Craft

TorchCraft: Unified binder design by inverting an all-atom structure predictor

TorchCraft Team*

Changping Laboratory, Beijing, China

Abstract

All-atom structure predictors model diverse molecular interactions, but using their learned structural priors for binder design remains challenging. Here we present TorchCraft, a unified binder-design framework that optimizes sequence logits through a frozen all-atom predictor. Implemented in TorchFold, TorchCraft combines confidence, contact, geometric, and sequence-prior objectives within a shared optimization procedure for minibinders, framework-conditioned VHHs, cyclic peptides, and ligand-binding proteins. Using pretrained AlphaFold3 weights, TorchCraft generated representative minibinders and VHHs with experimentally measured binding across four targets in each format, without post hoc sequence redesign. Computational benchmarks further demonstrated the framework's applicability to cyclic peptides and ligand-conditioned pocket design. TorchCraft extends predictor inversion to multiple binder formats and molecular contexts, providing a common framework for reusing all-atom structural priors in design.

Correspondence: Mingchen Chen (mingchenchen@cpl.ac.cn)
Project Page: https://torchx-cpl.github.io
Code: https://github.com/Mingchenchen/TorchX

1 Introduction

De novo binder design seeks sequences that fold reliably and recognize specified molecular targets.[1–3] Advances in structure prediction have made learned structural priors increasingly useful for this task. AlphaFold2 models protein structure and assembly, whereas all-atom predictors such as AlphaFold3 can additionally represent nucleic acids, small molecules, ions, and modified residues.[4–7] These capabilities create an opportunity to guide binder design within a common molecular representation, provided that the predictor's outputs can be converted into effective sequence-optimization objectives.

Current design workflows combine structure generation, sequence assignment, and evaluation in different ways. Backbone-generation methods, including RFdiffusion, are commonly paired with ProteinMPNN or LigandMPNN and subsequent structure prediction.[8–13] These workflows provide complementary ways to explore sequence and structure, with task scope depending on the molecular representations and constraints supported by each component.

Predictor-inversion methods instead optimize sequences directly against outputs of a pretrained structure predictor.[14, 15] BindCraft established an experimentally effective AF2-based workflow for protein binder design.[16] BoltzDesign extended all-atom predictor inversion to additional molecular contexts, while Germinal incorporated antibody-specific objectives and sequence guidance.[17, 18] Forward-pass approaches such as Protein Hunter and HalluDesign offer a related strategy by alternating structure hallucination and

*Full author list in Author Contributions.

sequence redesign without backpropagating through the predictor.[19, 20] These developments motivate further evaluation of how all-atom predictors can support effective design across binder formats and target chemistries.

A practical question is whether a shared predictor-inversion framework can accommodate different binder formats and molecular contexts while producing experimentally functional sequences. Addressing this question requires both task-specific constraints, such as a fixed antibody framework or peptide cyclization, and sequence-level controls that complement structural objectives. It also requires evaluation beyond the confidence scores used during optimization.

Here we present TorchCraft, a binder-design framework that optimizes sequence logits through a frozen all-atom structure predictor implemented in TorchFold.[21] All designs reported here, including experimentally validated binders and computational benchmarks, were generated using pretrained AF3 weights. The contribution is a shared optimization procedure with task-specific sequence masks, restraints, priors, and schedules, together with experimental minibinder and VHH demonstrations and computational extensions to cyclic peptides and small-molecule-binding proteins.

2 Results

2.1 TorchCraft as an all-atom hallucination framework

TorchCraft represents the editable binder sequence as a trainable logit matrix and updates it using gradients of design objectives computed through a frozen predictor (Figure 1). Each iteration evaluates the target–binder system and adjusts the sequence to improve predicted structural compatibility, interface formation, and task-specific sequence properties. Optimization progresses from continuous sequence representations to discrete amino-acid assignments, yielding a designed sequence and a corresponding predicted all-atom structure (Figure 2b).

The all-atom representation allows the target context to include protein and nonprotein components. In this study, we apply the framework to minibinders, framework-conditioned VHHs, head-to-tail and disulfide-linked cyclic peptides, and proteins designed around small-molecule ligands. These tasks share the sequence-optimization engine but use different editable positions, loss terms, molecular constraints, and optimization schedules (Supplementary Table S1).

TorchCraft uses the TorchFold implementation to perform sequence optimization with a frozen predictor. Computational optimizations, including conformer-generation parallelism, gradient checkpointing, caching, and fused attention, support repeated evaluation during optimization (Appendix A.3).

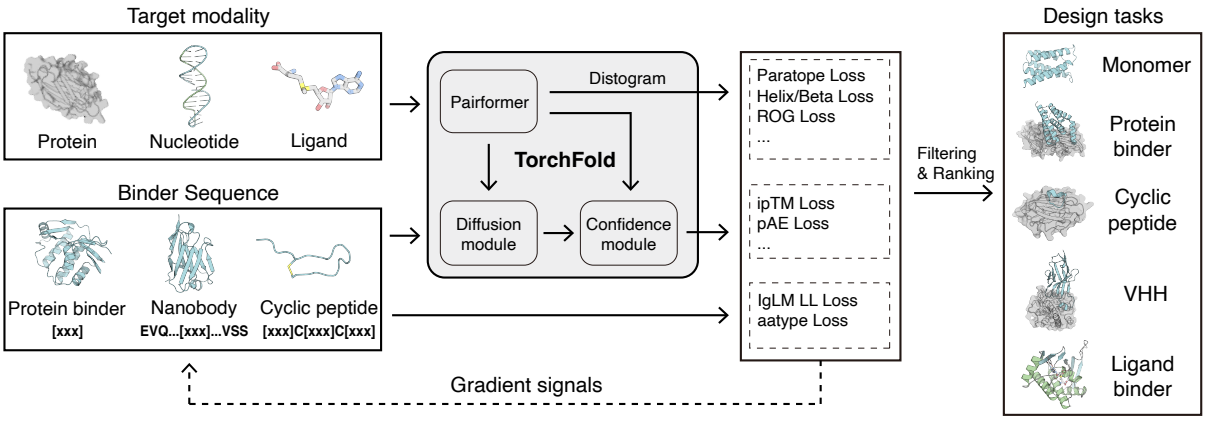

Figure 1 TorchCraft optimizes binder sequences through a frozen all-atom predictor. Target context and editable sequence representations are evaluated using the TorchFold implementation with pretrained AF3 weights, and gradients of task-specific objectives update the sequence logits. Experimental campaigns and computational benchmarks reported here used the same AF3-weight setting. The framework is applied to minibinders, framework-conditioned VHHs, cyclic peptides, and ligand-conditioned proteins by changing the editable positions, molecular constraints, and loss terms.

2.2 De novo design of protein binders against protein targets

We designed minibinders against IFNAR2, IL-7Rα, PDGFRβ, and PD-L1 using target structures, binder-length ranges, and hotspot definitions specified in Supplementary Table S2. Candidates were selected from 3,000 generated designs per target using the filters and ranking procedure in Appendix A.2.2.1. Multi-concentration BLI measurements identified binders in all four campaigns, with representative apparent dissociation constants of 60.2, 8.55, 2.04, and 23.3 nM, respectively (Figure 2a). The experimentally tested sequences were obtained directly from TorchCraft without post hoc sequence redesign. Predicted complexes of the analyzed designs displayed diverse binding configurations and differed from interactions identified in the PDB comparison (Figure 2c,d).

Structural objectives do not directly constrain all sequence features relevant to expression. We therefore added an amino-acid-composition regularizer that matches the average residue distribution to a reference estimated from ProteinMPNN-redesigned samples. The regularizer altered the composition of TorchCraft sequences (Figure 2e) and increased ESMFold confidence across the 12-target benchmark (Supplementary Figure S1a). In two PDGFRβ minibinder campaigns, the round incorporating composition regularization showed higher expression than the preceding unregularized round (Supplementary Figure S1b).

We next compared TorchCraft with BoltzGen and RFdiffusion across 12 protein targets and binder lengths of 75, 120, 160, and 200 residues. Designs were evaluated using TorchScore, a score-only AF3 evaluation of supplied coordinates (Appendix A.4.2). Computational pass rates varied by target and binder length (Figure 2f). Raw TorchCraft designs were competitive on several targets, while optional ProteinMPNN redesign of noninterface residues further improved pass rates in many settings. Because the evaluation uses predictor-derived scores, we additionally report Rosetta interface-score distributions (Supplementary Figure S2).

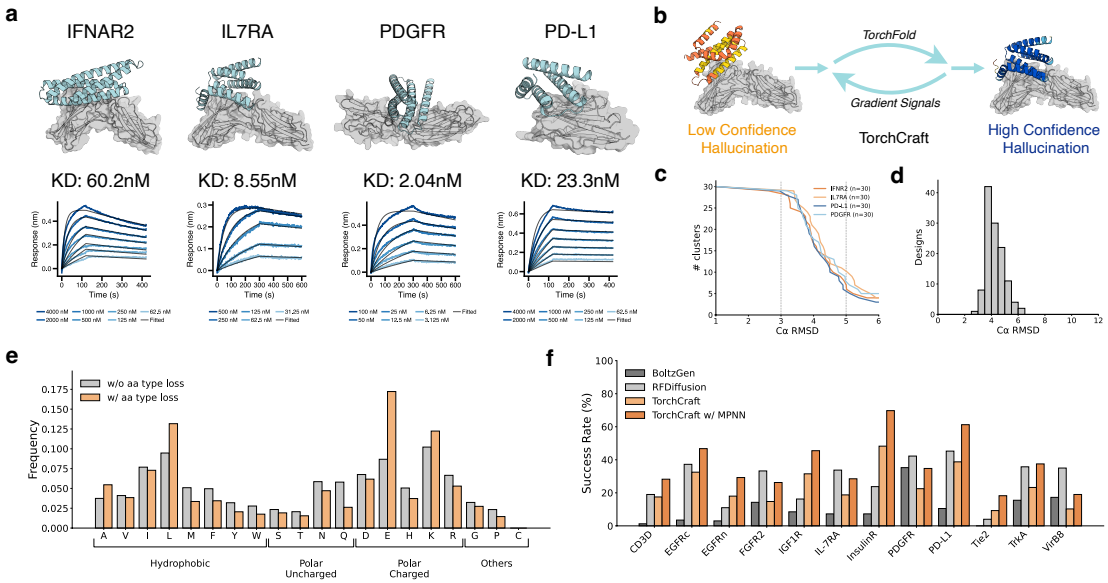

Figure 2 Experimental minibinder binding and computational evaluation of TorchCraft designs. (a) Predicted complexes and BLI measurements for representative minibinders against IFNAR2, IL-7Rα, PDGFRβ, and PD-L1. Apparent dissociation constants of the displayed binders were 60.2, 8.55, 2.04, and 23.3 nM, respectively. (b) Schematic of sequence optimization, showing conversion of a continuous sequence representation into a discrete designed sequence and predicted complex through iterative predictor evaluation and sequence updates. (c,d) Diversity and divergence of predicted binding configurations for the analyzed designs, quantified using target-aligned binder Cα RMSD clustering and comparison with interactions identified around similar target chains in the PDB. (e) Amino-acid composition of TorchCraft minibinders optimized with or without the composition regularizer, which matches the mean residue distribution to a ProteinMPNN-derived reference. (f) Computational pass rates across 12 protein targets, aggregated over binder lengths of 75, 120, 160, and 200 residues. TorchCraft, BoltzGen, and RFdiffusion designs were evaluated with TorchScore, which scores supplied coordinates using the same pretrained AF3 weights used for design and does not refold sequences.

2.3 De novo VHH design by framework-conditioned CDR optimization

We next applied TorchCraft to VHH design, holding a predefined framework sequence fixed while optimizing the three CDRs. This confines sequence search to the paratope-forming regions while retaining the selected scaffold context. Designs were generated against BHRF1, EFNA5, PDGFRβ, and S100A4 using the task specifications in Supplementary Table S3. BLI measurements identified binders in all four campaigns, with representative apparent dissociation constants of 24.3, 122, 268, and 354 nM, respectively (Figure 3a). These sequences were tested without post hoc sequence redesign. The corresponding predicted complexes were analyzed for diversity and divergence from PDB binding configurations (Figure 3g,h).

Unregularized VHH optimization produced CDR sequences with residue preferences that differed from the SAbDab reference set. To incorporate antibody-sequence constraints during design, we added an IgLM-derived objective evaluated separately for CDR1, CDR2, and CDR3.[22] For each CDR, the framework and the current sequences of the other CDRs provide context for the local infilling objective (Figure 3b; Algorithm 3). This concentrates sequence guidance on the editable regions. We compared this approach with the full-sequence guidance used in the Germinal implementation evaluated here.[18]

CDR-restricted IgLM guidance increased IgLM likelihoods and shifted position-specific residue preferences toward those observed in the SAbDab reference set, particularly in CDR1 and CDR2 (Figure 3c,d). Guided sequences also showed changes in TNP developability-associated profiles and higher OASis humanness scores (Figure 3e,f).[23, 24] These analyses indicate improved agreement with selected sequence-based reference distributions.

We compared TorchCraft with RFantibody, BoltzGen, and Germinal across 15 targets and five VHH frameworks

(Figure 3i; Supplementary Figure S5).[13, 18, 25] Designs were assessed using the joint TorchScore-confidence and ESM2-perplexity criterion described in Appendix A.4. TorchCraft achieved higher computational pass rates on many targets, with additional gains from optional AbMPNN redesign in several settings. Rosetta interface-score distributions provide an additional assessment of the predicted complexes (Supplementary Figure S3).

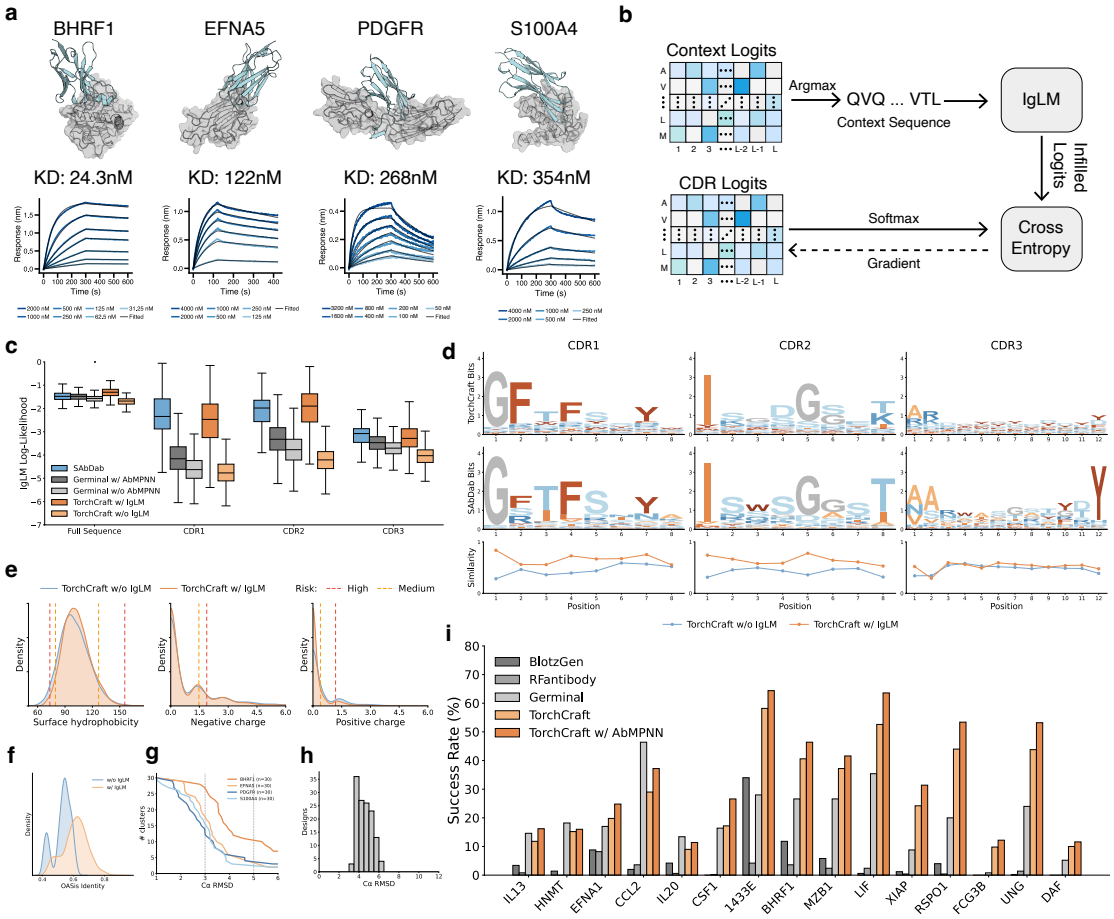

Figure 3 Framework-conditioned VHH design with CDR-specific sequence guidance. (a) Predicted complexes and BLI measurements for representative VHHs against BHRF1, EFNA5, PDGFRβ, and S100A4. Apparent dissociation constants of the displayed binders were 24.3, 122, 268, and 354 nM, respectively. (b) Schematic of CDR-restricted IgLM guidance: for each CDR, the framework and the remaining CDRs provide infilling context, and the IgLM-derived objective is applied only to the editable loop. (c) IgLM log-likelihood distributions for natural SAbDab VHHs, Germinal-generated sequences with or without CDR redesign, and TorchCraft sequences with or without CDR-restricted IgLM regularization. (d) Information-content sequence logos for HCDR1, HCDR2, and HCDR3, shown together with position-wise similarity to the SAbDab reference set. (e) Therapeutic Nanobody Profiler (TNP) comparison of developability-associated profiles for sequences generated with or without IgLM guidance.[23] (f) BioPhi/OASis humanness-score comparison of guided and unguided sequences.[24] (g,h) Diversity and divergence of predicted VHH binding configurations, quantified using target-aligned binder Cα RMSD clustering and comparison with PDB interactions. (i) Computational pass rates for VHH design across 15 targets, comparing TorchCraft with RFantibody, BoltzGen, and Germinal under a joint TorchScore-confidence and ESM2-perplexity criterion. TorchScore scores supplied coordinates using the same pretrained AF3 weights used for design and does not refold sequences. A framework-stratified analysis is shown in Supplementary Figure S5.

2.4 In silico benchmark and analysis of cyclic peptide design

We next evaluated cyclic-peptide design computationally using two parameterizations (Figure 4a). Head-to-tail designs used cyclic relative positional encoding to represent sequence adjacency across the termini. Disulfide-linked designs retained linear positional encoding and fixed the cysteine positions defining the intended linkage. Predicted structures were then classified according to whether they satisfied the intended cyclization and backbone-integrity criteria (Supplementary Figure S6). Both settings used contact-based objectives together with the additional terms specified in Supplementary Table S1.

We compared cyclic-peptide designs for eight target systems with outputs from RFpeptides and BoltzGen using the structural and confidence criteria defined in Appendix A.4 (Figure 4c). Performance depended on both the target and cyclization mode. Disulfide-linked TorchCraft designs showed higher computational pass rates in several settings, including PD-L1 and GABARAP, whereas head-to-tail designs displayed a different target-dependent profile. Optional ProteinMPNN redesign improved pass rates in several comparisons. Representative predicted complexes and Rosetta interface-score distributions illustrate the resulting interfaces (Figure 4b; Supplementary Figure S7).

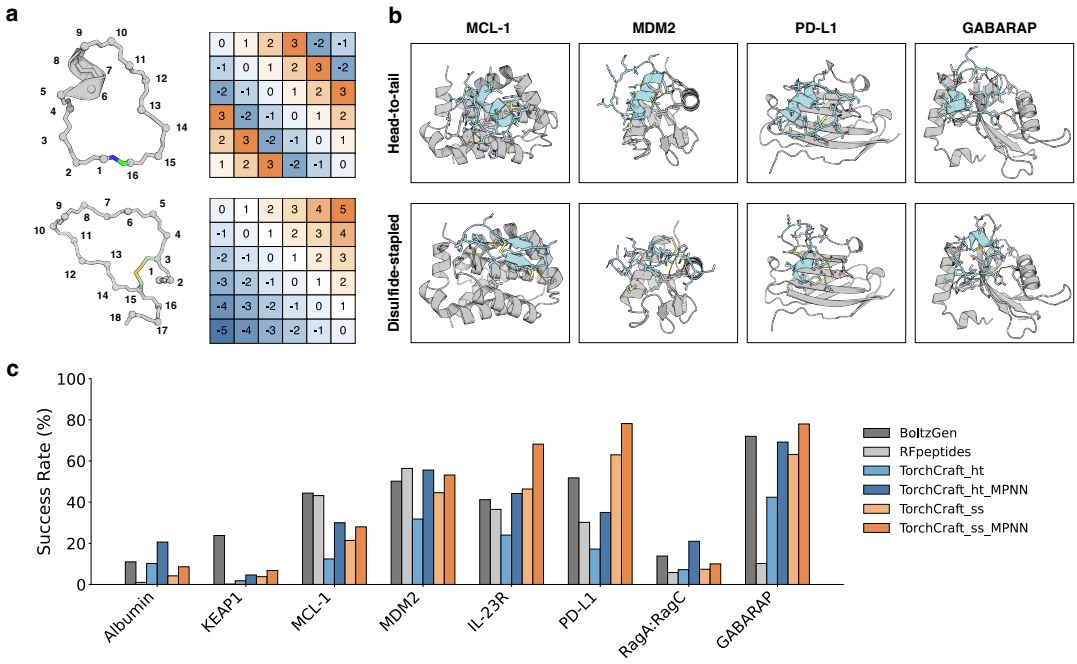

Figure 4 Computational evaluation of head-to-tail and disulfide-linked cyclic-peptide designs. (a) Schematic comparison of the two parameterizations: head-to-tail designs use cyclic relative positional encoding to represent sequence adjacency across the termini, whereas disulfide-linked designs retain linear positional encoding and fix the cysteine positions that define the intended linkage. (b) Representative predicted complexes for MCL-1, MDM2, PD-L1, and GABARAP, shown for both cyclization modes. (c) Computational pass rates across eight target systems: MCL-1, MDM2, PD-L1, IL-23R, albumin, KEAP1, GABARAP, and RagA:RagC. BoltzGen, RFpeptides, and TorchCraft designs were compared using the structural and confidence criteria defined in Appendix A.4.

2.5 Computational design of ligand-binding protein pockets

Designing proteins around small-molecule ligands requires compatible folding, pocket geometry, and local chemical interactions. We therefore combined ligand-contact objectives with controls on binder compactness and secondary-structure composition. An isotropy regularizer was used to bias the overall binder shape, while helix and β-structure terms modulated secondary-structure preferences (Figure 5a,b). These objectives were intended to support pocket formation without specifying a pre-existing scaffold.

Small-molecule design began with a ligand-free warm-up stage that optimized the binder before ligand-dependent objectives were introduced. The ligand was then added, and the weight of the interface objective was gradually increased before the final optimization stages (Appendix A.1.3). This schedule separates initial fold formation from subsequent pocket optimization.

We evaluated a panel of ten small-molecule targets spanning different chemical structures, including nucleotides, cofactors, metabolites, and drug-like compounds (Supplementary Figure S8).[17, 20, 26, 27] TorchCraft generated 500 independent 160-residue proteins per ligand without a supplied native scaffold. Raw outputs were evaluated without post hoc sequence redesign or score-based preselection. An optional LigandMPNN-redesigned setting was analyzed separately. Representative predicted pockets are shown in Figure 5e.

We evaluated the predicted ligand-binding proteins using AutoDock Vina scores and Boltz-2 affinity predictions (Figure 5c,d),[6, 28, 29] with TorchScore pass rates reported separately (Supplementary Figure S9). Score distributions varied across ligands and methods. These complementary computational assessments support the feasibility of generating ligand-conditioned pockets.

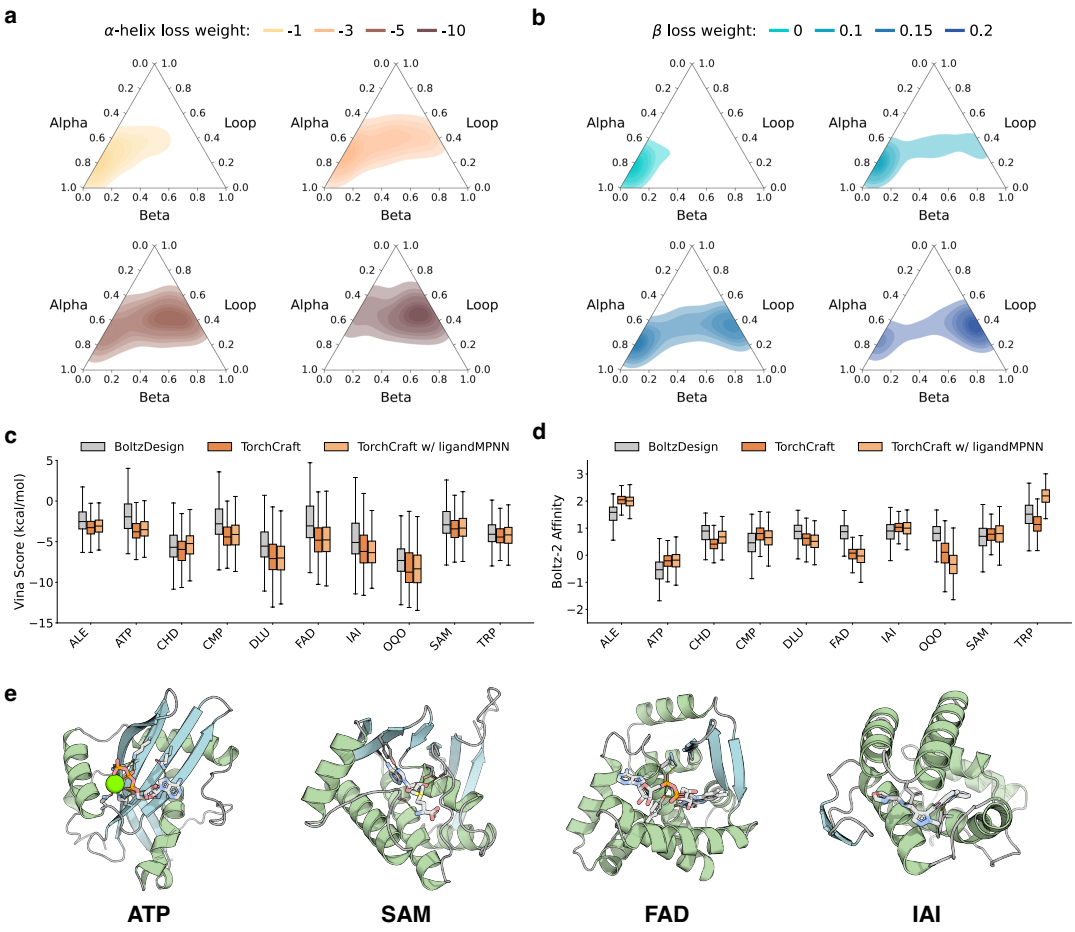

Figure 5 Computational generation and evaluation of ligand-conditioned protein pockets. (a,b) Helix and β-structure terms modulate the secondary-structure composition of designed binders. Ternary kernel density plots show compositions under different loss-weighting schemes. (c) AutoDock Vina docking scores (kcal mol^{-1}) for designed pockets across the 10-ligand panel. (d) Boltz-2 affinity predictions for the same designs. Score distributions vary by ligand and method. (e) Representative predicted 160-residue binders for four ligands: ATP, SAM, FAD, and IAI. Designs were generated without a supplied native scaffold. Raw outputs were evaluated without post hoc sequence redesign. An optional LigandMPNN-redesigned setting is analyzed separately.

3 Discussion

TorchCraft extends the predictor-inversion approach to binder design within an all-atom molecular representation. Building on the principle established by BindCraft and related methods,[16, 17] it uses a common optimization engine with task-specific constraints for minibinders, VHHs, cyclic peptides, and ligand-conditioned proteins. Experiments using pretrained AF3 weights established binding by designed minibinders and VHHs without post hoc sequence redesign. These results show that direct sequence optimization can produce experimentally functional binders in distinct protein formats, while computational studies support extension to additional molecular contexts.

A practical advantage of TorchCraft is that design tasks can be specified through editable residues, molecular context, restraints, and objective functions within the same framework. This allows new constraints to be evaluated without training a separate generator for every task. The present results support this approach across several design settings, although each requires appropriate objectives, sequence priors, and validation. Optional sequence redesign remains useful in some applications, even though it was not required for the experimentally tested binders reported here.

The current study has several limitations. Experimental validation is confined to minibinders and VHHs, whereas cyclic-peptide and small-molecule results remain computational. Predictor-derived pass rates are sensitive to the evaluator and should not be interpreted as experimental success rates. The functional consequences of sequence regularization also require matched experimental comparisons, and binding measurements do not by themselves establish the accuracy of the predicted interfaces. Further validation across targets, formats, and independent assays will clarify the framework's practical range.

A further methodological direction is to incorporate objectives defined directly on atomic interactions. Stable differentiation through coordinate generation could enable more explicit control of ligand-contact geometry, buried polar networks, and coordination sites. Such objectives would complement the confidence- and distogram-based signals used here, but their value must be assessed through both geometric validation and experimental function.

TorchFold's reported improvements in antibody–antigen structure prediction provide a promising direction for antibody design.[21] Its specialized checkpoint may offer more informative guidance during CDR optimization. Matched comparisons of experimental binding and expression under the same targets, frameworks, and candidate-selection procedures will be needed to test this potential.

4 Acknowledgments

This work was funded by Changping Laboratory (CPNL project 2026D-03-01).

5 Author Contributions

Yu Liu[1,*], Zhouhanyu Shen[1,*], Zhengyi Li[1], Xikun Huang[1], Jiaqi Liu[1], Shuxian Gao[2], Qilin Yu[1], Xiayan Qin[1], Yucheng Zhang[2], Mingchen Chen[1,†]

[1]Changping Laboratory, Beijing, China

[2]Huawei Technologies Co., Ltd.

[*]Equal contribution

[†]Corresponding author

Y.L. and Z.S. contributed equally to this work. M.C. designed research; Y.L., Z.S., and X.H. performed research and analyzed data; Z.L. and S.G. performed code engineering; J.L. performed computational cyclic-peptide analysis; Q.Y., X.Q., and Y.Z. participated in discussions; and Y.L. wrote the paper.

6 Data Availability

The TorchCraft inference implementation and design protocols are available at https://github.com/Mingchenchen/TorchX. The project page is available at https://torchx-cpl.github.io.

References

[1] A. M. Watkins and P. S. Arora. Structure-based inhibition of protein–protein interactions. European Journal of Medicinal Chemistry, 94:480–488, 2015.

[2] H. Lu, Q. Zhou, J. He, Z. Jiang, C. Peng, R. Tong, and J. Shi. Recent advances in the development of protein–protein interactions modulators: Mechanisms and clinical trials. Sig Transduct Target Ther, 5(1):213, 2020.

[3] H. Khakzad, I. Igashov, A. Schneuing, C. Goverde, M. Bronstein, and B. Correia. A new age in protein design empowered by deep learning. Cell Systems, 14(11):925–939, 2023.

[4] J. Jumper, R. Evans, A. Pritzel, T. Green, M. Figurnov, O. Ronneberger, K. Tunyasuvunakool, R. Bates, A. Žídek, A. Potapenko, A. Bridgland, C. Meyer, S. A. A. Kohl, A. J. Ballard, A. Cowie, B. Romera-Paredes, S. Nikolov, R. Jain, J. Adler, T. Back, S. Petersen, D. Reiman, E. Clancy, M. Zielinski, M. Steinegger, M. Pacholska, T. Berghammer, S. Bodenstein, D. Silver, O. Vinyals, A. W. Senior, K. Kavukcuoglu, P. Kohli, and D. Hassabis. Highly accurate protein structure prediction with AlphaFold. Nature, 596(7873):583–589, 2021.

[5] J. Abramson, J. Adler, J. Dunger, R. Evans, T. Green, A. Pritzel, O. Ronneberger, L. Willmore, A. J. Ballard, J. Bambrick, S. W. Bodenstein, D. A. Evans, C.-C. Hung, M. O'Neill, D. Reiman, K. Tunyasuvunakool, Z. Wu, A. Žemgulytė, E. Arvaniti, C. Beattie, O. Bertolli, A. Bridgland, A. Cherepanov, M. Congreve, A. I. Cowen-Rivers, A. Cowie, M. Figurnov, F. B. Fuchs, H. Gladman, R. Jain, Y. A. Khan, C. M. R. Low, K. Perlin, A. Potapenko, P. Savy, S. Singh, A. Stecula, A. Thillaisundaram, C. Tong, S. Yakneen, E. D. Zhong, M. Zielinski, A. Žídek, V. Bapst, P. Kohli, M. Jaderberg, D. Hassabis, and J. M. Jumper. Accurate structure prediction of biomolecular interactions with AlphaFold 3. Nature, 630(8016):493–500, 2024.

[6] S. Passaro, G. Corso, J. Wohlwend, M. Reveiz, S. Thaler, V. R. Somnath, N. Getz, T. Portnoi, J. Roy, H. Stark, D. Kwabi-Addo, D. Beaini, T. Jaakkola, and R. Barzilay. Boltz-2: Towards Accurate and Efficient Binding Affinity Prediction. bioRxiv, 2025.

[7] ByteDance AML AI4Science Team, X. Chen, Y. Zhang, C. Lu, W. Ma, J. Guan, C. Gong, J. Yang, H. Zhang, K. Zhang, S. Wu, K. Zhou, Y. Yang, Z. Liu, L. Wang, B. Shi, S. Shi, and W. Xiao. Protenix - Advancing Structure Prediction Through a Comprehensive AlphaFold3 Reproduction, 2025.

[8] J. L. Watson, D. Juergens, N. R. Bennett, B. L. Trippe, J. Yim, H. E. Eisenach, W. Ahern, A. J. Borst, R. J. Ragotte, L. F. Milles, B. I. M. Wicky, N. Hanikel, S. J. Pellock, A. Courbet, W. Sheffler, J. Wang, P. Venkatesh, I. Sappington, S. V. Torres, A. Lauko, V. De Bortoli, E. Mathieu, S. Ovchinnikov, R. Barzilay, T. S. Jaakkola, F. DiMaio, M. Baek, and D. Baker. De novo design of protein structure and function with RFdiffusion. Nature, 620(7976):1089–1100, 2023.

[9] W. Ahern, J. Yim, D. Tischer, S. Salike, S. M. Woodbury, D. Kim, I. Kalvet, Y. Kipnis, B. Coventry, H. R. Altae-Tran, M. S. Bauer, R. Barzilay, T. S. Jaakkola, R. Krishna, and D. Baker. Atom-level enzyme active site scaffolding using RFdiffusion2. Nat Methods, 23(1):96–105, 2026.

[10] J. Butcher, R. Krishna, R. Mitra, R. I. Brent, Y. Li, N. Corley, P. T. Kim, J. Funk, S. Mathis, S. Salike, A. Muraishi, H. Eisenach, E. Sehgal, B. Coventry, O. Zhang, B. Qiang, K. Didi, F. DiMaio, and D. Baker. De novo Design of All-atom Biomolecular Interactions with RFdiffusion3.

[11] J. Dauparas, I. Anishchenko, N. Bennett, H. Bai, R. J. Ragotte, L. F. Milles, B. I. M. Wicky, A. Courbet, R. J. De Haas, N. Bethel, P. J. Y. Leung, T. F. Huddy, S. Pellock, D. Tischer, F. Chan, B. Koepnick, H. Nguyen, A. Kang, B. Sankaran, A. K. Bera, N. P. King, and D. Baker. Robust deep learning–based protein sequence design using ProteinMPNN. Science, 378(6615):49–56, 2022.

[12] J. Dauparas, G. R. Lee, R. Pecoraro, L. An, I. Anishchenko, C. Glasscock, and D. Baker. Atomic context-conditioned protein sequence design using LigandMPNN. Nat Methods, 22(4):717–723, 2025.

[13] H. Stark, F. Faltings, M. Choi, Y. Xie, E. Hur, T. O'Donnell, A. Bushuiev, T. Uçar, S. Passaro, W. Mao, M. Reveiz, R. Bushuiev, T. Pluskal, J. Sivic, K. Kreis, A. Vahdat, S. Ray, J. T. Goldstein, A. Savinov, J. A. Hambalek, A. Gupta, D. A. Taquiri-Diaz, Y. Zhang, A. K. Hatstat, A. Arada, N. H. Kim, E. Tackie-Yarboi, D. Boselli, L. Schnaider, C. C. Liu, G.-W. Li, D. Hnisz, D. M. Sabatini, W. F. DeGrado, J. Wohlwend, G. Corso, R. Barzilay, and T. Jaakkola. Toward Universal Binder Design.

[14] I. Anishchenko, S. J. Pellock, T. M. Chidyausiku, T. A. Ramelot, S. Ovchinnikov, J. Hao, K. Bafna, C. Norn,

A. Kang, A. K. Bera, F. DiMaio, L. Carter, C. M. Chow, G. T. Montelione, and D. Baker. De novo protein design by deep network hallucination. Nature, 600(7889):547–552, 2021.

[15] C. A. Goverde, B. Wolf, H. Khakzad, S. Rosset, and B. E. Correia. De novo protein design by inversion of the AlphaFold structure prediction network. Protein Science, 32(6):e4653, 2023.

[16] M. Pacesa, L. Nickel, C. Schellhaas, J. Schmidt, E. Pyatova, L. Kissling, P. Barendse, J. Choudhury, S. Kapoor, A. Alcaraz-Serna, Y. Cho, K. H. Ghamary, L. Vinué, B. J. Yachnin, A. M. Wollacott, S. Buckley, A. H. Westphal, S. Lindhoud, S. Georgeon, C. A. Goverde, G. N. Hatzopoulos, P. Gönczy, Y. D. Muller, G. Schwank, D. C. Swarts, A. J. Vecchio, B. L. Schneider, S. Ovchinnikov, and B. E. Correia. One-shot design of functional protein binders with BindCraft. Nature, 646(8084):483–492, 2025.

[17] Y. Cho, M. Pacesa, Z. Zhang, B. E. Correia, and S. Ovchinnikov. Boltzdesign1: Inverting All-Atom Structure Prediction Model for Generalized Biomolecular Binder Design. bioRxiv, 2025.

[18] L. S. Mille-Fragoso, J. N. Wang, C. L. Driscoll, H. Dai, T. Widatalla, X. Zhang, B. L. Hie, and X. J. Gao. Efficient generation of epitope-targeted de novo antibodies with Germinal.

[19] Y. Cho, G. Rangel, G. Bhardwaj, and S. Ovchinnikov. Protein Hunter: Exploiting structure hallucination within diffusion for protein design.

[20] M. Fang, C. Wang, J. Shi, F. Lian, Q. Jin, Z. Wang, Z. Cui, Y. Wang, Y. Ke, Q. Han, and L. Cao. HalluDesign: Protein Optimization and de Novo Design via Iterative Structure Hallucination and Sequence Design.

[21] Y. Liu, D. Wang, Z. Li, S. Gao, H. Wang, Z. Shen, Z. Ma, Y. Zhang, and M. Chen. TorchFold: A PyTorch and NPU-compatible reimplementation of AlphaFold3, 2026.

[22] R. W. Shuai, J. A. Ruffolo, and J. J. Gray. IgLM: Infilling language modeling for antibody sequence design. Cell Systems, 14(11):979–989.e4, 2023.

[23] G. L. Gordon, J. Gervasio, C. Souders, and C. M. Deane. The Therapeutic Nanobody Profiler: Characterising and predicting nanobody developability to improve therapeutic design.

[24] D. Prihoda, J. Maamary, A. Waight, V. Juan, L. Fayadat-Dilman, D. Svozil, and D. A. Bitton. BioPhi: A platform for antibody design, humanization, and humanness evaluation based on natural antibody repertoires and deep learning. mAbs, 14(1):2020203, 2022.

[25] N. R. Bennett, J. L. Watson, R. J. Ragotte, A. J. Borst, D. L. See, C. Weidle, R. Biswas, Y. Yu, E. L. Shrock, R. Ault, P. J. Y. Leung, B. Huang, I. Goreshnik, J. Tam, K. D. Carr, B. Singer, C. Criswell, B. I. M. Wicky, D. Vafeados, M. G. Sanchez, H. M. Kim, S. Vázquez Torres, S. Chan, S. M. Sun, T. Spear, Y. Sun, K. O'Reilly, J. M. Maris, N. G. Sgourakis, R. A. Melnyk, C. C. Liu, and D. Baker. Atomically accurate de novo design of antibodies with RFdiffusion, 2024.

[26] K. Didi, D. Reidenbach, M. Penner, S. Ravichandran, M. Nichols, E. Swanson, A. Reis, M. Prescott, Y. Qian, D. Qian, J. Yang, W. Li, L. Li, D. Shonai, S. Gay, B. B. Mallik, H. Yeung, L. Chen, M. A. Juantay, H. Klein, A. U. Macintyre, D. Granata, Z. Cao, G. Zhou, T. Geffner, X. Chen, Z. Zhang, T. Zhang, K. Gion, M. M. Bronstein, K. Deibler, S. Soderling, A. Khmelinskaia, F. Hollfelder, C. Dallago, E. Kucukbenli, A. Vahdat, P. Ogden, and K. Kreis. Latent Generative Search unlocks de novo Design of Untapped Biomolecular Interactions at Scale.

[27] R. Krishna, J. Wang, W. Ahern, P. Sturmfels, P. Venkatesh, I. Kalvet, G. R. Lee, F. S. Morey-Burrows, I. Anishchenko, I. R. Humphreys, R. McHugh, D. Vafeados, X. Li, G. A. Sutherland, A. Hitchcock, C. N. Hunter, A. Kang, E. Brackenbrough, A. K. Bera, M. Baek, F. DiMaio, and D. Baker. Generalized biomolecular modeling and design with RoseTTAFold All-Atom. Science, 384(6693):eadl2528, 2024.

[28] O. Trott and A. J. Olson. AutoDock Vina: Improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading. J Comput Chem, 31(2):455–461, 2010.

[29] J. Eberhardt, D. Santos-Martins, A. F. Tillack, and S. Forli. AutoDock Vina 1.2.0: New Docking Methods, Expanded Force Field, and Python Bindings. J. Chem. Inf. Model., 61(8):3891–3898, 2021.

Appendix

Contents

A Methods

A.1 TorchCraft design protocol

A.1.1 General optimization framework

TorchCraft design trajectories are configured from a target molecular context, an editable binder sequence, and optional restraints. Target structures are supplied as templates, with crops and hotspot definitions recorded for each task. Editable positions may span the complete binder or selected residues within a fixed sequence framework. Task configurations also specify excluded residue types, secondary-structure objectives, and sequence priors. This setup allows the same optimization engine to be used across minibinders, VHHs, cyclic peptides, and ligand-binding proteins while changing the editable positions, loss terms, molecular constraints, and optimization schedules.

The editable sequence is parameterized by a logit matrix $\mathbf{L}$ with one row per editable position and one column per standard amino acid. At each step, the stage-specific sequence representation is passed through a frozen TorchFold predictor, and the weighted design objective is differentiated with respect to $\mathbf{L}$. Only the editable sequence variables are updated. Active objectives and their weights are specified separately for each task in Supplementary Table S1. All designs reported in this study used pretrained AF3 weights. The final sequence is obtained by discrete decoding, followed by optional sequence redesign only in the explicitly designated benchmark settings.

A.1.2 Four-stage optimization

TorchCraft uses a configurable sequence-optimization schedule adapted from prior predictor-inversion methods.[16, 17] The schedule can include a soft warm-up, a logits-to-probabilities transition, temperature annealing, and straight-through discrete optimization. Task-specific subsets of these stages are used, as listed in Supplementary Table S1. The sequence to be optimized is parameterized as a trainable logit matrix $\mathbf{L}$, which is then converted into the model input sequence representation $\mathbf{X}$ according to the optimization stage. Given $\mathbf{X}$, the frozen predictor produces the structural state $\hat{\mathcal{S}}$, and gradients of the design objective $\mathcal{L}(\hat{\mathcal{S}}, \mathbf{X})$ are backpropagated to update the sequence logits. Algorithm 1 summarizes the conversion from sequence logits to predictor inputs and the resulting gradient update.

Soft warm-up (Stage 0). During the optional soft warm-up, sequence logits are converted to probabilities to provide normalized inputs before subsequent optimization:

$$\mathbf{X} = \mathrm{Softmax}(\mathbf{L}).$$

The target context and active losses during warm-up are task-dependent, as specified in Appendix A.1.3.

Logits-to-soft transition (Stage 1). A linearly scheduled mixing coefficient a is introduced to interpolate between the raw logit representation and the softmax-normalized probabilistic representation:

$$\mathbf{X} = (1 - a)\mathbf{L} + a\,\mathrm{Softmax}(\mathbf{L}),$$

where a is gradually increased over the course of this stage. This transitions the input from the raw logit representation toward a normalized sequence representation.

Temperature annealing (Stage 2). A temperature parameter τ is used to control the sharpness of the softmax distribution:

$$\mathbf{X} = \mathrm{Softmax}(\mathbf{L}/\tau).$$

Given an initial temperature τ_0, τ is annealed according to

$$\tau = \max\left(\tau_{\min},\ \tau_0 \cdot \left(\frac{\tau_{\min}}{\tau_0}\right)^{t/T}\right).$$

As the temperature decreases, the sequence distribution sharpens from a smooth probability distribution toward a near one-hot representation. Here t is the step within the annealing stage and T is the number

of steps in that stage. The temperature decreases from τ_0 to $\tau_{\min}$, sharpening each position's amino-acid distribution.

Straight-through hard optimization (Stage 3). The sequence representation is explicitly converted into a one-hot form. Gradients are still propagated through the underlying softmax probabilities. Low-temperature softmax probabilities are computed as

$$\mathbf{P} = \mathrm{Softmax}(\mathbf{L}/\tau_{\min}),$$

and the amino acid with the highest probability is selected to form the hard one-hot representation $\mathbf{H} = \mathrm{OneHot}(\arg\max \mathbf{P})$. The final sequence representation used as model input is defined as

$$\mathbf{X} = (\mathbf{H} - \mathbf{P})_{\mathrm{stopgrad}} + \mathbf{P}.$$

The forward pass uses the one-hot sequence $\mathbf{H}$, while the backward pass uses the derivative of $\mathbf{P}$ with respect to the logits. This straight-through estimator permits additional sequence updates under a discrete forward representation.

Final decoding. After the final stage, each editable position is assigned its highest-logit amino acid. Optional sequence-redesign outputs are retained as a separate method setting.

Algorithm 1: Four-Stage Optimization

Input: Initial sequence logits $\mathbf{L}$, frozen predictor f_θ
Design objective $\mathcal{L}$, Stages $\{s_0, s_1, s_2, s_3\}$
Output: Optimized discrete sequence $S^\star$

for optimization step t **do**
  `// Stage 0: soft warm-up`
  if $t \in s_0$ **then**
    $\mathbf{X} \leftarrow \mathrm{Softmax}(\mathbf{L})$
  `// Stage 1: logits-to-soft transition`
  else if $t \in s_1$ **then**
    $a \leftarrow \mathrm{LinearSchedule}(t)$
    $\mathbf{P} \leftarrow \mathrm{Softmax}(\mathbf{L})$
    $\mathbf{X} \leftarrow (1-a)\mathbf{L} + a\mathbf{P}$
  `// Stage 2: temperature annealing`
  else if $t \in s_2$ **then**
    $\tau \leftarrow \mathrm{AnnealTemperature}(t)$
    $\mathbf{X} \leftarrow \mathrm{Softmax}(\mathbf{L}/\tau)$
  `// Stage 3: straight-through hard optimization`
  else if $t \in s_3$ **then**
    $\mathbf{P} \leftarrow \mathrm{Softmax}(\mathbf{L}/\tau_{\min})$
    $\mathbf{H} \leftarrow \mathrm{OneHot}(\arg\max \mathbf{P})$
    $\mathbf{X} \leftarrow (\mathbf{H} - \mathbf{P})_{\mathrm{stopgrad}} + \mathbf{P}$
  $\hat{\mathcal{S}} \leftarrow f_\theta(\mathbf{X})$
  $\mathbf{L} \leftarrow \mathrm{UpdateLogits}(\mathbf{L}, \nabla_{\mathbf{L}}\mathcal{L}(\hat{\mathcal{S}}, \mathbf{X}))$
$S^\star \leftarrow \arg\max \mathrm{Softmax}(\mathbf{L})$
return $S^\star$

A.1.3 Target-specific stage configurations

Minibinder and VHH runs used temperature annealing followed by straight-through optimization. Cyclic-peptide runs used all four stages. Stage lengths and learning rates are listed in Supplementary Table S1.

Small-molecule design used an initial ligand-free warm-up followed by gradual introduction of the interface objective. After this warm-up, the ligand was added and the complex was jointly optimized. We write the time-dependent objective as

$$\mathcal{L}(t) = \mathcal{L}_{\text{scaffold}} + w_{\text{interface}}(t) \cdot \mathcal{L}_{\text{interface}},$$

where $w_{\text{interface}}(t)$ increases from zero to its final value during the specified transition stage and remains fixed thereafter. The molecular inputs, active losses, and stage lengths are reported in Supplementary Table S1.

A.1.4 Design losses

Confidence losses. Loss terms based on the following confidence scores are constructed:

pLDDT loss. pLDDT (predicted Local Distance Difference Test) measures local structural confidence. This loss increases predicted local confidence of binder residues or atoms.

pTM loss. pTM (predicted TM-score) measures predicted global confidence. This loss increases predicted global confidence for the selected structure.

ipTM loss. ipTM (interface predicted TM-score) evaluates predicted confidence in the relative placement of different subunits. This loss increases predicted interface confidence between chains or molecular components.

PDE loss. PDE (predicted Distance Error) penalizes higher predicted distance error between residue or atom pairs.

PAE loss. PAE (predicted Aligned Error) reflects uncertainty in the relative spatial placement of residue or token pairs. This loss penalizes higher predicted alignment error.

iPAE loss. iPAE (interface predicted Aligned Error) focuses on predicted alignment error across chains or molecular components at the complex interface.

Contact loss. We impose soft contact constraints using the predicted distogram rather than the final inter-residue distances in the predicted structure. For each residue pair (i, j), the distogram logits are used to estimate a probability distribution over distance bins. Given a distance threshold, all bins below the threshold are treated as contact bins, and the contact loss is computed from the corresponding probabilities.

In the optional binary formulation, the loss mainly penalizes insufficient total probability mass within the contact region. Otherwise, the probability distribution is renormalized over the contact bins, and the corresponding cross-entropy term is computed. For each anchor residue, only a subset of partner residues with the lowest losses is retained. The best-performing fraction of anchor residues is then selected and averaged. This aggregation rewards a selected subset of high-probability contacts.

Intra-chain binder contact loss. The intra-chain binder contact loss encourages the designed binder to form a sufficient number of internal residue–residue contacts. This biases the binder toward a more compact fold.

Inter-chain target–binder contact loss. The inter-chain target–binder contact loss encourages the designed binder to form a sufficient number of cross-chain contacts with the target. This strengthens interface formation between the two chains. If hotspot residues are provided, the loss prioritizes contacts between the binder and the specified target sites. If the designable positions on the binder side are further restricted, interface contacts are mainly encouraged between those designable binder residues and the target or hotspot region.

α-Helix loss. Inspired by BoltzDesign,[17] an α-Helix loss is introduced to explicitly encourage, or suppress when assigned a negative weight, the formation of α-helical conformations. It uses contact probabilities derived from the predicted distance distribution as a differentiable signal, and rewards residue pairs i and $i + 3$ within 6 Å, as residues separated by three positions along the sequence are typically in close spatial proximity in α-helices. Formally, the α-Helix loss is equivalent to minimizing the binary contact negative log-likelihood over selected $(i, i + 3)$ pairs:

$$\mathcal{L}_{\alpha} = -\frac{1}{|H|} \sum_{(i,j)\in H} \log(p_{ij}^{\alpha} + \epsilon),$$

where the candidate helical contact set is defined within the binder chain as

$$H = \{(i, j) \mid |i - j| = 3\}.$$

Given the predicted pairwise distance distribution $P_{ij}(d)$, the confidence that a residue pair forms a local contact is defined as the probability mass below the distance threshold of 6 Å:

$$p_{ij}^{\alpha} = \sum_{d<6 \text{ Å}} P_{ij}(d).$$

β-structure loss. We introduced a β-structure loss to modulate the β-structure propensity of the binder, similar to Germinal.[18] Together with the α-helix loss, this provides more flexible control over the secondary-structure composition of designed samples. The β-structure loss consists of a β-strand term and a β-sheet term:

$$\mathcal{L}_{\beta} = \mathcal{L}_{\beta\text{-strand}} + \mathcal{L}_{\beta\text{-sheet}}.$$

For local β-strand geometry, the probability of a residue-pair distance falling within the interval $[9.75, 11.5)$ Å is defined as

$$p_{ij}^{\beta\text{-strand}} = \sum_{d\in[9.75,11.5) \text{ Å}} P_{ij}(d).$$

Let B denote the candidate β-design positions, define their neighboring position set as

$$N = \{i - 1, i + 1 \mid i \in B\},$$

and the local β-strand candidate pairs as

$$S = \{(i, j) \mid i, j \in N,\ j = i + 3\}.$$

Among these candidates, we select the top-K pairs with the largest per-pair losses, focusing optimization on the local β-strand constraints that are currently least satisfied, where

$$K = \left\lfloor \frac{|B|}{3} \right\rfloor.$$

The β-strand loss is then given by

$$\mathcal{L}_{\beta\text{-strand}} = -\frac{1}{K} \sum_{(i,j)\in S_{\text{top-}K}} \log(p_{ij}^{\beta\text{-strand}} + \epsilon).$$

For β-sheet formation, a shorter contact-distance interval, $[4.4, 6.0)$ Å, is used to characterize spatial contacts between two β-strands:

$$p_{ij}^{\beta\text{-sheet}} = \sum_{d\in[4.4,6.0) \text{ Å}} P_{ij}(d).$$

The β-sheet term enumerates a set of antiparallel β-sheet pairing patterns. For each candidate β-region R, we construct multiple reverse-diagonal pairing masks, each corresponding to a possible strand–strand pairing configuration. We denote the resulting candidate pairing sets as

$$A_R = \{A_1, A_2, \cdots, A_m\},$$

where each A is a set of residue pairs, $A \subseteq \{(i, j) \mid i, j \in \text{binder}\}$. For a given candidate pairing pattern A, the average β-sheet contact negative log-likelihood is defined as

$$l(A) = -\frac{1}{|A|} \sum_{(i,j)\in A} \log(p_{ij}^{\beta\text{-sheet}} + \epsilon).$$

For each β-region, we take the maximum loss over all candidate pairing patterns:

$$l_{\beta\text{-sheet}}(R) = \max_{A \in A_R} l(A).$$

The overall β-sheet loss is therefore

$$\mathcal{L}_{\beta\text{-sheet}} = \sum_R \max_{A \in A_R} \left[-\frac{1}{|A|} \sum_{(i,j)\in A} \log(p_{ij}^{\beta\text{-sheet}} + \epsilon) \right].$$

Full details are provided in Algorithm 2.

Algorithm 2: Beta Loss

Input: Predicted distance distribution $P_{ij}(d)$; candidate β design positions B; candidate β regions $\mathcal{R}$; numerical constant ϵ

Output: β structural loss $\mathcal{L}_\beta$

// β-strand loss

$\mathcal{L}_{\beta\text{-strand}} \leftarrow 0$

$N \leftarrow \{i-1, i+1 \mid i \in B\}$

$S \leftarrow \{(i,j) \mid i,j \in N,\ j = i+3\}$

$K \leftarrow \lfloor |B|/3 \rfloor$

for $(i,j) \in S$ **do**

  $p_{ij}^{\beta\text{-strand}} \leftarrow \sum_{d\in[9.75,11.5)\,\text{Å}} P_{ij}(d)$

  $\ell_{ij}^{\beta\text{-strand}} \leftarrow -\log\left(p_{ij}^{\beta\text{-strand}} + \epsilon\right)$

$S_{\text{top-}K} \leftarrow \text{TopK}\left(S,\ \ell_{ij}^{\beta\text{-strand}},\ K\right)$

$\mathcal{L}_{\beta\text{-strand}} \leftarrow -\frac{1}{K} \sum_{(i,j)\in S_{\text{top-}K}} \log\left(p_{ij}^{\beta\text{-strand}} + \epsilon\right)$

// β-sheet loss

$\mathcal{L}_{\beta\text{-sheet}} \leftarrow 0$

for $R \in \mathcal{R}$ **do**

  Construct antiparallel pairing masks $A_R = \{A_1, A_2, \cdots, A_m\}$ for R

  for $A \in A_R$ **do**

    for $(i,j) \in A$ **do**

      $p_{ij}^{\beta\text{-sheet}} \leftarrow \sum_{d\in[4.4,6.0)\,\text{Å}} P_{ij}(d)$

    $l(A) \leftarrow -\frac{1}{|A|} \sum_{(i,j)\in A} \log\left(p_{ij}^{\beta\text{-sheet}} + \epsilon\right)$

  $l_{\beta\text{-sheet}}(R) \leftarrow \max_{A\in A_R} l(A)$

  $\mathcal{L}_{\beta\text{-sheet}} \leftarrow \mathcal{L}_{\beta\text{-sheet}} + l_{\beta\text{-sheet}}(R)$

$\mathcal{L}_\beta \leftarrow \mathcal{L}_{\beta\text{-strand}} + \mathcal{L}_{\beta\text{-sheet}}$

return $\mathcal{L}_\beta$

Paratope loss. Inspired by Germinal,[18] a distogram-based paratope contact loss is introduced to restrict the designed binding interface to predefined CDR or paratope regions. The goal is to strengthen predicted contacts between CDR residues and the target epitope or hotspot residues, while discouraging framework–target contacts.

We compute three contact losses: the CDR–hotspot contact loss $\mathcal{L}_{\text{CDR-Hotspot}}$, the CDR–interface contact loss $\mathcal{L}_{\text{CDR-Interface}}$, and the framework–target contact loss $\mathcal{L}_{\text{framework}}$. Since a smaller contact loss indicates stronger predicted contact, minimizing $\mathcal{L}_{\text{CDR-Hotspot}}$ and $\mathcal{L}_{\text{CDR-Interface}}$ jointly promotes interface formation through the CDR regions. In contrast, contacts between the framework and the target should remain weak. We therefore place $\mathcal{L}_{\text{framework}}$ in the denominator as a suppression term:

$$\mathcal{L}_{\text{paratope}} = \frac{\mathcal{L}_{\text{CDR-Hotspot}} \cdot \mathcal{L}_{\text{CDR-Interface}}}{\text{ReLU}\left(\mathcal{L}_{\text{framework}} - \lambda\right) + \epsilon}.$$

The denominator is lower-bounded by ϵ when $\mathcal{L}_{\text{framework}} - \lambda$ is negative or close to zero.

Distogram-based compactness loss. To penalize overly extended conformations during sequence optimization, we constructed a differentiable radius-of-gyration (ROG) proxy using the predicted distogram probability distribution. For a binder chain containing N residues, the squared radius of gyration of an equally weighted point cloud can be expressed as the average over all pairwise Euclidean distances:

$$R_g^2 = \frac{1}{2N^2} \sum_i \sum_j \|\mathbf{r}_i - \mathbf{r}_j\|^2,$$

where $\mathbf{r}_i$ denotes the spatial position of residue i. For each residue pair (i, j), let b_k denote the k-th distogram bin, c_k its representative distance, taken as the bin center, and $p_{ij}(b_k)$ the corresponding predicted probability. The expected squared distance for this residue pair is approximated as

$$\mathbb{E}[d_{ij}^2] = \sum_k p_{ij}(b_k) c_k^2.$$

A distogram-based approximation of R_g^2 is then obtained by averaging the expected squared distances over all residue pairs in the binder:

$$R_g^2 = \frac{1}{2N^2} \sum_i \sum_j \mathbb{E}[d_{ij}^2].$$

We further define the radius of gyration as

$$R_g = \sqrt{R_g^2 + \epsilon}.$$

To impose a length-dependent compactness upper bound, we define the target radius of gyration as

$$R_{g,\text{target}} = \alpha N^\beta + \gamma.$$

The final ROG regularization term is defined as

$$\mathcal{L}_{\text{rog}} = \text{ReLU}\left(R_g - R_{g,\text{target}}\right).$$

Isotropy loss. A radius-of-gyration penalty controls overall size but does not uniquely specify shape. Elongated rods, thin disks, and spherical conformations can exhibit similar values of R_g. To bias the binder toward more similar extents along its principal axes, we additionally defined an isotropy objective from a distogram-derived distance matrix. From the distogram, we first compute the expected squared distance for each residue pair and assemble the intra-binder squared distance matrix:

$$\mathbf{D}_{ij}^{(2)} = \mathbb{E}[d_{ij}^2].$$

The matrix is then symmetrized as

$$\mathbf{D}^{(2)} \leftarrow \frac{1}{2}\left(\mathbf{D}^{(2)} + \mathbf{D}^{(2)T}\right),$$

and its diagonal entries are set to zero:

$$D_{ii}^{(2)} = 0.$$

We next recover a centered Gram matrix from the squared distance matrix using the standard double-centering formula:

$$\mathbf{G} = -\frac{1}{2}\mathbf{J}\mathbf{D}^{(2)}\mathbf{J},$$

where

$$\mathbf{J} = \mathbf{I} - \frac{1}{N}\mathbf{1}\mathbf{1}^T.$$

When $\mathbf{D}^{(2)}$ is derived from real three-dimensional coordinates, $\mathbf{G}$ is equivalent to the inner-product matrix of the centered coordinate matrix, $\mathbf{G} = \mathbf{X}\mathbf{X}^T$. The eigenvalues of $\mathbf{G}$ reflect the variance scale of the point cloud

along its principal axes. We take the three largest non-negative eigenvalues, $\lambda_1 \geq \lambda_2 \geq \lambda_3$, which correspond to the extent of the structure along the three principal axes, and penalize deviations among them:

$$\mathcal{L}_{\text{isotro}} = \frac{(\lambda_1 - \lambda_2)^2 + (\lambda_2 - \lambda_3)^2 + (\lambda_1 - \lambda_3)^2}{(\lambda_1 + \lambda_2 + \lambda_3)^2 + \epsilon}.$$

For a near-spherical conformation, the three principal extents should be comparable, such that $\lambda_1 \approx \lambda_2 \approx \lambda_3$.

Amino-acid composition loss. An aa-type loss regularizes the global residue distribution of the optimized chain. A reference composition is estimated from ProteinMPNN-redesigned samples, and the current sequence probabilities are averaged over editable positions and matched to this reference by cross-entropy. This term constrains global residue usage and is evaluated separately from position-specific structural and language-model objectives. Let $\mathcal{E}$ denote the editable residue positions, and let $\mathbf{P} = \text{Softmax}(\mathbf{L})$ be the amino-acid probability matrix obtained from the current sequence logits. The empirical amino-acid composition of the optimized region is

$$\bar{p}_a = \frac{1}{|\mathcal{E}|} \sum_{i \in \mathcal{E}} P_{i,a}, \qquad a \in \mathcal{A},$$

where $\mathcal{A}$ is the 20-amino-acid alphabet. Given a ProteinMPNN-derived reference composition q_a, the amino-acid composition loss is

$$\mathcal{L}_{\text{AA}} = -\sum_{a \in \mathcal{A}} q_a \log\left(\bar{p}_a + \epsilon\right).$$

CDR-restricted IgLM loss. In nanobody CDR design, the CDR sequences to be optimized are represented as a continuous logit matrix $\mathbf{L}_{\text{CDR}}$. Inspired by Germinal,[18] we adopt a decoupled dual-path strategy to incorporate antibody language-model priors while preserving differentiable sequence optimization.[22]

The first path constructs the discrete context for IgLM. A pseudo sequence is obtained by taking the arg max of the current logits; equivalently, of a temperature-scaled softmax of those logits. For fixed logits, this discrete context does not depend on the softmax temperature. For each CDR, we then construct an infilling input independently: the CDR region being evaluated is replaced with mask tokens, while the framework regions and the remaining CDRs are provided as context to the frozen IgLM. Under this local context, IgLM predicts the conditional amino acid distribution at each masked position, yielding a context-dependent target distribution $\hat{\mathbf{P}}_{\text{IgLM}}$.

The second path is used for differentiable gradient propagation. The original logits are directly passed through a softmax with temperature 1 to obtain the continuous predicted distribution $\mathbf{P}_{\text{pred}}$. The local conditional distribution predicted by IgLM is then used as the teacher distribution, and a cross-entropy loss is computed over the CDR region. Because the loss is backpropagated only through the continuous $\mathbf{P}_{\text{pred}}$ path, gradients can act stably on the original logits, while the discrete pseudo sequence is used solely to provide context for IgLM. This avoids the gradient interruption introduced by arg max discretization, while still using the antibody sequence prior learned by IgLM to guide CDR optimization.

We evaluated the sequence-prior objective separately over the three editable CDRs, allowing their contributions to be weighted independently. This differs from a full-sequence objective in both the positions scored and the conditioning context (Algorithm 3).

Algorithm 3: CDR-restricted IgLM Loss

Input: Sequence logits $\mathbf{L}_{\text{CDR}}$
Fixed framework sequence S_{fw}
CDR regions $\mathcal{R} = \{R_1, \cdots, R_m\}$
CDR weights $\{w_i\}_{i=1}^{m}$
Output: IgLM infilling loss $\mathcal{L}_{\text{IgLM}}$

// Path 1: Discrete target for IgLM Context
$\tau_{step} \leftarrow \text{ExponentialDecay}(\tau_{start}, \tau_{min}, \text{step})$
$\mathbf{P}_{hard} \leftarrow \text{Softmax}(\mathbf{L}_{\text{CDR}}/\tau_{step})$
$S_{pseudo} \leftarrow \text{Argmax}(\mathbf{P}_{hard})$ // Pseudo-one-hot sequence
// Path 2: Differentiable prediction for gradient flow
$\mathbf{P}_{\text{pred}} \leftarrow \text{Softmax}(\mathbf{L}_{\text{CDR}})$
$S \leftarrow$ combine S_{fw} and S_{pseudo}
$\mathcal{L}_{\text{IgLM}} \leftarrow 0$
for each CDR region $R_i \in \mathcal{R}$ **do**
  // Define Context for R_i using other discrete CDRs
  $Context_i \leftarrow \{S_{\text{fw}}\} \cup \{S_{pseudo}[R_j] \text{ for } j \neq i\}$
  // Construct infilling prompt with R_i masked
  $\hat{\mathbf{P}}_{\text{IgLM},R_i} \leftarrow f_\theta^{\text{IgLM}}([\texttt{MASK}] \mid Context_i)$
  $\mathcal{L}_i \leftarrow \text{CrossEntropy}\left(\hat{\mathbf{P}}_{\text{pred},R_i}, \mathbf{P}_{\text{iglm},R_i}\right)$
  $\mathcal{L}_{\text{IgLM}} \leftarrow \mathcal{L}_{\text{IgLM}} + w_i \mathcal{L}_i$
return $\mathcal{L}_{\text{IgLM}}$ // Update via Path 2

A.2 Experimental-candidate design protocol

A.2.1 Design task specification

Minibinder design. Experimental minibinders were designed against the four protein targets reported in Figure 2a using pretrained AF3 weights through TorchFold. For each target, TorchCraft generated 3,000 candidate molecules using the target crops, hotspot definitions, and binder-length ranges listed in Table S2. The design task allowed the full binder sequence to be optimized while keeping the target template fixed. The candidates selected for experimental validation were raw TorchCraft outputs and were not subjected to post hoc MPNN sequence redesign.

VHH design. Experimental VHHs were designed against the four protein targets reported in Figure 3a using pretrained AF3 weights through TorchFold. The scaffold framework, CDR length ranges, target crops, and hotspot definitions followed the settings in Table S3. For VHH designs, TorchCraft optimized the CDR regions under the fixed framework context, with the interface objective restricted toward CDR-driven target recognition. The candidates selected for experimental validation were raw TorchCraft outputs under this framework-conditioned setting, without post hoc MPNN sequence redesign.

A.2.2 Design experiments

A.2.2.1 Minibinder candidate generation and selection For each minibinder target, 3,000 TorchCraft candidates were sampled. After generation, candidates were filtered by radius of gyration (ROG < 17), binder pTM (> 0.7), and TorchCraft ipTM (> 0.7). Passing candidates were then ranked using Rosetta Interface Analyzer. Candidates were selected for experimental validation using the composite score,

$$\text{score}_{\text{composite}} = 100 \times \text{ipTM}_{\text{TorchCraft}} - \text{score}_{\text{Rosetta interface}}.$$

Higher composite scores were prioritized for wet-lab validation.

A.2.2.2 VHH candidate generation and selection For each VHH target, 12,500 TorchCraft candidates were generated. Candidates were filtered using CDR interface rate (> 0.6), CDR3 hotspot coverage rate (> 0), and TorchCraft ipTM (> 0.7). Passing candidates were ranked using Rosetta Interface Analyzer with the same composite score used for minibinders,

$$\text{score}_{\text{composite}} = 100 \times \text{ipTM}_{\text{TorchCraft}} - \text{score}_{\text{Rosetta interface}}.$$

This ranking favors designs that combine high TorchCraft interface confidence with favorable Rosetta interface scores, and the top-ranked designs were selected for experimental validation.

A.2.3 Structural novelty and diversity analyses

Structural novelty. We assessed the divergence of predicted binding configurations from interactions identified around similar target chains in the PDB. Target-similar chains were identified by taking the union of a sequence search (`mmseqs easy-search -s 7.5 --max-seqs 1000`) and a structure search (`foldseek easy-search -s 7.5 --alignment-type 1 --max-seqs 1000`) of the target chain against the PDB. For each target-similar chain, all protein chains with one or more heavy-atom contacts (< 5 Å) were taken as its binders. After superposing the design target onto the target-similar chain with CEAlign, structural divergence was measured as the nearest-neighbour binder Cα RMSD; for each design, we report the minimum target-aligned binder Cα RMSD over all interactions with target-similar chains.

Structural diversity. We clustered predicted binding configurations using average-linkage hierarchical clustering of the target-aligned binder Cα RMSD matrix. Using the same structural distance applied between pairs of designs, we computed the pairwise distance matrix D over all designs in a cohort, built a dendrogram by UPGMA average-linkage clustering (`scipy.cluster.hierarchy.linkage(squareform(D), method="average")`), and obtained the number of clusters at a given RMSD threshold by cutting the dendrogram at that linkage threshold (`scipy.cluster.hierarchy.fcluster`, `criterion="distance"`). The number of clusters as a function of the RMSD threshold is reported per cohort.

A.3 Implementation optimizations

A.3.1 Multithreaded parallel conformer generation

CPU multithreading is implemented for residue-level parallel computing. Parallel determinism is supported through pre-extraction of random-state snapshots, and each thread maintains an independent RandomState instance to avoid race conditions during generation of protein reference conformations.

A.3.2 Gradient rematerialization

The memory bottleneck of TorchCraft stems from the large number of activations, complex computational graphs, and high-complexity structure-related intermediate tensors. We used gradient checkpointing to reduce activation memory by recomputing selected intermediate tensors during backpropagation. For TorchCraft, a reentrant gradient-rematerialization implementation was used together with native `torch.utils.checkpoint` for selected modules.

Furthermore, the Template and MSA modules exhibit no significant computational-graph nesting that leads to memory accumulation. For these modules, non-reentrant gradient checkpointing from native `torch.utils.checkpoint` was used.

A.3.3 Caching mechanism

Caching was restricted according to feature dependence. Features that remained unchanged were reused, while sequence-dependent conditioning was recomputed when the sequence representation changed. Within a single diffusion prediction, conditioning could be reused across denoising steps.

Before diffusion begins, the model runs the trunk network, such as Pairformer, once on the current sequence representation and target context, producing `single_cond` and `pair_cond`. These conditioning tensors are reused across denoising steps within the same prediction and refreshed at the next design update.

A.3.4 NPU Fusion Attention

The NPU fusion-attention operator was modified to support diverse input layouts, reducing data movement and transpose operations. Fused attention reduced runtime for several larger sequence settings in the reported NPU benchmark, while its effect depended on sequence length and batch size (Supplementary Table S9).

A.3.5 Other optimizations

Operator substitutions and memory-layout adjustments were used on Ascend NPU, including reduced random memory access and contiguous allocations where possible. On GPU, optional substitution of equivariant kernels from cuequivariance was evaluated. Runtime benchmarks for NPU with and without fused attention, and for GPU with and without cuequivariance, are summarized in Supplementary Table S9.

A.4 Benchmark Settings

A.4.1 Target information

The target structures, crop regions, and hotspot residues used in the wet-lab validation campaigns and computational benchmarks are summarized in Supplementary Tables S2–S6.

A.4.2 TorchScore

TorchScore is a score-only loading of the TorchFold implementation.[21] Given input coordinates, it bypasses the diffusion-based structure module and evaluates the complex through the confidence pathway, returning predictor-derived scores including pLDDT, pTM, and ipTM. No separate scoring network is trained. In this study, TorchScore used pretrained AF3 weights, the same checkpoint used for design. Designed sequences that were rebuilt with FlowPacker were scored from the resulting coordinates; they were not refolded from sequence by TorchScore.

A.4.3 Method setup

A.4.3.1 Minibinder benchmark. For de novo minibinder design, we compared BoltzGen, RFdiffusion, and TorchCraft. For each method, at each benchmark binder length and for each target, 100 designs were first generated using the default settings of the corresponding method. For each BoltzGen and RFdiffusion design, we then redesigned the full binder sequence with ProteinMPNN without fixing any residue, sampling eight redesigned sequences per generated backbone. Each redesigned sequence was rebuilt into an all-atom side-chain model with FlowPacker. The eight rebuilt models were scored with TorchScore, and the redesign with the highest pTM + ipTM was selected as the final structure and score for that original design. For TorchCraft, we evaluated raw and MPNN-polished settings separately. In the raw setting, which corresponds to the default TorchCraft core procedure, outputs were scored directly with TorchScore without sequence redesign. In the MPNN-polished setting, following the BindCraft-style protocol, TorchCraft-designed interface residues were fixed and all remaining positions were redesigned eight times with ProteinMPNN; FlowPacker side-chain rebuilding and TorchScore-based selection were then performed exactly as for BoltzGen and RFdiffusion. A minibinder design was counted as passing the computational criterion when both TorchScore pTM and TorchScore ipTM exceeded 0.7.

A.4.3.2 VHH benchmark. For VHH design, we compared BoltzGen, RFantibody, Germinal, and TorchCraft. For each method, at each benchmark framework and for each target, 100 designs were first generated using the default settings of the corresponding method. For BoltzGen and RFantibody, the full designed sequence was redesigned eight times with AbMPNN without fixing any residue. For Germinal, we followed the original pipeline by fixing CDR residues that contacted the target interface and applying AbMPNN redesign only to residues outside the CDR regions. Each redesigned sequence was rebuilt with FlowPacker to recover an all-atom side-chain structure. The eight rebuilt models were scored with TorchScore, and the redesign with the highest ipTM was selected as the final structure and score for that original design. For TorchCraft, we again evaluated raw and MPNN-polished settings separately. Raw TorchCraft outputs were scored directly with TorchScore without sequence redesign. In the MPNN-polished setting, interface

CDR residues designed by TorchCraft were fixed and all other positions were redesigned eight times with AbMPNN. The subsequent FlowPacker rebuilding and TorchScore-based selection were identical to those used for BoltzGen, RFantibody, and Germinal. A VHH design passed the computational criterion only when it satisfied both a structural-confidence criterion and an expression proxy: TorchScore ipTM had to exceed 0.7, and ESM2-150M perplexity had to be below 6.31. The perplexity threshold was obtained from a retrospective nanobody-expression analysis, in which public expression measurements were combined with accumulated TorchCraft expression data. Designs with total expression of at least 0.08 mg were classified as high-expression cases, ESM2-150M perplexity was swept as a one-dimensional classifier, and the operating threshold was selected by maximizing the Matthews correlation coefficient.

A.4.3.3 Cyclic-peptide benchmark. For cyclic-peptide design, we compared BoltzGen, RFpeptides, and TorchCraft. In the head-to-tail cyclization benchmark, each method generated 100 designs per target using its default settings. For BoltzGen and RFpeptides, the full peptide sequence of each generated structure was redesigned eight times with ProteinMPNN without fixing any residue. Each redesigned sequence was rebuilt with FlowPacker, scored with TorchScore, and the redesign with the highest pLDDT + 100 × ipTM was selected as the final structure and score for that original design. For head-to-tail TorchCraft outputs, we evaluated raw designs scored directly with TorchScore and an optional ProteinMPNN-polished variant in which all peptide positions were redesigned eight times; the subsequent FlowPacker and TorchScore selection procedure was identical to that used for BoltzGen and RFpeptides. TorchCraft was also evaluated in a disulfide-linked cyclization setting. For this setting, raw outputs were scored directly, or all positions except the two cysteines were redesigned eight times with ProteinMPNN, followed by the same FlowPacker rebuilding and TorchScore selection protocol used for the head-to-tail setting. A cyclic-peptide design passed the computational criterion only if the predicted structure formed the intended cycle without chain breaks and satisfied TorchScore pLDDT > 70 and ipTM > 0.5.

A.4.3.4 Small-molecule binder benchmark. For small-molecule binder design, we compared BoltzDesign and TorchCraft. For each method and each target, 500 designs were generated using the default settings of the corresponding method. BoltzDesign and the raw TorchCraft setting were evaluated without sequence redesign. The score distributions of the resulting designs were assessed with AutoDock Vina and Boltz-2. For TorchCraft, we additionally reported a TorchScore-based computational pass rate, where a design passed the computational criterion when TorchScore pLDDT exceeded 70 and TorchScore ipTM exceeded 0.7. We also tested a TorchCraft redesign variant in which ligand-interface residues were fixed, all remaining positions were redesigned eight times with LigandMPNN, and the redesigned sequences were refolded with TorchFold. The refolded structure with the highest pTM was selected as the redesigned structure and was then evaluated with AutoDock Vina and Boltz-2.

A.5 Experimental validation methods

A.5.1 Binder experimental methods

The following procedures were used for the wet-lab validation of binder designs reported in Figure 2a.

A.5.1.1 Gene synthesis and subcloning The DNA sequences encoding the designed minibinders were codon-optimized and synthesized by GenScript Biotechnology Co., Ltd. (Nanjing, China). The synthesized gene was subcloned into the pIVEX vector for protein expression in a cell-free system. Subcloning into the pIVEX vector yielded the following product: MSG-[protein]-GSGSHHWGSTHHHHHH, containing a C-terminal SNAC cleavage tag and a 6×His affinity tag.

A.5.1.2 Cell-free protein expression Cell-free protein synthesis reactions were assembled by combining S30 cell lysate, synthesis buffer, required enzymes, and plasmid DNA in a 24-deep-well plate. Each reaction had a final volume of 5 mL and was incubated at 25 °C for 16 hours. The reaction mixture was then centrifuged at 4000 rpm for 5 minutes and the supernatant was collected for purification.

A.5.1.3 Cell-free protein purification and analysis Designed minibinders were obtained by Ni-column purification using Hamilton instruments (Hamilton, MICRO LAB Starlet). The purified protein was then dialyzed in the desired buffer (either PBS, pH 7.4 or 50 mM Tris, 150 mM NaCl, 10% glycerol, pH 8.0). Before aliquoting and storage, the protein was filtered through a 0.22 μm filter. Protein concentration was determined by the A280 protein assay using bovine serum albumin (BSA) as the standard. Protein purity was verified by standard SDS-PAGE electrophoresis. For SDS-PAGE analysis: samples were mixed with 5× reducing loading buffer (300 mM Tris-HCl pH 6.8, 10% SDS, 30% glycerol, 0.5% bromophenol blue, and 250 mM DTT). Proteins were separated on a 12% homogeneous SDS-PAGE gel (GenScript, Cat. No. M00668) and visualized accordingly.

A.5.1.4 Target protein preparation The four minibinder campaigns used the Fc-fusion target proteins listed in Table S7. His-tagged designed binders were immobilized for BLI measurements.

A.5.1.5 Binding characterization of binders Initial screening of binder designs for target protein binding was performed using Biolayer Interferometry (BLI, Gator Bio). For target proteins without the His-tag (IFNAR2, PD-L1, PDGFRβ, and IL-7Rα), His-tagged binders were prepared at 5 μg/mL in PBSTA buffer (10 mM $Na_2HPO_4{\cdot}12H_2O$, 2 mM KH_2PO_4, 137 mM NaCl, 2.7 mM KCl, 0.05% Tween-20, 0.1% BSA, pH 7.4) for loading. Target proteins were diluted to 1000 nM in PBSTA buffer. Anti-His sensors (Gator Bio, Cat. No. 20-5066) were saturated with His-tagged binders. The BLI assay sequence was as follows: 60 s baseline, 120 s ligand loading, 60 s post-loading baseline, 120 s association phase, 300 s dissociation phase, and 30 s regeneration. Baseline-subtracted signals (in nanometers) were calculated to prioritize binders for further characterization.

To determine apparent affinities of selected binder designs: His-tagged binders (5 μg/mL in PBSTA) were immobilized onto Anti-His sensors for 120 s. Serial dilutions of target proteins were prepared as follows: IFNAR2 (4000 nM to 62.5 nM or 125 nM to 1.95 nM), PDGFRβ (100 nM to 3.125 nM), PD-L1 (4000 nM to 62.5 nM), and IL-7Rα (500 nM to 31.25 nM). Diluted target proteins were allowed to associate with the immobilized ligand for 120 s, followed by dissociation in PBSTA buffer for 300 s. After background subtraction using buffer-only (PBSTA) control curves, apparent dissociation constants (K_D) were determined using the 1:1 binding model in the Results & Analysis curve-fitting module.

A.5.2 VHH experimental methods

The following procedures were used for the wet-lab validation of VHH designs reported in Figure 3a.

A.5.2.1 Framework selection VHH frameworks were randomly selected from 7EOW (caplacizumab), 8Z8V (ozoralizumab), 5JDS (envafolimab), 7XL0 (vobarilizumab), and 8COH (gefurulimab).

A.5.2.2 Gene synthesis and subcloning VHH DNA sequences were codon-optimized and synthesized by GenScript Biotechnology Co., Ltd. (Nanjing, China). The synthesized genes were subcloned into the pCDNA3.4 vector for protein expression in a CHO cell system. Subcloning into the pCDNA3.4 vector yielded the following product: MGWSCIILFLVATATGVHS (signal peptide)-[Nanobody]-GGGGSHHHHHH, with a C-terminal 6×His affinity tag.

A.5.2.3 VHH expression in CHO cells Transfection-grade plasmids were prepared in large quantities using a GenScript self-developed plasmid extraction kit for CHO cell expression. Cells were maintained at 37 °C with 5% CO_2 on an orbital shaker. One day before transfection, the cells were seeded at an appropriate density. On the day of transfection, DNA and Reagent (self-developed by GenScript) were mixed at an optimal ratio and then added into cells ready for transfection. Approximately 24 hours post transfection, Feed was added to each sample.

A.5.2.4 Protein purification and analysis Cell culture broth was centrifuged and followed by filtration. Filtered cell culture supernatant was loaded onto an affinity purification column at an appropriate flowrate.

After washing and elution with appropriate buffers, the eluted fractions were pooled and buffer exchanged to the final formulation buffer (PBS, pH 7.2). The purified protein was analyzed by SDS-PAGE and SEC-HPLC analysis to determine the molecular weight and purity. The concentration was determined by A280 method.

A.5.2.5 Target protein preparation EFNA5, PDGFRβ, and S100A4 used the Fc-fusion constructs listed in Table S8. For BHRF1, commercially available proteins lacking alternative tags were unavailable. Therefore, BHRF1 was expressed with a His-SUMO tag in HEK293 cells, followed by tag cleavage.

A.5.2.6 Binding characterization of VHHs Initial screening of VHH designs for target protein binding was performed using BLI (Gator Bio). For target proteins without His-tag (PDGFRβ, EFNA5, BHRF1, and S100A4), His-tagged VHHs were prepared at 5 μg/mL in PBSTA buffer. Target proteins were diluted to 1000 nM in PBSTA buffer. Anti-His sensors (Gator Bio, Cat. No. 20-5066) were saturated with His-tagged VHHs. The BLI assay sequence was as follows: 60 s baseline, 120 s ligand loading, 60 s post-loading baseline, 120 s association phase, 300 s dissociation phase, and 30 s regeneration. Baseline-subtracted signals (in nanometers) were calculated to prioritize VHHs for further characterization.

To determine apparent affinities of selected designs, 5 μg/mL His-tagged design VHHs prepared in PBSTA were immobilized onto Anti-His sensor for 120 s. Serial dilutions of the target proteins (BHRF1: from 2000 nM diluted to 31.25 nM, PDGFRβ: from 3200 nM diluted to 50 nM, S100A4 and EFNA5 from 4000 nM diluted to 125 nM) were then allowed to associate with the immobilized ligand for 120 s, followed by a dissociation step in PBSTA for 300 s. After background subtraction using buffer-only (PBSTA) control curves, apparent K_D values were determined using the 1:1 binding model in the Results & Analysis curve-fitting module.

B Supplementary Tables

Supplementary tables collect the target definitions, sampling settings, benchmark panels, and experimental-validation reagents used in this study.

Table S1 Design tasks and corresponding hyperparameters.

Design task	Loss weight	Stage iterations	Learning rate
Minibinder	Intra-chain binder contact loss = 1 Inter-chain target–binder contact loss = 1 Binder PAE loss = 1 Amino-acid composition loss = 5	Stage 0 = 0 Stage 1 = 0 Stage 2 = 80 Stage 3 = 5	1
VHH	Paratope loss = 1 ipTM loss = 10 Binder–target interface PAE loss = 1 CDR1-restricted IgLM loss = 1.5 CDR2-restricted IgLM loss = 3.5 CDR3-restricted IgLM loss = 5	Stage 0 = 0 Stage 1 = 0 Stage 2 = 80 Stage 3 = 5	0.1
Cyclic peptide	Intra-chain binder contact loss = 1 Inter-chain target–binder contact loss = 1 Binder PAE loss = 1 Amino-acid composition loss = 5	Stage 0 = 30 Stage 1 = 60 Stage 2 = 45 Stage 3 = 5	0.1
Small-molecule binder	Intra-chain binder contact loss = 1 Inter-chain target–binder contact loss = 1 α-Helix loss = 0.01 Beta loss = 0.15 Isotropy loss = 0.3 Amino-acid composition loss = 5	Stage 0 = 30 Stage 1 = 75 Stage 2 = 80 Stage 3 = 5	0.1

* All design tasks were optimized using the same SGD optimizer.

Table S2 Mini-binder experimental-validation sampling settings for the designs reported in Figure 2a.

Design target	PDB ID	Target chain/residues	Hotspots	Length range
IFNAR2	3se4*	C34–232	C46,C48,C76,C80,C98,C100,C133,C138,C186, C189,C190	90–150
IL-7Rα	7opb	B36–231	B50,B76,B81,B97,B103,B155,B158,B162, B183,B187,B211,B212	90–150
PDGFRβ	3mjg	X124–312	X136,X138,X186,X246,X259,X264	90–150
PD-L1	8znl	B20–133	B50,B64,B118,B126	60–130

* For IFNAR2 (PDB: 3se4), missing residues at the protein interface were repaired using homology modeling via SWISS-MODEL.

Table S3 VHH experimental-validation sampling settings for the designs reported in Figure 3a. The length range lists HCDR1, HCDR2, and HCDR3, respectively.

Design target	PDB ID	Target chain/residues	Hotspots	CDR length range
BHRF1	2wh6	A2–158	A65,A72,A81,A88,A107	8,8,9–21
EFNA5	2x11	B32–171	B101,B130,B132	8,8,9–21
PDGFRβ	3mjg	X124–312	X136,X138,X270	8,8,9–21
S100A4	5lpu	C2–90	C45,C58,C78	8,8,9–21

Table S4 Mini-binder benchmark targets. Hotspots are listed separately for each binder-length setting.

Target	PDB ID	Target chain/residues	75-aa hotspots	120-aa hotspots	160/200-aa hotspots
IL-7Rα	7opb	B36–231	B78,B100,B159	B79,B104,B155,B159, B184,B212	B79,B104,B155,B159, B184,B187,B212
InsulinR	4zxb	E6–155	E64,E88,E96	E34,E37,E65,E86,E142, E147	E9,E19,E34,E37,E65,E86, E142,E147
PD-L1	8znl	B20–133	B57,B116,B124	B59,B70,B74,B120,B122, B126	B30,B59,B70,B74,B107, B120,B122,B126
TrkA	1www	X282–382	X294,X296,X333	X287,X304,X306,X334, X339,X347	X287,X291,X294,X296, X304,X306,X327,X329, X334,X339,X347
CD3D	1xiw	B2–72	B6,B17,B38	B4,B13,B17,B29,B32,B41	B4,B9,B13,B17,B29,B32, B41
EGFRc	1mox	A301–500	A384,A436,A468	A384,A406,A411,A418, A463,A468	A318,A357,A384,A406, A411,A418,A463,A468
EGFRn	1mox	A6–162	A45,A47,A127	A16,A20,A93,A101,A128, A153	A16,A20,A30,A93,A101, A128,A148,A153
FGFR2	1djs	A251–362	A289,A347,A350	A285,A289,A311,A317, A352,A354	A285,A289,A311,A317, A352,A354,A356,A360
IGF1R	5u8r	A1–144	A8,A59,A83	A28,A58,A108,A114,A136, A138	A10,A28,A58,A79,A108, A114,A136,A138
PDGFRβ	3mjg	X124–312	X207,X242,X267	X136,X138,X186,X246, X259,X264	X136,X138,X186,X245, X246,X257,X259,X264
Tie2	2gy7	B23–445	B152,B161,B192	B66,B68,B95,B161,B167, B197	B61,B66,B68,B71,B92, B95,B101,B161,B162, B167,B197
VirB8	4o3v	A90–231	A124,A140,A152	A134,A138,A142,A186, A189,A193	A134,A138,A142,A186, A189,A193,A196,A200, A217

Table S5 VHH benchmark targets. All designs used CDR length settings of HCDR1 = 8 aa, HCDR2 = 8 aa, and HCDR3 = 9–21 aa.

Target	PDB ID	Target chain/residues	Hotspots
IL13	1ijz	A1–113	A11,A14,A15,A101,A107,A108
HNMT	1jqd	A5–292	A175,A221,A269,A280
1433E	2br9	A3–232	A176,A199,A219,A230
BHRF1	2wh6	A2–158	A65,A72,A81,A88,A107
EFNA1	3hei	B18–149	B46,B90,B114
CCL2	4dn4	M9–69	M28,M39,M55,M61
IL20	4doh	A24–176	A62,A68,A160,A167,A171
CSF1	4wrm	B2–146	B10,B55,B56,B78,B86
MZB1	7aah	A39–185	A71,A86,A144,A147
LIF	7n0a	C10–181	C14,C20,C127,C135
XIAP	1g3f	A263–343	A307,A323,A324
RSPO1	4bsp	A40–126	A106,A110
FCGR3B	1fnl	A3–175	A88,A90,A113
UNG	2hxm	A82–304	A147,A158,A167
DAF	1h03	P5–129	P26,P47,P50

Table S6 Cyclic-peptide benchmark targets.

Target	PDB ID	Target chain/residues	Hotspots	Peptide length
MDM2	4hfz	A26–108	A54,A58,A61	12–18
MCL-1	2pqk	A172–197,A203–321	A224,A227,A231,A235,A249,A253,A263	12–18
GABARAP	7zkr	A1–115	A48,A50,A51,A52,A62,A65	12–18
PD-L1	5o45	A17–132	A56,A115,A123	12–18
KEAP1	2flu	A325–609	A334,A380,A382,A415,A483,A530	12–18
RagA:RagC	6s6d	A4–303,B59–369	B80,B226	12–18
IL-23R	5mzv	C24–C123	C67,C100	12–18
albumin	1bj5	A207–A385	A212,A228,A325	12–18

Table S7 Target proteins used for binder experimental validation.

Protein	Company	Description	Cat.	Accession No.	Protein structure
IFN-α/β R2	ACRO	Human IFN-α/β R2 Protein, Fc Tag	IF2-H5255	P48551-2	IFNAR2(Ile27-Lys243)_hFc
PDGFRβ	SinoBiological	Recombinant Human PDGF beta receptor/PDGFRB Protein (ECD), HPLC-verified	10514-H02H1	NP_002600.1	(Met1-Phe530)_hFc
PD-L1	SinoBiological	Recombinant Human PD-L1/B7-H1 Protein (ECD, hFc Tag), HPLC-verified	10084-H02H	NP_054862.1	(Met1-Thr239)_hFc
IL-7Rα	ACRO	Human IL-7Rα/CD127 Protein, Fc Tag	ILA-H525a	P16871-1	IL-7Rα(Glu21-Asp239)_Fc(Pro100-Lys330)

Table S8 Target proteins used for VHH experimental validation.

Protein	Company	Description	Cat.	Accession No.	Protein structure
BHRF1	Oritop	BHRF1	H2025111402	P03182	His-SUMO-(Met1-Ser165)
EFNA5	SinoBiological	Recombinant Human Ephrin A5 Protein (hFc Tag)	10192-H02H	NP_001953.1	(Met1-Asn203)_hFc
PDGFRβ	SinoBiological	Recombinant Human PDGF beta receptor/PDGFRB Protein (ECD), HPLC-verified	10514-H02H1	NP_002600.1	(Met1-Phe530)_hFc
S100A4	SinoBiological	Recombinant Human S100A4 Protein (hFc Tag)	10185-H01H	NP_002952.1	hFc_(Met1-Lys101)

Table S9 TorchCraft runtime benchmark across sequence lengths on NPU (with/without fused attention, FA) and GPU (with/without cuequivariance). Batch size 2 is the recommended setting. Runtime is reported in minutes per design, obtained by dividing the total wall-clock time of a batch-size-2 run by 2. GPU timings are the mean of three independent runs.

Code	Operator	Machine	100 aa	200 aa	300 aa	400 aa	500 aa
NPU	w/ FA	910B (64G)	3.1	3.75	6.23	11.29	16.56
NPU	w/o FA	910B (64G)	3.0	4.13	8.63	14.64	33.47*
GPU	w/ cuequivariance	A100 (80G)	3.92	4.27	4.54	5.85	7.59
GPU	w/o cuequivariance	A100 (80G)	4.76	7.96	13.08	21.65	33.3

*Batch size 1; batch size 2 exceeded memory (OOM).

C Supplementary Figures

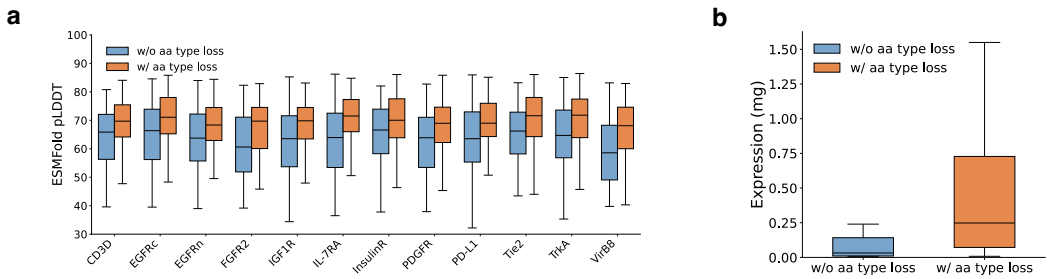

Figure S1 Sequence-composition regularization, predicted confidence, and observed minibinder expression. (a) ESMFold pLDDT comparison for TorchCraft minibinders optimized with or without the composition regularizer across the 12-target benchmark. (b) PDGFRβ minibinder expression in two experimental rounds. The round incorporating composition regularization showed higher expression than the preceding unregularized round.

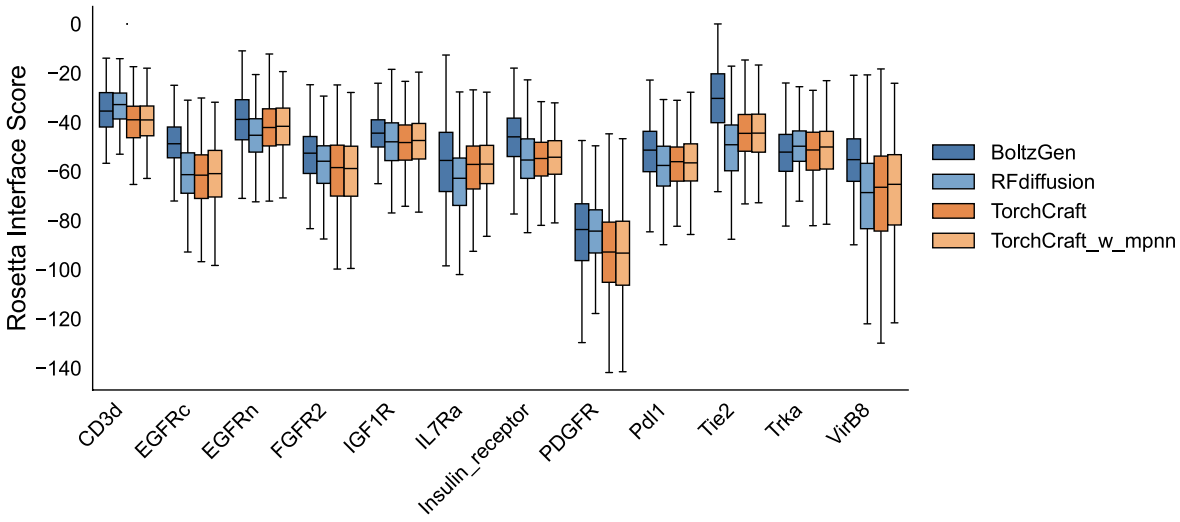

Figure S2 Rosetta interface-score benchmark for de novo minibinders. Distribution of Rosetta interface scores for minibinder benchmark designs.

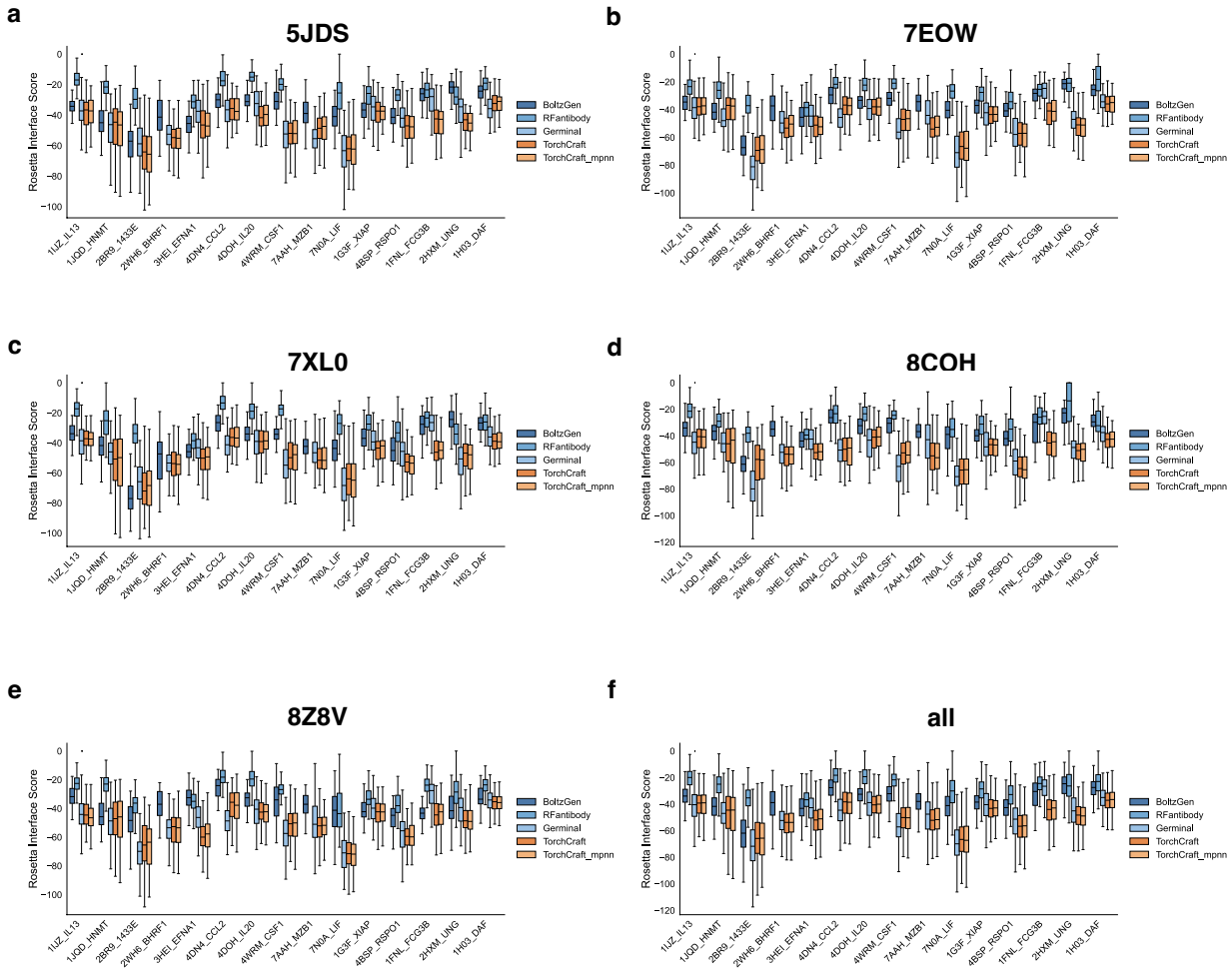

Figure S3 Rosetta interface-score benchmark for framework-conditioned nanobody designs. Distribution of Rosetta interface scores for nanobody benchmark designs. Panels a–e show results for each of the five VHH frameworks, and panel f shows the average across frameworks.

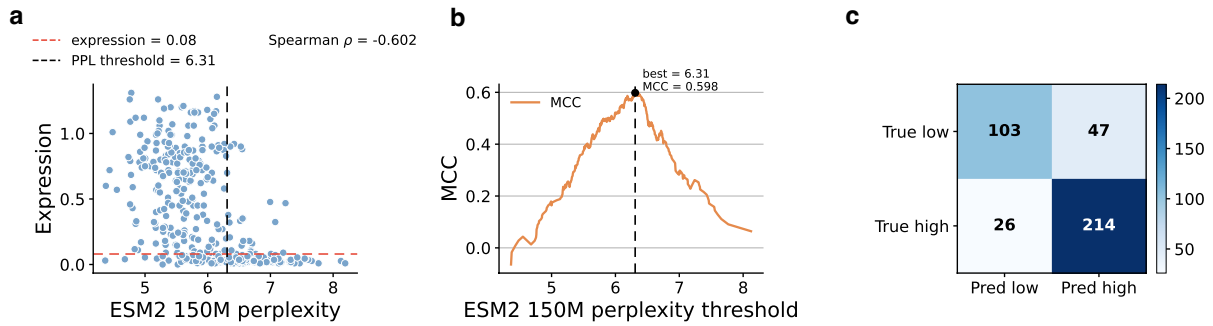

Figure S4 ESM2-150M perplexity retrospectively predicts VHH expression. (a) Retrospective analysis of public expression measurements and accumulated TorchCraft expression data. ESM2-150M perplexity is correlated with expression, with lower perplexity generally associated with higher expression. The horizontal dashed line marks the high-expression cutoff of 0.08 mg. (b) Threshold sweep for classifying high-expression designs by ESM2-150M perplexity. The selected threshold of 6.309 maximizes the Matthews correlation coefficient (MCC = 0.598; Pearson $r = -0.538$; Spearman $\rho = -0.602$), with designs below the threshold classified as high-expression candidates. (c) Confusion matrix obtained using the maximum-MCC perplexity threshold.

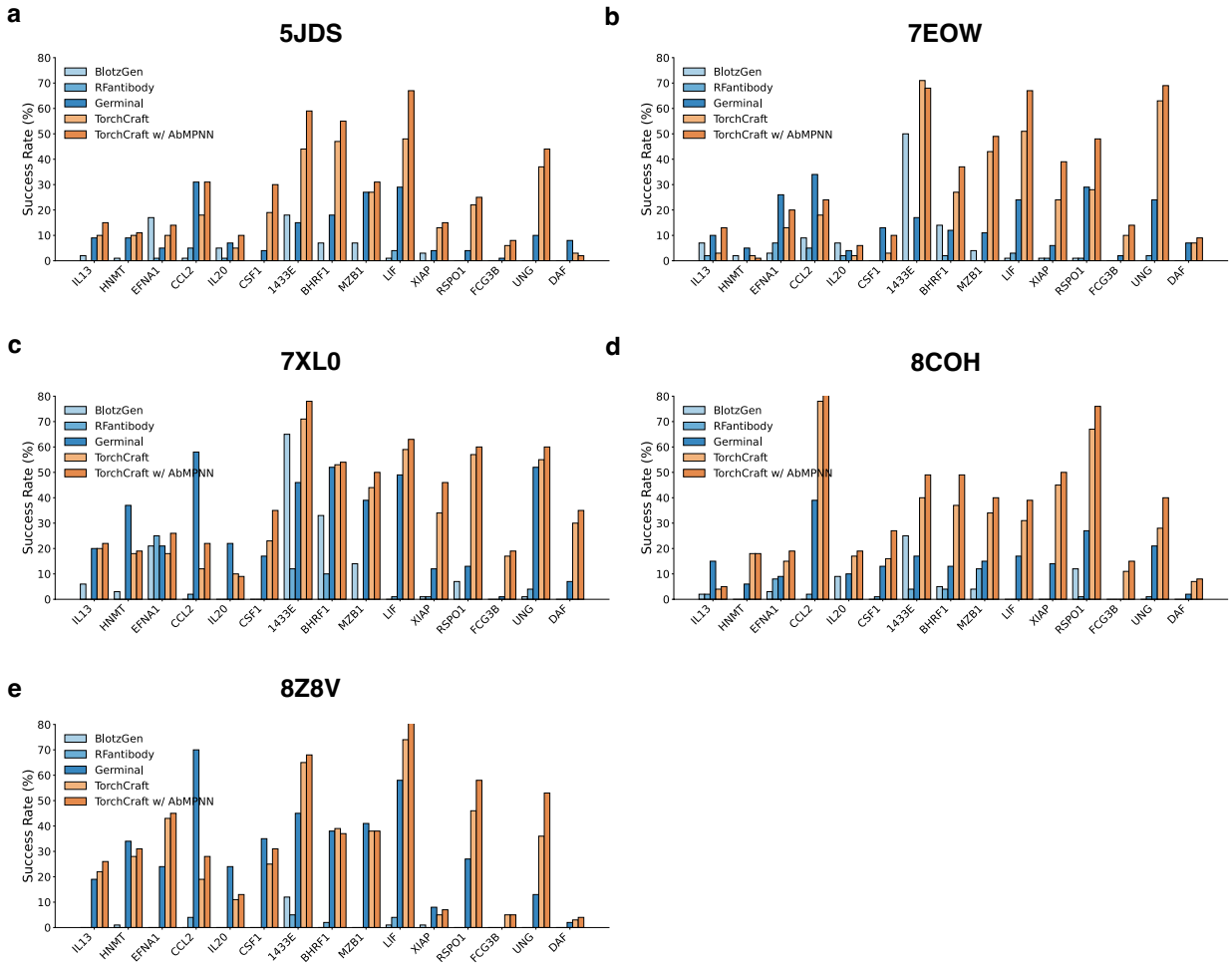

Figure S5 Framework-stratified VHH benchmark across five nanobody scaffolds. TorchScore-based VHH design benchmark split by framework, showing how TorchCraft performance varies across five framework choices while using the same target panel and computational pass-rate criterion.

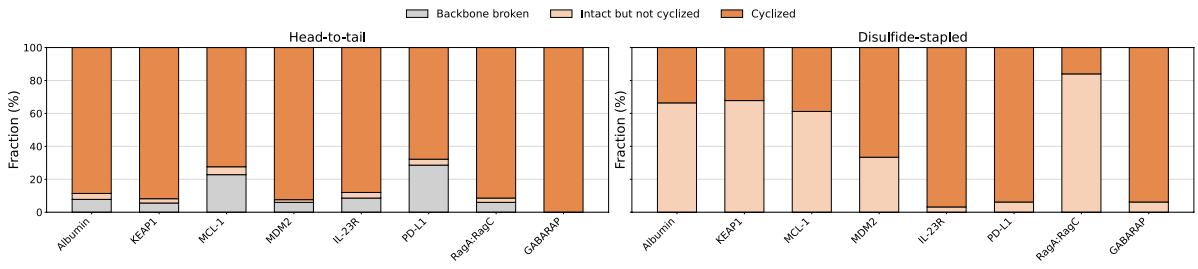

Figure S6 Cyclization outcomes for cyclic-peptide designs. Fractions of successfully cyclized, intact-but-not-cyclized, and backbone-broken predictions across cyclic-peptide benchmark targets for the head-to-tail and disulfide-linked TorchCraft settings.

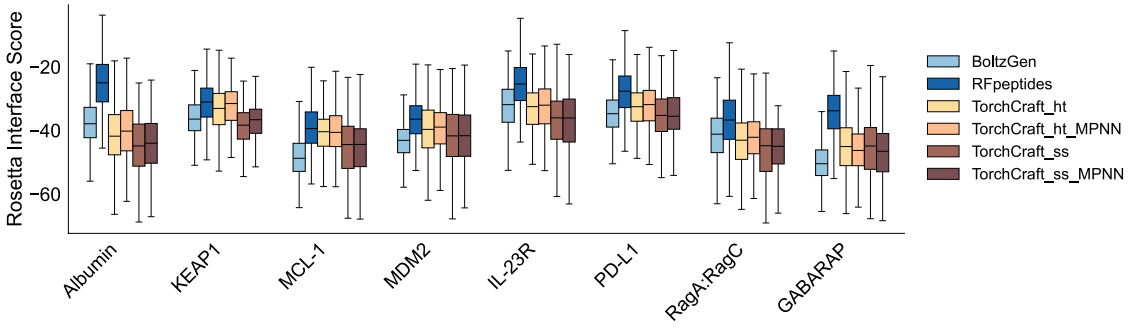

Figure S7 Rosetta interface-score benchmark for cyclic-peptide designs. The cyclic-peptide benchmark from Figure 4c evaluated using Rosetta interface score as an orthogonal assessment of predicted target-binding quality.

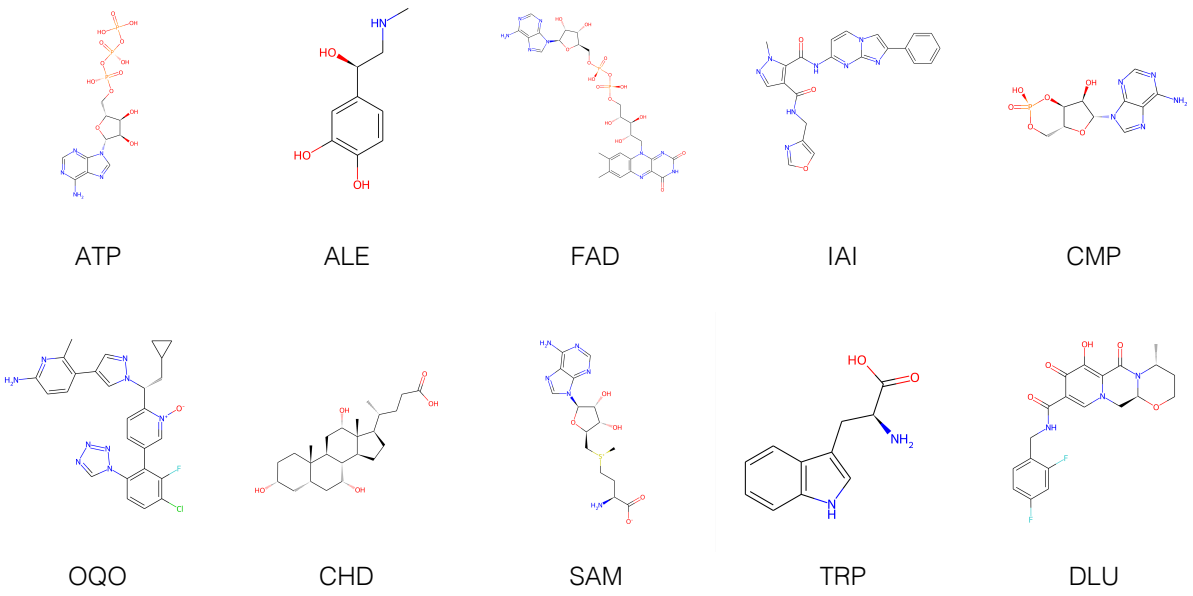

Figure S8 Small-molecule targets used in the SMI benchmark. Benchmark panel of chemically diverse ligand targets used for small-molecule interaction design, including ALE, ATP, CHD, CMP, DLU, FAD, IAI, OQO, SAM, and TRP.

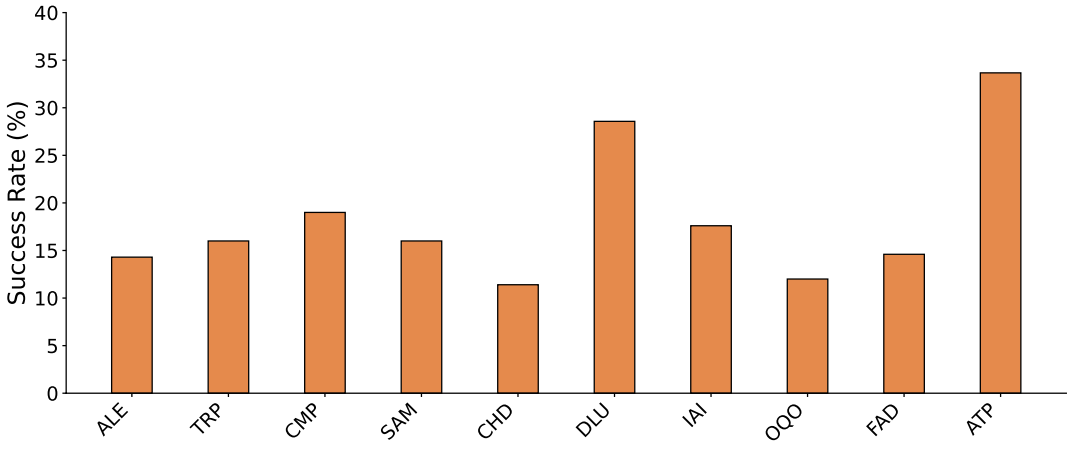

Figure S9 TorchScore computational pass rates in the small-molecule benchmark. Pass-rate evaluation of TorchCraft ligand-conditioned protein designs across the 10-ligand panel using the TorchScore criterion.