Summary We are grateful to the reviewers for their thoughtful feedback, and for acknowledging the novelty of our approach and its potential impact for computational protein design. Based on their suggestions, we benchmark our method against additional non-deep-learning, state-of-the-art baselines. Our method achieves competitive accuracy with significantly accelerated and streamlined computation.

Reviewer #1

Comparing to state-of-art baselines We have extended our experiments to include two benchmarks comparing against Rosetta, the leading framework for computational protein design. Following the suggestions of reviewer #4, we focus on 'native sequence recovery', which measures the model's ability to accurately recover sequences given a backbone structure alone. We evaluate native sequence recovery on two different training sets of proteins and find that our method is competitive on both (Table 1). In the first, we used the latest version of Rosetta (3.10) to design sequences specific to our test set with the *fixbb* fixed-backbone design protocol and default parameters (Table 1, left). In the second, we also compared to a prior benchmark from members of the Rosetta community (Kortemme group, PLOS one, 2015) across 40 diverse proteins (Table 1, right). To test our model against this, we re-split our dataset to form new training/validation sets that have no CATH topology overlap with their benchmark. This reduced the size of the training set from ∼18,000 chains to ∼10,000 chains, but we still found our model to be competitive with Rosetta.

We believe that achieving performance competitive with Rosetta (for this specific task) is a significant accomplishment, given it is built on several million lines of code developed by over 50 labs for two decades. We note that the Rosetta `fixbb` program emits an approximately 14,000-line usage message describing options if you add the flag `-help`.

Method	Recovery (%)	Speed (residues/s)
Rosetta 3.10 `fixbb`	17.9	4.88×10^{-1}
Ours ($T = 0.1$)	**28.5**	$\mathbf{1.08 \times 10^4}$

(a) Single chain test set

Method	Recovery (%)
Rosetta, `fixbb` 1	33.1
Rosetta, `fixbb` 2	38.4
Ours ($T = 0.1$)	**38.6**

(b) Ollikainen 40 benchmark

Table 1: **Evaluation against Rosetta for native sequence recovery** (Left) Our model more accurately recovers native sequences than Rosetta `fixbb` (median sequence similarity to native across 111 structures, 100 designs per structure). We note that these numbers are generally low because our test set is enriched for difficult examples that come from NMR-based templates. (Right) Evaluation with a prior benchmark of 40 structures, 100 designs per structure.

Reviewer #3

Error bars and attention ablations Thank you for these great suggestions. We agree that the presented framework might also be realized with message passing neural networks and that it would be interesting to understand the tradeoffs of non-attentive aggregation and other message nonlinearities. While we were not able to report those results at this point, we will include them in the camera-ready if accepted.

Explanation of SPIN2 Thank you for this suggestion. We will expand on our discussion of methods behind baselines (including Rosetta). Briefly, SPIN2 uses a neural network based on local molecular environment features (local angles, contacts, fragment profiles) to predict the identity of that specific amino acid (rather than the joint like ours).

Reviewer #4

Decoding strategies We found that we could generate sequences with considerably higher likelihoods than native state simply via biased sampling with a softmax temperature $T < 1$ (Figure 1). We agree that this is an important consideration and will add both the experiment and disscussion of strategies (such as beam search and top-k sampling) in the paper.

Benchmarking against Rosetta Please see our response to reviewer # 1.

Redesign We will discuss two methods for redesign: first, because the likelihood calculation is reasonably fast (16,000 residues / s on a consumer GPU), the log-likelihood could be used out-of-the-box for MCMC-based sampling (e.g. Gibbs). Second, the model could be retrained on randomized permutations at training time (Uria et al, ICML, 2014), and then conditioned at test time on autoregressive orderings that put the designed residues last.

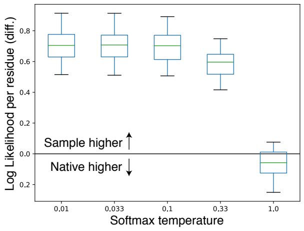

Figure 1: **Decoding.** Sampling with a low-temperature softmax (x-axis) generates sequences with higher normalized log likelihoods (y-axis) than native (horizontal line).