the effects of mutations on protein function," *bioRxiv*, p. 2021.07.09.450648, 2021.

[7] P. Notin, A. W. Kollasch, D. Ritter, L. V. Niekerk, S. Paul, H. Spinner, N. J. Rollins, A. Shaw, R. Orenbuch, R. Weitzman, J. Frazer, M. Dias, D. Franceschi, Y. Gal, and D. S. Marks, "Proteingym: Large-scale benchmarks for protein fitness prediction and design," in *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023.

[8] M. Cagiada, S. Ovchinnikov, and K. Lindorff-Larsen, "Predicting absolute protein folding stability using generative models," *bioRxiv*, 2024.

[9] R. Rao, J. Liu, R. Verkuil, J. Meier, J. F. Canny, P. Abbeel, T. Sercu, and A. Rives, "MSA Transformer," *bioRxiv*, p. 2021.02.12.430858, 2021.

[10] T. Krick, N. Verstraete, L. G. Alonso, D. A. Shub, D. U. Ferreiro, M. Shub, and I. E. Sánchez, "Amino Acid Metabolism Conflicts with Protein Diversity," *Molecular Biology and Evolution*, vol. 31, no. 11, pp. 2905–2912, 2014.

[11] V. Gapsys, S. Michielssens, D. Seeliger, and B. L. d. Groot, "Accurate and Rigorous Prediction of the Changes in Protein Free Energies in a Large-Scale Mutation Scan," *Angewandte Chemie International Edition*, vol. 55, no. 26, pp. 7364–7368, 2016.

[12] L. M. Blaabjerg, M. M. Kassem, L. L. Good, N. Jonsson, M. Cagiada, K. E. Johansson, W. Boomsma, A. Stein, and K. Lindorff-Larsen, "Rapid protein stability prediction using deep learning representations," *eLife*, vol. 12, p. e82593, 2023.

[13] K. Tsuboyama, J. Dauparas, J. Chen, E. Laine, Y. Mohseni Behbahani, J. J. Weinstein, N. M. Mangan, S. Ovchinnikov, and G. J. Rocklin, "Mega-scale experimental analysis of protein folding stability in biology and design," *Nature*, vol. 620, pp. 434–444, Aug 2023.

[14] P. Fariselli, P. L. Martelli, C. Savojardo, and R. Casadio, "INPS: predicting the impact of non-synonymous variations on protein stability from sequence," *Bioinformatics*, vol. 31, no. 17, pp. 2816–2821, 2015.

[15] Y. Dehouck, J. M. Kwasigroch, D. Gilis, and M. Rooman, "PoPMuSiC 2.1: a web server for the estimation of protein stability changes upon mutation and sequence optimality," *BMC Bioinformatics*, vol. 12, no. 1, p. 151, 2011.

[16] S. Wang, H. Tang, P. Shan, Z. Wu, and L. Zuo, "ProS-GNN: Predicting effects of mutations on protein stability using graph neural networks," *Computational Biology and Chemistry*, vol. 107, p. 107952, 2023.

[17] C. Pancotti, S. Benevenuta, G. Birolo, V. Alberini, V. Repetto, T. Sanavia, E. Capriotti, and P. Fariselli, "Predicting protein stability changes upon single-point mutation: a thorough comparison of the available tools on a new dataset," *Briefings in Bioinformatics*, vol. 23, no. 2, p. bbab555, 2022.

[18] B. J. Wittmann, Y. Yue, and F. H. Arnold, "Informed training set design enables efficient machine learning-assisted directed protein evolution," *Cell Systems*, vol. 12, no. 11, pp. 1026–1045.e7, 2021.

[19] C. Hsu, H. Nisonoff, C. Fannjiang, and J. Listgarten, "Learning protein fitness models from evolutionary and assay-labeled data," *Nature Biotechnology*, vol. 40, no. 7, pp. 1114–1122, 2022.

[20] J. Frazer, P. Notin, M. Dias, A. Gomez, J. K. Min, K. Brock, Y. Gal, and D. S. Marks, "Disease variant prediction with deep generative models of evolutionary data," *Nature*, vol. 599, no. 7883, pp. 91–95, 2021.

[21] E. H. Kellogg, A. Leaver-Fay, and D. Baker, "Role of conformational sampling in computing mutation-induced changes in protein structure and stability," *Proteins: Structure, Function, and Bioinformatics*, vol. 79, no. 3, pp. 830–838, 2011.

[22] B. Frenz, S. M. Lewis, I. King, F. DiMaio, H. Park, and Y. Song, "Prediction of Protein Mutational Free Energy: Benchmark and Sampling Improvements Increase Classification Accuracy," *Frontiers in Bioengineering and Biotechnology*, vol. 8, p. 558247, 2020.

[23] L. Montanucci, E. Capriotti, Y. Frank, N. Ben-Tal, and P. Fariselli, "DDGun: an untrained method for the prediction of protein stability changes upon single and multiple point variations," *BMC Bioinformatics*, vol. 20, no. Suppl 14, p. 335, 2019.

[24] M. Steinegger, M. Meier, M. Mirdita, H. Vöhringer, S. J. Haunsberger, and J. Söding, "HH-suite3 for fast remote homology detection and deep protein annotation," *BMC Bioinformatics*, vol. 20, no. 1, p. 473, 2019.

[25] S. Benevenuta, C. Pancotti, P. Fariselli, G. Birolo, and T. Sanavia, "An antisymmetric neural network to predict free energy changes in protein variants," *Journal of Physics D: Applied Physics*, vol. 54, no. 24, p. 245403, 2021.

[26] J. Schymkowitz, J. Borg, F. Stricher, R. Nys, F. Rousseau, and L. Serrano, "The FoldX web server: an online force field," *Nucleic Acids Research*, vol. 33, no. suppl_2, pp. W382–W388, 2005.

A Appendix

A.1 Correction to ESMIF-DDG

$$f^{ddG}_{i,X\to Y} = \log \frac{p(Y|s_{\mathbb{U}\setminus i}, x_{\mathbb{U}})}{p(X|s_{\mathbb{U}\setminus i}, x_{\mathbb{U}})} - \log \frac{p(Y|x_i)}{p(X|x_i)} + c_M * (\delta(Y, M) - \delta(X, M)) \tag{4}$$

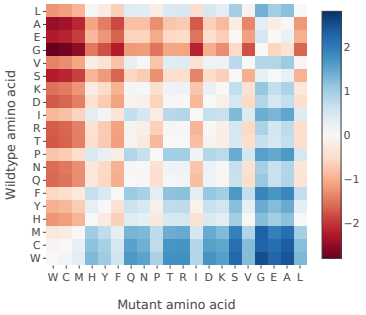
where M is methionine, $c_M = 4.18$ is the trained coefficient and δ is the Dirac delta function.

A.2 Approximate theoretical slowdown of decode last

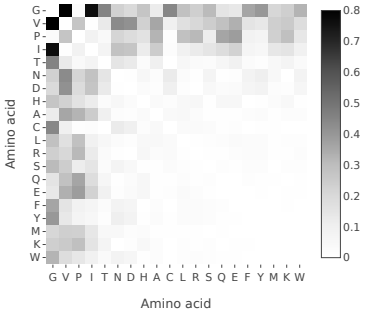
We estimated the relative cost of the tied decoding scheme to a single pass of PROTEINMPNN. PROTEINMPNN is a message passing neural network on a sparse graph, where each node has at most 48 edges. The bound number of edges per node leads to constant decode cost per residue. The encoder and decoder stages share the same hidden dimension and number of layers, though the decoder updates both edges and nodes while the encoder only updates nodes. We approximate decoder and encoder as equal cost. The encoder stage is independent of decoding order so has no increased cost, with N decodings for both. In the decoder stage the number of decodings increases from N to $(N \log_2 N + N)$ in the tied decoding scheme if N is a power of two. This gives a theoretical bound of the slowdown to use full sequence context rather than partial as $\frac{1}{2}\log_2 N + 1$ (Equation 5). Our timings benchmark were all within theoretical bounds (Figure 1).

The relative slowdown of tied decoding to a single pass, S, is given by:

$$S = \frac{t_{\text{tied}}}{t_{\text{single pass}}} = \frac{N + (N \log_2 N + N)}{N + N} = \frac{1}{2}\log_2 N + 1 \tag{5}$$



(a) $\delta_{X \to Y}$ for PROTEINMPNN, amino acids ordered by frequency in training set of PROTEINMPNN



(b) Deviation from antisymmetry of $\delta_{X \to Y}$ for PROTEINMPNN, $|\delta_{X \to Y} + \delta_{Y \to X}|$, amino acids ordered by degree of deviation.

Figure 2: Values and deviation from antisymmetry for $\delta_{X \to Y}$ of PROTEINMPNN

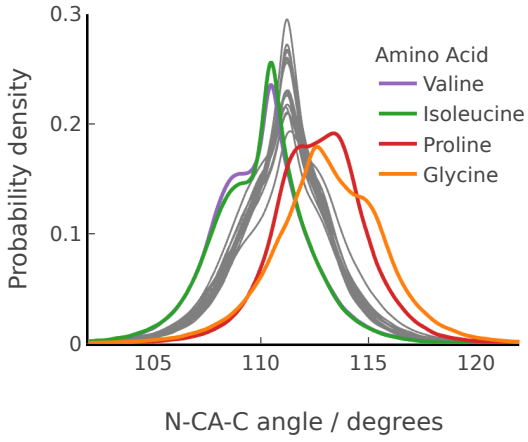


Figure 3: N-Cα-C angle distributions for all 20 natural amino acids

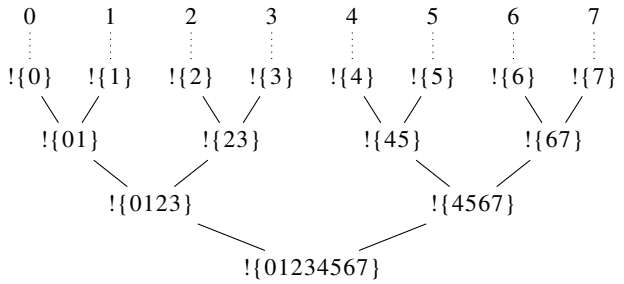
6	4	7	5	3	2	1	**0**
7	3	0	5	6	4	2	**1**
0	5	4	1	6	3	7	**2**
1	5	0	7	2	6	4	**3**
3	6	5	3	7	2	0	**4**
4	1	2	0	7	1	6	**5**
2	0	7	5	1	3	4	**6**
5	1	4	6	2	3	0	**7**

(a) Random decoding order, resulting in $8^2 = 64$ decodings

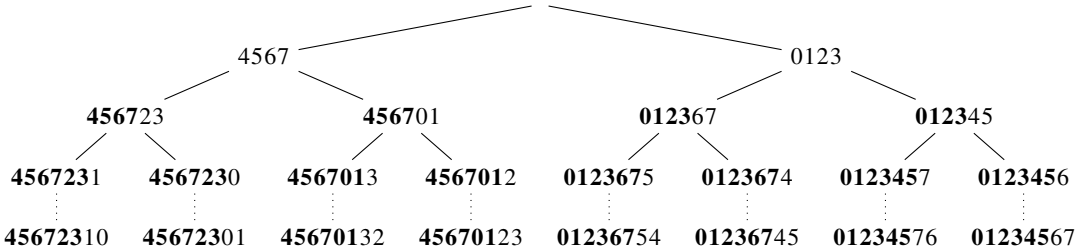
4	5	6	7	2	3	1	**0**
4	5	6	7	2	3	0	**1**
4	5	6	7	0	1	3	**2**
4	5	6	7	0	1	2	**3**
0	1	2	3	6	7	5	**4**
0	1	2	3	6	7	4	**5**
0	1	2	3	4	5	7	**6**
0	1	2	3	4	5	6	**7**

(b) Tied decoding orders for 8 residue system, resulting in 32 decodings

Figure 4: Comparison of decoding schemes which end in a particular residue being decoded last which are random vs tied to minimise compute required, decoding proceeds from left to right, with different final residues decoded in the far right column



(a) A particular merging of decoding order constraints for 8 residue system proceeding backwards,
!{XY...} defines any order in which all residues except X,Y,... are decoded



(b) A solution of decoding order constraints for 8 residue system proceeding forwards

Figure 5: Shared decoding order visualisation

Table 4: Ablation results for modifications of ProteinMPNN

Model	Tsuboyama		
	Success@10	auROC	auPRC
ProteinMPNN	66%	0.78	0.49
+ decode last	72%	0.79	0.52
+ shift by single residue prior for observed backbone (ProteinMPNN-ddG, ours)	**77%**	**0.82**	**0.56**

Table 5: Results with all mutations involving methionine removed

Model	Tsuboyama, S164524			S2648		S669	
	Success@10	auROC	auPRC	auROC	auPRC	auROC	auPRC
Unsupervised							
ProteinMPNN	65%	0.78	0.50	0.75	0.44	0.65	0.35
ProteinMPNN-ddG (ours)	**76%**	**0.82**	**0.56**	**0.80**	**0.52**	**0.71**	0.42
ESMif	64%	0.77	0.47	0.70	0.41	0.64	0.39
ESMif-ddG (ours)	69%	0.79	0.50	0.72	0.45	0.68	0.40
Supervised							
RaSP	51%	0.73	0.41	0.76	0.42	0.69	0.42
ACDC-NN-Seq	49%	0.70	0.37	0.75	0.43	0.70	0.39
ACDC-NN-Struct	42%	0.71	0.37	0.76	0.39	0.70	**0.43**
DDGun	49%	0.69	0.37	0.71	0.39	0.69	0.34
DDGun3D	41%	0.70	0.35	0.73	0.38	0.69	0.38