Broadly applicable and accurate protein design by integrating structure prediction networks and diffusion generative models

Joseph L. Watson[‡1,2], David Juergens[‡1,2,3], Nathaniel R. Bennett[‡1,2,3], Brian L. Trippe[‡2,4], Jason Yim[‡2,5], Helen E. Eisenach[1,2], Woody Ahern[1,2,6], Preetham Venkatesh[1,2,7], Susana Vázquez Torres[1,2,7], Andrew J. Borst[1,2], Basile I. M. Wicky[1,2], Robert J. Ragotte[1,2], Lukas F. Milles[1,2], Alexis Courbet[1,2,8], William Sheffler[1,2], Jue Wang[1,2], Isaac Sappington[1,2,7], Samuel J. Pellock[1,2], Nikita Hanikel[1,2], Anna Lauko[1,2,7], Regina Barzilay[5], Tommi S. Jaakkola[5], Frank DiMaio[1,2], Minkyung Baek[9], David Baker*[1,2,8]

‡Equal contribution
*To whom correspondence should be addressed

1. Department of Biochemistry, University of Washington, Seattle, WA 98105, USA
2. Institute for Protein Design, University of Washington, Seattle, WA 98105, USA
3. Graduate Program in Molecular Engineering, University of Washington, Seattle, WA 98105, USA
4. Columbia University, Department of Statistics, New York, NY 10027, USA
5. Massachusetts Institute of Technology, Cambridge, MA 02139, USA
6. Paul G. Allen School of Computer Science and Engineering, University of Washington, Seattle, WA 98105, USA
7. Graduate Program in Biological Physics, Structure and Design, University of Washington, Seattle, WA 98105, USA
8. Howard Hughes Medical Institute, University of Washington, Seattle, WA 98105, USA
9. School of Biological Sciences, Seoul National University, Seoul 08826, Republic of Korea

Abstract

Deep learning methods for protein design have shown considerable promise for sequence design[1–3], scaffolding functional sites[4,5], and building new monomers[6], cyclic oligomers[7], and antibody loops[8,9]. Despite this progress, a general framework for protein design that enables solution of a wide range of design challenges, including *de novo* binder design and design of higher order symmetric architectures, has yet to be described. Diffusion models[10,11] have had considerable success in image and language generative modeling, and have been applied to the protein monomer generation problem, but with limited success, likely due to the complexity of protein backbone geometry and sequence-structure relationships. Here we show that by utilizing powerful structure prediction methods as diffusion denoising networks, we can leverage the protein representations they have learned. We demonstrate state of the art performance on unconditional and topology constrained protein monomer design, protein and peptide binder design, symmetric oligomer design, enzyme active site scaffolding, and symmetric motif scaffolding for therapeutic and metal-binding protein design. We demonstrate the power and generality of the method, called RoseTTAFold Diffusion (RF*diffusion*), by experimentally characterizing hundreds of new designs. Highlights include a picomolar binder to parathyroid

hormone, considerably higher affinity than any previous computational designed binder prior to experimental optimization, and a series of not-previously-observed symmetric assemblies experimentally confirmed by electron microscopy. In a manner somewhat reminiscent of networks which produce images from user-specified inputs, RF*diffusion* makes accessible the design of diverse and complex protein architectures and functions from simple semantic molecular specifications.

Main

Denoising diffusion probabilistic models (DDPMs) have emerged as a powerful class of generative models to sample from complex data distributions[10,11]. DDPMs are trained to reconstruct data (for instance images or text) corrupted with varying amounts of added noise. After this training, new samples are generated by feeding the model random noise and then refining it by iterative application of the trained denoising network[10,11]. The power of DDPMs is illustrated in the context of computer graphics by the generation of novel, photorealistic images in response to text prompts[12,13]. DDPMs have a number of qualities that naturally lends them to protein design: they (1) generate highly diverse outputs, due to the stochasticity of the inputs and subsequent denoising trajectory (2) can be guided at each step of the iterative data generation process towards specific design objectives, either through provision of conditioning information or through external guide potentials, and (3) unlike methods that design proteins through generation or optimization of protein *sequences* alone[2,4,6,7,14], DDPMs can be formulated to generate protein *structures* directly, enabling more direct control over structural properties. Recent work has sought to adapt DDPMs for protein monomer design by conditioning on small protein "motifs"[5,9] or on secondary structure and adjacency ("fold") information[8]. While showing promise, these attempts have thus far had limited success in generating sequences that are predicted to fold to the intended structures *in silico*[5,15], and have not been tested experimentally.

We reasoned that improved diffusion models for protein design could be developed by taking advantage of the deep understanding of protein structure implicit in powerful structure prediction methods like AlphaFold2 (AF2) and RoseTTAFold (RF). The power of fine-tuning pretrained structure prediction networks for protein design was previously illustrated by a version of RoseTTAFold (called RF_{joint} Inpainting [4]) that was trained to recover missing sequence and structure information. Experimental characterization showed that the method can scaffold a wide range of protein functional sites with atomic accuracy[16], but the approach fails on minimalist site descriptions with insufficient topological information, and because it is deterministic, can produce only a limited diversity of designs for a given problem. We reasoned that by instead fine-tuning RoseTTAFold as the denoising network in a generative diffusion model, we could overcome both problems: because the starting point is random noise, each denoising trajectory yields a different solution, and because structure is built up progressively through many denoising iterations, little to no starting information should be required.

We formulate the diffusion model in a manner well-suited to fine-tuning from pre-trained RoseTTAFold (Fig 1A). As in ref [[8]], at each timestep we predict the final protein structure given the current noised structure. We then generate the slightly denoised input to the next timestep

via a noisy interpolation from the current (input) structure toward the predicted final structure. The correspondence between RoseTTAFold structure prediction and a RF*diffusion* denoising step is highlighted in Fig. 1A: in both cases, input sequence and structure information is transformed by the model into a prediction of the native protein structure. During classical structure prediction with RoseTTAFold, structural inputs to the model come from homologous template structures, each of which have associated per-residue "confidence" values[17]. In RF*diffusion*, structural inputs are derived from the partially (de-)noised structure, and the confidence feature is reparameterized to represent the current denoising timestep, on which the model conditions its prediction (see Methods 2.3). To generate noised protein structures for training or inference, we perform "forward" diffusion on all or some subset of the amino acid residues in a protein over backbone N-C_α-C frame translations and rotations. For translations, we perturb the C_α coordinates with 3D Gaussian noise. For rotations, we use Brownian motion on the manifold of rotation matrices, SO(3) (building on refs [18,19]). The noised structures are input to the network via the structure (3D) track of RF. We trained RF*diffusion* with losses similar to those described in previous work for image generation[10] (Fig. S1A, methods section 1.3). While in this study we use RoseTTAFold as the basis for the denoising network architecture, our approach is quite general, and it should be possible to substitute in other structure prediction networks that manipulate 3D coordinates (AF2[12], Omegafold[21], ESMfold[22], etc.).

We explored two different strategies for training RF*diffusion*: 1) in a manner akin to "canonical" diffusion models, with predictions at each timestep independent of predictions at previous timesteps (as in previous work[5,8,9,15]), and 2) with self-conditioning[23], where the model can condition on previous predictions between timesteps (Fig. 1A bottom row). The latter strategy was inspired by the success of "recycling" in both AF2 and RF_{joint} Inpainting. We found that self-conditioning within RF*diffusion* dramatically improved performance on *in silico* benchmarks encompassing both conditional and unconditional protein design tasks (Fig. S1B, Methods 3.1, 3.2). Fine-tuning RF*diffusion* from a pre-trained RF model was far more successful than training from scratch (Fig. S1C). For all *in silico* benchmarks in this paper, we use the AF2 structure prediction network[20] for validation and define *in silico* "success" as an RF*diffusion* output for which the AF2 structure predicted from a single sequence (1) has high confidence (mean predicted aligned error, pAE, < 5), (2) is globally within 2 Å backbone-RMSD of the designed structure, and (3) is within 1 Å backbone-RMSD on the scaffolded functional-site. We choose these metrics, which are more stringent than metrics described elsewhere[5,8,15,24] (e.g., TM score between design and subsequent structure prediction > 0.5, see Fig. S2A-B), because they have been demonstrated to be good predictors of experimental success[4,7,25]. Because RoseTTAFold and AF2 are different networks, AF2 serves as a reasonably independent arbitrator of the success of a design calculation. To design amino acid sequences that encoded the RF*diffusion*-generated backbones, we chose to use the ProteinMPNN network[1], which allows the rapid and robust generation of many high-quality sequences for each backbone. We generate 8 ProteinMPNN sequences per backbone, and select those predicted to fold to the target structure most accurately by AF2 (in line with previous work[5,15]).

Unconditional protein monomer generation

Unconstrained generation of diverse protein monomers is difficult to address with physically-based protein design methods due to the magnitude of the conformational sampling problem, and has been a primary test of deep learning based protein design approaches[5,6,8,14,15,26]. As illustrated in Fig. 1C-E, Fig. S3A, RF*diffusion* can readily generate complex protein structures with little overall structural similarity to any known protein structures, indicating considerable generalization beyond the PDB training set. The designs span a wide range of alpha-, beta- and mixed alpha-beta- topologies, with AF2 predictions very close to the design structure models for *de novo* designs with as many as 600 residues (we found that ESMFold[24] often more closely recapitulated the design structures - Fig. S1I, S2A, but given the experimental success in using AF2 for design validation[4,7,25], we used AF2 as the primary *in silico* validation for the design challenges described in this study). RF*diffusion* generates plausible structures for even very large proteins, but these are difficult to validate *in silico* as they are likely beyond the single sequence prediction capabilities of AF2. The quality and diversity of designs that are sampled is inherent to the model, and does not require *any* auxiliary conditioning input (for example secondary structure information[8]). RF*diffusion* strongly outperforms Hallucination (Fig. 1F), the only experimentally validated deep learning approach for unconditional generation, with success rates for Hallucination deteriorating beyond 100 amino acids. RF*diffusion* is also more compute efficient than unconstrained hallucination, requiring ~2.5 minutes on an NVIDIA RTX A4000 GPU to generate a 100 residue structure compared to ~8.5 minutes for Hallucination. Computational efficiency can be further improved by taking larger steps at inference time, and by truncating trajectories early - an advantage of predicting the *final* structure at each timestep (Fig. S2C-D). For design problems where a particular fold or architecture is desired (such as TIM barrels or cavity-containing NTF2s for small molecule binder and enzyme design[27,28]), we further fine-tuned RF*diffusion* to condition on (partial) input secondary structure and/or fold information, enabling rapid and accurate generation of diverse designs with the desired topologies or folds (Fig. S3B-D). *In silico* success rates were 42.5% and 54.1% for TIM barrels and NTF2 folds respectively (Fig. S3C).

Higher order oligomer design through denoising with explicit symmetrization

There is considerable interest in designing new higher order symmetric oligomers which can serve as vaccine platforms[29], delivery vehicles[30], and catalysts[31]. Cyclic oligomers have been generated using structure prediction networks by starting from a random sequence and carrying out a Monte Carlo search for sequences predicted to fold to the desired cyclic symmetry[7]. This “hallucination” approach fails with higher order dihedral, tetrahedral, octahedral, and icosahedral symmetries, likely because these architectures require multiple distinct sets of monomer-monomer interactions. We reasoned that this limitation could be overcome by leveraging two aspects of RF*diffusion*; first, RF*diffusion* acts directly on amino acid coordinates (as opposed to input sequence tokens) and so allows explicit symmetrization throughout the denoising process, and second the equivariance properties of the RosettaFold architecture with respect to global rotation of coordinate inputs and chain annotations ensures that the targeted symmetry is maintained in denoising predictions (see Methods 1.7). We experimented with

arranging multiple copies of a starting random Gaussian monomer coordinate distribution with the desired symmetry as the input, and explicitly symmetrizing the denoising updates at each step (Fig. 1B, second row). For octahedral and icosahedral architectures, to reduce the computational cost and memory footprint, we explicitly model only the smallest subset of monomers required to generate the full assembly (in the icosahedral case, the subunits at the five-fold, three-fold, and two-fold symmetry axes).

We found that despite not being trained on symmetric inputs, RF*diffusion* was able to generate higher order symmetric oligomers with high *in silico* success rates (Fig. S4B), particularly when guided by an auxiliary inter- and intra-chain contact potential (Fig. S4C). As illustrated in Fig. 2 and Fig. S4D,E, RF*diffusion*-generated cyclic (C3, C5, C6, C8, C10, C12), dihedral (D2, D3, D4, D5), tetrahedral, octahedral and icosahedral designs are nearly indistinguishable from AF2 predictions of the structures adopted by the designed sequences (for the full assemblies for the cyclic and dihedral designs, and trimeric substructures of the octahedral and icosahedral designs). These include a number of topologies not seen in nature, including two-layer beta strand barrels (Fig. 2A, bottom row) and complex mixed alpha/beta topologies (Fig. 2A). We selected 376 designs for experimental characterization, and found using size exclusion chromatography that at least 37 had oligomerization states closely consistent with the design models (Fig. S8, S9). We collected negative stain electron microscopy (nsEM) data on six of the 37 designs with the highest total molecular weights (ranging from 70-110 kilodaltons), and for all six, distinct particles were evident with shapes resembling the design models (Fig. 2C, and Fig S4D).

The structures of these assemblies are, to our knowledge, unprecedented in nature. HE0626 is a C6 hexameric ring composed of an inner ring of 18 strands and an outer ring of 18 helices. The helices are packed in a flower-shaped arrangement, and nsEM micrographs, 2D class averages, and 3D reconstruction are in agreement with the computational design model. The inner beta ring and the outer helical ring can be distinguished in both the 2D averages and in the 3D reconstruction. HE490 is a hexameric D3 ring composed of helical subunits and resembles a trimeric ring of dimers. The original micrographs and the 2D class averages for HE0490 have the overall triangular shape of this design, with the 3D reconstruction further confirming the overall topology. The side-view of the reconstruction shows the two distinct hemispheres represented by the dimeric substructures. HE0675 is a C8 octameric ring composed of an inner ring of 16 strands and an outer ring of 16 helices. The helices –similar to HE0626–form a flower-like arrangement, with somewhat more distinguishable lobes. The electron microscopy individual particle images, corresponding 2D class averages, and resulting 3D reconstruction are again closely consistent with the design model. HE0537 is a D4 octameric dihedral assembly resembling a dimer of tetramers with an overall rectangular prism shape (5x5x6 nanometers) formed by a largely alpha helical monomer. The electron microscopy images clearly indicate the rectangular prism shape in both top down and side views. The 3D reconstruction of HE0537 (Fig. 2C, bottom row) closely matches the design model, recapitulating the approximate 45° offset between tetramic subunits. Taken together, these data demonstrate the efficacy of RF*diffusion* for the accurate design of symmetric homo-oligomers across a wide range of symmetry groups and structural topologies.

Functional-site scaffolding with RFdiffusion

We next investigated the use of RF*diffusion* for scaffolding protein structural motifs that carry out binding and catalytic functions, where the role of the scaffold is to hold the site in precisely the 3D geometry needed for optimal function. A number of deep learning methods have been recently developed to address this problem, including RF_{joint} Inpainting[4], constrained hallucination[4], and diffusion generative models[5,8,24]. To rigorously evaluate the performance of these methods in comparison to RF*diffusion* across a representative set of design challenges, we established an *in silico* benchmark test comprising all functional site scaffolding design problems described in six recent publications[4,5,24,32–34] encompassing both deep learning-based and conventional design methodologies. There are 25 challenges in total, spanning a broad range of functional sites, including simple “inpainting” problems, viral epitopes, receptor traps, small molecule binding sites, binding interfaces and enzyme active sites. Full details of this benchmark are described in Table 1. RF*diffusion*, with no hyperparameter tuning or external potentials, on the problem set, outperforms Hallucination (where some preliminary optimization was used) and Inpainting in all but one design problem, and provides solutions to six problems for which hallucination and inpainting, even with the aid of ProteinMPNN, fail to generate successful designs under these conditions *in silico* (Fig. 3A-C). In 17/23 of the problems, RF*diffusion* generated successful solutions with higher success rates when noise was not added during the reverse diffusion trajectories (see Fig S1E-F for further discussion of the effect of noise on design quality).

Scaffolding enzyme active sites

A grand challenge in protein design is the ability to scaffold minimalist descriptions of enzyme active sites (typically just a few single amino acids). While some *in silico* success has been reported previously[4], a general solution that can readily produce high-quality, orthogonally-validated outputs is not currently available. Following fine tuning for 4 epochs on training examples involving scaffolding of the relative orientations and geometries of 2-3 residues close in Euclidean space, but discontinuous in sequence space, RF*diffusion* was able to scaffold enzyme active sites comprised of multiple sidechain and backbone functional groups with high accuracy and *in silico* success rates across a range of enzyme classes (Fig. 3D-F), illustrating the ease with which RF*diffusion* can be fine-tuned to solve problems beyond those in the original training set. While RF*diffusion* is currently unable to *explicitly* model bound small molecules (see conclusion), the substrate can be *implicitly* modeled using an external potential to guide the generation of “pockets” around the active site. As a demonstration we scaffold the triadic active site of a retro-aldolase while implicitly modeling its substrate (Fig. S5).

Symmetric functional-site scaffolding for metal mediated assemblies and antiviral therapeutics and vaccines

A number of important design challenges involve the scaffolding of multiple copies of a functional motif in symmetric arrangements. For example, many viral glycoproteins are trimeric, and symmetry matched arrangements of inhibitory domains can be extremely potent[35–38].

Conversely, symmetric presentation of viral epitopes in an arrangement that mimics the virus could induce new classes of neutralizing antibodies[39,40]. To explore this general direction, we sought to design trimeric multivalent binders to the SARS-CoV-2 spike protein. In previous work, flexible linkage of a design that binds to the ACE2 binding site on the receptor binding domain of the spike to a trimerization domain yielded a high-affinity inhibitor that had potent and broadly neutralizing antiviral activity in animal models[35]. Rigidly fusing or oligomerizing the binder could in principle improve its affinity for the target by reducing the entropic cost of binding while maintaining the avidity benefits from multivalency. We used RF*diffusion* to design C3 symmetric trimers which rigidly hold three binding domains (the "functional-site" in this case) so they exactly match the ACE2 binding sites on the SARS-CoV-2 spike protein trimer. Design models were confidently recapitulated by AF2 to both assemble as C3-symmetric oligomers, and to scaffold the AHB2 SARS-CoV-2 binder interface with sub-angstrom accuracy (Fig. 3G).

The ability to scaffold functional sites with any desired symmetry opens up new approaches to designing metal coordinating protein assemblies. Divalent metal ions exhibit distinct preferences for specific coordination geometries - square planar (C4), tetrahedral, and octahedral - with ion-specific optimal sidechain-metal bond lengths. RF*diffusion* provides a general route to building up symmetric protein assemblies around such sites. As a first test of this, we sought to design square planar nickel binding sites. We designed C4 protein assemblies with four central histidine imidazoles arranged in ideal nickel binding geometry. Designs starting from six different C4-symmetric histidine functional (Fig. 3H, Fig. S6A) sites showed high *in silico* design success rates (Fig. S6C), with the histidine residues in near ideal geometries for coordinating metal in the AF2 predicted structures (Fig. 3H rightmost panel, Fig S6B,D)

De novo protein and peptide binder design

The design of high-affinity binders to target proteins is a grand challenge in protein design, with numerous therapeutic applications[41]. The ability to design *de novo* binders using the physically based Rosetta method was recently described[42], and subsequently, the utility of ProteinMPNN and AF2 for sequence design and design filtering respectively has improved design success rates[25]. However, experimental success rates are typically low, requiring many thousands of designs to be screened for each design campaign[42]. Further, this work relied on pre-specifying a particular set of protein scaffolds as the basis for the designs, inherently limiting the diversity and shape complementarity of possible solutions[42]. We reasoned that RF*diffusion* might be able to address this challenge by directly generating diverse and target-compatible protein binders. To our knowledge, no deep-learning method has yet demonstrated general experimental success in designing completely *de novo* binders.

For many therapeutic applications, for example blocking a protein-protein interaction, it is desirable to bind to a particular site on a target protein. To enable this, we fine-tuned RF*diffusion* on protein complex structures, providing as input a subset of the residues on the target chain to which the diffused chain binds (Fig. S7A, B, see Methods 2.5). With this fine-tuned model, we were able to design putative binders confidently predicted by AF2 to bind their target[25]. These could be generated without any fold/topology information, with success rates several orders of

magnitude higher than with our previous Rosetta-based approach (Fig. 4A-B). To enable control over binder scaffold topology, we also fine-tuned a model to condition binder diffusion on secondary structure adjacency information[8] (Fig. S7C, D), and in cases where compatible folds for putative binders were known, this model typically further improved *in silico* success rates (Fig. 4B, bottom row).

An outstanding challenge in protein design is the design of binders to flexible helical peptides, which are challenging targets due to their general lack of structure in solution and therefore the entropic cost of binding in a rigid conformation. For two such peptides, the apoptosis-related peptide Bim and parathyroid hormone (PTH), we experimented with unconditional binder design - providing RF*diffusion* only with the sequence and structures of the two peptides in helical conformations, and leaving the topology of the binding protein and the binding mode completely unspecified. From this minimal starting information, RF*diffusion* generated designs predicted by AF2 to fold and bind to the targets with high *in silico* success rates. We obtained synthetic genes encoding 96 designs for each target. Using yeast surface display, we found that 25 of the 96 designs bound to Bim (10nM, no avidity). The highest affinity design (Fig. 4C), which purified as a soluble monomer, bound too tightly for steady state estimates of the dissociation constant (K_D); global fitting of the association and dissociation kinetics suggest a K_D of ~100pM (Fig 4D). For parathyroid hormone, we found that 56/96 of the designs bound by yeast surface display with sub-micromolar affinities. The highest affinity design (Fig. 4E) again bound too tightly for accurate K_D estimation; instead fluorescence polarization data provides an approximate upper bound for the K_D of 350pM (Fig. 4F). To our knowledge, these Bim and PTH binding proteins are the highest affinity binders to any target (protein, peptide, or small molecule) achieved directly by computational design with no experimental optimization.

Conclusion

RFdiffusion is a major improvement over current physically-based and deep learning protein design methods over a wide range of design challenges. Substantial progress was recently made using Rosetta in designing binding proteins from target structural information alone, but this required testing tens of thousands of designs – with RF*diffusion* high affinity binders to the targets experimentally characterized here can be identified through testing of dozens of designs (more experimental testing will be required to determine success rates over a broader range of targets). There has also been progress in scaffolding protein functional motifs using deep learning methods (hallucination, inpainting and diffusion), but hallucination becomes very slow for complex systems, inpainting fails when insufficient starting information is provided, and previous diffusion methods had quite low accuracy; our benchmark tests show that RF*diffusion* considerably outperforms all previous methods in the complexity of the motifs that can be scaffolded, the ability to precisely position sidechains (for catalysis and other functions), and the accuracy of motif recapitulation by AF2. For the classic unconstrained protein structure generation problem, RF*diffusion* readily generates novel protein structures with as many as 600 residues that are accurately predicted by AF2 (and ESMFold), far exceeding the complexity and accuracy achieved by previously described diffusion and other methods. The versatility and control provided by diffusion models enabled extension of RF*diffusion* unconditional generation

to higher order architectures with any desired symmetry (hallucination methods are primarily limited to cyclic symmetries); experimental characterization of a subset of these designs using electron microscopy revealed structures very similar to the design models (which are without precedent in nature). Combining the accurate motif scaffolding with the ability to design symmetric assemblies, we were able to scaffold functional sites spanning multiple symmetrically arranged chains which has not been previously possible. Overall, the complexity of the problems solvable with RF*diffusion* and the robustness and accuracy of the solutions (validated both *in silico* and experimentally) far exceeds what has been achieved previously. In a manner somewhat reminiscent of the generation of images from text prompts, RF*diffusion* makes possible, with minimal specialist knowledge, the generation of proteins from very simple, semantic specifications (for example, from a specification of target peptide, high affinity binders to that peptide, and from specification of a desired symmetry, diverse protein assemblies with that symmetry).

The power and scope of RF*diffusion* can be extended in several directions. RF has recently been extended to nucleic acids and protein-nucleic acid complexes[17], which should enable RF*diffusion* to design nucleic acid binding proteins, and perhaps folded RNA structures. Extension of RF to incorporate ligands should similarly enable RF*diffusion* to design small molecule binding proteins. The ability to customize RF*diffusion* to specific design challenges by addition of external potentials and by fine-tuning (as illustrated here for catalytic site scaffolding, binder-targeting and fold-specification), along with continued improvements to the underlying methodology, should enable protein design to achieve still higher levels of complexity, to approach and in some cases surpass what natural evolution has achieved.

Acknowledgements

We thank Namrata Anand, Doug Tischer, Valentin De Bortoli and Emile Mathieu for helpful discussions. We thank Ian Haydon for help with graphics. We also thank Luki Goldschmidt and Kandise VanWormer, respectively, for maintaining the computational and wet lab resources at the Institute for Protein Design.

This work was supported by gifts from Microsoft (D.J., M.B., D.B.), Amgen (J.L.W.), the Audacious Project at the Institute for Protein Design (B.L.T., I.S., J.Y., H.E., D.B.), the Washington State General Operating Fund supporting the Institute for Protein Design (P.V., I.S.), grant INV-010680 from the Bill and Melinda Gates Foundation Grant (W.B.A., D.J., J.W., D.B.), grant DE-SC0018940 MOD03 from the U.S. Department of Energy Office of Science (A.J.B., D.B.), grant 5U19AG065156-02 from the National Institute for Aging (S.V.T., D.B.), an EMBO long-term fellowship ALTF 139-2018 (B.I.M.W.), the Open Philanthropy Project Improving Protein Design Fund (R.J.R., D.B.), The Donald and Jo Anne Petersen Endowment for Accelerating Advancements in Alzheimer's Disease Research (N.R.B.), a Washington Research Foundation Fellowship (S.J.P.), a Human Frontier Science Program Cross Disciplinary Fellowship (LT000395/2020-C, L.F.M.), an EMBO Non-Stipendiary Fellowship (ALTF 1047-2019, L.F.M.), the Defense Threat Reduction Agency grants HDTRA1-19-1-0003 (N.H., D.B.) and HDTRA12210012 (F.D.), the Institute for Protein Design Breakthrough Fund (A.C., D.B.), an EMBO Postdoctoral Fellowship (ALTF 292-2022, J.L.W.) and the Howard Hughes Medical Institute (A.C., W.S., R.R., D.B.), an NFS-GRFP (J.Y), an NSF Expeditions grant (1918839, J.Y, R.B., T.S.J.), the Machine Learning for Pharmaceutical Discovery and Synthesis consortium (J.Y, R.B., T.S.J.), the Abdul Latif Jameel Clinic for Machine Learning in Health (J.Y, R.B., T.S.J), the DTRA Discovery of Medical Countermeasures Against New and Emerging threats program (J.Y, R.B., T.S.J), the DARPA Accelerated Molecular Discovery program and the Sanofi Computational Antibody Design grant (J.Y, R.B., T.S.J.). We thank Microsoft and AWS for generous gifts of cloud computing resources.

Figures

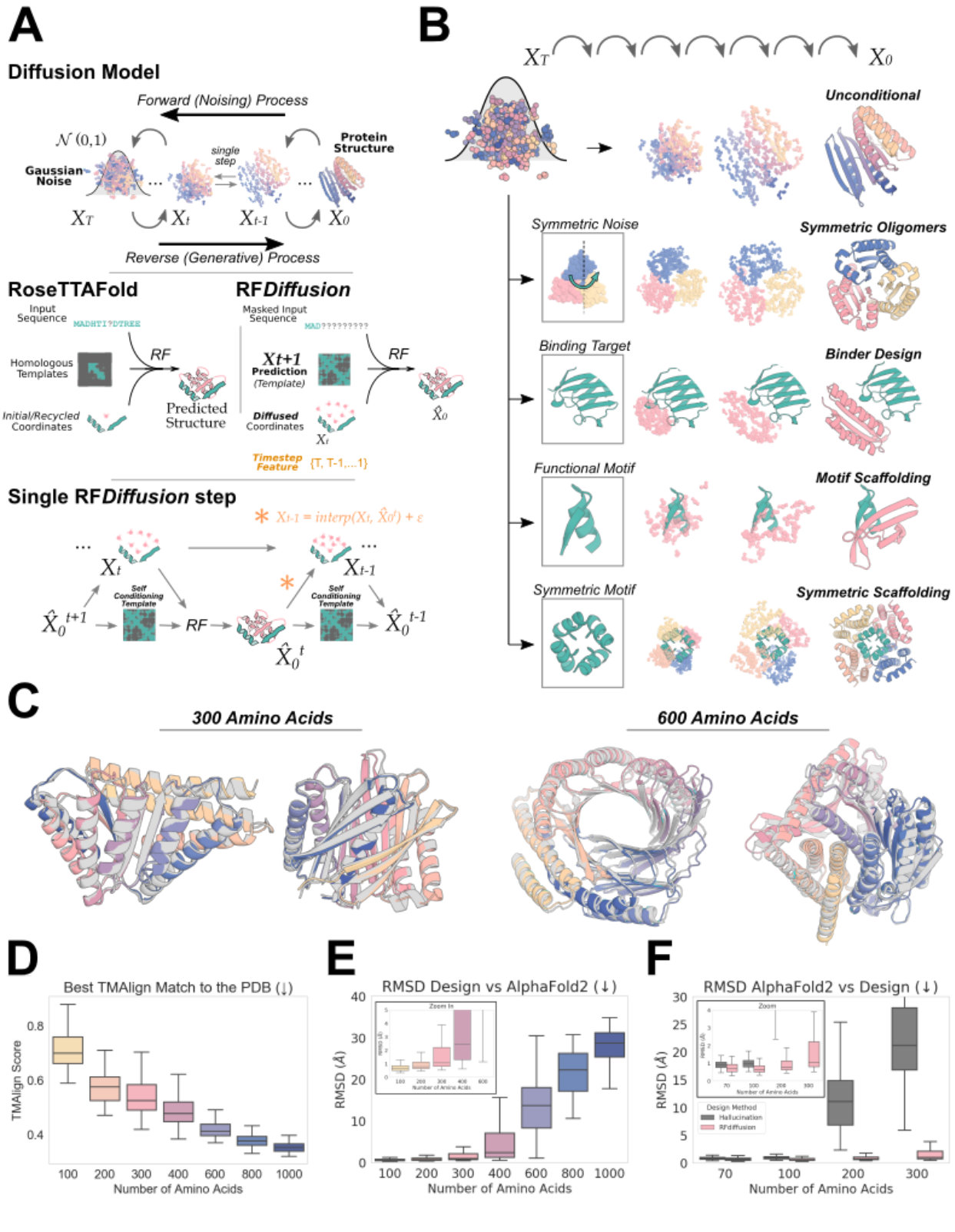

A
Diffusion Model
Forward (Noising) Process
Gaussian Noise
single step
Protein Structure
Reverse (Generative) Process
RoseTTAFold
RFDiffusion
Input Sequence
Homologous Templates
Initial/Recycled Coordinates
Predicted Structure
Masked Input Sequence
Prediction (Template)
Diffused Coordinates
Timestep Feature
Single RFDiffusion step
Self Conditioning Template
B
Unconditional
Symmetric Noise
Symmetric Oligomers
Binding Target
Binder Design
Functional Motif
Motif Scaffolding
Symmetric Motif
Symmetric Scaffolding
C
300 Amino Acids
600 Amino Acids
D
Best TMAlign Match to the PDB (↓)
TMAlign Score
Number of Amino Acids
E
RMSD Design vs AlphaFold2 (↓)
RMSD (Å)
Number of Amino Acids
F
RMSD AlphaFold2 vs Design (↓)
Design Method
Hallucination
RFdiffusion
Number of Amino Acids

Figure 1: Incorporating diffusion into RoseTTAFold addresses a broad range of protein design problems. A) Top panel: Diffusion models for proteins are trained to recover structures of proteins corrupted with noise, and generate new structures by reversing the corruption process through iterative denoising of initially random noise X_T into a realistic structure X_0. Middle panel: Diffusion models can be incorporated into RoseTTAFold (RF, left), a rotationally equivariant protein structure prediction network that maps sequence, template structures and initial coordinates into accurate structure predictions. RF*diffusion* (right) is trained from a *pre-trained* RF network with minimal architectural changes. The input sequence is (partially) masked and the model's previous prediction provided in place of the template ("self-conditioning", see Methods 1.6). A noised structure at timestep "t" (X_t) is provided as input coordinates, with the timestep also provided to the model. The output from RF*diffusion*, just as in RF, is the prediction of the true protein structure (now denoted X_0). Bottom panel: At each timestep "t" of a design trajectory (typically 200 steps), RF*diffusion* takes X_t and $\hat{X}_0^{t+1}$ from the previous step and then predicts an updated X_0 structure ($\hat{X}_0^{t}$) and the coordinate input to the model at the next time step (X_{t-1}) is generated by a noisy interpolation toward $\hat{X}_0^{t}$. **B)** RF*diffusion* is of broad applicability to protein design. RF*diffusion* generates protein structures either without conditioning (top row), or conditioning on: symmetric inputs to design symmetric oligomers (second row); a structure of a binding target (third row); protein functional sites (fourth row); symmetric functional sites to design symmetric oligomers scaffolds (bottom row). In each case, random noise, along with conditioning information, is input to RF*diffusion*, which iteratively refines that noise until a final protein is designed. **C)** RF*diffusion* can generate new monomeric proteins of different lengths (left: 300, right: 600) with no conditioning information. Gray=design model; colors= AlphaFold2 (AF2) prediction. RMSD AF2 vs design (Å), left to right: 0.90, 0.98, 1.15, 1.67. **D)** Unconditional designs from RF*diffusion* are novel and not present in the training set as quantified by highest TM score to the protein databank (PDB). Designs are increasingly diverse with increasing length. **E)** Unconditional samples are closely re-predicted by AF2. Beyond 400 amino acids, the recapitulation by AF2 deteriorates. **F)** RF*diffusion* significantly outperforms hallucination (with RoseTTAFold) at unconditional monomer generation (two-way ANOVA & Tukey's test, $p < 0.001$). While hallucination successfully generates designs up to 100 amino acids in length, success rates rapidly deteriorate beyond this length.

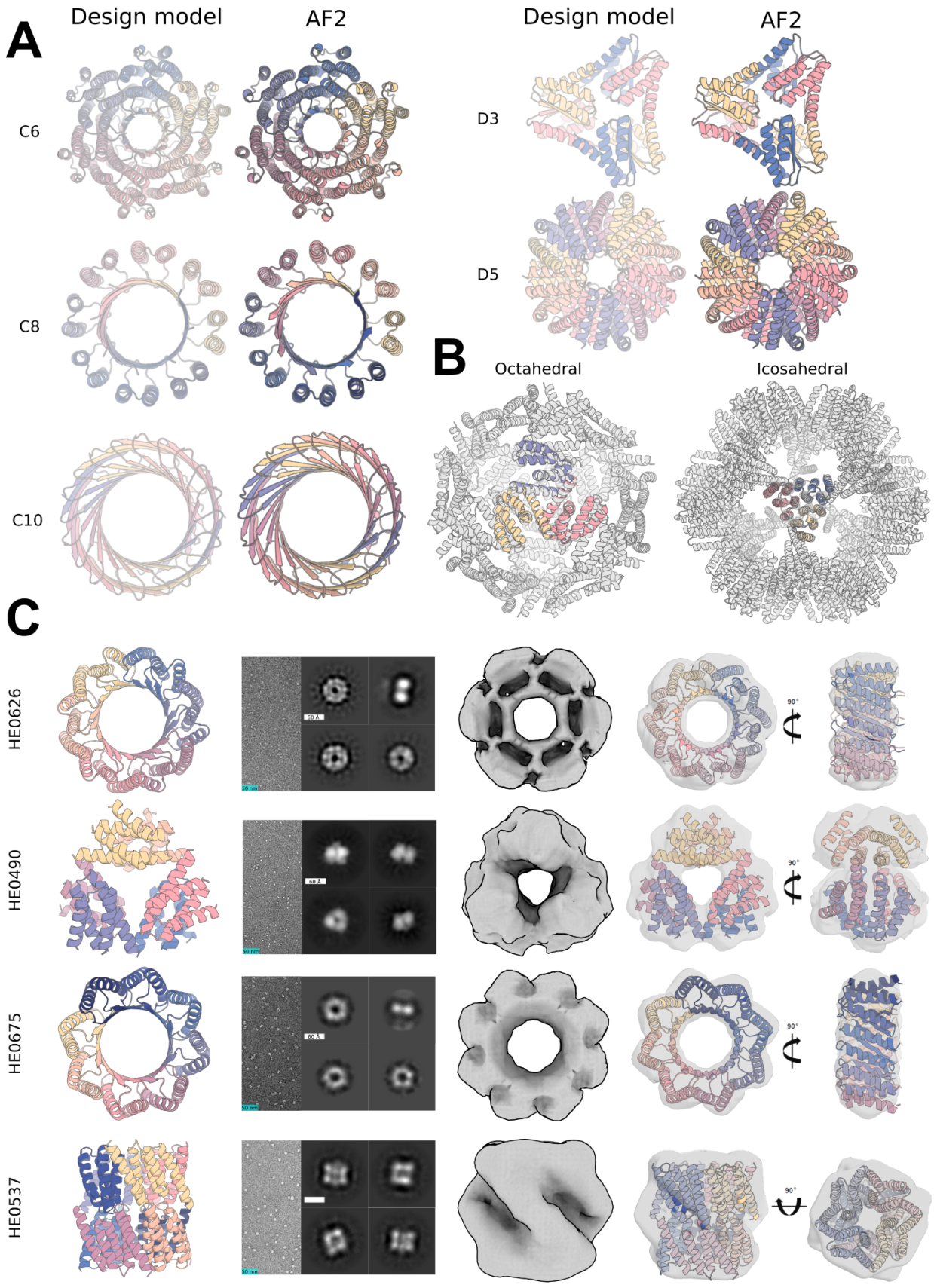

A
Design model
AF2
C6
C8
C10
Design model
AF2
D3
D5
B
Octahedral
Icosahedral
C
HE0626
HE0490
HE0675
HE0537
60 Å
50 nm
90°

Figure 2: Design and experimental validation of high-order symmetric oligomers. A) RF*diffusion*-generated cyclic and dihedral assemblies (left) compared to AF2 structure predictions based on the designed sequences (right); in all 5 cases they are nearly indistinguishable (backbone RMSDs vs AF2 for C6, C8, C10, D3, D5 are 1.04, 0.45, 0.60, 0.66, 0.72, respectively, with total amino acids 1200, 480, 600, 480, 1000, respectively). Symmetries are indicated to the left of the design models. **B)** Octahedral (left) and icosahedral (right) assemblies generated by RF*diffusion* (gray). These structures are too large to be predicted by AF2 in their entirety; instead AF2 predictions for trimeric substructures are shown superimposed on the models (colors). **C)** Designed assemblies validated by single molecule electron microscopy. From left to right: 1) symmetric design model, 2) raw micrographs and 2D particle class averages demonstrating homogeneous samples, 3) 3D reconstructions from class averages, and 4) AF2 predictions fitted into 3D reconstruction. The overall shapes are closely consistent with the design models. As in **A)**, the AF2 predictions of each design are nearly indistinguishable from the original diffusion model (backbone RMSDs for HE0626, HE0490, HE0675, HE0537, are 1.03, 0.60 0.74, 0.75, respectively). Model symmetries from top to bottom are C6 (HE0626, 100 AA/ chain), D3 (HE0490, 80 AA/ chain), C8 (HE0675, 60 AA/ chain), and D4 (HE0537, 100 AA/ chain).

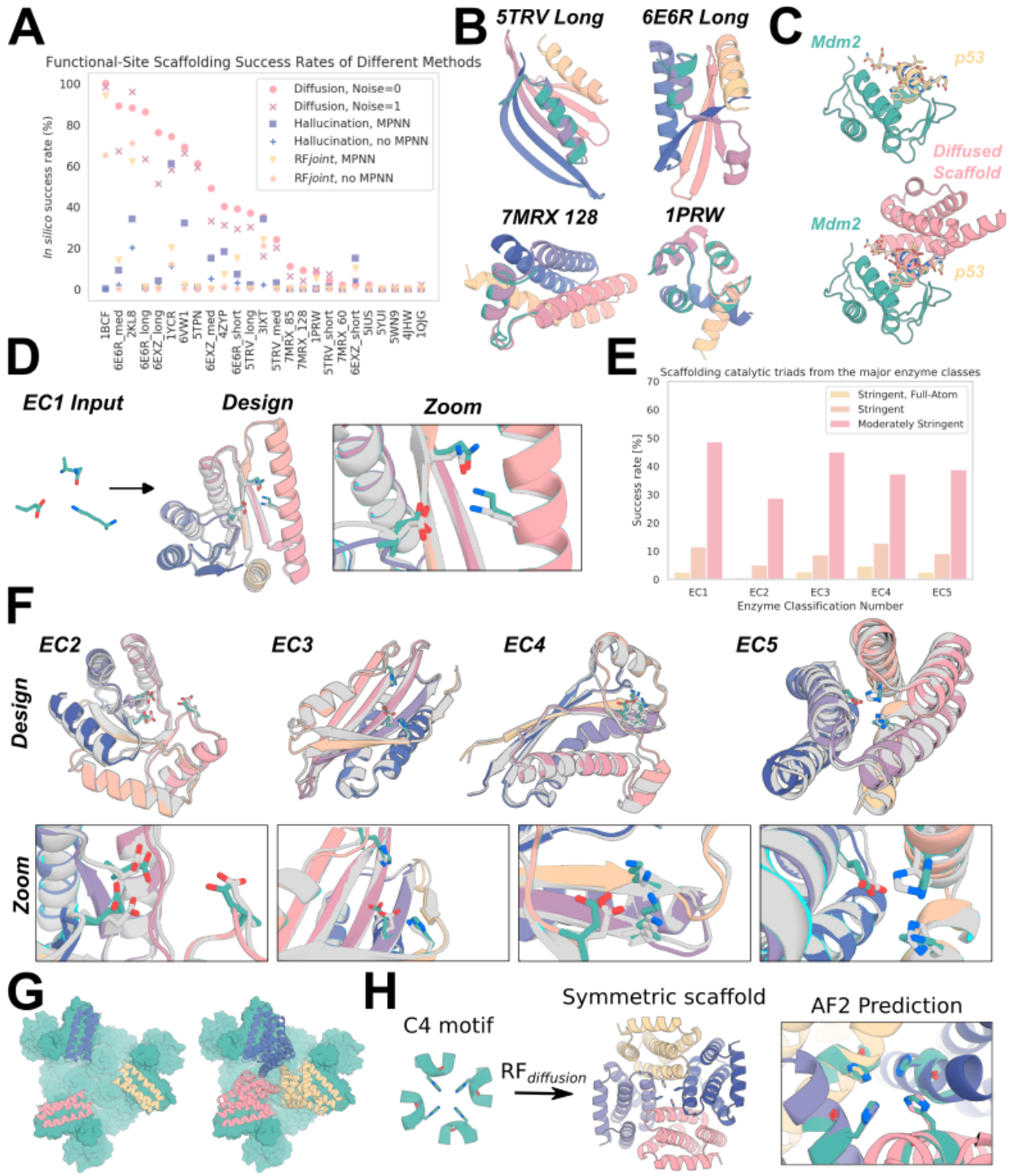
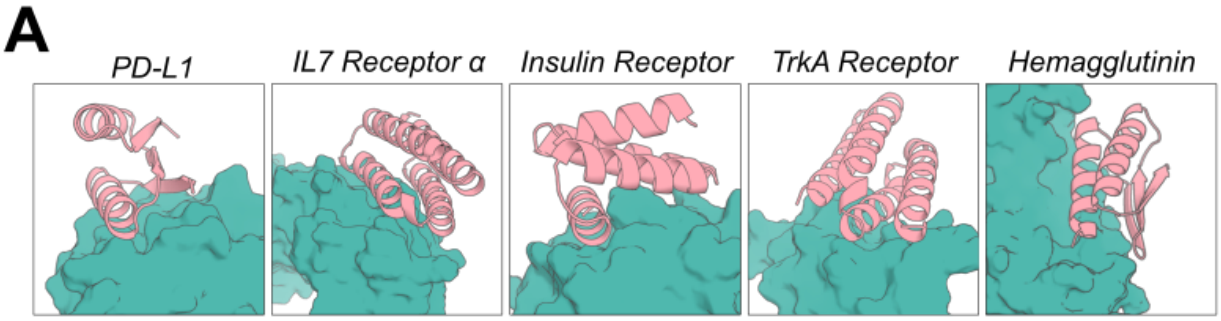

A
Functional-Site Scaffolding Success Rates of Different Methods
In silico success rate (%)
Diffusion, Noise=0
Diffusion, Noise=1
Hallucination, MPNN
Hallucination, no MPNN
RFjoint, MPNN
RFjoint, no MPNN
1BCF
6E6R_med
2KL8
6E6R_long
6EXZ_long
1YCR
6VW1
5TPN
6EXZ_med
4ZYP
6E6R_short
5TRV_long
3IXT
5TRV_med
7MRX_85
7MRX_128
1PRW
5TRV_short
7MRX_60
6EXZ_short
5IUS
5YUI
5WN9
4JHW
1QJG
B
5TRV Long
6E6R Long
7MRX 128
1PRW
C
Mdm2
p53
Diffused Scaffold
Mdm2
p53
D
EC1 Input
Design
Zoom
E
Scaffolding catalytic triads from the major enzyme classes
Stringent, Full-Atom
Stringent
Moderately Stringent
Success rate [%]
EC1
EC2
EC3
EC4
EC5
Enzyme Classification Number
F
EC2
EC3
EC4
EC5
Design
Zoom
G
H
C4 motif
RFdiffusion
Symmetric scaffold
AF2 Prediction

Figure 3 - Scaffolding of diverse functional-sites with RF*diffusion.* A) RF*diffusion* is state of the art across a diverse set of benchmark functional-site scaffolding problems. The 25 problems, collected from six recent publications, encompass a broad range of functional sites, including enzyme active sites, binding interfaces and viral epitopes (Table 1). Success was defined as AF2 RMSD to design model < 2 Å , AF2 RMSD to the native functional site (the "motif") < 1 Å , and AF2 predicted alignment error (pAE) < 5, and the examples are ordered by success rate with RF*diffusion* (with noise scale = 0). 100 designs were generated per problem, with no prior optimization on the benchmark set (some optimization was necessary and permitted for the hallucination data). RF*diffusion* solves 23/25 problems, outperforming existing methods in all but one. Table 2 presents full results. **B)** Four examples of designs for benchmarking problems where RF*diffusion* significantly outperforms existing methods. Teal: native motif; colors: AF2 prediction of an RF*diffusion* design. Metrics (RMSD AF2 vs Design, RMSD AF2 vs native motif, AF2 pAE): 5TRV Long: 1.17 Å , 0.57 Å , 4.73; 6E6R Long: 0.89 Å , 0.27 Å , 4.56; 7MRX Long: 0.84 Å , 0.82 Å 4.32; 1PRW: 0.77 Å , 0.89 Å , 4.49. **C)** RF*diffusion* can scaffold the native p53 helix that binds to Mdm2 and makes additional contacts with the target. The designed scaffold (pink) is confidently predicted to interact with Mdm2 by AF2, and to scaffold the native p53 helix with atomic accuracy (Interaction pAE: 4.65, Monomer pAE: 4.93, AF2 Motif RMSD: 0.52 Å , AF2 vs design RMSD: 0.43 Å). *In silico* successful designs had, on average, 31% higher contacting surface area than the original helix. **D)** RF*diffusion* can be fine-tuned for specific and highly challenging design tasks, including the design of scaffolds supporting minimalist enzyme active sites (see Methods 2.6). The input to RF*diffusion* is a few individual residues (left, in this case from the first enzyme class) and the network scaffolds these sites, with designs often accurately re-predicted by AF2 (middle and right, gray: design model; colors: AF2 prediction. Motif backbone RMSD 0.53 Å , Motif full-atom RMSD 1.05 Å , AF2 vs Design RMSD: 0.88 Å ; AF2 pAE: 4.47). **E)** After fine-tuning on motifs close in Euclidean space, but discontinuous in sequence-space, RF*diffusion* is able to scaffold a broad range of enzyme active sites from the five major enzyme classes (a random triadic active site from an apo structure belonging to each of the five classes in the M-CSA database[43]). Three degrees of stringency for success are reported: *Stringent, Full-Atom/Stringent, Backbone/Moderately Stringent*: AF2 vs design RMSD (backbone) < 2 Å /2 Å /3 Å ; AF2 vs design Motif RMSD (backbone) < 1 Å /1 Å /1.5 Å , AF2 pAE < 5/5/7.5; AF2 vs design Motif RMSD (full-atom) 1.5 Å /na/na. For all cases, RF*diffusion* generated designs that passed our most stringent filters (EC1: 2.6%; EC2: 0.6%; EC3: 2.7%; EC4: 4.7%; EC5: 2.6%). Without fine-tuning RF*diffusion,* produces no successful designs passing the Full-Atom/Stringent filter for these enzyme benchmarks. **F)** An example (top row) and zoomed view (bottom row) of successful designs generated to the other four enzyme classes, demonstrating high-accuracy scaffolding of the active sites. Gray: design model, colors: AF2 model, Teal: motif structure prediction. Metrics (AF2 vs design backbone RMSD, AF2 vs design motif backbone RMSD, AF2 vs design motif full-atom RMSD, AF2 pAE): EC2: 0.93 Å , 0.50 Å , 1.29 Å , 3.51; EC3: 0.92 Å , 0.60 Å , 1.07 Å , 4.59; EC4: 0.93 Å , 0.80 Å , 1.03 Å , 4.41; EC5: 0.78 Å , 0.44 Å , 1.14 Å , 3.32. **G-H)** RF*diffusion* can scaffold symmetric functional sites. **G)** The SARS-CoV-2 spike protein is a C3-symmetric trimer. AHB2, a previously-described ACE2 mimic, can bind to a single spike protein subunit. To increase avidity, we symmetrized AHB2 around the C3 axis (left) and used RF*diffusion* to design bespoke C3-symmetric oligomers to allow rigid scaffolding of the AHB2 interface in a position well-suited to interacting with all three

spike subunits (right). Teal: SARS-CoV-2 structure (from PDB: 7JZL), colors: symmetrized AHB2 (left) and AF2 model of RF*diffusion* design (right). Metrics: AF2 pAE (monomer): 7.18; AF2 RMSD vs design (monomer/triple): 1.07 Å /1.28 Å ; AF2 Motif RMSD (monomer/triple): 0.53 Å /2.36 Å . **H)** Nickel can be coordinated in a square-planar geometry. We generated C4 symmetric motifs scaffolding histidine residues in positions ideal for coordinating nickel (left). RF*diffusion* generates C4 symmetric oligomers scaffolding these motifs, with AF2 predictions recapitulating the site with high confidence (pAE < 10) and full-atom RMSD on the coordinating histidine residues between AF2 and the ideal motif < 1Å.



B ***In Silico* Success Rate (%) [raw / +FastRelax]**

	Noise Scale	PD-L1	IL7 Receptor α	Insulin Receptor	TrkA Receptor	Hemagglutinin
Unconditional	0.0	39.3/47.6	9.1/18.8	**17.9**/17.9	14.7/12.1	4.2/3.9
	0.5	25.4/29.9	6.6/9.9	7.1/6.7	6.5/6.0	0.6/1.0
	1.0	10.7/14.4	2.1/4.4	2.8/3.2	1.5/1.6	0.1/0.1
Fold-Conditioned	0.0	44.0/46.0	11.7/**21.9**	10.6/10.8	**29.2**/24.2	**14.6**/13.9
	0.5	37.0/**49.0**	11.9/15.0	3.7/3.7	16.3/15.2	3.2/4.7
	1.0	22.0/26.0	5.3/10.7	2.5/3.0	6.6/7.2	0.2/2.3

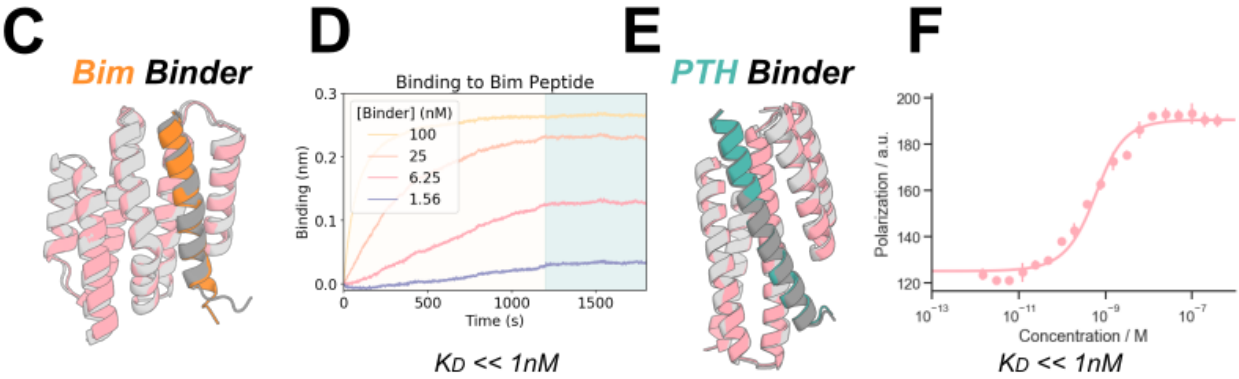


Figure 4: Design of protein and peptide binders. A-B) *De novo* binders were designed to five protein targets; PD-L1, IL7 Receptor α, Insulin Receptor, TrkA receptor and Influenza Hemagglutinin, and tested *in silico* with AF2 prediction. **A)** An example structure for each of the five targets, highlighting the diversity and complementarity of designs to their respective targets. AF2 models are shown (teal: target, pink: design). Metrics (Monomer pLDDT, Interaction pAE, Monomer RMSD AF2 vs Design): PD-L1: 87.9, 4.35, 0.56 Å ; IL7-Rα: 94.9, 7.33, 0.23 Å ; Insulin: 94.0, 4.84, 0.37 Å ; TrkA Receptor: 95.3, 4.62, 0.37 Å ; Hemagglutinin: 91.9, 9.20, 0.71 Å . **B)** Full *in silico* success rates for the protein binders designed to five targets. In each case, the best fold-conditioned results are shown (i.e. from the most target-compatible input fold), and the success rates at each noise scale are shown. In line with current best practice[25], we tested using Rosetta FastRelax[44] before designing the sequence with ProteinMPNN, but found that this did not systematically improve designs. Success is defined in line with current best practices[25]: AF2 pLDDT of the monomer > 80, AF2 interaction pAE < 10, AF2 RMSD monomer vs design < 1 Å . **C-F)** RF*diffusion* can design binders to helical peptides. **C)** Design model (gray) and AF2 prediction (colors) of an experimentally validated binder to the apoptosis-related peptide Bim. Orange: Bim peptide, Pink: designed binder. Metrics: RMSD AF2 vs Design: 0.80 Å ; interaction pAE: 4.50; Binder pLDDT: 96.6 **D)** Biolayer interferometry measurement of Bim binding indicate a sub-nanomolar affinity, and notably slow dissociation kinetics. **E)** Design model (gray) and AF2 prediction (colors) of an experimentally validated binder to the helical peptide parathyroid hormone (PTH). Teal: PTH peptide, Pink: designed binder. Metrics: RMSD AF2 vs Design: 0.78 Å ; interaction pAE: 4.40; Binder pLDDT: 94.3. **F)** Fluorescence polarization measurements with TAMRA-labeled PTH indicate a sub-nanomolar binding affinity.

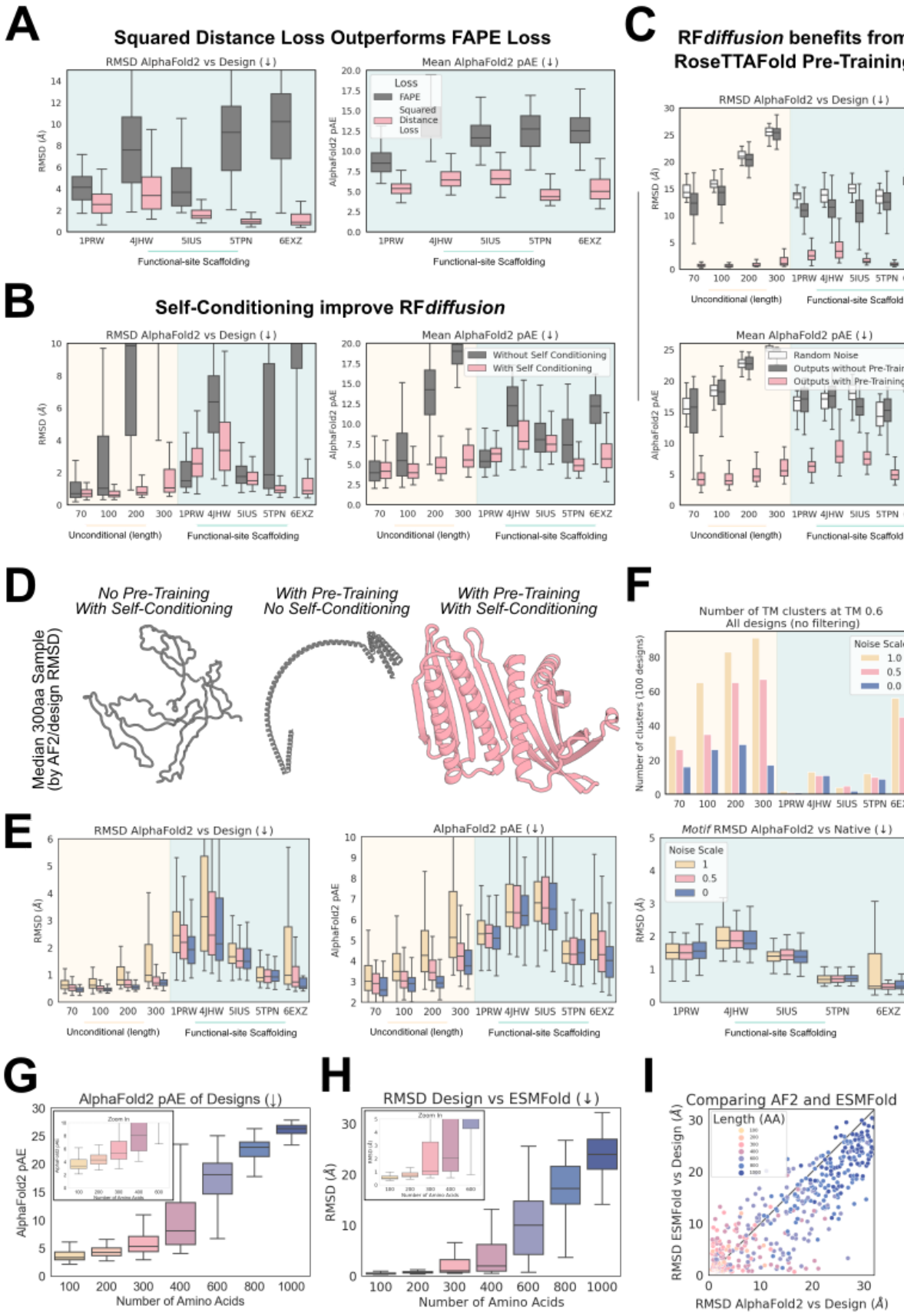

A
Squared Distance Loss Outperforms FAPE Loss
RMSD AlphaFold2 vs Design (↓)
RMSD (Å)
Mean AlphaFold2 pAE (↓)
AlphaFold2 pAE
Loss
FAPE
Squared Distance Loss
1PRW
4JHW
5IUS
5TPN
6EXZ
Functional-site Scaffolding
B
Self-Conditioning improve RFdiffusion
Without Self Conditioning
With Self Conditioning
70
100
200
300
Unconditional (length)
C
RFdiffusion benefits from RoseTTAFold Pre-Training
Random Noise
Outputs without Pre-Training
Outputs with Pre-Training
D
No Pre-Training With Self-Conditioning
With Pre-Training No Self-Conditioning
With Pre-Training With Self-Conditioning
Median 300aa Sample (by AF2/design RMSD)
E
AlphaFold2 pAE (↓)
Motif RMSD AlphaFold2 vs Native (↓)
Noise Scale
1
0.5
0
F
Number of TM clusters at TM 0.6
All designs (no filtering)
Number of clusters (100 designs)
Noise Scale
1.0
0.5
0.0
G
AlphaFold2 pAE of Designs (↓)
Number of Amino Acids
H
RMSD Design vs ESMFold (↓)
I
Comparing AF2 and ESMFold
Length (AA)
RMSD ESMFold vs Design (Å)
RMSD AlphaFold2 vs Design (Å)

Figure S1: Training ablations reveal determinants of RF*diffusion* success A) Comparing RF*diffusion* trained with squared distance losses on C_α atoms and N-C_α-C backbone frames (see methods), rather than with FAPE loss[8,20]. The two models were benchmarked on functional-site scaffolding problems (see Methods 3.4 for justification of this decision), and across all cases, AF2 recapitulation of the structure (left) and AF2 confidence (right) was improved when RF*diffusion* was trained with squared distance losses. Two-way ANOVA: Success rate $p < 0.001$. Henceforth, these losses were used for all models described in this paper. **B)** Allowing the model to condition on its X_0 prediction at the previous timestep (see methods) improves designs. Designs with self-conditioning (pink) have improved recapitulation by AF2 (left) and better AF2 confidence in the prediction (right). Two-way ANOVA, *in silico* success rate: $p < 0.001$. **C)** RF*diffusion* leverages the protein representations learned during RF pre-training. RF*diffusion* fine-tuned from pre-trained RF (pink) comprehensively outperforms a model trained for an equivalent amount of time, from untrained weights (gray). Indeed, training RF*diffusion* without pre-training showed no significant improvement (in terms of *in silico* success rates) compared with generating ProteinMPNN sequences from random Gaussian-sampled coordinates (white, two-way ANOVA & Tukey's test, *$p<0.001$; Random noise vs no pre-training, $p=0.9$ (n.s.); Random noise vs with pre-training, $p<0.001$; Pre-training vs not, $p<0.001$*). Note that the data in pink in **A-C** is the same data, reproduced in each plot for clarity. **D)** The median (by AF2 RMSD vs design) 300 amino acid unconditional sample highlighting the importance of self-conditioning and pre-training. Without pre-training, RF*diffusion* outputs bear little resemblance to proteins (gray, left). Without self-conditioning, outputs show characteristic protein secondary structures, but lack core-packing and ideality (gray, middle). With pre-training and self-conditioning, proteins are diverse and well-packed (pink, right). **E-F)** During the reverse (generation) process, the noise added at each step can be scaled (reduced). Reducing the noise scale comprehensively improves the *in silico* design success rates (two-way ANOVA & Tukey's test: $p < 0.001$, 0 vs 0.5: $p=0.13$, 0 vs 1: $p < 0.001$; 0.5 vs 1: $p < 0.001$). This comes at the expense of diversity, with the number of unique clusters at a TM score cutoff of 0.6 reduced when noise is reduced (**F**). **G-I)** RF*diffusion* (without reducing the added noise) can generate high quality large unconditional monomers. Designs are routinely accurately recapitulated by AF2 (see also Fig. 1E), with high confidence (**G**) for proteins up to approximately 400 amino acids in length. **H)** Further orthogonal validation of designs by ESMFold demonstrates the quality of unconditional RF*diffusion* designs. **I)** Recapitulation of the design structure is often better with ESMFold compared with AF2. For each backbone, the best of 8 ProteinMPNN sequences is plotted, with points therefore paired by backbone rather than sequence.

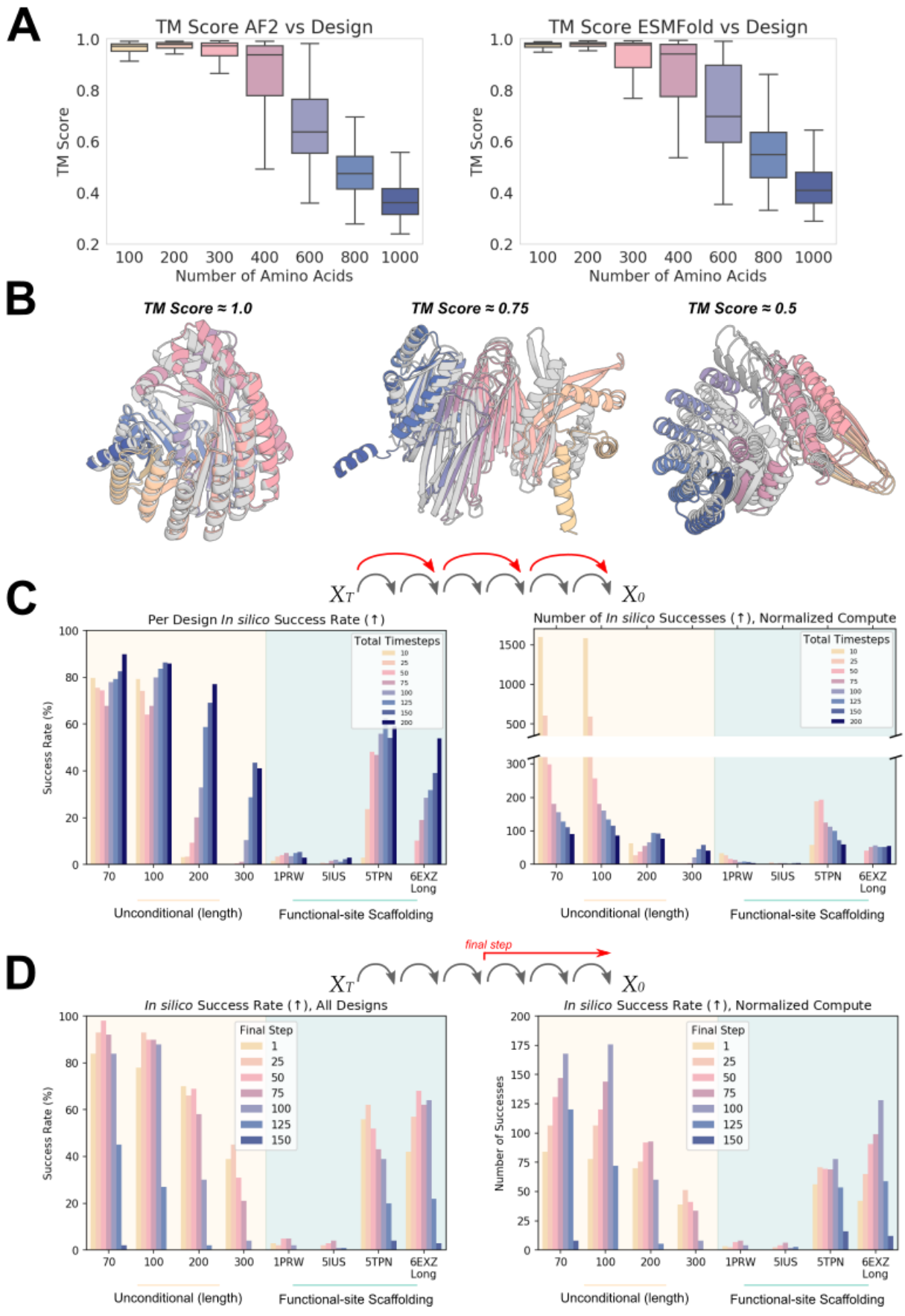

A
TM Score AF2 vs Design
TM Score ESMFold vs Design
TM Score
Number of Amino Acids
B
TM Score ≈ 1.0
TM Score ≈ 0.75
TM Score ≈ 0.5
C
Per Design In silico Success Rate (↑)
Number of In silico Successes (↑), Normalized Compute
Total Timesteps
Success Rate (%)
Unconditional (length)
Functional-site Scaffolding
D
final step
In silico Success Rate (↑), All Designs
In silico Success Rate (↑), Normalized Compute
Final Step
Number of Successes

Figure S2: Optimizing inference and improving metrics for *in silico* success. A-B) TM score between a design and a subsequent orthogonal prediction (e.g. AF2), has been previously used, typically with a threshold of > 0.5, as a metric for design success. **A)** RF*diffusion* designs have high TM score agreement to both the AF2 (left) and ESMFold (right) predictions of the unconditional structures, with TM > 0.5 for a significant fraction of designs even up to 1000 amino acids in length. **B)** TM score is, however, much less stringent than RMSD alignment. Depicted here are three unconditional RF*diffusion* designs of 600 amino acids in length (gray), overlaid with the AF2 prediction (colors), with TM scores of 0.983, 0.757 and 0.506 respectively. While a TM score of 0.5 clearly shows some resemblance to the designed structure, it differs significantly and should not be classed as "successfully designed". RMSD with a strict threshold (for example, 2 Å) is significantly more stringent. RMSDs for the displayed designs are 1.15 Å, 9.78 Å and 21.4 Å respectively. **C-D)** While RF*diffusion* is trained to generate samples over 200 timesteps, in many cases, trajectories can be shortened to improve computational efficiency. **C)** Bigger steps can be taken between timesteps at inference. While decreasing the number of timesteps typically reduces the per-design success rate (left), when normalized for compute budget (right), it is often more efficient to run more trajectories with fewer timesteps. For example, while generating 100 amino acid unconditional proteins, using a schedule with just 10 timesteps (as opposed to 200) allows the generation of 1584 *in silico* successful designs in the time taken to generate 86 successful designs with 200 timesteps. As problems get more challenging, however, this no longer remains the case (for example, fourth column, with generation of 300 amino acid designs). **D)** An alternative to taking larger steps is to stop trajectories early (possible because RF*diffusion* predicts X_0 at every timestep). In many cases, trajectories can be stopped at timestep 50-75 with little effect on the final success rate of designs (left), and when normalized by compute budget (right), success rates per unit time are typically higher generating more designs with early-stopping. For example, in the *6EXZ_Long* benchmarking motif-scaffolding problem, stopping trajectories at t=100 allows the generation of 128 *in silico* successful designs in the time it takes to generate 42 successful designs running full trajectories.

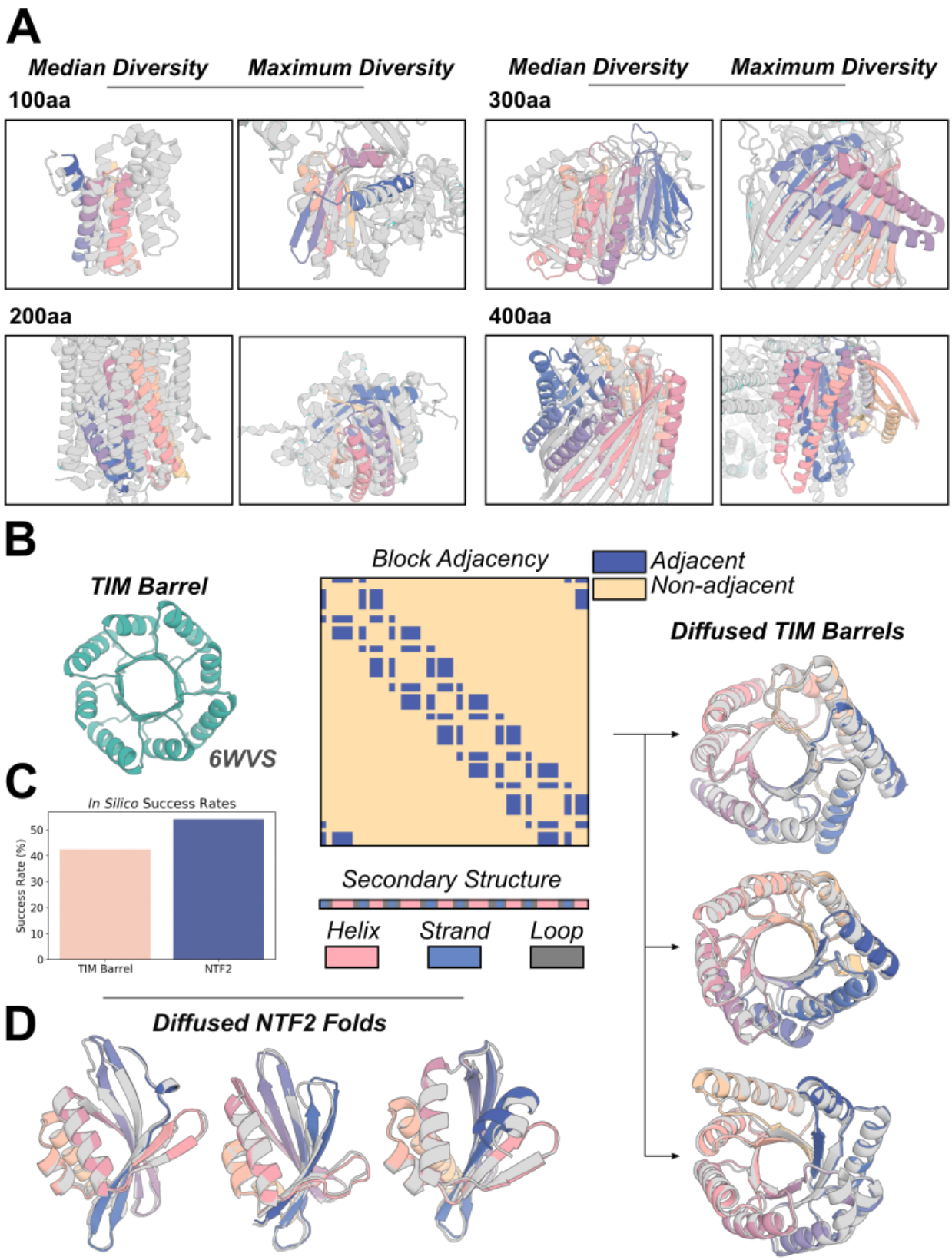

A
Median Diversity
Maximum Diversity
Median Diversity
Maximum Diversity
100aa
300aa
200aa
400aa
B
TIM Barrel
6WVS
Block Adjacency
Adjacent
Non-adjacent
Diffused TIM Barrels
C
In Silico Success Rates
Success Rate (%)
0
10
20
30
40
50
TIM Barrel
NTF2
Secondary Structure
Helix
Strand
Loop
D
Diffused NTF2 Folds

Figure S3: RF*diffusion* designs are novel without conditional information, or can be conditioned to generate specific folds. A) Example designs demonstrating extrapolation beyond the training set for generating novel folds. Gray: closest protein in the PDB by TM score, colors: RF*diffusion* design model, overlaid by TM alignment. For each protein length, the median and most diverse samples are shown. While for short proteins, designs typically show some similarity to known protein folds, with increasing length, designs become increasingly dissimilar to the PDB. TM score (closest PDB, TM score; median, most diverse): 100aa: 5WVE_A, 0.71; 4W5T_A, 0.59; 200aa: 4AV3_A, 0.58; 4CLY_A, 0.47; 300aa: 4PEW_B, 0.53; 4RDR_A, 0.46; 400aa: 4AIP_A, 0.49; 6R9T_A, 0.42. **B-D)** Designs can also be generated by conditioning on protein fold information. **B)** 6WVS is a previously-described *de novo* designed TIM barrel (left). A fine-tuned RF*diffusion* model can condition on 1D and 2D inputs representing this protein fold, specifically secondary structure (middle, bottom) and block adjacency information (middle, top, see Methods 2.5). RF*diffusion* readily conditions on this information and generates a diverse set of TIM barrels (right). Gray: RF*diffusion* design, colors: AF2 prediction. **C)** TIM barrels are generated with an *in silico* success rate of 42.5% (left bar). Success incorporates AF2 metrics and a TM score vs 6WVS > 0.5. **C-D)** NTF2 folds are useful scaffolds for *de novo* enzyme design, and can also be readily generated with fold-conditioned RF*diffusion*. Designs are diverse (**D**) and designed with an *in silico* success rate of 54.1% (**C**, right bar). NTF2 fold design success also included both AF2 metrics and a TM score vs PDB: 1GY6 > 0.5. Gray: RF*diffusion* design, colors: AF2 prediction.

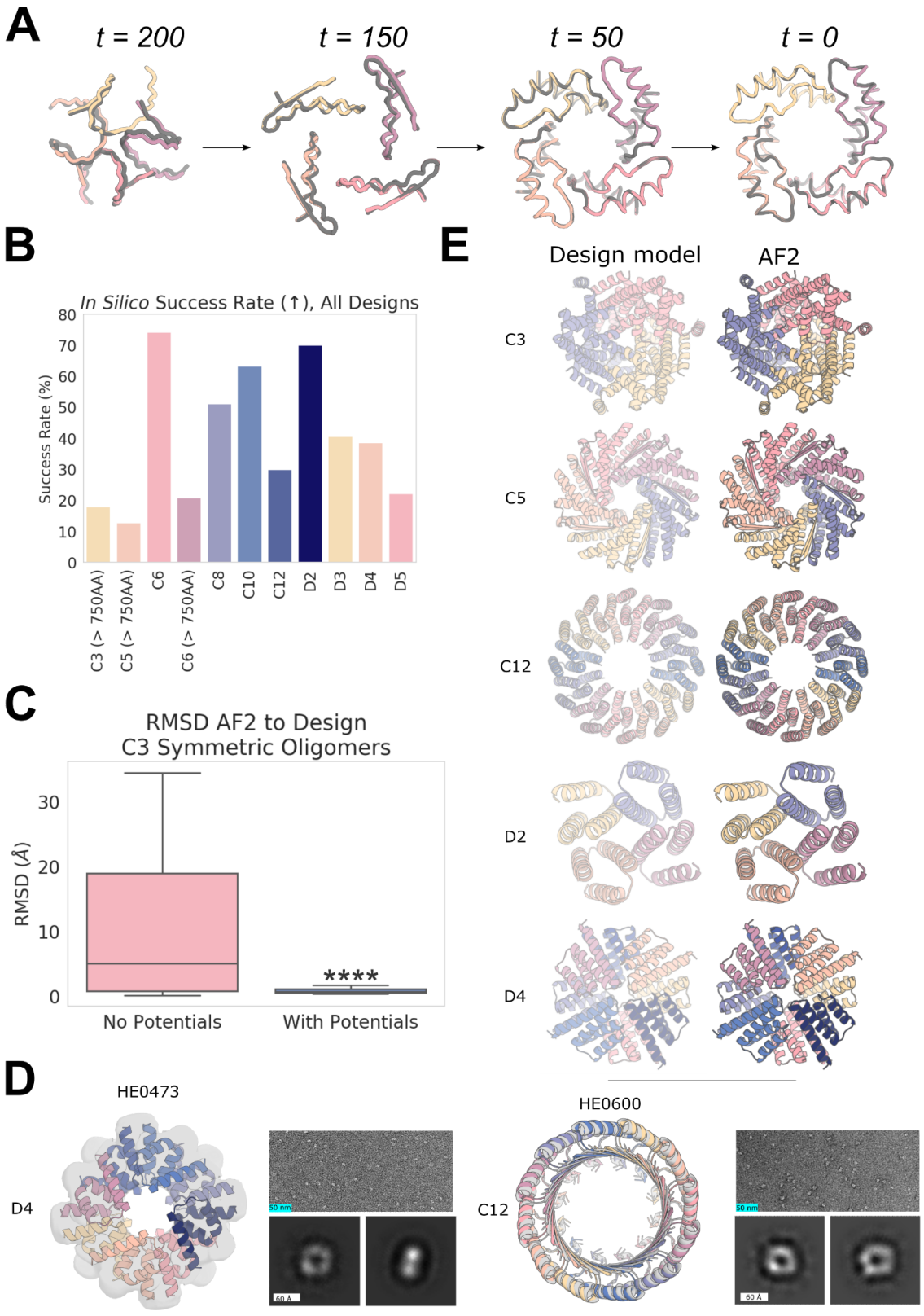

A
t = 200
t = 150
t = 50
t = 0
B
In Silico Success Rate (↑), All Designs
Success Rate (%)
0
10
20
30
40
50
60
70
80
C3 (> 750AA)
C5 (> 750AA)
C6
C6 (> 750AA)
C8
C10
C12
D2
D3
D4
D5
C
RMSD AF2 to Design
C3 Symmetric Oligomers
RMSD (Å)
0
10
20
30
****
No Potentials
With Potentials
D
HE0473
D4
HE0600
C12
E
Design model
AF2
C3
C5
C12
D2
D4

Figure S4: Symmetric oligomer design with RF*diffusion.* A) Due to the (near-perfect - see Methods 1.7) equivariance properties of RF*diffusion*, X_0 predictions from symmetric inputs are also symmetric, even at very early timepoints (and becoming more symmetric through time; RMSD vs symmetrized: *t=200* 1.20 Å ; *t=150* 0.40 Å ; *t=50* 0.06 Å ; *t=0* 0.02 Å). Gray: symmetrized (top left) subunit; colors: RF*diffusion* X_0 prediction. **B)** *In silico* success rates for symmetric oligomer designs of various cyclic and dihedral symmetries. Success is defined here as the proportion of designs for which AF2 yields a prediction from a single sequence that has mean pLDDT > 80 and backbone RMSD over the oligomer between the design model and AF2 < 2 Å . Note that 16 sequences per RF*diffusion* design were sampled. **C)** Box plots of the distribution of backbone RMSDs between AF2 and the RF*diffusion* design model with and without the use of external potentials during the trajectory. The external potentials used are the “inter-chain” contact potential (pushing chains together), as well as the “intra-chain” contact potential (making chains more globular)o. Using these potentials dramatically improves *in silico* success (Student’s unpaired t-test, $p < 0.001$). **D)** Additional symmetric oligomers structurally verified by negative stain electron microscopy (nsEM). D4 symmetric HE0473 (left) design model fitted into the 3D reconstruction, with micrograph and two 2D class averages. C12 symmetric HE0600 (right) design model (gray) overlaid with its AF2 prediction (color), with micrograph and two 2D class averages. **E)** Additional examples of design models (left) against AF2 predictions (right) for C3, C5, C12, D2, and D4 symmetric designs (the symmetries not displayed in Fig. 2) with backbone RMSDs against their AF2 predictions of 0.82, 0.63, 0.79, 0.43, 0.78 with total amino acids 750, 900, 960, 240, 640.

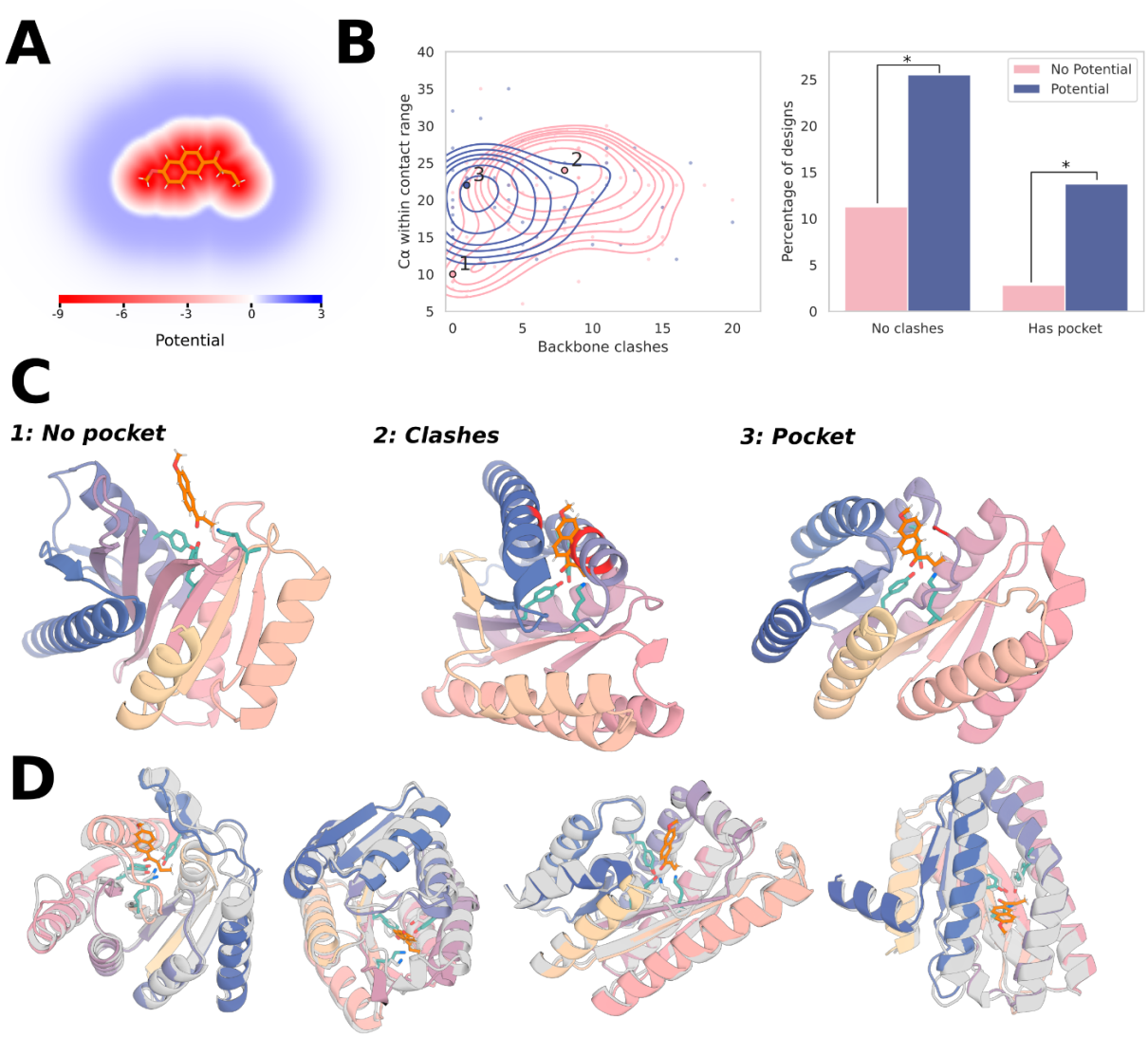


Figure S5: External potentials for generating pockets around substrate molecules. Enzymes generated from the triadic active site [TYR1051-LYS1083-TYR1180] of a retro-aldolase: PDB: 5AN7. **A)** The potential used to implicitly model the substrate, which has both a repulsive and attractive field (see Methods 2.7). **B)** Left: Kernel densities demonstrate that without using the external potential (pink), designs often fall into two failure modes: (1) no pocket, and (2) clashes with the substrate. Right: clashes (substrate < 3A of the backbone) & pockets (no clash and > 16 Cα within 3-8A of substrate) with and without the potential. Two-proportion z-test: clashes $p<0.03$, pocket $p<0.02$. Each datapoint represents a design already passing the stringent success metrics (AF2 motif RMSD < 1 Å , AF2 backbone RMSD < 2 Å , AF2 pAE < 5). **C)** Designs close to the labeled local maxima of the kernel density estimate. Without the potential, the catalytic triad is predominantly (*1*) exposed on the surface with no residues available to provide substrate stabilization or (*2*) buried in the protein core, preventing substrate access. With the potential, the catalytic triad is predominantly (*3*), partially buried in a concave pocket with shape complementary to the substrate. Backbone atoms within 3 Å of the

substrate are shown in red. **D)** A variety of diverse designs with pockets made using the potential, with no clashes between the substrate and the AF2-predicted backbone. The functional form and parameters used for the pocket potential are discussed in Methods 2.7. In each case, the substrate is superimposed on the AF2 prediction of the catalytic triad.

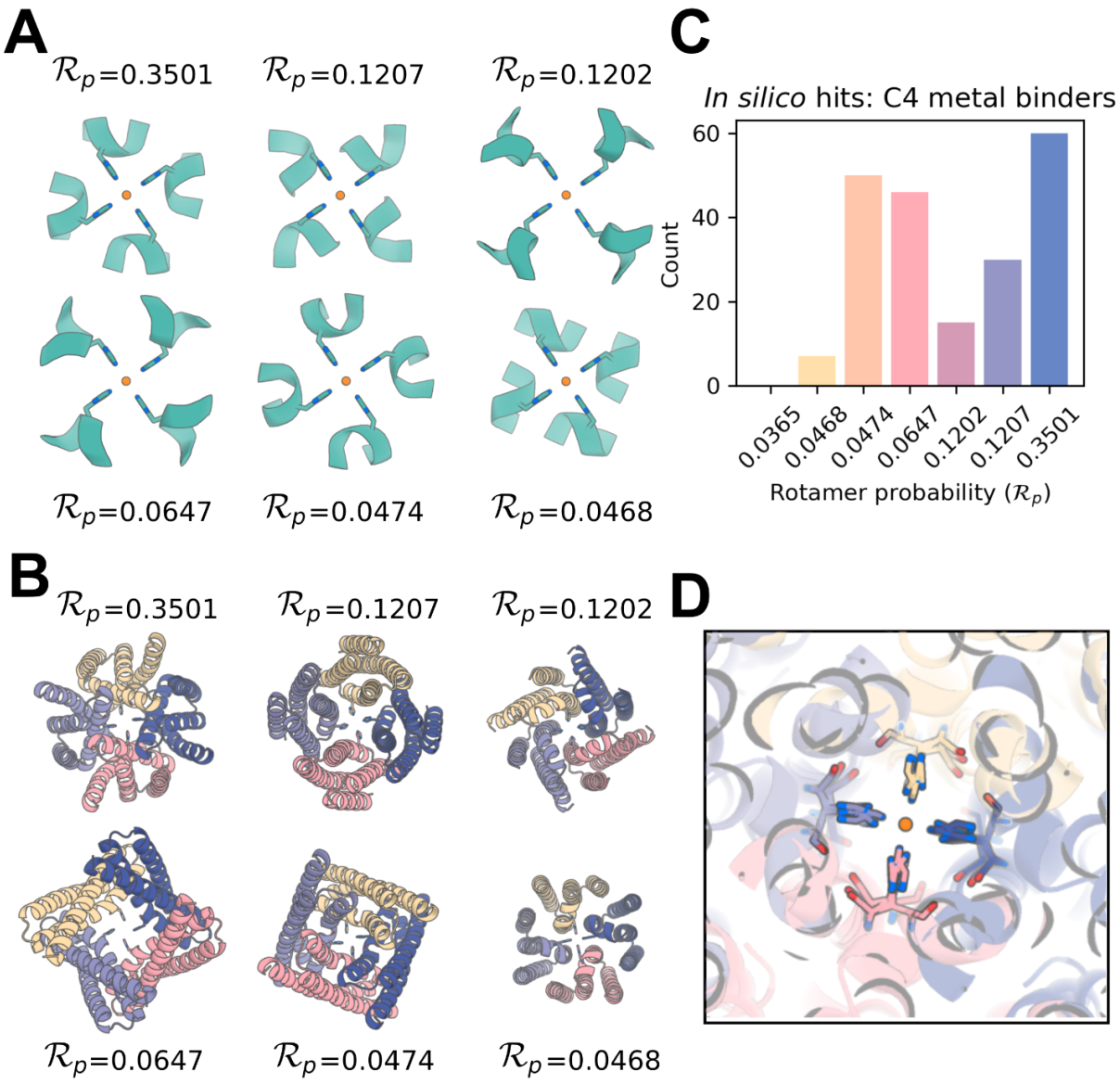


Figure S6: Symmetric motif scaffolding for square planar nickel binding. A) Depiction of a set of C4-symmetrized backbone-dependent (φ = -40°, ψ = -60°) inverse rotamers[45] used as motifs input to RF*diffusion* for symmetrically scaffolding a theoretical nickel binding site. The inverse rotamers (teal) all have identical placements of the imidazole group within histidine relative to an ideal nickel atom placement (orange), but different positions of backbone atoms which could yield this imidazole placement.. **B)** AF2 predictions of selected *in silico* hits for scaffolding the C4 inverse rotamers show significant structural diversity in RF*diffusion* solutions. **C)** *In silico* success count for the various inverse rotamers depicted in panel A. An *in silico* "success" here is defined as an AF2 prediction for a single sequence which has (1) full-atom RMSD over the four histidine residues between the AF2 prediction and the ideal C4 motif of < 1.0 Å and (2) an AF2 pAE < 10. **D)** Overlay of the 6 solutions in panel B reveals that the diverse array of RF*diffusion* solutions all place the imidazole groups with near atomic accuracy of the desired conformation (according to AF2).

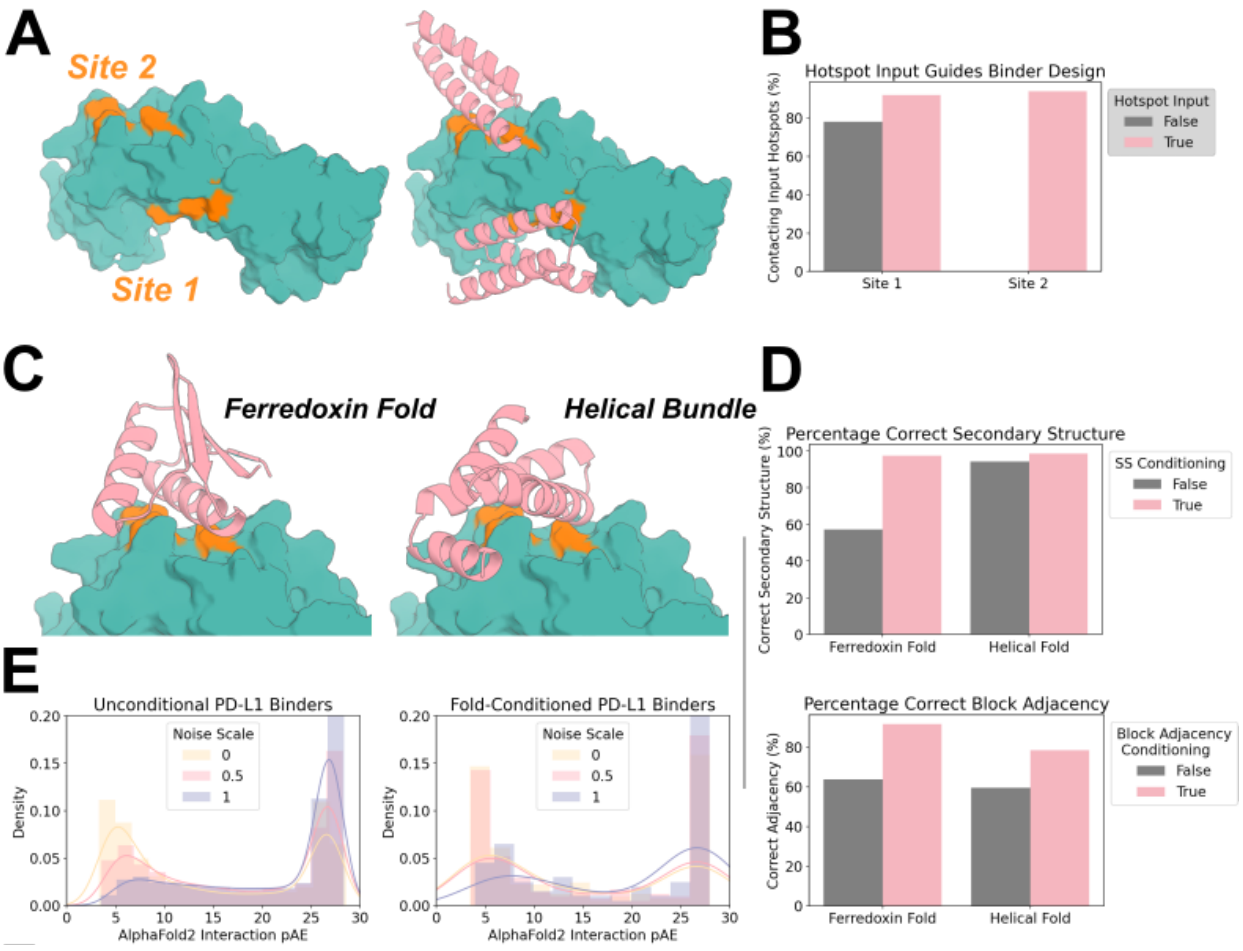


Figure S7: Targeted unconditional and fold-conditioned protein binder design. A-B) The ability to specify where on a target a designed binder should bind is crucial. Specific "hotspot" residues can be input to a fine-tuned RF*diffusion* model, and with these inputs, binders almost universally target the correct site. **A)** IL7-Rα (PDB: 3DI3) has two patches that are optimal for binding, denoted Site 1 and Site 2 here. For each site, 100 designs were generated (without fold-specification). **B)** Without guidance, designs typically target Site 1 (left bar, gray), with contact defined as C_α-C_α distance between binder and hotspot reside < 10 Å. Specifying Site 1 hotspot residues increases further the efficiency with which Site 1 is targeted (left bar, pink). In contrast, specifying the Site 2 hotspot residues can completely redirect RF*diffusion*, allowing it to efficiently target this site (right bar, pink). **C-D)** As well as conditioning on hotspot residue information, a fine-tuned RF*diffusion* model can also condition on input fold information (secondary structure and block-adjacency information - see Methods 2.5). This effectively allows the specification of a (for instance, particularly compatible) fold that the binder should adopt. **C)** Two examples showing binders can be specified to adopt either a ferredoxin fold (left) or a particular helical bundle fold (right). **D)** Quantification of the efficiency of fold-conditioning. Secondary structure inputs were accurately respected (top, pink). Note that in this design target and target site, RF*diffusion* without fold-specification made generally helical designs (right, gray bar). Block adjacency inputs were also respected for both input folds (bottom, pink). **E)**

Reducing the noise added at each step of inference improves the quality of binders designed with RF*diffusion*, both with and without fold-conditioning. As an example, the distribution of AF2 interaction pAEs (known to indicate binding when pAE < 10) is shown for binders designed to PD-L1. In both cases, the proportion of designs with interaction pAE < 10 is high (blue curve), and improved when the noise is scaled by a factor 0.5 (pink curve) or 0 (yellow curve).

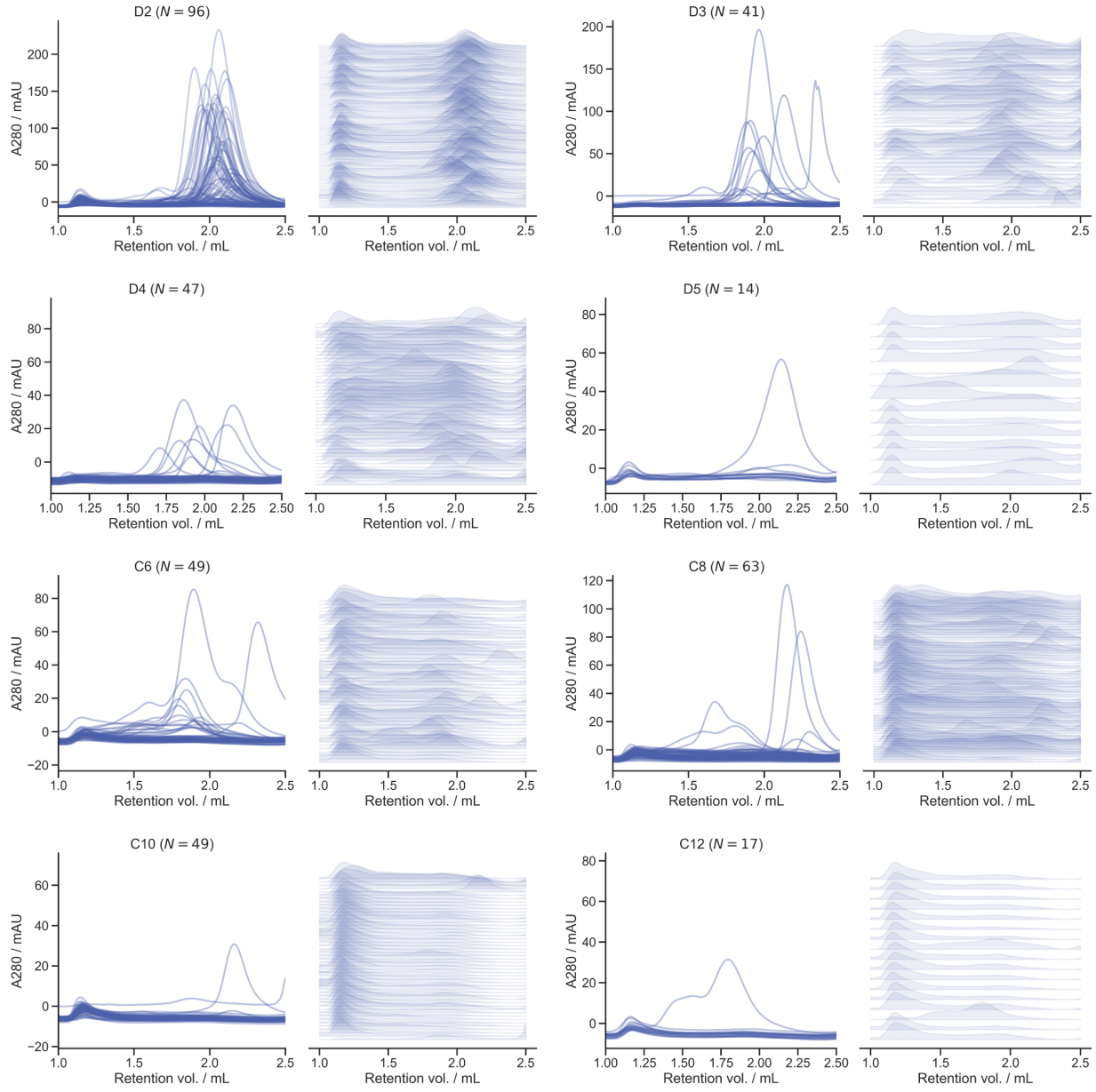


Figure S8: Size exclusion chromatography of symmetric oligomers. Size exclusion chromatography (SEC) was used as a primary screening method for all RF*diffusion*-generated oligomers. Here, SEC traces from 376 oligomers are shown for each of the eight experimentally tested symmetry groups, excluding the void volume. On the left, SEC traces are overlaid for all designs, and on the right, traces are normalized and stacked. As designs increase in complexity (higher number of individual subunits), the amount of soluble protein shown by SEC visibly decreases.

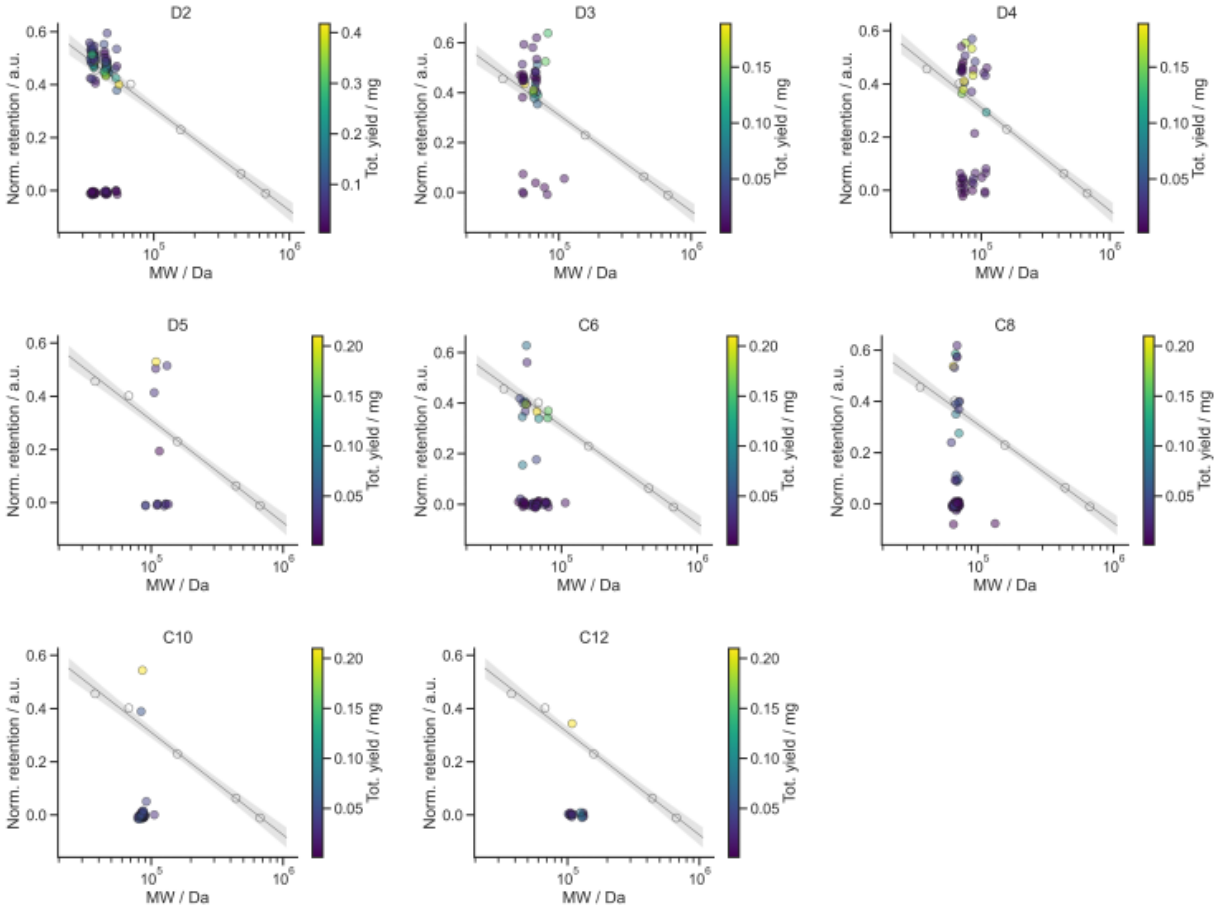


Figure S9: SEC data for symmetric oligomers with respect to calibration curves. Retention volume for the major SEC peak versus molecular weight for each design are plotted in comparison to a known calibration curve. The calibration curve is shown in gray, with shading representing the 95% confidence interval. Total yield of each design is indicated by the scale bar on the right of the graphs. Given that MW is being used as a proxy for hydrodynamic radius, we expect that some designs (e.g. cycles with large pores) may be true to their design model, but deviate from the standard curve. These calibration curves provide a rough estimate of the success rate of each symmetry group, and help guide the selection process for downstream analysis of any design.

Name, Reference	Description	Input	Total Length	Sequence to be redesigned*
1PRW[4]	Double EF-hand motif	5-20,**A16-35**,10-25,**A52-71**,5-20	60-105	A16-19,A21,A23,A25,A27-30,A32-35,A52-55,A57, A59,A61,A63-66,A68-71
1BCF[4]	Di-iron binding motif	8-15,**A92-99**,16-30,**A123-130**,1 6-30,**A47-54**,16-30,**A18-25**,8-15	96-152	A19-25,A47-50,A52-53,A92-93,A95-99,A123-126, A128-129
5TPN[4]	RSV F-protein Site V	10-40,**A163-181**,10-40	50-75	A163-168,A170-171,A179,A189
5IUS[4]	PD-L1 binding interface on PD-1	0-30,**A119-140**,15-40,**A63-82**, 0-30	57-142	A63,A65,A67,A69,A71,A72,A76,A79,A80,A82,A11 9,A120,A121,A122,A123,A125,A127,A129,A130,A 131,A133,A135,A137,A138,A140
3IXT[33]	RSV F-protein Site II	10-40,**P254-277**,10-40	50-75	P255,P258-259,P262-263,P268,P271-272,P275-2 76
5YUI[4]	Carbonic anhydrase active site	5-30,**A93-97**,5-20,**A118-120**,10-35,**A198-200**,10-30	50-100	A93,A95,A97,A118,A120
1QJG[4]	Delta5-3-ketosteroid isomerase active site	10-20,**A38**,15-30,**A14**,15-30,**A9 9**,10-20	53-103	n/a
1YCR[4]	P53 helix that binds to Mdm2	10-40,**B19-27**,10-40	40-100	B17-18,B20-22,B24-25
2KL8[4,24]	*De novo* designed protein	**A1-7**,20,**A28-79**	79	n/a
7MRX_60[24]	Barnase ribonuclease inhibitor	0-38,**B25-46**,0-38	60	n/a
7MRX_85[24]		0-63,**B25-46**,0-63	85	n/a
7MRX_128[24]		0-122,**B25-46**,0-122	128	n/a
4JHW[32]	RSV F-protein Site 0	10-25,**F196-212**,15-30,**F63-69**, 10-25	60-90	F196,F198,F203,F211-212,F63,F69
4ZYP[32]	RSV F-protein Site 4	10-40,**A422-436**,10-40	30-50	A422-427,A430-431,A433-436
5WN9[33]	RSV G-protein 2D10 site	10-40,**A170-189**,10-40	35-50	A170-175,A188-189
6VW1[4,34]	ACE2 interface binding SARS-CoV-2	***E400-510***/20-30,**A24-42**,4-10, **A64-82**,0-5†	62-83	A25-26,A29-30,A32-34,A36-42,A64-82
5TRV_short[5]	*De novo* designed protein	0-35,**A45-65**,0-35	56	n/a
5TRV_med[5]		0-65,**A45-65**,0-65	86	n/a
5TRV_long[5]		0-95,**A45-65**,0-95	116	n/a
6E6R_short[5]	Ferridoxin Protein	0-35,**A23-35**,0-35	48	n/a
6E6R_med[5]		0-65,**A23-35**,0-65	78	n/a
6E6R_long[5]		0-95,**A23-35**,0-95	108	n/a
6EXZ_short[5]	RNA export factor	0-35,**A28-42**,0-35	50	n/a
6EXZ_med[5]		0-65,**A28-42**,0-65	80	n/a
6EXZ_long[5]		0-95,**A28-42**,0-95	110	n/a

Table 1: A benchmarking set of recently published functional-site scaffolding problems. To benchmark RF*diffusion* at functional-site scaffolding, against existing methods, we generated a benchmark set encompassing problems described in six recent publications[4,5,24,32–34], which utilize a range of design methodologies to address these problems. For each problem, named by PDB accession (and, where applicable, the length of the designs to be generated, left column), we recapitulated the inputs as closely as possible with respect to details available in each publication. So that others can test methods on this benchmark, the exact input is specified in the third column. In bold, prefixed by a letter, are the inputs (chain, residues) from the PDB structure provided to the model (the "functional-site"). In non-bold text are the lengths that the different methods randomly sampled to generate good designs. The final lengths of the proteins were either specified by the input to the model, or were provided as constraints (for example, for 6EXZ_Long, the model could sample any N- and C-terminal length between 0 and 95 residues, but the total length of the output had to equal 110 amino acids). For each design challenge, 100 designs were generated, and, where ProteinMPNN was used, 8 sequences were designed, with the best sequence chosen for each backbone. *Both the RF*joint* and RoseTTAFold constrained hallucination approaches can simultaneously redesign sequences during generation, which can, in some cases, be helpful (if extracting the "functional-site" exposes hydrophobic residues which may subsequently end up as surface residues in the output designs, for example). Therefore, in this benchmark, these methods were allowed to redesign non-functional residues, listed in the right-most column. † This example is multi-chain generation (scaffolding a functional-site in the presence of a second chain). All methods benchmarked here can represent chain breaks (with large residue index jumps). Full results are shown in Fig. 3A, and tabulated in Table 2.

Problem Name	RF*diffusion* (noise=0)	RF*diffusion* (noise=1)	RF*joint*	RF*joint* + ProteinMPNN	RF Hallucination	RF Hallucination + ProteinMPNN
1BCF	**100**	98	65	94	0	0
6E6R_med	**89**	67	0	14	2	9
2KL8	88	**96**	71	62	20	34
6E6R_long	**86**	63	0	1	0	1
6EXZ_long	**76**	51	0	0	1	4
1YCR	**74**	58	12	20	11	61
6VW1	**69**	66	0	2	2	32
5TPN	**61**	59	0	1	0	1
6EXZ_med	**49**	33	0	0	5	15
4ZYP	**40**	31	1	7	1	18
6E6R_short	**39**	29	0	15	3	7
5TRV_long	**37**	30	0	0	0	2
3IXT	**25**	16	21	24	2	34
5TRV_med	**24**	20	0	0	0	3
7MRX_85	**11**	6	0	0	0	0
7MRX_128	**9**	4	0	0	0	0
1PRW	8	**9**	0	5	0	0
5TRV_short	4	**7**	0	0	0	1
7MRX_60	**2**	0	0	0	0	0
6EXZ_short	2	4	1	10	4	**15**
5IUS	**2**	0	0	0	0	0
5YUI	0	0	0	0	0	0
5WN9	0	**1**	0	0	0	0
4JHW	0	0	0	0	0	0
1QJG	0	**1**	0	0	0	0

Table 2: Functional-site scaffolding benchmark results. Full results for the benchmark test described in Fig. 3A and Table 1. In each case, values represent the success rate (%) in a set of 100 designs generated with each method.

Materials & Methods

Section 1: Motivation for and explanation of RF as the neural network in a generative diffusion model

In this section we describe in greater detail how we have repurposed RoseTTAFold (RF) as a generative model of protein structure.

1.1 Preliminaries

Machine learning models for protein structure design must confront two major challenges to representing protein structures: (1) protein structure is most naturally represented by coordinates in a semantically arbitrary 3D coordinate system, yet (2) each amino acid which lives in this subspace has (effectively) two degrees of freedom (the ϕ and ψ backbone torsions angles) as opposed to the canonical six for a free rigid body. To navigate these challenges, most previous works on generative models of protein structure[26,46] have represented proteins as "maps" of pairwise distances between amino acids, followed by realizing chemically plausible 3D structures from these maps. However, given the remarkable representative power and accuracy of networks like AlphaFold2 (AF2) and RF which manipulate a "gas" of rigid bodies in 3D space in an SE(3) equivariant manner to produce a final 3D protein structure, we chose to formulate the protein generation task in a way that was compatible with this representation strategy. Moreover, design methods that directly parameterize structures in 3D are appealing for design because they allow specification of both rigid structural constraints such as the presence of functional motifs or existence of a desired symmetry by direct manipulations of structure[5,8].

We next give a brief overview of our diffusion modeling framework and how we have adapted it to protein structures in 3D. We then detail how we have applied it to the different components of our representation of structure. Lastly, we describe how we train conditional variants of the diffusion model for motif-scaffolding and generation with secondary structure constraints.

1.2 Diffusion probabilistic modeling of protein structure

Our approach builds on denoising diffusion probabilistic models (DDPMs)[10,11]. We follow & adapt the conventions and notation set by [10], which we review here. DDPMs are a class of generative models based on a reversible, discrete-time diffusion process. The *forward process* starts with a sample $x_0 \sim q(x_0)$ from an unknown data distribution q. Noise is added at each step, to obtain a sequence of increasingly noisy samples x_t such that the final step $x_T \sim q(x_T)$ is indistinguishable from a *reference* distribution that has no dependence on the data. DDPMs approximate $q(x_0)$ with a second distribution $p_\theta(x_0)$ defined by transition distributions of the *reverse process* $p_\theta(x_{t-1} | x_t)$ at each t which are parameterized by a neural network. The neural network is trained such that $p_\theta(x_{t-1} | x_t)$ approximates $q(x_{t-1} | x_t)$. One then draws from $p_\theta(x_0)$ by first

sampling from the reference distribution $x_T \sim p_\theta(x_T) \approx q(x_T)$, and then for each $t < T$ iteratively denoising by sampling $x_{t-1} \sim p_\theta(x_{t-1}|x_t)$ until $x_0 \sim p_\theta(x_0)$ is obtained.

In our case, we consider $q(x_0)$ to be a distribution over the structures of backbones of native proteins. We adopt the "residue-gas" representation of backbones used by RF[47]. This representation consists of the 3D coordinates (z) of the central carbon (C_α) and 3x3 rotation matrices (r) representing the rigid-body orientation of each residue in a global reference frame, thereby additionally defining the coordinates of the N and C backbone atoms. $x = [z, r]$. We defined a forward process that applies noise independently both across residues and across these two components of residue geometry. We similarly model the reverse process transitions as independent across these components:

$$p_\theta(x_{t-1}|x_t) = p_\theta(z_{t-1}|x_t)p_\theta(r_{t-1}|x_t).$$

While $q(x_{t-1}|x_t)$ can in general be correlated across these different components of structure, standard practice has found it beneficial to ignore the correlation across dimensions in the reverse diffusion process[10]. Indeed, in the limiting regime where the number of steps in the forward process tends to infinity, and the forward process is viewed as a discretization of a continuous time diffusion process, one can see that the correlation between different dimensions is absent in the reverse process as well[48].

To address the challenge of the arbitrary reference frames we build on previous work[5], and seek to learn a distribution over protein structure that is invariant to global rotation; that is, we require that any protein structure is modeled as equally likely upon a rigid rotation. More formally, this means that for any structure x_0 and rotation R we desire to have that $p_\theta(x_0) = p_\theta(R * x_0)$, where $R * x_0$ represents the structure obtained by rotating x_0 about the origin by R (for each residue $R * x = [Rz, Rr]$).

Following prior work[5,49], we incorporate this invariance by (1) using a rotation invariant reference distribution (i.e. satisfying $p_\theta(x_T) = p_\theta(R * x_T)$) and (2) constraining the reverse diffusion model to be equivariant to rotations (i.e. satisfying $p_\theta(x_t|x_{t+1}) = p_\theta(R * x_t|R * x_{t+1})$). To this end, we leverage the geometric equivariance and invariance properties inherent to RoseTTAFold. In particular, RF uses the SE(3)-transformer architecture[50,51] to provide equivariant updates to intermediate predictions of structure across recycling steps; we use these same input channels to obtain *equivariant* updates of C_α coordinates (Methods 1.4) and rotations (Methods 1.5) for each residue.

1.3 Training RosettaFold with rotation and translation distance losses

Our approach to learning the reverse process transition is to train RoseTTAFold to denoise noisy protein structures. For each step of training, we first choose an example protein structure

x_0 and a time step t uniformly at random between 1 and T, and then simulate the forward process to obtain $x_t \sim q(x_t|x_0)$. We next apply RoseTTAFold to obtain a prediction of the denoised structure, which we denote by $\hat{x}_0(x_t) = (\hat{z}_0, \hat{r}_0)$. We then compute a loss on this output consisting of, for each residue, the squared Euclidean distance (i.e. the squared L_2 error) on the C_α coordinates ($L_{trans.} = ||z_0 - \hat{z}_0||^2$) and the square of a metric on the space of rotation matrices ($L_{rot.} = ||I_3 - \hat{r}_0^T r_0||_F^2$, where $|| \cdot ||_F$ denotes the Frobenius norm of a matrix)[52]. Algorithm 1 summarizes the training procedure.

The approach above takes inspiration from Ho *et al*[10]. In particular, Ho *et al*[10] (section 3.2) comments that when the forward process consists of adding Gaussian noise, the training objective of minimizing the KL divergence of $q(x_t|x_{t-1})$ to $p_\theta(x_{t-1}|x_t)$ can be rewritten as a rescaling of the expected squared error of a prediction of x_0 from noisy observations x_t:

$$E_{x_0, x_t \sim q}[KL(q(x_t|x_{t-1})||p_\theta(x_{t-1}|x_t))] \propto E_{x_0, x_t \sim q}[||x_0 - \hat{x}_0(x_t)||^2] + c \quad (1)$$

where c is a constant that does not depend on θ. Consequently when one minimizes the right-hand-side of equation (1), they maximize a weighted variational lower bound on the likelihood of the data[10] that is globally minimized only when each $p_\theta(x_{t-1}|x_t)$ matches $q(x_{t-1}|x_t)$, and $p_\theta(x_0)$ therefore matches the data-distribution. Although ref [10] found better performance in generative modeling of images when predicting the noise added in the forward process (rather than x_0), we reasoned that by predicting x_0 we could better leverage the inductive biases of RoseTTAFold pre-trained for structure prediction to produce realistic structures (as in RF_{joint} [4]).

However, the equivalence of learning to optimally denoise according to average squared L_2 error and matching the reverse process in equation (1) applies only when the forward process consists of Gaussian noise. This presents a challenge for learning the reverse process for the rotations, because neither the Gaussian distribution nor Euclidean distance are well-defined on the space of rotation matrices (SO(3)), and so prior work did not offer an obvious choice of loss for the $\hat{r}_0$ prediction. However the squared Frobenius norm metric ($L_{rot.}$) seemed to be a sensible choice because (1) our chosen forward noising process for rotations is approximately Gaussian in the tangent space of SO(3) at r_0 for t close to zero (see Methods 1.5), and (2) L_{rot} is approximately equal to squared L_2 distance in the tangent space of SO(3) when r_0 is close to $\hat{r}_0$ (ref [53]).

Algorithm 1 RFDiffusion Training.

function FORWARDNOISE(x_0, t)
  $[(z_{0,1}, r_{0,1}), \ldots, (z_{0,M}, r_{0,M})] = x_0$
  for $m = 1, \ldots, M$ **do**
    $z_{t,m} \sim \mathcal{N}\left(z_{t,m}; \sqrt{\bar{\alpha}_t} z_{0,m}, (1 - \bar{\alpha}_t) I_3\right)$
    $r_{t,m} \sim \text{IGSO3}\left(r_{t,m}; r_{0,m}, \sigma_t^2\right)$
  end for
  $x_t = [(z_{t,1}, r_{t,1}), \ldots, (z_{t,M}, r_{t,M})]$
return x_t
end function

function LOSS($x_0, \hat{x}_0, w_{\text{trans.}} = 1, w_{\text{rot.}} = 5$)
  $[(z_{0,1}, r_{0,1}), \ldots, (z_{0,M}, r_{0,M})] = x_0$
  $[(\hat{z}_{0,1}, \hat{r}_{0,1}), \ldots, (\hat{z}_{0,M}, \hat{r}_{0,M})] = \hat{x}_0$

  $L_{\text{trans.}} = \frac{1}{M} \sum_{m=1}^{M} \|z_{0,m} - \hat{z}_{0,m}\|^2$
  $L_{\text{rot.}} = \frac{1}{M} \sum_{m=1}^{M} \|I_3 - r_{0,m}^\top \hat{r}_{0,m}\|_F^2$
  $L_{\text{total}} = w_{\text{trans.}} L_{\text{trans.}} + w_{\text{rot.}} L_{\text{rot.}}$
return L_{total}
end function

function TRAIN(x_0, t)
  while not converged **do**
    $x_0 \sim \text{TrainingSet}$
    $t \sim \text{Uniform}(\{1, \ldots, T\})$
    $x_t = \text{ForwardNoise}(x_0, t)$
    $\hat{x}_{0,\text{prev.}} = \vec{0}$ ▷ Self-conditioning variable
    if $\text{Uniform}(0, 1.0) < 0.5$ **and** $t < T$ **then**
      $x_{t+1} = \text{ForwardNoise}(x_0, t+1)$
      $\hat{x}_{0,\text{prev.}} = \text{RFDiffusion}_\theta(x_t, \vec{0}, t)$
      $\hat{x}_{0,\text{prev.}} = \text{StopGradient}(\hat{x}_{0,\text{prev.}})$
      $x_t = \text{ReverseStep}(x_{t+1}, \hat{x}_{0,\text{prev.}}, t)$
    end if
    $\hat{x}_0 = \text{RFDiffusion}_\theta(x_t, \hat{x}_{0,\text{prev.}}, t)$
    Take gradient step on $\nabla_\theta \text{Loss}\left(x_0, \hat{x}_0\right)$
  end while
end function

Algorithm 2 RFDiffusion reverse step and rotation score approximation

function $\text{F}(\omega, \epsilon^2, L = 2000)$

▷ IGSO3 density factor, truncated to L terms

return $\sum_{l=0}^{L}(2l+1)e^{-l(l+1)\epsilon^2}\frac{\sin((l+\frac{1}{2})\omega)}{\sin(\omega/2)}$

end function

function RotationScoreApproximation$(r_t, \hat{r}_0, \sigma_t^2)$

$\vec{r}_{0t} = \text{Log}(r_t\hat{r}_0^\top)$ ▷ Log map from SO(3) to rotation vector, $\vec{r}_{0t} \in \mathbb{R}^3$

$\omega = \|\vec{r}_{0t}\|_2$ ▷ $\omega \in [0, \pi]$

▷ Compute score approximation as rotation vector

$s = \frac{r_t\vec{r}_{0t}}{\sqrt{2}\omega} \cdot \frac{d}{d\omega}\log\text{F}(\omega; \sigma_t^2)$ ▷ $s \in \mathbb{R}^3$

return score_approx

end function

function ReverseStep$(x_{t+1}, \hat{x}_0)$

▷ One step of reverse diffusion

$[(z_{t+1,1}, r_{t+1,1}), \ldots, (z_{t+1,M}, r_{t,M})] = x_{t+1}$

$[(\hat{z}_{0,1}, \hat{r}_{0,1}), \ldots, (\hat{z}_{0,M}, \hat{r}_{0,M})] = \hat{x}_0$

for $m = 1, \ldots, M$ **do**

▷ Update translations

$z_{t,m} \sim \mathcal{N}(\frac{\sqrt{\bar{\alpha}_t}\beta_t}{1-\bar{\alpha}_t}\hat{z}_{0,m} + \sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})z_{t+1,m}, \beta_t)$

▷ Update rotations

$s_m = \text{RotationScoreApproximation}(r_{t,m}, \hat{r}_{0,m}, \sigma_t^2)$

$\Delta r_m = \text{Exp}\{(\sigma_{t+1}^2 - \sigma_t^2)s_m\}$ ▷ Exp map from Lie algebra to SO(3)

$r_{t,m} \sim IGSO3(\Delta r_m r_{t+1,m}, \sigma_{t+1}^2 - \sigma_t^2)$

end for

$x_t = [(z_{t,1}, r_{t,1}), \ldots, (z_{t,M}, r_{t,M})]$

return x_t

end function

Algorithm 3 RFDiffusion generation of monomers and symmetric oligomers

function SAMPLEREFERENCE(M)
▷ Random initial structure for M residues
for $m = 1, \ldots, M$ **do**
$z_T \sim \mathcal{N}(0, I_3)$
$r_T \sim \text{Uniform}(SO(3))$
end for
$x_T = [(z_{T,1}, r_{T,1}), \ldots, (z_{T,M}, r_{T,M})]$
return x_T
end function

function SAMPLE(M)
▷ RFDiffusion generation of M-residue backbone structure
$x_T = \text{SampleReference}(M)$
$\hat{x}_{0,\text{prev.}} = \vec{0}$ ▷ Initialize self-conditioning
for $t = T-1, \ldots, 0$ **do**
$\hat{x}_0 = \text{RFDiffusion}_\theta(x_{t+1}, \hat{x}_{0,\text{prev.}})$
$x_t = \text{ReverseStep}(x_{t+1}, \hat{x}_0)$
$\hat{x}_{0,\text{prev.}} = \hat{x}_0$
end for
return $\hat{x}_0$
end function

function SAMPLESYMMETRIC($M, \mathfrak{R} = \{R_k\}_{k=1}^K$)
▷ RFDiffusion generation of oligomer with symmetry $\mathfrak{R}$
$x_T^1 = \text{SampleReference}(M)$
$\hat{X}_0 = [\vec{0}, \ldots, \vec{0}]$ ▷ Initialize self-conditioning
for $t = T-1, \ldots, 0$ **do**
$X_{t+1} = [R_1 x_{t+1}^1, \ldots, R_K x_{t+1}^1]$ ▷ Symetrization of chains
$\hat{X}_0 = \text{RFDiffusion}_\theta(X_{t+1}, \hat{X}_0)$
$[x_t^1, \ldots, x_t^K] = \text{ReverseStep}(X_{t+1}, \hat{X}_0)$
end for
return $\hat{X}_0$
end function

1.4 Details of forward and reverse diffusion of backbone residue translations

In this subsection we describe our forward diffusion over backbone Cα coordinates (z), and how we relate predictions $\hat{x}_\theta(x_t)$ of x_0 to our approximation $p_\theta(z_{t-1}|x_t)$ of $q(z_t|x_{t-1})$. Our development and notation follows ref [10]. We let $\beta_1, \beta_2, \cdots, \beta_T$ be scalars between 0 and 1 that define a variance schedule such that for each $t = 1, 2, \cdots, T$ the transition density of the forward process is $q(z_t \mid z_{t-1}) = N(z_t; \sqrt{1-\beta_t} z_{t-1}, \beta_t I_3)$. Define $\alpha_t = 1 - \beta_t$ and $\bar{\alpha}_t = \prod_{s=1}^{t} \alpha_t$. To sample z_t during training, rather than sampling $z_s|z_{s-1}$ from $s = 1$ all the way up to $s = t$, we draw z_t directly from the marginal distribution,

$$q(z_t|z_0) = N(z_t; \sqrt{\bar{a}_t} z_0, (1 - \bar{\alpha}_t) I_3). \quad (2)$$

Given that $q(z_{t-1}|z_t, z_0) = N(z_{t-1}; \tilde{\mu}(z_t, z_0), \tilde{\beta}_t I_3)$

for $\tilde{\mu}(z_t, z_0) = \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t} z_0 + \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t} z_t$ and $\tilde{\beta}_t = \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t \approx \beta_t$, we choose to parameterize the reverse transitions by

$p_\theta(z_t|x_{t-1}) = N(z_t; \mu_\theta(x_t), \beta_t I_3)$ for $\mu_\theta(x_t) = \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t} \hat{z}_0(x_t) + \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t} z_t$,

where $\hat{z}_0(x_t)$ are the predicted C_α coordinates obtained from $\hat{x}_0(x_t)$. We choose β_t according to a linear variance schedule as in[26,54] with parameters $\beta_0 = 0.01$ and $\beta_T = 0.07$. We chose these parameters such that signal remaining in x_0 (as quantified by $\bar{a}_t$) decayed slowly toward zero as t approaches T=200 .

1.5 Details of forward and reverse diffusion on backbone residue rotations

We model the remaining two backbone atoms (N and C) with a diffusion process on rigid body rotations that map an axis-aligned residue with idealized internal geometry (i.e. bond lengths and angle) to the positions of these atoms relative to central C_α. Specifically, for any backbone atom coordinates z_c, $z_{c\alpha}$ and z_N for a given residue we may apply a Gram-Schmidt process to compute a 3x3 rotation matrix r with rows

$$r_1 = (z_c - z_{c\alpha}) / ||z_c - z_{c\alpha}||,$$
$$r_2 = ((z_N - z_{c\alpha}) - (z_N - z_{c\alpha}) \cdot r_1)/||(z_N - z_{c\alpha}) - (z_N - z_{c\alpha}) \cdot r_1||, \text{ and}$$
$$r_3 = r_1 \times r_2.$$

where · and × are the dot and cross-products, respectively. 3D backbone coordinates can then be reconstructed by multiplication of idealized coordinates (with $z^*_{c\alpha}$ at the origin, $z^*_c - z^*_{c\alpha}$ along the x-axis, and $z^*_N - z^*_{c\alpha}$ in the xy-plane) by R:

$$[z_c,\ z_N,\ z_{c\alpha}] \ = \ R\,[z^*_c, z^*_N,\ z^*_{c\alpha}] \ + z_{c\alpha}\vec{1}_3\ ,$$

where $\vec{1}_3 \ = \ [1,\ 1,\ 1]$. Accordingly, modeling the coordinates of a triplet of backbone atoms is equivalent to modeling the $C\alpha$ coordinate z and the rotation matrix r.

However, modeling rotation matrices introduces challenges not addressed by ref [10]; the space of 3x3 rotation matrices (known as the special orthogonal group of dimension 3, or SO(3)) is a compact Riemmanian manifold on which the typical Gaussian distribution is not well-defined and the so the associated techniques of [10] do not apply. To this end we adapt the approach of [18], who extend diffusion generative modeling to Riemannian manifolds. In brief, they build on the continuous-time diffusion framework[48] and define their forward corruption process as the Brownian motion on the manifold of interest, and characterize the time-reversal of this process through the Stein score of the noised data distribution at each t.

In the case of SO(3), the marginal distribution of a rotation matrix r_t evolving according to Brownian motion for time t from an initial rotation r_0 is given by the $IGSO3$ distribution[19,55], and we may write $r_t \sim IGSO3(\mu = r_0, \epsilon^2 = t)$. The density of the $IGSO3$ distribution with respect to the uniform distribution on SO(3) is given by

$$IGSO3(r\,;\mu\,,\,\epsilon^2) \ = \ f(\omega(r\mu^\top);\ \epsilon^2),\ for\ f(\omega;\ \epsilon^2) \ = \ \sum_{l=0}^{\infty}(2l\ +\ 1)e^{-l(l+1)\epsilon^2}\ \frac{\sin((l+\frac{1}{2})\omega)}{\sin(\omega/2)},\ (3)$$

where μ is 3x3 mean rotation matrix and $\omega(r)$ denotes the angle of rotation in radians associated with a rotation r (i.e. its length if written in the axis-angle parameterization). We approximate the power series in equation (3) by truncating after 2000 terms. We formulate a discrete-time forward noising by discretizing the Brownian motion, which provides:

$q(r_t\,|\,r_{t-1}) \ = \ IGSO3(r_t\,;r_{t-1}\,,\,\epsilon^2 = \sigma_t^{\,2} - \sigma_{t-1}^{\,2})$ and marginally
$q(r_t\,|\,r_0) \ = \ IGSO3(r_t\,;r_0\,,\,\epsilon^2 = \sigma_t^{\,2})$

where $\sigma_t^{\,2}$ is a variance schedule for rotations. In particular, we set this variance schedule by using a continuous time analogue of a linear β schedule in a variance exploding SDE[48], defined as

$$\sigma_t \ = \ t * \beta_{min} + \tfrac{1}{2}(\tfrac{t}{T})^2(\beta_{max} - \beta_{min}).$$

We chose $\beta_{min} = 1.06$ and $\beta_{max} = 1.77$ so that the rotations are corrupted at a rate similar to the forward process on translations. In contrast to the translations, which converge to a Gaussian distribution as t increases, the rotations converge to the uniform distribution on $SO(3)$.

To approximate the reverse transitions for the rotations, we appeal to De Bortoli *et al*[18] (Theorem 1), which shows that (up to error from discretization of the continuous time process) the reverse process transitions have the form,

$r_{t-1}|x_t \sim \exp\{\Delta r_t\}\, r_t$ for $\Delta r_t \sim IGSO3(\exp\{(\sigma_t^2 - \sigma_{t-1}^2) \cdot \nabla_{r_t} \log q(x_t)\},\ \sigma_t^2 - \sigma_{t-1}^2)$, (4)

where $\nabla_{r_t} \log q(x_t)$ denotes the "Stein score" of the forward process at time t, and $\exp\{\cdot\}$ denotes the exponential map to SO(3) from the Lie algebra of SO(3) (the space in which the score is defined).

Equation 4 describes how one could sample from the reverse process using the IGSO(3) distribution based on the score of the forward process. One could in principle learn this score function directly by score matching training[18]. However, we instead rely on an approximation that directly leverages RoseTTAFold's ability to produce denoised structures when trained according to Algorithm 1. For a given t and r_t we note that we may write

$$\begin{aligned}
\nabla_{r_t} \log q(x_t) &= \mathbb{E}_q \left[\nabla_{r_t} \log q(x_t \mid x_0) \mid x_t \right] \\
&= \mathbb{E}_q \left[\nabla_{r_t} \log q(r_t \mid r_0) \mid x_t \right] \\
&\approx \nabla_{r_t} \log q(r_t \mid r_0 = \hat{r}_0) \\
&= \nabla_{r_t} \log \mathrm{IGSO3}(r_t; \hat{r}_0, \sigma_t^2), \qquad (5)
\end{aligned}$$

where the first line is known as the denoising score matching identity[18,56], the second line obtains from the conditional independence structure of the forward process, the third line is an approximation that can be thought of as replacing $q(r_0|r_t)$ with a point mass on the noiseless rotation $\hat{r}_0$ predicted by RoseTTAFold, and the final line recognizes the approximation as the tractable IGSO3 density. In particular,

$$\begin{aligned}
\nabla_r \log IGSO3(r; \hat{r}, \epsilon^2) &= \nabla_r \omega(r\hat{r}^T) \cdot \tfrac{d}{d\omega} \log f(\omega; \epsilon^2)|_{\omega=\omega(r\hat{r}^T)} \qquad (6) \\
&= \tfrac{1}{\sqrt{2}}\, r\, log(r\hat{r}^T)/\omega(r\hat{r}^T) \tfrac{d}{d\omega} \log f(\omega; \epsilon^2)|_{\omega=\omega(r\hat{r}^T)}
\end{aligned}$$

where $log(r\hat{r}^T)$ is the logarithmic map from SO(3) to the Lie algebra of SO(3) [57], $\omega(r\hat{r}^T)$ is the angle of rotation associated with $r\hat{r}^T$, and f is the IGSO3 density factor in equation (3). $r\ \log(r\hat{r}^T)\ /\omega(r\hat{r}^T)$ is a unit length perturbation in the direction of $\log(r\hat{r}^T)$ applied in the tangent space of SO(3) at r, and $\frac{d}{d\omega} \log f(\omega,\ \epsilon^2)|_{\omega=\omega(r\hat{r}^T)}$ is a scaling of this direction. Notably, when computed with RF*diffusion*, $\nabla_{r_t} \log IGSO3(r_t; \hat{r}_0, \sigma_t^2)$ is rotationally equivariant with respect to x_t. Computation of this score approximation and our use of it in the reverse process is described in Algorithm 2.

We reasoned that approximation in equation (5) may be reasonably accurate for two reasons. First, in the case of Gaussian diffusion probabilistic models where optimizing to convergence would provide $\hat{z}_0(z_t) = E_q[z_0|x_t]$, this approximation holds exactly in the sense that $E_q[\nabla_{z_t} \log q(z_t|z_0)|x_t] = \nabla_{z_t} \log q(z_t|z_0 = \hat{z}_0(x_t))$ Though this does not hold with equality with the IGSO(3), because SO(3) is a Riemannian manifold and is therefore locally Euclidean the IGSO(3) closely resembles a Gaussian for small t. Second, again when t is near to zero, x_t will be close to an un-noised structure and, if the model is trained well, $q(r_0|x_t)$ will be concentrated near $\hat{r}_0$. Finally, we note that this rule has beneficial qualitative behavior -- as with the Gaussian score, the magnitude of $\nabla_{r_t} \log q(r_t|r_0 = \hat{r}_0(x_t))$ will grow roughly linearly with the distance between r_t and $\hat{r}_0$. Consequently, this leads to larger steps when r_t is farther from $\hat{r}_0$.

In summary we approximate reverse transitions by $p_\theta(r_{t-1} \mid x_t) = IGSO3(r_{t-1}; \hat{r}_{t-1}, \sigma_t^2 - \sigma_{t-1}^2)$, where $\hat{r}_{t-1} = \exp\{(\sigma_t^2 - \sigma_{t-1}^2) \nabla_{r_t} \log IGSO3(r_t; \hat{r}_0, \sigma_t^2)\} r_t$ with $\nabla_{r_t} \log IGSO3(r_t; \hat{r}_0, \sigma_t^2)$ computed as in Equation (6).

1.6 Self-Conditioning

Self-conditioning was introduced previously[23] where it was shown to dramatically improve text diffusion. We implement self-conditioning in the manner described in Chen *et al*[23], which we review here.

For sampling in diffusion generative models without self-conditioning, once a denoising step x_t has been sampled the prediction of the denoised data from the previous step ($\hat{x}_{0,prev} = \hat{x}_0(x_{t+1})$) is discarded. However, since each denoising step is typically small, the predictions $\hat{x}_0(x_t)$ can be similar, so much of the denoising computation must be repeated. By contrast, with self-conditioning one saves the denoising predictions at each step and provides them as an input to the denoising model at the next iteration, instead predicting x_0 as $\hat{x}(x_t, \hat{x}_{0,prev})$. When training *with* self-conditioning, on 50% of examples one performs a usual denoising step, setting $\hat{x}_{0,prev}$ = 0 and computing a loss as $L(x_0, \hat{x}_0(x_t, \hat{x}_{0,prev} = 0))$. The other 50% of the time, one (1) simulates an additional forward noising step to obtain $x_{t+1} \sim q(x_{t+1}|x_t)$, (2) computes $\hat{x}_{0,prev} = \hat{x}(x_t, \hat{x}_{0,prev} = 0)$, and (3) computes a loss as $L(x_0, \hat{x}_0(x_t, \hat{x}_{0,prev} = 0))$, backpropagating gradients only through the second denoising step. Training and sampling with self-conditioning are described in Algorithms 1 and 3, respectively

In RF*diffusion*, we input $\hat{x}_{0,prev}$ through the template structure feature and we input x_t as coordinates to the 3D track of RF. Inputting x_t as coordinates, as opposed to the distogram and anglegram used in the template structure feature, allows the network to keep the motif fixed in coordinate space.

1.7 Symmetric diffusion

As discussed in the main text, generating oligomeric assemblies obeying desired point-group symmetry constraints is a design goal. In what follows we describe how we have leveraged RF*diffusion* to design symmetric oligomers.

Point group symmetries may be represented by a finite collection of rotation matrices that form a mathematical group with respect to matrix multiplication as the group operation[31]. For example, we may represent the cyclic symmetry group of order K by the set of rotation matrices that rotate increments of (360/K)° about the z-axis, $C_K = \{R_z^{(k/K)360°}\}_{k=0}^{K-1}$. Analogous representations exist for all other point groups (including dihedral, tetrahedral, octahedral, and icosahedral). Without loss of generality, we set the first rotation to be the identity $R_1 = I_3$. We represent an oligomer with K monomer subunits each with M residues by $X = [x^1, \cdots, x^K]$ where each subunit k consists of the translations and rotations $x^k = ([z^k{}_1, \ldots, z^k{}_M], [r^k{}_1, \ldots r^k{}_M])$. Then, we say an oligomer obeys a point group symmetry $\mathfrak{R} = \{R_1, \ldots, R_K\}$, if $X = [R_1 * x^1, \cdots, R_K * x^1]$ where $R * x^1 = ([Rx^1{}_1, \ldots, Rx^1{}_M], [Rr^1{}_1, \ldots, Rr^1{}_M])$ denotes the rotation of the monomer backbone structure by R.

Previous work has demonstrated some success generating designs with symmetry through hallucination with the inclusion of penalty terms on the deviation of predicted structures from the desired symmetry, but this work suffered from large computational cost (on the order of 1 GPU day per design) and low success rates, presumably due to the inability to precisely control the desired symmetry. We hypothesized that RF*diffusion* by contrast could provide improved control over symmetries in design by enforcing hard constraints during the *reverse process*.

In contrast to the hallucination approach, the desired symmetry is enforced from the beginning of the design trajectory and preserved throughout (Algorithm 3). Although exact symmetry is enforced through explicit symmetrization at each denoising step, we observe that RF*diffusion* provides predictions of the denoised oligomer structures that preserve the desired symmetry nearly exactly, even in the first denoising steps (Fig. S4A). This property of denoised predictions owes to the exact equivariance of RoseTTAFold with respect to global rotations and the approximate equivariance with respect to permutation (i.e. relabeling) of chains. In particular, in Section M.6II we provide a proposition that guarantees that rotation and permutation equivariance of a neural network are sufficient conditions for maintenance of point group

symmetries of the neural network's output. In RoseTTAFold diffusion, exact rotation equivariance is inherited from the SE(3)-transformer architecture used in the structure module of RoseTTAFold[47]. Permutation equivariance by contrast arises if the intermediate representations and outputs for each residue are unaffected by the ordering of chains. This is nearly the case with RF diffusion, with the exception that the RoseTTAFold pair representation contains directional sequence distance feature inputs for each pair of residues, clipped between -32 and 32 residues away; since oligomers are presented to RoseTTAFold by incrementing the sequence position index at the start of each chain[47], the sign of these features breaks exact permutation symmetry. However, we find empirically that deviation from exact symmetry in RF*diffusion* predictions is minimal even at the early steps.

1.8 Proposition on rotation symmetry

We here provide a proposition that provides a mechanism by which predictions of denoised structures maintain the desired symmetry at each step. Here, we consider functions F: $[x_1, \ldots, x_K] \rightarrow [y_1, \ldots, y_K]$ that transform K rigid objects.

Proposition 1: *Consider any function* F: $[x_1, \ldots, x_K] \rightarrow [y_1, \ldots, y_K]$ *and point group symmetry* $\mathfrak{R} = \{R_1, \ldots, R_K\}$. *If* F *is both*

(1) rotation equivariant, that is $F([Rx_1, \ldots, Rx_K]) = [Ry_1, \ldots, Ry_K]$ *for every rotation matrix* R *and*

(2) permutation equivariant, that is $F([x_{\sigma(1)}, \ldots, x_{\sigma(K)}]) = [y_{\sigma(1)}, \ldots, y_{\sigma(K)}]$ *for every permutation* σ*,*

then F *is symmetry preserving. In particular for any* x, $F([R_1x, \ldots, R_Kx]) = [R_1y, \ldots, R_Ky]$ *for some* y.

Notably Proposition 1 holds for any neural network satisfying the assumption on F above. We now prove the proposition.

Proof:

We first establish some basic properties about permutations of point groups. First note that every member $R_k \in \mathfrak{R}$ defines a permutation of $\mathfrak{R}$ since $\{R_kR_1, R_kR_2, \ldots, R_kR_K\} = \mathfrak{R}$. Let σ_k denote the permutation associated with $R_1R_k^T \in \mathfrak{R}$. In particular, σ_k is the permutation such that

for each m, $R_{\sigma_k(m)} = (R_1 R_k^T) R_m$. Notably, $\sigma_k(k) = 1$ because $R_{\sigma_k(k)} = (R_1 R_k^T) R_k = R_1$. Lastly, for any permutation σ, we let $\bar{\sigma}$ denote its inverse, the permutation such that $\bar{\sigma}(\sigma(k)) = k$ for every k, and note that for bar $R_{\bar{\sigma}_k(m)} = (R_k R_1^T) R_m$.

Assume without loss of generality that $F(R_1 x, \ldots, R_K x)_1 = R_1 y$. To prove the proposition, it suffices to show that for any k, $F(R_1 x, \ldots, R_K x)_k = R_k y$. Consider σ_k as defined above. We can write

$$F(R_1 x, \ldots, R_K x)_k \;=\; F(R_{\bar{\sigma}_k(1)} x, \ldots, R_{\bar{\sigma}_k(K)} x)_{\sigma_k(k)} \;=\; F((R_k R_1^T) R_1 x, \ldots, (R_k R_1^T) R_K x)_1,$$

where the first equality follows from the permutation equivariance of F, and the second equality follows from the definitions of σ_k and $\bar{\sigma}_k$. Finally, by the rotation equivariance of F,

$F((R_k R_1^T) R_1 x, \ldots, (R_k R_1^T) R_K x)_1 = [(R_k R_1^T) R_1 y, \ldots, (R_k R_1^T) R_k y]_1 = R_k y.$
Therefore $F(R_1 x, \ldots, R_K x)_k = R_k y$, as desired.

Section 2: Training RF*diffusion*

Supplementary Section 1 described conceptually how RF*diffusion* was trained and used for unconditional generation of protein backbones. In this section we describe specific training details of the initial RoseTTAFold2 architecture and its subsequent fine-tuning. We then describe how we have leveraged RF*diffusion* for generation subject to specific design criteria.

2.1 RoseTTAFold2 Architecture

RoseTTAFold2 (RF2) (referred to simply as RoseTTAFold throughout this paper) is an updated version of the original RoseTTAFold network[47] with multiple architectural improvements: 1) use of a three-track architecture with initial coordinates from a template structure, 2) use of biased axial attention to update 2D pair features by considering geometric constraints between residues inferred from the current 3D structure, 3) communication between 1D, 2D, and 3D tracks through attention biasing, and 4) use of recycling that executes the network multiple times with the updated input embeddings based on outputs from the previous cycle. RF2 contains two major types of architecture blocks: main three-track blocks and the final structure refinement blocks. The 3-track blocks consist of layers of biased row and column attention over the 1D and 2D features, SE(3)-equivariant layers[50] to update 3D coordinates, and layers to communicate between 1D, 2D, and 3D features. The structure refinement block is based on SE(3)-equivariant network which gives refined 3D coordinates based on given 1D and 2D features.

2.2 RoseTTAFold2 Training

RF2 was trained based on a mixture of datasets including 1) monomer/homo-oligomer structures in the PDB, 2) hetero-oligomer structures in the PDB (date cutoff August 2nd, 2021), 3) AlphaFold2 structural models having pLDDT > 0.7[58], and 4) negative protein-protein interaction examples generated by random pairing. The training examples were sampled from each database with a ratio of 2:1:4:1. The model was trained using the masked language model (MLM) loss, distogram (dist) prediction loss, FAPE loss, accuracy estimation loss, bond geometry loss and van der Waals (vdW) energy loss. For the initial round of training, only the first four loss terms were used with crop size 256. After 200 epochs of initial round training, we performed fine-tuning with all the loss terms with crop size 384 for 100 epochs. The entire training took ~4 weeks of training using 64 V100 GPUs on Microsoft Azure. The training details are summarized in Table 3.

Table 3: Details for RoseTTAFold2 training

	Initial training	Fine-tuning
Crop size	256	384
Batch size	64	64
Loss function	$3.0*Loss_{MLM} + 1.0*Loss_{dist} + 10.0*Loss_{FAPE} + 0.1*Loss_{accuracy}$	$3.0*Loss_{MLM} + 1.0*Loss_{dist} + 10.0*Loss_{FAPE} + 0.1*Loss_{accuracy} + 0.1*Loss_{bond} + 0.1*Loss_{vdW}$
Learning rate & scheduling	0.001 Linear warm-up for first 1000 optimization steps, then decay learning rate by 0.95 after every 15000 optimization steps	0.0005 No warm-up. Decay learning rate by 0.95 after every 15000 optimization steps
Examples per epoch	25600	25600
Number of epochs	200	100

2.3 RF*diffusion* Training

RF*diffusion* was trained on only the dataset of monomer structures in the PDB that was used for RF2 training. 20% of examples shown to RF*diffusion* contain no fixed motif – this is the unconditional-generation task. The other 80% of examples shown to RF*diffusion* contain a contiguous motif where the true sequence and structure are provided to the model. RF*diffusion* is trained starting from the final RF2 weights. We train RF*diffusion* using the hyperparameters in Table 4 (distogram CCE loss was used for stability). Although the coordinate inputs and outputs of RoseTTAFold are in units of Angstroms, we define the diffusion process in a downscaled

space by dividing all coordinate values of x_t and $\hat{x}_0$ before performing each diffusion step by a factor of 4 (chosen empirically). Subsequent steps are scaled back up to Angstroms. We reparametrize the “template confidence” feature in RF to input the timestep.

Table 4: Details for RF*diffusion* training

Crop size	384
Pseudo-batch size	64
Loss function	$0.5*\text{Loss}_{\text{trans}} + 1.0*\text{Loss}_{\text{rot}} + 1.0*\text{Loss}_{\text{dist}}$
Learning rate	0.0005 No warm-up. Decay learning rate by 0.95 after every 10000 optimization steps
Examples per epoch	25600
Number of diffusion timesteps	200
Beta schedule	Linear interpolation between $b_0 = 0.01$ and $b_T = 0.07$
Fraction of protein residues masked (when motif is provided)	Randomly picked from a uniform distribution between 20% and 100%, inclusive
Probability of motif being contiguous or discontiguous	0.5
Probability of providing self-conditioning information	0.5
Coordinate scaling	0.25

RF*diffusion* trains to convergence when initialized from RF2 weights in ~5 epochs. Training takes ~3 days on 8 NVIDIA A100 GPUs.

2.4 Conditional training for functional-site scaffolding

Our approach to scaffolding functional motifs with RF*diffusion* follows [5] and treats motif scaffolding as a conditional sampling problem. We partition the residues of a structure into the motif and consider the remainder of the backbone as the scaffold that supports it. For a structure with L residues, we let M denote the (potentially discontiguous) set of indices corresponding to the motif and S be the remaining indices, such that the union of M and S is the set of indices up to L (i.e. $M \cup S = \{1, \cdots, L\}$). We write x^M to denote the structure of the motif residues and x^S to be the scaffold residues such that we may write the whole (un-noised) protein

structure as $x_0 = [x_0^M, x_0^S]$. Our goal is to sample scaffold backbones from the conditional distribution $q(x_0^S | x_0^M)$. To do this, we aim to learn the reversal of the forward noising process applied only to scaffold residues, with the motif held fixed, $p_\theta(x_{t-1}^S | x_t^S, x_0^M) \approx q(x_{t-1}^S | x_t^S, x_0^M)$, where $q(x_{t-1}^S | x_t^S, x_0^M)$ is the conditional forward noising process described in Methods 1.4 and 1.5.

We build on previous work[4] demonstrating that RoseTTAFold may be trained to respect motif constraints provided as inputs through the template structure input features through retraining. Because the division of residues into motif and scaffold is specific to each design problem, we desired to train *RFdiffusion* such that it could be used for any location of the motif within the sequence. To this end we took an amortized training approach, wherein for each motif-scaffolding training example we (1) begin with a pdb structure x_0, (2) choose a random division into "motif" and "scaffold" ($x_0 = [x_0^M, x_0^S]$) following the masking strategy outlined in Table 4, (3) apply noise to the scaffold to obtain $x_t^S \sim q(x_t^S \mid x_0^S)$, and (4) compute a loss on the *RFdiffusion* prediction $\hat{x}_0([x_0^M, x_t^S])$ of $x_0 = [x_0^M, x_0^S]$. In order to encourage *RFdiffusion* to not move the motif, we set the time-step input for motif residues to t=0, and compute the loss on both the motif and the scaffold. Because motif side-chain geometry is crucial for most motif-scaffolding problems, we additionally provide the amino acid sequence and side chain torsion angles for motif residues as inputs to RF*diffusion* (provided through RoseTTAFold's template feature inputs). Overall this strategy is akin to the diffusion model inpainting training and generation described by, who use randomly generated image masks [ref [59]].

Generation of scaffolds conditional on a motif with RF*diffusion* differs from unconditional generation only in (1) the inclusion of noise-free motif backbone and sidechain structure in the template inputs and (2) replacement of the motif backbone coordinates in x_t with un-noised motif coordinates at each step and (3) setting of the timestep for motif residues to 0.

2.5 Fine-tuning RF*diffusion* on protein complexes and with fold information

The version of RF*diffusion* fine-tuned on protein complexes, is trained starting from the base version of RF*diffusion* trained for 5 epochs. The training task consists of monomer examples (50%) and complex examples (50%). When the model is shown a complex example, only one side of the complex is noised, the other side is kept fixed (this is in keeping with established PPI design methods[25] where the target protein is kept fixed). When the model is shown a complex example the model is provided with the residue indices of 0-20% of the residues ("hotspot residues") in the interface on the fixed chain side (the interface is defined as all residues within 10 Å Cβ-Cβ distance of another chain), to permit targeting of the designed binder at inference time. In a separate model, also trained on protein complexes, during both complex and monomer training the model is provided with secondary structure 50% of the time and

(independently) block adjacency information 50% of the time for the noised region. The junctions between blocks of secondary structure and their corresponding entries in the block adjacency matrix are masked during training, such that at inference time, one does not need to specify exact, *per residue* secondary structure and block adjacency matrices. Specifically, 0-75% of secondary structure (and corresponding adjacency, when provided), is masked, with this masking occurring over junctions in secondary structure (mask length 1-8 residues).

2.6 Fine-tuning RF*diffusion* on enzyme active site scaffolding

The version of RF*diffusion* fine-tuned for enzyme active site scaffolding is trained starting from the base version of RF*diffusion* trained for 5 epochs. During fine-tuning 30% of tasks are from the base model task set (Table 4) and the other 70% are the triple contact task, in which a random set of 3 residues all >10 residues apart in sequence space but with pairwise Cβ-Cβ distances < 6 Å is selected to form a model "active site". These three residues are included in the motif, and for each, there is a 50% chance of including one flanking residue. If no such triad is found in the monomer (as is the case for approximately 23% of training PDBs), the task would fall back to the base model training task. In addition, the motif-specific displacement loss is upweighted by a factor of 10 to encourage the network to keep the motif fixed, compensating for the fact that otherwise motif recapitulation would comprise a significantly lower portion of the overall loss due to the much shorter motif length in this task. The network was fine-tuned for 5 epochs in this manner.

2.7 Guiding RF*diffusion* inference with external potentials

In addition to the network's inbuilt ability to condition on structural motifs, the inference process can be steered by external potential functions to generate proteins which possess arbitrary desired properties, such as the existence of contacts with another protein or a desired surface concavity. Previous work has demonstrated that diffusion models can be made to sample conditionally from $p_\theta(x_0|y)$ without retraining if given a classifier able to operate on noisy samples, $p(y|x_t)$. In particular, $p(y = 1|x_t)$ may be understood as a predicted probability that an example x_0 has a property of interest (or is in a given "class") given only the noised observation x_t. In contrast to unguided generation, wherein one noisily moves in the direction $\nabla_{x_t} \log p_\theta(x_t)$ (which points toward $\hat{x}_0$), with guidance on instead follows $\nabla_{x_t} \log p_\theta(x_t) + \nabla_{x_t} \log p(y = 1|x_t)$ in the reverse step[11,48]. In the present work, we construct heuristic approximations of these classification log probabilities $P(x_t) \approx \log p(y = 1|\, x_t)$ for two protein conditional generation objectives, symmetric oligomer design (Fig. S4C) and enzyme design with concave pockets (Fig. S5). We now describe the details of choices of $P(x_t)$in these applications.

When designing symmetric oligomers, we employ an inter-chain and intra-chain contact potential to promote the formation of contacts between subunits. Letting $Z = [z_1, \ldots, z_K]$ denote

the C_α coordinates in oligomer with K subunits and L residues in each subunit (so for each k, $z_k = [z_{k,1}, \ldots, z_{k,L}]$ with each $z_{k,l} \in R^3$) we set

$$P_{sym}(Z) = \sum_{1\le k,k'\le K} \sum_{1\le l,l'\le L} (1[k \neq k']w_{intra} + 1[k = k']w_{intra})\, Switch(||z_{k,l} - z_{k',l'}||_2^2),$$

where w_{inter} and w_{intra} weight the inter-chain and intra-chain potentials and are set to 2 and 0.2 respectively to prioritize the formation of inter-subunit contacts while encouraging individual subunits to be well-packed.

$Switch(r) = \frac{1-(\frac{r-d_0}{r_c})^n}{1-(\frac{r-d_0}{r_c})^m}$, is a switching function which smoothly transitions from 1 if two atoms are within contact range to 0 when they are out of range. We set the hyperparameters that control its functional dependence on distance as $n = 6$, $m = 12$, $d_0 = 8$, and $r_c = 4$, based on physical intuition of the contact distances we would expect between interacting sidechains. It is sufficient to bias the distribution at high t to promote contacts in the higher order structure, and unnecessary to continue to do so at low t as the quaternary structure has already been sufficiently determined, so we scale the potential by a guide-scale $g(t)$:

$P_{sym}'(x_t, t) = g(t)P_{sym}(x_t)$, for $g(t) = (\frac{t}{T})^2$.

When designing enzymes, in addition to recapitulating the sidechain geometry of the active site, a pocket must be formed which has shape complementarity to the substrate. This condition can be captured effectively by a simple attractive-repulsive potential parameterized by the minimum distance between enzyme alpha-carbons and substrate atoms. Denoting the coordinates of a substrate with K atoms by $s = \{s_k\}_{k=1}^K$ and the coordinates of alpha-carbons by $z = [z_1, \ldots, z_L]$, we set:

$$P_{enzyme}(z, s) = w_{attr}[\sum_{1\le l\le L} Switch(\min_{1\le k\le K} ||z_l - s_k||_2^2)] - w_{rep}[\sum_{1\le l\le L} Rep(\min_{1\le k\le K} ||z_l - s_k||_2^2)],$$

where $Rep(r; r_0, p) = max(0, \frac{|r-r_0|^p}{pr_0^{p-1}})$ and we set $w_{attr} = 1$, $w_{rep} = 4$, $r_c = 2$.

The gradient of $Rep(r; r_0, p)$ decays smoothly from -1 at $r = 0$ to 0 at $r = r_0$, penalizing clashes between the protein backbone and the substrate. We do not use a guide scale with P_{enzyme}, as the potential relates to fine-grained details of the structure which are not fully determined until late in the reverse diffusion process

Experimentally, the model is sufficiently receptive and robust to bespoke potentials based on physical intuition that we were able to achieve our objectives of interface production in the case of symmetric oligomer design and implicit substrate modeling in the case of enzyme design without exhaustive hyperparameter tuning.

Section 3: *In silico* experimental methods

3.1 Unconditional benchmarking

To test RF*diffusion* on unconditional generation of monomers (Fig. 1C-F), we generated 100 designs for lengths 70, 100, 200, 300, 400, 600, 800 and 1000 amino acids. For each backbone, we generated 8 sequences with ProteinMPNN and subsequently predicted their structures with AF2 (or ESMFold - Fig. S1I). The best sequence (by alignment of the predicted structure to the design model) was taken for each backbone. We benchmarked against the recently-published RoseTTAFold Hallucination[4]. As some knowledge of how best to use RoseTTAFold for Hallucination is required, these samples were generated by the respective expert. ProteinMPNN was used to design sequences for all benchmarking designs. For ProteinMPNN, a sampling temperature of 0.1 was used, and cysteines were omitted from the designs (as these are often problematic for protein purification).

3.2 Conditional benchmarking

The full conditional benchmark is described in Table 1, and encompasses 25 design challenges from six recent publications[4,5,24,32–34]. RF*diffusion* was compared to RoseTTAFold Hallucination and RF*joint* Inpainting. While both Hallucination and Inpainting are able to generate sequences directly, for the fairest comparison, we also redesigned the sequence with ProteinMPNN, and took the best of 8 sequences per backbone. Both RF*joint* Inpainting and RF Hallucination are able to scaffold structure without sequence, so in cases where functional-site residues were not required for function, these methods were permitted to redesign the sequence of the non-functional residues, which is generally beneficial for design. Finally, as Hallucination requires some expert knowledge and empirical hyperparameter tuning, some exploration of the benchmark set was permitted, and these designs were generated by the respective expert.

For a number of comparisons made in the paper (Fig. S1, S2C-D), a smaller benchmark encompassing a subset of unconditional and conditional benchmark problems described above was used.

3.3 Assessing diversity of designs

Designs were assessed for their diversity both to each other, and to the PDB (PDB100 April 19, 2022), using the TM score[60]. In Fig. S1F, designs were clustered at a 0.6 pairwise TM score cutoff.

3.4 Assessing choice of losses

Previous work on using DDPMs for protein design[8] has used Frame Aligned Point Error (FAPE) as the loss function. FAPE was introduced in AF2 and was also used to train RF2. FAPE is SE3 invariant but not invariant to reflections, this makes it an ideal loss for protein structure prediction where the exact global orientation of the predicted structure is arbitrary, but chirality within the structure is important. With a DDPM, however, $\hat{x}_0$ must be in the same global frame as x_t since $\hat{x}_0$ and x_t are interpolated between to generate x_{t-1}. We reasoned that, as FAPE is SE3 *invariant*, a model trained with FAPE would not learn to make predictions in the same global frame as the inputs. We tested this by comparing a model trained with FAPE to a model trained with the C_α and rotation squared distance losses described in Section 1.3. By contrast these losses are not SE3 invariant.

We found that the model trained with FAPE was unable to perform unconditional generation; the model was unable to preserve a global frame and this caused the denoising process to diverge after repeatedly interpolating between coordinates in different frames. In the motif scaffolding task, $\hat{x}_0$ and x_t can be aligned to one another using the fixed motif, this effectively eliminates the global frame problem as any arbitrary SE3 action applied by the model can be reversed by this motif-alignment step. In the motif-scaffolding task we found that the model trained with displacement losses dramatically outperformed the model trained with FAPE (Fig S1A). We attribute the performance difference to better agreement of the squared distance losses with the objective of matching the reversal of the forward process, which thereby yields a model that better matches the data distribution (see Sections 1.3-1.5).

3.5 Design of symmetric oligomers

To better understand RF*diffusion*'s capacity for designing symmetric oligomers, we generated backbones for the following groups: dihedral (D2, D3, D4, D5), cyclic (C3, C5, C6, C8, C10, C12), tetrahedral, octahedral, and icosahedral. We tested both RF*diffusion*'s ability to design symmetry for these groups with and without a guiding potential function for inter- and intra-chain contacts, weighting in all cases the intrachain contacts over the interchain. For dihedral, cyclic, and tetrahedral symmetries, protomers had 60-110 AA per chain, and for a subset of the cyclic symmetries (C3, C5, C6), additional models were designed with large protomers (150-400 AA per chain) to test RF*diffusion*'s ability to design unconditional yet large oligomers. The octahedral and icosahedral models were designed by modeling the minimal number of subunits (100-200 AA per protomer) required to capture all axes of symmetry (O: 4-, 3-, and 2-fold; I: 5-, 3-, and 2-fold).

Original backbones were filtered by sufficient oligomeric interfaces (determined by C_α-C_α backbone distances between chains) to enrich for backbones with a higher likelihood for assembly following design. Cyclic and D2 symmetries were filtered for backbones consisting of protomers forming at least two distinct 10 residue interfaces, whereas all other symmetries required at least three distinct 10 residue interfaces. Following filtering, all backbones were redesigned with ProteinMPNN, and then sequences were validated by AF2 (for the cyclic and

dihedral symmetries). Given the complexity and challenge these symmetries present, we provided AF2 with an initial guess[25] and increased the number of recycles the model could use in the predictions. Tetrahedra were predicted using RoseTTAFold, and octahedron and icosahedron were predicted with AF2 along their C3 axes of symmetry only. Designs were considered successful (success rates for cyclic and dihedral shown in Fig. S4) if the structure predictions had a mean pLDDT > 80 and an RMSD between prediction and design model of < 2 Å. This same filtering regime was also used for the cage symmetries, but applied to the C3 predictions (for octahedra and icosahedra), and the monomer predictions (for tetrahedra).

3.6 Design of p53 helix scaffolds

To design scaffolds able to hold the Mdm2-binding helix of p53, we used the version of RF*diffusion* fine-tuned on protein complexes (see Methods 2.5), and provided the network with both the p53 helix and the whole Mdm2 protein structure from PDB: 1YCR. To encourage extra contacts with the target protein, we used an external potential to encourage inter-chain contacts (see Methods 2.7). No fold information was provided to the network in this case.

3.7 Design of symmetric nickel binding oligomers

To design the C4-symmetric Nickel binding proteins (Fig 3H,S6), we started from a set of backbone dependent inverse rotamers[45] sampled for pieces of ideal alpha-helix (φ = -40°, ψ = -60°) containing the Histidine rotamers in the middle, and an Alanine residue on either side of the Histidine (three residues total per asymmetric unit going into the model). The rotamers chosen were of probability 0.3502, 0.1207, 0.0647, 0.0474, 0.0469, and 0.0365. The 3-residue inverse rotamers were then positioned relative to the Z-axis such that (A) the imidazole group was perfectly vertical/flat with respect to the Z-axis, and (B) the NE2 atom of the imidazole group was 2.3 Angstroms away from the Z-axis (a common coordination distance for His-containing square-planar nickel coordinating sites in the MetalPDB[61]).

Once the 3-residue inverse rotamers were aligned with respect to the Z-axis via the above procedure, the rotamers were symmetrized around the Z-axis and fed to the model as separate chains, each with a fixed motif. 100 reverse diffusion trajectories were run for the full T=200 steps for all 7 symmetric motifs, with 50 residues designed on either side of the inverse rotamer helix chunks in each chain (total complex length 412 residues). As in Methods section 2.7, an intra-chain guiding potential was used during the trajectory with a weight of 1, an inter-chain guiding potential with a weight of .06, and a global multiplicative factor of 2x. Half (50) the designs per motif were designed such that the effect of the external potential decayed quadratically during the trajectory, while the other half having potentials decay cubicly. Importantly, multiple models from the training session which produced *RFdiffusion* were tested to see which checkpoint could scaffold the sites most accurately, and pilot experiments suggested the set of weights after the 8th epoch, rather than the 5th epoch (standard used for this paper) should be used.

Before sequence design with ProteinMPNN, RF*diffusion* outputs were filtered to only allow designs for which the backbone RMSD from the model < 1 Å RMSD from the true motif. This yielded 199 backbones, and ProteinMPNN was then used to perform symmetric sequence design on all residues except the Histidines (including the original Alanines), with 16 sequences per backbone. AF2 was then used to predict the structure of all designed sequences.

3.8 Design of protein binders to rigid targets

To test the ability of RF*diffusion* to design *de novo* binders to rigid targets, we designed binders to five targets: PD-L1 (PDB: 5O45), IL7 Receptor α (PDB: 3DI3), Insulin Receptor (PDB: 4ZXB), TrkA Receptor (PDB: 1WW7) and Flu Hemagglutinin (PDB: 5VLI). We generated designs both with and without fold conditioning, with the folds used derived from scaffold sets typically used for Rosetta-based protein binder design[42]. In all cases, we targeted binders, using input "hotspot" residues, to a specific site on the target protein. In line with current best practice[25], we tried using Rosetta FastRelax[44] before running a single ProteinMPNN, although we found that this was not systematically helpful for design success rates. For the five design cases, we generated several thousand designs. We classed a design as successful if it had AF2 pAE of interaction between binder and target < 10 (this has been shown to be highly indicative of design success), as well as RMSD between the designed binder and the AF2 prediction < 1 Å, and AF2 pLDDT > 80. Success rates are reported in Fig. 4B, and were several orders of magnitude higher than with traditional Rosetta binder design.

3.9 Design of helical peptide binders

Binders were designed to two helical peptides: Bim (DMRPEIWIAQELRRIGDEFNAYYARR; PDB: 6X8O) and PTH (SVSEIQLMHNLGKHLNSMERVEWLRKKLQDVHNF; PDB: 1ET1). External potentials were used to promote interactions between the binder and target (see Methods 2.7). Note that for these designs, an earlier version of RF*diffusion* was used. Briefly, this network was trained with a coordinate scaling of 1/15, and without self-conditioning. It was also trained with autoregressive sequence decoding over the final 40 diffusion steps. The model was trained for 4 epochs.

3.10 Figures and statistics shown in the paper

Protein structures depicted in this paper were rendered in PyMOL[62], and graphs were plotted with Matplotlib[63] and Seaborn[64]. Note that for all boxplots displayed in the paper, for aesthetic reasons, outliers are not displayed. Appropriate statistical tests were performed using SciPy[65], as indicated in figure legends.

Section 4: *In vitro* experimental methods

4.1 Plasmid construction

Symmetric oligomer designs were ordered as synthetic genes (eBlocks, Integrated DNA Technologies) with compatible BsaI overhangs to the target cloning vector, LM0627 (ref [7]) for

Golden Gate assembly. LM0627 is a modified expression vector containing a Kanamycin resistance gene and a ccdb lethal gene between BsaI cut sites to select out failed clones. Subcloning into LM0627 results in the following product:
MSG-[***protein***]-GSGSHHWGSTHHHHHH, with the C-terminal SNAC cleavage tag and 6XHis affinity tag respectively underlined. Helical peptide binders were ordered in a similar format, except for the addition of adaptors (GGGSGGGGSASHMRS, SSEISFCSEPPPSRRS) permitting cloning into the pETcon3 vector (as well as LM0627), to permit both purification in *E. coli* and yeast surface display.

4.2 Protein expression and purification

For the oligomeric and protein binder expression screens, a previously reported protocol was followed[7], with some modifications as denoted. In short, Golden Gate subcloning reactions of designs were carried out in 96-well PCR plates in 1µL volume. Reaction mixtures were then transformed into a chemically competent expression strain (BL21(DE3)), and 1-hour outgrowths were split directly into four 96-deep well plates containing 0.9-1.0mL of auto-induction media (autoclaved TBII media supplemented with Kanamycin, 2mM MgSO4, 1X 5052) for a final total volume of approximately 4mL. The following day (20-24 hrs later), cells were harvested and lysed, and clarified lysates were applied directly to a 50µL bed of Ni-NTA agarose resin in a 96-well fritted plate equilibrated with a Tris wash buffer. After sample application and flow through, the resin was thoroughly washed, and samples were eluted in 200µL of a Tris elution buffer containing 300mM imidazole. For oligomers, 0.5 M EDTA was spiked into the eluates (10 mM final) to reduce self-association due to the 6XHis tag. All eluates were sterile filtered with a 96-well 0.22µm filter plate (Agilent 203940-100) prior to size exclusion chromatography.

Protein designs were then screened via SEC using an AKTA FPLC outfitted with an autosampler capable of running samples from a 96-well source plate. The symmetric oligomers were run on a Superdex200 Increase 5/150 GL column (Cytiva 28990945) (10,000 to 400,000 Da separation range). For the cyclic and dihedral symmetric oligomers, a running buffer of 20 mM NaPhos pH 7.4, 100 mM NaCl was used. For the tetrahedral, octahedral, and icosahedral oligomers, samples were run in 20 mM Tris pH 8, 50 mM NaCl, 100 mM Glycine. To improve peak resolution, the SEC column was connected directly in line from the autosampler to the UV detector. 0.25 mL fractions were collected from each run, and selected fractions were pooled for further analysis (native mass spectrometry, negative stain EM, SDS-page).

Purification of helical peptide binders was performed similarly, except that the wash buffer comprised 20 mM Tris pH 8, 100 mM NaCl, and SEC purification was performed on a 10/300 Superdex 75 column.

4.3 Negative-Stain EM sample preparation

De novo designed oligomeric proteins were diluted to 0.1mg/mL for negative stain. 3µL of the diluted complexes were immediately negatively stained after diluting using Gilder Grids overlaid with a thin layer of carbon and 2% uranyl formate.

4.4 Negative-Stain EM data collection, processing, and validation

Data were collected on an Talos L120C 120kV electron microscope equipped with a CETA camera. A total of ~150 images were collected per sample by using a random defocus range of 1.3–2.3 µm, with a total exposure of between 30 and 50 e−/A2, with a pixel size of either 1.54 or 2.49 Å/pixel. All data were automatically acquired using EPU (ThermoFisher Scientific). All data processing was performed using CryoSPARC V4.0.3 (PMID: 28165473). The parameters of the contrast transfer function (CTF) were estimated using Patch CTF, with minimal and maximal fitting resolutions set to 40Å and 8Å, respectively. Particles were picked initially in a reference-free manner using blob picker, followed by template picking using well-defined 2D classes of intact oligomers. Particles were extracted after correcting for the effect of the CTF for each micrograph with a box size of 80 pixels. Extracted particles were sorted by reference-free 2D classification. Given the small size of these particles, 2D classification was performed both in the presence and absence of CTF correction, with the best parameters and resulting classes selected for subsequent 3D *ab initio* reconstruction. 3D *ab initio* jobs for each RF*diffusion* construct were performed by sorting into 3-4 classes in the presence and absence of the appropriate symmetry operator and compared. Resulting *ab initio* maps which exhibited a striking degree of similarity to both the 2D class average projections and computational design model were next rigid-body docked against the AlphaFold2 prediction model for further validation. For HE0537, the *ab initio* map and corresponding particles which demonstrated highest agreement to the design model were homogeneously refined in the presence of applied symmetry, with a maximum alignment resolution set to 15Å, followed by rigid-body docking against the AF2 predictive model, similar to the other designs.

4.5 Yeast surface display screening of peptide binding

EBY100 *S. cerevisiae* were transformed with 50ng digested pETcon3 and 100ng insert DNA following a protocol described previously[42]. EBY100 cultures were grown in C-Trp-Ura medium supplemented with 2% (w/v) glucose (CTUG). For induction of expression, saturated cultures were diluted into SGCAA medium supplemented with 0.2% (w/v) glucose and induced at 30°C for 16-24h. Cells were washed with PBS supplemented with 1% (w/v) bovine serum albumin (PBSF), and labeled for 40 minutes at room temperature with 10nM biotinylated peptide (no-avidity conditions). After incubation, cells were washed and resuspended in PBSF and sorted on an Attune NxT Flow Cytometer (Thermo Fisher Scientific).

4.6 Fluorescence polarization

Fluorescence polarization binding assays were carried out in 96-well plates (Corning 3686), with two-fold serial dilution of designed peptide binders in the presence of 0.5nM TAMRA-labelled peptide targets. Protein and peptide were diluted from their stock concentration into 20mM Tris pH 8, 100mM NaCl, 0.1% v/v Tween 20. After incubating the peptide and binder for one hour at room temperature, the fluorescence polarization was measured at the excitation and emission wavelengths of the TAMRA dye (530/590nm), in a Synergy Neo2 multi-mode platereader. Titrations were conducted in replicate, and the K_D was fitted with SciPy[65]. Specifically, curves

were fit to N observations of an observed signal, $Signal_i$, at titrated concentrations $[A_{tot}]_i$ according to the following equation:

$$Signal_i = Baseline + Amplitude \frac{AB_{conc}([A_{tot}]_i, [B_{tot}], K_d)}{[B_{tot}]},$$

Where $[B_{tot}]$ is the known total concentration of the binder, $Baseline$ and $Amplitude$ are free parameters, and the concentration of the bound state $[AB]$ is computed as

$$AB_{conc}([A_{tot}]_i, [B_{tot}], K_d) = (([A_{tot}] + [B_{tot}] + K_d) \pm \sqrt{([A_{tot}] + [B_{tot}] + K_d)^2 - 4[A_{tot}][B_{tot}]})/2.$$

The unknown parameters (K_D, $Baseline$ and $Amplitude$) were fit using scipy.optimize.curve_fit, $[B_{tot}]$ was additionally fit in the optimization, but only allowed to within 0.5 nM $\pm$ 0.1%.

4.7 Bio-layer Inferometry (BLI) Binding Experiments

BLI experiments were performed on an Octet Red96 (ForteBio) instrument, with streptavidin coated tips (Sartorius Item no. 18-5019). Buffer comprised 1X HBS-EP+ buffer (Cytiva BR100669) supplemented with 0.1% w/v bovine serum albumin. Tips were pre-incubated in buffer for at least 10 minutes before use. Tips were then sequentially incubated in 50nM biotinylated Bim peptide (loading, 500s), buffer (baseline, 150s), designed binder (association, 1200s) and buffer (dissociation, 600s). Due to the extremely slow dissociation of Bim from the designed binders, it was not possible to calculate a precise K_D, but estimates suggest significantly sub-nanomolar affinity.

Bibliography

1. Dauparas, J. *et al.* Robust deep learning-based protein sequence design using ProteinMPNN. *Science* eadd2187 (2022) doi:10.1126/science.add2187.
2. Ferruz, N., Schmidt, S. & Höcker, B. ProtGPT2 is a deep unsupervised language model for protein design. *Nat. Commun.* **13**, 4348 (2022).
3. Singer, J. M. *et al.* Large-scale design and refinement of stable proteins using sequence-only models. *PLOS ONE* **17**, e0265020 (2022).
4. Wang, J. *et al.* Scaffolding protein functional sites using deep learning. *Science* **377**, 387–394 (2022).
5. Trippe, B. L. *et al.* Diffusion probabilistic modeling of protein backbones in 3D for the motif-scaffolding problem. (2022) doi:10.48550/ARXIV.2206.04119.
6. Anishchenko, I. *et al.* De novo protein design by deep network hallucination. *Nature* **600**, 547–552 (2021).
7. Wicky, B. I. M. *et al.* Hallucinating symmetric protein assemblies. *Science* **378**, 56–61 (2022).
8. Anand, N. & Achim, T. Protein Structure and Sequence Generation with Equivariant Denoising Diffusion Probabilistic Models. (2022) doi:10.48550/ARXIV.2205.15019.
9. Luo, S. *et al.* Antigen-Specific Antibody Design and Optimization with Diffusion-Based Generative Models. 13.
10. Ho, J., Jain, A. & Abbeel, P. Denoising Diffusion Probabilistic Models. Preprint at https://doi.org/10.48550/arXiv.2006.11239 (2020).
11. Sohl-Dickstein, J., Weiss, E. A., Maheswaranathan, N. & Ganguli, S. Deep Unsupervised Learning using Nonequilibrium Thermodynamics. Preprint at https://doi.org/10.48550/arXiv.1503.03585 (2015).

12. Ramesh, A. *et al.* Zero-Shot Text-to-Image Generation. Preprint at http://arxiv.org/abs/2102.12092 (2021).
13. Saharia, C. *et al.* Photorealistic Text-to-Image Diffusion Models with Deep Language Understanding. Preprint at http://arxiv.org/abs/2205.11487 (2022).
14. Jendrusch, M., Korbel, J. O. & Sadiq, S. K. AlphaDesign: A de novo protein design framework based on AlphaFold. 2021.10.11.463937 Preprint at https://doi.org/10.1101/2021.10.11.463937 (2021).
15. Wu, K. E. *et al.* Protein structure generation via folding diffusion. (2022) doi:10.48550/arXiv.2209.15611.
16. Watson, J. L., Bera, A., Juergens, D., Wang, J. & Baker, D. X-ray crystallographic validation of design from this paper | Science | AAAS. (2022).
17. Baek, M., McHugh, R., Anishchenko, I., Baker, D. & DiMaio, F. Accurate prediction of nucleic acid and protein-nucleic acid complexes using RoseTTAFoldNA. 2022.09.09.507333 Preprint at https://doi.org/10.1101/2022.09.09.507333 (2022).
18. De Bortoli, V. *et al.* Riemannian Score-Based Generative Modelling. Preprint at https://doi.org/10.48550/arXiv.2202.02763 (2022).
19. Leach, A., Schmon, S. M., Degiacomi, M. T. & Willcocks, C. G. Denoising Diffusion Probabilistic Models On SO(3) For Rotational Alignment. 8 (2022).
20. Jumper, J. *et al.* Highly accurate protein structure prediction with AlphaFold. *Nature* **596**, 583–589 (2021).
21. Wu, R. *et al.* High-resolution de novo structure prediction from primary sequence. 2022.07.21.500999 Preprint at https://doi.org/10.1101/2022.07.21.500999 (2022).
22. Lin, Z. *et al.* Language models of protein sequences at the scale of evolution enable accurate structure prediction. 31.
23. Chen, T., Zhang, R. & Hinton, G. Analog Bits: Generating Discrete Data using Diffusion Models with Self-Conditioning. Preprint at https://doi.org/10.48550/arXiv.2208.04202 (2022).

24. Lee, J. S. & Kim, P. M. ProteinSGM: Score-based generative modeling for de novo protein design. *bioRxiv* (2022) doi:10.1101/2022.07.13.499967.
25. Bennett, N. *et al.* Improving de novo Protein Binder Design with Deep Learning. 2022.06.15.495993 Preprint at https://doi.org/10.1101/2022.06.15.495993 (2022).
26. Anand, N. & Huang, P. Generative modeling for protein structures. in *Advances in Neural Information Processing Systems* vol. 31 (Curran Associates, Inc., 2018).
27. Basanta, B. *et al.* An enumerative algorithm for de novo design of proteins with diverse pocket structures. *Proc. Natl. Acad. Sci.* **117**, 22135–22145 (2020).
28. Pan, X. *et al.* Expanding the space of protein geometries by computational design of de novo fold families. *Science* **369**, 1132–1136 (2020).
29. Marcandalli, J. *et al.* Induction of Potent Neutralizing Antibody Responses by a Designed Protein Nanoparticle Vaccine for Respiratory Syncytial Virus. *Cell* **176**, 1420-1431.e17 (2019).
30. Butterfield, G. L. *et al.* Evolution of a designed protein assembly encapsulating its own RNA genome. *Nature* **552**, 415–420 (2017).
31. Goodsell, D. S. & Olson, A. J. Structural symmetry and protein function. *Annu. Rev. Biophys. Biomol. Struct.* **29**, 105–153 (2000).
32. Sesterhenn, F. *et al.* De novo protein design enables the precise induction of RSV-neutralizing antibodies. *Science* **368**, (2020).
33. Yang, C. *et al.* Bottom-up de novo design of functional proteins with complex structural features. *Nat. Chem. Biol.* **17**, 492–500 (2021).
34. Glasgow, A. *et al.* Engineered ACE2 receptor traps potently neutralize SARS-CoV-2. *Proc. Natl. Acad. Sci. U. S. A.* **117**, 28046–28055 (2020).
35. Hunt, A. C. *et al.* Multivalent designed proteins neutralize SARS-CoV-2 variants of concern and confer protection against infection in mice. *Sci. Transl. Med.* **14**, eabn1252 (2022).
36. Silverman, J. *et al.* Multivalent avimer proteins evolved by exon shuffling of a family of

human receptor domains. *Nat. Biotechnol.* **23**, 1556–1561 (2005).

37. Detalle, L. *et al.* Generation and Characterization of ALX-0171, a Potent Novel Therapeutic Nanobody for the Treatment of Respiratory Syncytial Virus Infection. *Antimicrob. Agents Chemother.* **60**, 6–13 (2016).
38. Strauch, E.-M. *et al.* Computational design of trimeric influenza-neutralizing proteins targeting the hemagglutinin receptor binding site. *Nat. Biotechnol.* **35**, 667–671 (2017).
39. Boyoglu-Barnum, S. *et al.* Quadrivalent influenza nanoparticle vaccines induce broad protection. *Nature* **592**, 623–628 (2021).
40. Walls, A. C. *et al.* Elicitation of Potent Neutralizing Antibody Responses by Designed Protein Nanoparticle Vaccines for SARS-CoV-2. *Cell* **183**, 1367-1382.e17 (2020).
41. Quijano-Rubio, A., Ulge, U. Y., Walkey, C. D. & Silva, D.-A. The advent of de novo proteins for cancer immunotherapy. *Curr. Opin. Chem. Biol.* **56**, 119–128 (2020).
42. Cao, L. *et al.* Design of protein-binding proteins from the target structure alone. *Nature* **605**, 551–560 (2022).
43. Ribeiro, A. J. M. *et al.* Mechanism and Catalytic Site Atlas (M-CSA): a database of enzyme reaction mechanisms and active sites. *Nucleic Acids Res.* **46**, D618–D623 (2018).
44. Leaver-Fay, A. *et al.* ROSETTA3: an object-oriented software suite for the simulation and design of macromolecules. *Methods Enzymol.* **487**, 545–574 (2011).
45. Shapovalov, M. V. & Dunbrack, R. L. A Smoothed Backbone-Dependent Rotamer Library for Proteins Derived from Adaptive Kernel Density Estimates and Regressions. *Structure* **19**, 844–858 (2011).
46. Lin, Z., Sercu, T., LeCun, Y. & Rives, A. Deep generative models create new and diverse protein structures. 17.
47. Baek, M. *et al.* Accurate prediction of protein structures and interactions using a three-track neural network. *Science* **373**, 871–876 (2021).
48. Song, Y. *et al.* Score-Based Generative Modeling through Stochastic Differential Equations.

Preprint at http://arxiv.org/abs/2011.13456 (2021).

49. Xu, M. *et al.* GeoDiff: a Geometric Diffusion Model for Molecular Conformation Generation. Preprint at http://arxiv.org/abs/2203.02923 (2022).
50. Fuchs, F. B., Worrall, D. E., Fischer, V. & Welling, M. SE(3)-Transformers: 3D Roto-Translation Equivariant Attention Networks. Preprint at http://arxiv.org/abs/2006.10503 (2020).
51. Thomas, N. *et al.* Tensor field networks: Rotation- and translation-equivariant neural networks for 3D point clouds. Preprint at https://doi.org/10.48550/arXiv.1802.08219 (2018).
52. Larochelle, P. M., Murray, A. P. & Angeles, J. A Distance Metric for Finite Sets of Rigid-Body Displacements via the Polar Decomposition. *J. Mech. Des.* **129**, 883–886 (2007).
53. Huynh, D. Q. Metrics for 3D Rotations: Comparison and Analysis. *J. Math. Imaging Vis.* **35**, 155–164 (2009).
54. Dhariwal, P. & Nichol, A. Diffusion Models Beat GANs on Image Synthesis. Preprint at http://arxiv.org/abs/2105.05233 (2021).
55. Nikolayev, D. I. & Savyolova, T. I. Normal Distribution on the Rotation Group SO(3). 33.
56. Vincent, P. A Connection Between Score Matching and Denoising Autoencoders. *Neural Comput.* **23**, 1661–1674 (2011).
57. Solà, J., Deray, J. & Atchuthan, D. A micro Lie theory for state estimation in robotics. Preprint at https://doi.org/10.48550/arXiv.1812.01537 (2021).
58. Hsu, C. *et al.* Learning inverse folding from millions of predicted structures. in *Proceedings of the 39th International Conference on Machine Learning* 8946–8970 (PMLR, 2022).
59. Saharia, C. *et al.* Palette: Image-to-Image Diffusion Models. Preprint at http://arxiv.org/abs/2111.05826 (2022).
60. Zhang, Y. & Skolnick, J. Scoring function for automated assessment of protein structure template quality. *Proteins* **57**, 702–710 (2004).
61. Andreini, C., Cavallaro, G., Lorenzini, S. & Rosato, A. MetalPDB: a database of metal sites

in biological macromolecular structures. *Nucleic Acids Res.* **41**, D312–D319 (2013).

62. Schrödinger, L. & DeLano, W. PyMOL. (2020).
63. Hunter, J. D. Matplotlib: A 2D graphics environment. *Comput. Sci. Eng.* **9**, 90–95 (2007).
64. Waskom, M. *et al.* mwaskom/seaborn: v0.8.1 (September 2017). (2017) doi:10.5281/zenodo.883859.
65. Virtanen, P. *et al.* SciPy 1.0: fundamental algorithms for scientific computing in Python. *Nat. Methods* **17**, 261–272 (2020).