MONTE CARLO TREE HALLUCINATION FOR ALL-ATOM BIOMOLECULAR SEQUENCE–STRUCTURE CO-DESIGN

Anonymous authors
Paper under double-blind review

ABSTRACT

Biomolecular design underpins applications from molecular recognition to therapeutics and synthetic biology, yet de novo interaction design remains challenging—especially for DNA/RNA, underexplored non-protein modalities with scarce, heterogeneous complex data and sharper geometric and chemical constraints. We introduce MCTH (Monte Carlo Tree Hallucination), an inference-only framework that casts all-atom sequence–structure co-design as uncertainty-aware planning over hallucinated states from pretrained folding and inverse-folding models, with optional biophysical control within the same decision loop. MCTH treats these models as frozen black-box operators and uses Monte Carlo Tree Search to allocate a fixed inference budget across competing design trajectories, incorporating model confidence and uncertainty, as well as cross-expert consensus/disagreement when multiple predictors are available. Across protein–RNA, protein–DNA, protein–protein, and protein–ligand design, matched-budget experiments show that adaptive search improves over simpler sampling and cycling strategies, while held-out AlphaFold3 and Chai-1 evaluations demonstrate transfer beyond the search-time oracle. MCTH provides a shared planning layer across modalities while allowing task-specific folding, inverse-folding, and biophysical modules, requiring no fine-tuning or backpropagation through component models.

1 INTRODUCTION

Biomolecular design underpins modern molecular biology, synthetic biology, and biomedicine, enabling applications from catalysis and therapeutics to biosensing and programmable systems. A core goal is to design biomolecular interactions—protein–protein, protein–small-molecule, protein–nucleic-acid, and nucleic-acid–nucleic-acid complexes—with high structural fidelity and functional precision. Despite decades of progress, de novo design remains challenging, especially beyond proteins to DNA/RNA, small molecules, ions, and post-translational modifications.

A core difficulty is the intrinsically coupled nature of sequence and structure: in a rugged, high-dimensional landscape, small sequence changes can induce large structural or functional shifts. Capturing interaction specificity while jointly optimizing sequence and structure is computationally demanding and often difficult to control. Physics-based modeling (e.g., Rosetta-style energy functions) provides explicit chemical priors but can be costly and brittle under local optimization (Alford et al., 2017; Leaver-Fay et al., 2011). Learning-based methods offer expressive data-driven priors, yet often struggle with generalization, exploration, and optimization stability across heterogeneous modalities. These issues are amplified for DNA/RNA systems, where complex data are scarce and success hinges on fine-grained interaction fidelity (e.g., base stacking, ion/ligand coordination, atomic contacts) rather than global fold alone.

Recent advances in structure prediction, including AlphaFold2, AlphaFold-Multimer, and AlphaFold3, have expanded biomolecular modeling from proteins to diverse complexes (Jumper et al., 2021; Evans et al., 2021; Abramson et al., 2024). However, structure prediction alone does not solve design. Backbone generators such as RFdiffusion typically pair generation with inverse folding and refolding-based verification (Watson et al., 2023; Dauparas et al., 2022), while SCUBA-D learns a generative model independently of pretrained structure predictors (Liu et al., 2024b). Alternatively, hallucination methods optimize sequences against predictor-derived objectives (Anishchenko et al.,

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

2021; Pacesa et al., 2024; Cho et al., 2025a). These approaches offer different forms of control, but repeated structure evaluation remains a major computational cost.

An emerging alternative paradigm exploits a property of diffusion-based structure prediction models often viewed as a limitation: hallucination. Given underspecified or out-of-distribution inputs—such as all-X sequences or partially defined complexes—these models can hallucinate plausible conformations guided by learned priors. Cycling between hallucinated structure prediction and sequence redesign can yield designable biomolecules without fine-tuning (Cho et al., 2025b; Fang et al., 2025). Fixed-schedule cycling does not adaptively distribute evaluations across competing trajectories, motivating a planning layer that decides which hypotheses to revisit under a finite budget. Moreover, high model confidence need not imply *biophysical feasibility*: unconstrained hallucinations can violate fine-grained constraints (e.g., steric clashes, strained geometries, incompatible ligand/ion coordination).

In this work, we propose MCTH, an inference-time planning framework that reframes biomolecular design as structured search over hallucinated sequence–structure states. Pretrained folding and inverse-folding models act as frozen black-box operators that propose and evaluate candidate transitions, while Monte Carlo Tree Search allocates a fixed inference budget across competing design trajectories. Unlike fixed-depth cycling or gradient-based optimization through a structure predictor, MCTH requires only model outputs—sequences, structures, and confidence or feasibility scores—and does not access, update, or differentiate through component-model parameters.

The planner is uncertainty-aware: model confidence, expert consensus/disagreement when multiple predictors are available, and optional biophysical feasibility signals can jointly guide selection, expansion, and backup. The same planning logic is reused across modalities, while the folding, inverse-folding, and biophysical modules are dispatched according to the task. This yields a unified *co-hallucination* strategy: biophysical cues enter the same selection/expansion loop as learned proposals, rather than acting only as a post-hoc filter, which is particularly relevant for nucleic-acid and ligand-conditioned settings where data are limited and geometric constraints are sharp. In the main experiments we use a single all-atom folding model with modality-specific inverse folding; we additionally validate the multi-folding-expert capability in protein-binder design using Boltz-2 and AlphaFold2 jointly during search.

Contributions and novelty.

- **Inference-time planning over sequence–structure states.** We cast hallucination-based biomolecular design as a finite-budget planning problem and use MCTS to adaptively allocate expensive folding evaluations across competing sequence–structure trajectories, rather than following fixed cycling or independent best-of-N sampling.
- **Composable black-box model coordination.** MCTH treats pretrained folding, inverse-folding, and scoring models as frozen operators and requires only their emitted sequences, structures, and scalar signals—without fine-tuning or backpropagation through component models. We also validate the combination of multiple folding experts through consensus, disagreement, and uncertainty signals in a protein-binder setting.
- **Biophysics–ML co-hallucination:** We integrate optional geometric, thermodynamic, and energy-like feasibility cues with learned model confidence within the same planning loop, enabling task-specific control during selection, proposal refinement, and evaluation rather than only post-hoc filtering.
- **General planning layer across biomolecular design tasks.** The MCTS state, selection, expansion, and backup mechanism is shared across protein–RNA, protein–DNA, protein–protein, and protein–ligand design, while inverse-folding backends and biophysical terms remain modality-specific. Matched-budget and held-out cross-predictor evaluations test both search efficiency and robustness beyond the search-time oracle.

2 Background and Problem Setup

Related method families and positioning. Physics-first frameworks such as Rosetta/MD provide explicit energetic priors but scale poorly in large multimodal design spaces (Alford et al., 2017; Leaver-Fay et al., 2011; Lindorff-Larsen et al., 2011; Rohl et al., 2004; Shaw et al., 2010). Learning-based structure predictors (AF2/Multimer, all-atom AF3/RFAA) expanded biomolecular modeling

beyond proteins (Abramson et al., 2024; Evans et al., 2021; Jumper et al., 2021; Krishna et al., 2024); diffusion and inverse-folding tools such as RFdiffusion, ProteinMPNN, and RNA pipelines enabled generation/redesign (Anand et al., 2025; Dauparas et al., 2022; Joshi et al., 2024; Watson et al., 2023). Hallucination/cycling and planning methods, including BindCraft/BoltzDesign and ERP/MCTD-ME, improve optimization or exploration but are often local, fixed-schedule, or single-modality (Anishchenko et al., 2021; Cho et al., 2025a;b; Fang et al., 2025; Liu et al., 2024a; 2025; Pacesa et al., 2024).

Relative to these lines, MCTH occupies a distinct point in the design stack: it is a *planning layer* that wraps pretrained folding and inverse-folding experts with plug-in biophysical cues and allocates expensive inference through MCTS. Its key distinction is *co-hallucination*: learned experts propose sequence–structure hypotheses, while biophysical cues enter the same decision loop for selection, expansion, and proposal reweighting, rather than acting only as post-hoc filters. Complexa also combines a generative prior with inference-time optimization for protein binder design (Didi et al., 2026); MCTH instead searches through composable folding and inverse-folding transitions. A fuller discussion of related work is provided in Appendix B, with a compact positioning summary in Appendix B.1.

Sequence–structure design as planning. We represent a biomolecular system by sequence(s) S and all-atom coordinates X; for complexes, the state includes all interacting components, e.g., $s = (S, X)$ under task conditioning C such as binding partners, motifs, or ligands. We view design as finite-horizon planning in an MDP $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, H, P, R \rangle$ (Puterman, 2014), where states $s \in \mathcal{S}$ are biomolecular sequence–structure hypotheses and actions $a \in \mathcal{A}$ are stochastic hallucination operators such as inverse-fold proposals, folding calls, or biophysical refinement/filtering steps. Transitions are induced by pretrained experts, $s' \sim P(\cdot \mid s, a)$, and rewards are computed from the resulting state, e.g., $R(s, a) = V(s')$.

Search and guided proposal view. MCTS iteratively expands a partial search tree through selection, expansion, evaluation, and backpropagation. In our setting, nodes are biomolecular states $s = (S, X)$, and evaluations come directly from model-derived value signals $V(s)$, such as confidence, interface quality, and optional feasibility terms, rather than long stochastic rollouts. Diffusion-style guidance can be viewed as reweighting proposals by an auxiliary score, $p^{\text{guide}}(\xi' \mid \xi) \propto p(\xi' \mid \xi) \exp\{\beta f(\xi')\}$; in MCTH, this score is instantiated by the same task-aware value signal used by planning. Practically, we implement this without gradients by sampling proposal candidates from different experts, seeds, or noise schedules and selecting/resampling according to their value.

Design objectives. We encode task objectives such as foldability, binding confidence, constraint satisfaction, and lead optimization through the state value $V(s)$. Thus, MCTH searches over hallucinated sequence–structure states using learned confidence and optional biophysical feasibility signals, rather than performing explicit end-to-end energy minimization. Additional preliminaries are provided in Appendix F.

3 Methodology

Figure 1 summarizes MCTH: inverse folding proposes sequences, folding predicts their structures, and a task-aware score guides tree search under a finite evaluation budget. Optional biophysical terms provide task-specific control within this loop.

3.1 Composable Folding and Inverse-Folding Operators

MCTH treats pretrained folding and inverse-folding models as frozen black-box operators that induce transitions in the search tree under task conditioning C (e.g., binding partners, motifs, or ligands). The planner interacts with these models only through forward-inference outputs—predicted structures, proposed sequences, and associated confidence or feasibility signals—without accessing model parameters or backpropagating through the component models. This interface allows the same search procedure to compose different folding and inverse-folding backends while keeping the planner itself unchanged.

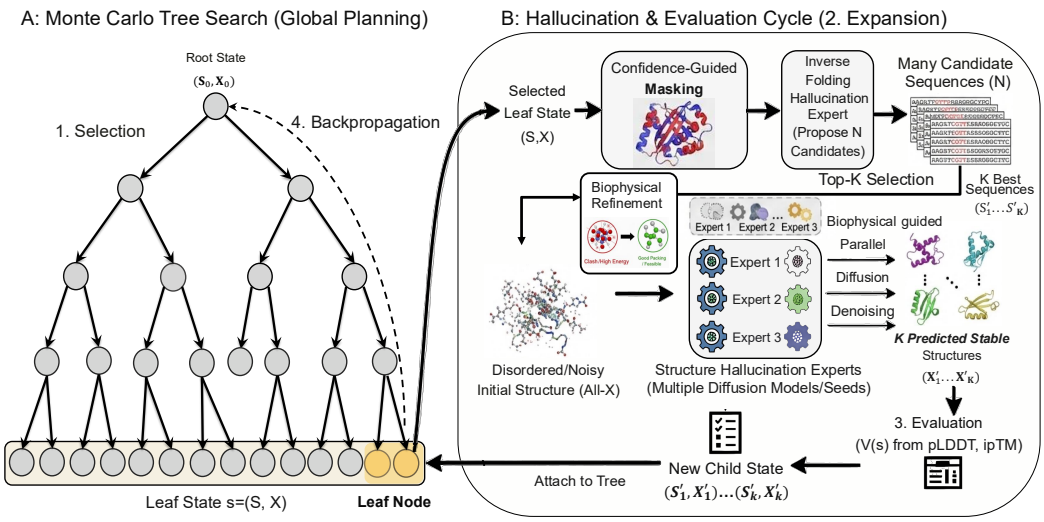


Figure 1: The MCTH framework: search-guided sequence–structure co-design. Left: UCT-style Monte Carlo Tree Search allocates computation over states $s = (S, X)$ via selection and backpropagation. Right: each expansion instantiates a hallucination–evaluation cycle that proposes new sequences (inverse folding), generates corresponding structures (diffusion-based hallucination), and evaluates candidates using model confidence signals (e.g., ipTM/pLDDT), which are then backed up to guide subsequent search. Confidence-guided masking is an optional refinement operator and is disabled in the minimal plug-and-play configuration used in this work.

Structure hallucination via diffusion models. Given a sequence hypothesis S and optional conditioning information C, a folding or structure-prediction model generates a candidate all-atom conformation

$$X \sim p_\theta(X \mid S, C). \tag{1}$$

When the sequence or complex specification is underspecified—for example, an all-X sequence or a partially defined interface—the predictor may generate plausible conformations from its learned structural prior. MCTH uses these predictions as candidate sequence–structure states for subsequent evaluation and search.

The main experiments use Boltz-2 with modality-specific inverse folding. A separate protein-binder experiment combines Boltz-2 and AlphaFold2; cross-expert terms apply only when multiple predictors are instantiated.

Sequence redesign via inverse folding. Given a hallucinated structure X and conditioning information C (e.g., binding partners, fixed motifs, ligand identity/pose constraints, or other task-specific constraints), inverse-folding experts propose compatible sequences

$$S \sim q_\phi(S \mid X, C), \tag{2}$$

improving foldability and interaction consistency while preserving the specified constraints. Unlike fixed iterative pipelines, inverse folding in MCTH is applied adaptively, allowing the planner to refine promising structural hypotheses or repair inconsistencies introduced by hallucination.

3.2 Uncertainty-Aware MCTS over Sequence–Structure States

Each tree node is a sequence–structure hypothesis $s = (S, X)$ under fixed task conditioning C. An action proposes a sequence S' by inverse folding and predicts the corresponding complex X' by folding, yielding a child $s' = (S', X')$. When multiple folding experts are used, the candidate additionally carries the ensemble $\{X'^{(e)}\}_{e=1}^{E}$. Thus, search evaluates both a sequence and the complex it induces. Retaining alternative states allows the planner to revisit an earlier structural hypothesis and redesign its sequence after another branch stalls. Greedy cycling instead commits to a single trajectory, while independent sampling does not reuse intermediate improvements. The tree therefore provides a mechanism for deciding where another expensive folding evaluation is most useful.

State value. Each expanded child is assigned a task-aware value

$$V(s') = f(m_1(s'), \ldots, m_J(s'))\,, \tag{3}$$

where the m_j's are predictor-derived signals such as pLDDT, pTM, ipTM, interface quality, and optional geometric or biophysical feasibility terms. The aggregation function f maps these signals to a scalar value used for evaluation and backup. Unless otherwise stated, we use a fixed task-level aggregation across targets rather than target-specific tuning. All signals are obtained through forward inference; MCTH does not require gradients or backpropagation through the component models.

Uncertainty-aware selection. At each MCTS iteration, MCTH traverses the current tree from the root to a leaf. We use a heuristic *consensus–uncertainty–physics* PUCT-style score, CU-PUCT:

$$\begin{aligned}\text{CU-PUCT}(s,a) = Q(s,a) + c_p \frac{\sqrt{\log N(s)}}{1+N(s,a)} \cdot \pi_{\text{cons},\tau}(a \mid s) \\ \cdot \Big(w_{\text{inv}} U_{\text{inv}}(s,a) + w_{\text{fold}} U_{\text{fold}}(s,a)\Big) \cdot \exp[-w_{\text{phys}} E_{\text{phys}}(s,a)]\,.\end{aligned} \tag{4}$$

We select

$$a^\star = \arg\max_{a \in \mathcal{A}(s)} \text{CU-PUCT}(s,a), \tag{5}$$

where $Q(s,a)$ is the empirical action value, $N(s)$ and $N(s,a)$ are visit counts, and c_p controls exploration.

The selection score combines the value already observed on a branch with an exploration bonus. The prior $\pi_{\text{cons},\tau}$ favors sequences compatible with the predicted structure and, when multiple experts are used, agreement across their predictions. U_{inv} measures ambiguity in sequence redesign, while U_{fold} measures low folding confidence or cross-expert structural disagreement. The optional penalty E_{phys} down-weights physically implausible proposals. Together, these signals allow the planner to revisit uncertain but promising alternatives instead of committing to the currently highest-scoring trajectory. The rule is heuristic; normalization, uncertainty definitions, and consensus construction are specified in Appendix C.1.

Expansion, evaluation, and backup. Once selection reaches a leaf, MCTH expands up to K candidate children. Each child is constructed by inverse folding followed by folding/refolding, yielding a new sequence–structure state and the signals used to evaluate $V(s')$. The folding call also supplies the child evaluation, so no separate folding rollout is required. The resulting value is backed up along the selected path, incrementing visit counts and updating $Q(s,a)$ as a running mean of backed-up values. Repeated iterations therefore allocate the fixed inference budget toward promising and informative branches and, when applicable, toward transitions supported across experts and consistent with biophysical constraints.

3.3 Optional Biophysical Control

For DNA aptamers, E_{phys} is a soft window penalty on nearest-neighbor self-folding free energy at 37 °C. The window encodes a task-specific trade-off: strong self-structure may require disruption to expose a binding interface, whereas weak pre-organization may increase the ordering cost upon binding. It serves as a design prior rather than a binding-affinity estimate. The window and its calibration from random sequences are specified in Appendix C.3.

This penalty can be complemented by sequence refinement at non-interface positions and, for compatible backends, soft contact restraints during refolding. These operators let physical preferences influence both which states are explored and how candidates are refined; their definitions and backend settings appear in Appendices C.7 and C.5.

3.4 Multi-modality Representation and Expert Assignment

For the modality-spanning main experiments, we use Boltz-2 as a shared all-atom folding expert, whose representation supports proteins, nucleic acids, small molecules, and their complexes (Passaro et al., 2025). Sequence redesign is dispatched by modality: ProteinMPNN for proteins and peptides (Dauparas et al., 2022), NA-MPNN for nucleic acids (Kubaney et al., 2025), and LigandMPNN

for ligand-conditioned protein design (Dauparas et al., 2025). Task-specific biophysical terms and refinement operators are enabled only where applicable.

The MCTS state, selection, expansion, and backup rules remain shared across modalities. Appendix C.4 summarizes the dispatch rules and supported interfaces.

4 EXPERIMENTS

Evaluation protocol. We evaluate MCTH from three complementary perspectives: optimization under the search-time oracle, matched-budget comparisons, and held-out evaluation. Matched-budget controls use the same initialization, folding oracle, inverse-folding backend, and evaluation budget to isolate search-guided compute allocation. Baseline comparisons use a common refolding protocol wherever applicable. AlphaFold3 re-scores nucleic-acid designs, Chai-1 (Chai Discovery, 2024) evaluates the dual-expert protein-binder experiment, and Rosetta InterfaceAnalyzer provides an additional physical check. For protein binders, we also report archive pass fractions under explicit held-out Chai-1 confidence and clash criteria.

4.1 NUCLEIC ACID DESIGN

4.1.1 RNA APTAMER DESIGN

Protein–RNA aptamer benchmark. We evaluate MCTH on a protein–RNA aptamer design benchmark consisting of 7 protein–RNA chain-pair targets derived from PDB complexes (7WKP, 7YEW, 7YGL, 8TG4, 8GXB, 9C7A), spanning RNA lengths from 4 to 61 nucleotides. Given a protein structure and the RNA length specification, the goal is to design an RNA aptamer sequence that yields a confident protein–RNA interface.

Baselines and matched-budget evaluation. We compare against ODesign (Zhang et al., 2025) and RNA-FrameFlow (Anand et al., 2025) + NA-MPNN (Kubaney et al., 2025). To control for the effect of repeated structure evaluation, both baselines are run with 100 candidate designs and re-scored using the same Boltz-2 refolding protocol. MCTH uses up to 100 folding calls, with early stopping, whereas each baseline supplies 100 candidates for the common refold evaluation. This aligns the maximum evaluator budget, not total generation cost.

Results under the search-time oracle. Under this protocol (Table 1A), MCTH achieves a mean best ipTM of 0.946, compared with 0.894 for ODesign and 0.923 for RNA-FrameFlow + NA-MPNN. MCTH obtains the highest ipTM on 6 of 7 targets; the only exception is 8GXB, where RNA-FrameFlow is higher by 0.001 (0.880 vs. 0.879).

Held-out AlphaFold3 evaluation. We additionally re-score the final designs using AlphaFold3, which is never used during search. Under this held-out predictor, MCTH retains the highest mean ipTM (0.884), compared with 0.824 for ODesign and 0.794 for RNA-FrameFlow + NA-MPNN, and is top-ranked on 6 of 7 targets (including one tie). The persistence of the ranking under a held-out predictor suggests that the gain is not confined to Boltz-2-specific preferences.

4.1.2 DNA APTAMER DESIGN

Protein–DNA aptamer benchmark. We evaluate MCTH on 10 protein–ssDNA chain-pair targets derived from four PDB complexes (7XVN, 7YSF, 7YUK, and 8PMF). Given a fixed protein target and a DNA length specification, the goal is to design an ssDNA aptamer sequence that forms a confident protein–DNA interface.

Baselines and matched-budget evaluation. We compare against ODesign as the native nucleic-acid design baseline. To control for repeated evaluator access, ODesign is run with 100 candidate designs and re-scored using the same Boltz-2 refolding protocol. Two 7XVN chain-B cases collapse to ipTM ≈ 0 for both ODesign and MCTH under Boltz-2; we therefore exclude these two oracle-failed cases from the matched-budget comparison and report the remaining eight chain-A targets. BoltzDesign is not included in this matched-budget table because its released design procedure

A RNA aptamers

Target	ODesign	RFF	MCTH
7WKP A:B	.883	.932	**.941**
7YEW A:B	.928	.935	**.960**
7YGL A:B	.918	.924	**.963**
7YGL A:C	.928	.927	**.953**
8TG4 A:C	.924	.925	**.963**
8GXB C:B	.835	**.880**	.879
9C7A A:B	.841	.940	**.960**
Boltz-2 mean	.894	.923	**.946**
Held-out AF3	.824	.794	**.884**

B DNA aptamers

Target	ODesign	MCTH
7XVN A:N	.936	**.952**
7XVN A:P	**.957**	.950
7YSF A:B	.890	**.898**
7YSF A:C	.869	**.876**
7YUK A:C	.904	**.906**
7YUK A:D	.901	**.912**
8PMF A:B	.914	**.956**
8PMF A:D	.895	**.947**
Boltz-2 mean	.908	**.925**
Held-out AF3	.644	**.721**

Table 1: **Aptamer design across targets and predictors.** Target rows show best Boltz-2 ipTM at up to 100 folding calls for MCTH and 100 commonly refolded candidates per baseline. RFF denotes RNA-FrameFlow + NA-MPNN; A:B specifies receptor and aptamer chains. The last row reports mean held-out AF3 ipTM: seven RNA targets and all ten DNA targets, including the two chain-B cases excluded from Boltz-2 because both methods fail under that oracle. Bold marks the highest value in each row. Full held-out results are in Tables 4b and 6b.

optimizes the protein chain in protein–DNA complexes rather than directly generating the DNA strand.

Results under the search-time oracle. On the eight non-degenerate Boltz-2 targets (Table 1B), MCTH achieves a mean best ipTM of 0.925 versus 0.908 for ODesign@100 and obtains the highest score on 7 of 8 targets. The gap is relatively small on several already high-scoring targets, indicating that strong best-of-N sampling can approach search when the oracle landscape saturates, while adaptive allocation still improves the overall fixed-budget result.

Held-out AlphaFold3 evaluation. We additionally re-score designs from all 10 targets using AlphaFold3, which is never used during search. Under AF3, the two 7XVN chain-B cases are no longer degenerate, indicating that their collapse under Boltz-2 is evaluator-specific rather than evidence that the design targets themselves are intrinsically invalid. Across all 10 targets, MCTH achieves the highest mean ipTM (0.721), compared with 0.665 for BoltzDesign and 0.644 for ODesign, and obtains the top score on 6 of 10 targets. The comparison with ODesign supports transfer beyond Boltz-2, while individual target rankings still differ across predictors. BoltzDesign uses a different designable-chain protocol and is reported separately as context in Table 6b.

4.2 Protein Binder Design

Benchmark and setup. We evaluate MCTH on de novo protein–protein binder design using the 12-target CaoData benchmark. For each target, the receptor is fixed with its native sequence and structure, while the binder is designed de novo at the reference length. Targets span diverse receptor and binder sizes, including the multi-chain antibody-style case 5U8R.

The main experiment uses Boltz-2 and ProteinMPNN, starting from a random binder sequence at the reference length, with a budget of 100 Boltz-2 evaluations per target.

Matched-budget optimization. We compare against BoltzDesign1, ODesign, and Complexa under a matched budget of up to 100 folding/evaluator calls per target. ODesign and Complexa candidates are compared under the common Boltz-2 refolding protocol, while BoltzDesign1 uses its native Boltz evaluation during optimization.

Across CaoData, MCTH achieves the highest mean best ipTM (0.864), compared with 0.853 for Complexa, 0.816 for ODesign, and 0.539 for BoltzDesign1 (11 available targets). The margin over Complexa is small and target-dependent: Complexa leads on 3FKD (0.958 vs. 0.686), 2GY7 (0.930 vs. 0.862), and 5U8R (0.783 vs. 0.576). These results show a modest average advantage

A Matched evaluator budget

Target	ODesign	Complexa	MCTH
1DJS	**.944**	.907	.933
1MOX	.901	.845	**.929**
1XIW	.940	**.946**	**.946**
2GY7	.672	**.930**	.862
2IFG	.730	.356	**.806**
3DI3	.901	.926	**.947**
3FKD	.556	**.958**	.686
3MJG	.878	.888	**.904**
3ZTJ	.872	**.941**	.914
4O3V	.944	.852	**.953**
4OGA	**.916**	.904	.912
5U8R	.532	**.783**	.576
Mean	.816	.853	**.864**

B Transfer and design yield

Mean best ipTM: 11 shared targets

Predictor	Single	Dual	RFdiff.
Boltz-2	.69	.74	.47
AlphaFold2	.19	.61	.58
Chai-1 (held out)	.18	.57	.60
Retained/target	3	3	10

Equal outputs: 10 designs per target

Chai-1 criterion	Dual	RFdiff.
Passing designs	41/110	22/110
Pass rate	**37.3%**	20.0%
Targets with a pass	8/11	8/11

Table 2: **Protein-binder optimization and transfer. A:** Best Boltz-2 ipTM at up to 100 evaluator calls on CaoData, using common refolding; bold marks row maxima. BoltzDesign1 uses its native evaluator (mean .539 on 11 targets; Table 7). **B:** Predictor-wise maxima use the indicated retained pools; pass rates use ten outputs per target, with MCTH ranked by in-loop reward. Passing requires held-out Chai-1 ipTM $\geq .50$, pTM $\geq .50$, and no interchain clash. Equal output counts do not imply equal total compute.

with substantial target-specific variation (Table 2A). Appendix E.6 reports descriptive Chai-1 pass fractions for Complexa's native retained outputs (1–3 per target) and MCTH's fixed-size archives, reflecting their respective retention policies.

Dual-expert cross-predictor evaluation. We next ask whether agreement between search-time predictors translates into stronger support from an unseen predictor. The dual-expert variant retains ProteinMPNN and MCTS while using both Boltz-2 and AlphaFold2 for evaluation and selection. Chai-1 is held out entirely from search.

Across the 11 targets shared by all configurations, the original Boltz-only MCTH obtains mean ipTM values of $0.69/0.19/0.18$ under Boltz-2/AF2/Chai-1, respectively. The dual-expert variant improves these to $0.74/0.61/0.57$. In particular, held-out Chai-1 improves from 0.18 to 0.57, with the dual-expert variant outperforming the single-expert configuration on 9 of 11 targets and tying on one. This supports improved transfer to a third predictor outside the search loop. Table 2B separates these predictor-wise maxima from the equal-output comparison below; target-level results appear in Table 8.

Matched final-design pass rate. Best-per-target scores describe discovery of a strong candidate, but not design yield. We compare MCTH and RFdiffusion using ten final designs per target on the same 11 targets and the Chai-1 criterion in Table 2B (Appendix E.6). MCTH candidates are ranked only by recorded in-loop joint reward, without held-out metrics.

Dual-expert MCTH yields 41 passing designs out of 110 (37.3%), compared with 22/110 (20.0%) for RFdiffusion, corresponding to a 17.3 percentage-point or $1.86\times$ higher pass rate. Both methods produce at least one passing design on 8 of 11 targets. The gain is therefore in the fraction of qualifying designs, with unchanged target coverage.

4.3 Search Efficiency and Additional Evaluation

Search versus repeated sampling. With initialization, proposals, scoring, and folding calls fixed, MCTH improves over independent sampling, greedy cycling, and exploration-off search (Table 3A). The advantage over cycling is larger for RNA and DNA than for proteins, supporting the value of retaining alternative trajectories in these tasks. The convergence diagnostic records the last improvement at fold call 46.6 on average for MCTH; every run improves by more than 0.002 beyond

A Controlled search ablation

Strategy	Protein	RNA	DNA
Best-of-N	.851	.778	.829
Greedy cycling	.928	.784	.828
Exploration-off	.925	.839	.914
MCTH	**.938**	**.913**	**.955**
Targets	2	3	3
Seeds/target	3	3	3

B Protein budget trajectory

Fold calls	ODesign	MCTH
1	.518	.614
5	.668	.724
10	.713	.768
25	.758	.803
50	.796	.830
100	.816	.864

Table 3: **How additional evaluations improve design. A:** Mean best ipTM under 100 folding calls, with identical initialization, proposal engine, scoring, and oracle within each task. Per-target variability is reported in Tables 15–17. **B:** Mean held-forward best Boltz-2 ipTM across the 12 protein targets. These full pipelines differ in proposal generation; panel A separately controls for that difference when testing allocation.

initialization, whereas 44–62% of reduced-search runs do not. Unique-state counts do not exceed best-of-N, suggesting that the gain comes from allocating evaluations to useful continuations rather than simply covering more states (Appendix E.8).

Ligand-conditioned design and physical checks. The same planner with LigandMPNN reaches mean ipTM 0.901 on ten protein–ligand targets, compared with 0.850 for RFdiffusion-AA and 0.460 for ODesign under Boltz-2 refolding (Appendix D.1). Replacing ligand-aware proposals with ligand-blind ProteinMPNN reduces mean ipTM to 0.826, illustrating the role of task-specific proposals within the shared planner. Separately, Rosetta InterfaceAnalyzer yields comparable area-normalized interface energies for MCTH and RFdiffusion (-3.12 and -3.13 per $100\,\text{Å}^2$); BindCraft scores -3.40 on its five available targets. These are interface-quality checks, not affinity measurements (Appendix E.7).

Compute. Mean realized folding calls are 98.1, 52.4, and 66.1 for protein, RNA, and DNA design, with approximate wall-clock times of 9.0, 0.8, and 1.1 hours, respectively. Early stopping explains the lower aptamer costs. Protein designs continue to improve across the budget: MCTH rises from 0.614 at one call to 0.864 at 100, while ODesign rises from 0.518 to 0.816 (Table 3B). The initial difference reflects the full pipelines; the controlled ablations above isolate search allocation. Hardware and stopping details are in Appendix E.8.

5 Conclusion

We presented MCTH, an inference-only framework that reframes biomolecular design as an *uncertainty-aware* search over hallucinated sequence–structure states. By coordinating frozen folding, inverse-folding, and optional biophysical operators with MCTS, MCTH allocates expensive structure evaluations using confidence, uncertainty, and, when multiple experts are available, consensus/disagreement signals. Across protein–RNA, protein–DNA, protein–protein, and protein–ligand design, matched-budget experiments demonstrate effective search-guided compute allocation, while held-out AlphaFold3 and Chai-1 evaluation shows that improvements can transfer beyond the search-time oracle. The planning layer is shared across tasks, while expert backends and biophysical terms remain modality-specific. Our evaluation remains entirely computational and does not replace experimental validation. Future work should prioritize wet-lab validation, stronger physics-based objectives, and broader multi-expert evaluation across modalities. Overall, MCTH suggests that closed-loop, biophysics-informed planning is a promising route for controllable biomolecular co-design across heterogeneous modalities.

Limitations. Our evaluation is computational, with small target sets and limited seed coverage. Binding, specificity, and stability require experimental validation. CU-PUCT is heuristic, and the present ablations do not isolate every scoring term. Broader target coverage and matched-filter comparisons are important next steps.

AI USE STATEMENT

Generative AI tools assisted with manuscript organization and editing, technical exposition, and experimental code development. The authors developed the research ideas and methodology, verified all AI-assisted material, and take full responsibility for the final text, claims, code, and artifacts.

ETHICS STATEMENT

This work studies computational biomolecular sequence–structure co-design without human participants, animal experiments, or clinical deployment. Improvements in model confidence and computational biophysical scores do not establish biological function, binding affinity, or therapeutic safety, which require experimental validation. Potential downstream applications should account for the dual-use risks of biomolecular design and undergo appropriate biosafety review before synthesis and testing.

REPRODUCIBILITY STATEMENT

Sections 2 and 3 describe the problem formulation and MCTH algorithm. Appendix C details the search procedure, pretrained model configurations, scoring functions, biophysical controls, and hyperparameters. Appendices D and E provide additional experiments, matched-budget ablations, held-out evaluations, and compute diagnostics. Table 11 specifies the main configuration, random seeds, and hardware.

REFERENCES

Josh Abramson, Jonas Adler, Jack Dunger, Richard Evans, Tim Green, Alexander Pritzel, Olaf Ronneberger, Lindsay Willmore, Andrew J Ballard, Joshua Bambrick, et al. Accurate structure prediction of biomolecular interactions with alphafold 3. *Nature*, 630(8016):493–500, 2024. doi: 10.1038/s41586-024-07487-w. 1, 3, 16

Rebecca F Alford, Andrew Leaver-Fay, Jeliazko R Jeliazkov, Matthew J O'Meara, Frank P DiMaio, Hahnbeom Park, Maxim V Shapovalov, P Douglas Renfrew, Vikram K Mulligan, Kalli Kappel, Jason W Labonte, Michael S Pacella, Richard Bonneau, Philip Bradley, Roland L Dunbrack, Rhiju Das, David Baker, Brian Kuhlman, Tanja Kortemme, and Jeffrey J Gray. The Rosetta all-atom energy function for macromolecular modeling and design. *Journal of Chemical Theory and Computation*, 13(6):3031–3048, 2017. doi: 10.1021/acs.jctc.7b00125. 1, 2, 16

Obtin Alkhamis, Caleb Byrd, Juan Canoura, Adara Bacon, Ransom Hill, and Yi Xiao. Exploring the relationship between aptamer binding thermodynamics, affinity, and specificity. *Nucleic Acids Research*, 53(6):gkaf219, 2025. doi: 10.1093/nar/gkaf219. 20

Rishabh Anand, Chaitanya K. Joshi, Alex Morehead, Arian R. Jamasb, Charles Harris, Simon V. Mathis, Kieran Didi, Rex Ying, Bryan Hooi, and Pietro Liò. RNA-FrameFlow: Flow matching for de novo 3d RNA backbone design. *Transactions on Machine Learning Research*, 2025. URL https://openreview.net/forum?id=wOc1Yx5s09. 3, 6, 16

Ivan Anishchenko, Samuel J Pellock, Tamuka M Chidyausiku, Theresa A Ramelot, Sergey Ovchinnikov, Jingzhou Hao, Khushboo Bafna, Christoffer Norn, Alex Kang, Asim K Bera, Frank DiMaio, Lauren Carter, Cameron M Chow, Gaetano T Montelione, and David Baker. De novo protein design by deep network hallucination. *Nature*, 600(7889):547–552, 2021. doi: 10.1038/s41586-021-04184-w. 1, 3, 17

Minkyung Baek, Frank DiMaio, Ivan Anishchenko, Justas Dauparas, Sergey Ovchinnikov, Gyu Rie Lee, Jue Wang, Qian Cong, Lisa N Kinch, R Dustin Schaeffer, et al. Accurate prediction of protein structures and interactions using a three-track neural network. *Science*, 373(6557):871–876, 2021. doi: 10.1126/science.abj8754. 16

Emmanuel Bengio, Moksh Jain, Maksym Korablyov, Doina Precup, and Yoshua Bengio. Flow network based generative models for non-iterative diverse candidate generation. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 27381–27394, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/e614f646836aaed9f89ce58e837e2310-Abstract.html. 17

Chai Discovery. Chai-1: Decoding the molecular interactions of life. *bioRxiv*, 2024. doi: 10.1101/2024.10.10.615955. 6

Yehlin Cho, Martin Pacesa, Zhidian Zhang, Bruno Correia, and Sergey Ovchinnikov. Boltzdesign1: Inverting all-atom structure prediction model for generalized biomolecular binder design. *bioRxiv*, pp. 2025.04.06.647261, 2025a. doi: 10.1101/2025.04.06.647261. 2, 3, 17

Yehlin Cho, Griffin Rangel, Gaurav Bhardwaj, and Sergey Ovchinnikov. Protein hunter: exploiting structure hallucination within diffusion for protein design. *bioRxiv*, pp. 2025.10.10.681530, 2025b. doi: 10.1101/2025.10.10.681530. 2, 3, 17

Gabriele Corso, Hannes Stärk, Bowen Jing, Regina Barzilay, and Tommi Jaakkola. DiffDock: Diffusion steps, twists, and turns for molecular docking. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2210.01776. 17

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. doi: 10.1126/science.add2187. 1, 3, 5, 16, 21

Justas Dauparas, Gyu Rie Lee, Robert Pecoraro, Linna An, Ivan Anishchenko, Cameron Glasscock, and David Baker. Atomic context-conditioned protein sequence design using LigandMPNN. *Nature Methods*, 22(4):717–723, 2025. doi: 10.1038/s41592-025-02626-1. 6

Pierre Del Moral, Arnaud Doucet, and Ajay Jasra. Sequential monte carlo samplers. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 68(3):411–436, 2006. doi: 10.1111/j.1467-9868.2006.00553.x. URL https://www.stats.ox.ac.uk/~doucet/delmoral_doucet_jasra_sequentialmontecarlosamplersJRSSB.pdf. 22, 24

Kieran Didi, Zuobai Zhang, Guoqing Zhou, Danny Reidenbach, Zhonglin Cao, Sooyoung Cha, Tomas Geffner, Christian Dallago, Jian Tang, Michael M. Bronstein, Martin Steinegger, Emine Kucukbenli, Arash Vahdat, and Karsten Kreis. Scaling atomistic protein binder design with generative pretraining and test-time compute. *arXiv preprint arXiv:2603.27950*, 2026. URL https://arxiv.org/abs/2603.27950. 3

Richard Evans, Michael O'Neill, Alexander Pritzel, Natasha Antropova, Andrew Senior, Tim Green, Augustin Žídek, Russ Bates, Sam Blackwell, Jason Yim, Olaf Ronneberger, Sebastian Bodenstein, Michal Zielinski, Alex Bridgland, Anna Potapenko, Andrew Cowie, Kathryn Tunyasuvunakool, Rishub Jain, Ellen Clancy, Pushmeet Kohli, John Jumper, and Demis Hassabis. Protein complex prediction with AlphaFold-Multimer. *bioRxiv*, 2021. doi: 10.1101/2021.10.04.463034. 1, 3, 16

Minchao Fang, Chentong Wang, Jungang Shi, Fengbai Lian, Qihan Jin, Zhe Wang, Yanzhe Zhang, Peipei Chen, Zhanyuan Cui, Yanjun Wang, Ze Zhang, Yitao Ke, Qingzheng Han, and Longxing Cao. HalluDesign: Protein optimization and de novo design via iterative structure hallucination and sequence design. *bioRxiv*, pp. 2025.11.08.686881, 2025. doi: 10.1101/2025.11.08.686881. 2, 3, 17

Chaitanya Joshi, Arian Jamasb, Ramon Viñas, Charles Harris, Simon Mathis, Alex Morehead, Rishabh Anand, and Pietro Liò. gRNAde: Geometric deep learning for 3d RNA inverse design. *bioRxiv*, pp. 2024.03.31.587283, 2024. doi: 10.1101/2024.03.31.587283. 3, 16

John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, et al. Highly accurate protein structure prediction with alphafold. *nature*, 596(7873):583–589, 2021. doi: 10.1038/s41586-021-03819-2. 1, 3, 16

Rohith Krishna, Jue Wang, Woody Ahern, Pascal Sturmfels, Preetham Venkatesh, Indrek Kalvet, Gyu Rie Lee, Felix S Morey-Burrows, Ivan Anishchenko, Ian R Humphreys, Ryan McHugh, Dionne Vafeados, Xinting Li, George A Sutherland, Andrew Hitchcock, C Neil Hunter, Alex Kang, Evans Brackenbrough, Asim K Bera, Minkyung Baek, Frank DiMaio, and David Baker. Generalized biomolecular modeling and design with RoseTTAFold All-Atom. *Science*, 384(6693):eadl2528, 2024. doi: 10.1126/science.adl2528. 3, 16

Andrew Kubaney, Andrew Favor, Lilian McHugh, Raktim Mitra, Robert Pecoraro, Justas Dauparas, Cameron Glasscock, and David Baker. Rna sequence design and protein–dna specificity prediction with na-mpnn. *bioRxiv*, pp. 2025.10.03.679414, 2025. doi: 10.1101/2025.10.03.679414. 5, 6, 21

Andrew Leaver-Fay, Michael Tyka, Steven M Lewis, Oliver F Lange, James Thompson, Ron Jacak, Kristian Kaufman, P Douglas Renfrew, Colin A Smith, Will Sheffler, Ian W Davis, Seth Cooper, Adrien Treuille, Daniel J Mandell, Florian Richter, Yih-En Andrew Ban, Sarel J Fleishman, Jacob E Corn, David E Kim, Sergey Lyskov, Monica Berrondo, Stuart Mentzer, Zoran Popović, James J Havranek, John Karanicolas, Rhiju Das, Jens Meiler, Tanja Kortemme, Jeffrey J Gray, Brian Kuhlman, David Baker, and Philip Bradley. ROSETTA3: an object-oriented software suite for the simulation and design of macromolecules. *Methods in Enzymology*, 487:545–574, 2011. doi: 10.1016/b978-0-12-381270-4.00019-6. 1, 2, 16

Zeming Lin, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, Nikita Smetanin, Robert Verkuil, Ori Kabeli, Yaniv Shmueli, et al. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637):1123–1130, 2023. doi: 10.1126/science.ade2574. 16

Kresten Lindorff-Larsen, Stefano Piana, Ron O Dror, and David E Shaw. How fast-folding proteins fold. *Science*, 334(6055):517–520, 2011. doi: 10.1126/science.1208351. 2, 16

Xuefeng Liu, Chih-chan Tien, Peng Ding, Songhao Jiang, and Rick L. Stevens. Entropy-reinforced planning with large language models for drug discovery. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24, pp. 31917–31932. JMLR.org, 2024a. URL https://proceedings.mlr.press/v235/liu24be.html. 3, 17

Xuefeng Liu, Mingxuan Cao, Songhao Jiang, Xiao Luo, Xiaotian Duan, Mengdi Wang, Tobin R Sosnick, Jinbo Xu, and Rick Stevens. Monte carlo tree diffusion with multiple experts for protein design. *arXiv preprint arXiv:2509.15796*, 2025. URL https://arxiv.org/abs/2509.15796. 3, 17

Yufeng Liu, Sheng Wang, Jixin Dong, Linghui Chen, Xinyu Wang, Lei Wang, Fudong Li, Chenchen Wang, Jiahai Zhang, Yuzhu Wang, Si Wei, Quan Chen, and Haiyan Liu. De novo protein design with a denoising diffusion network independent of pretrained structure prediction models. *Nature Methods*, 21(11):2107–2116, 2024b. doi: 10.1038/s41592-024-02437-w. 1

Martin Pacesa, Lennart Nickel, Christian Schellhaas, Joseph Schmidt, Ekaterina Pyatova, Lucas Kissling, Patrick Barendse, Jagrity Choudhury, Srajan Kapoor, Ana Alcaraz-Serna, et al. Bindcraft: one-shot design of functional protein binders. *bioRxiv*, pp. 2024.09.30.615802, 2024. doi: 10.1101/2024.09.30.615802. 2, 3, 17

Saro Passaro, Gabriele Corso, Jeremy Wohlwend, Mateo Reveiz, Stephan Thaler, Vignesh Ram Somnath, Noah Getz, Tally Portnoi, Julien Roy, Hannes Stark, David Kwabi-Addo, Dominique Beaini, Tommi Jaakkola, and Regina Barzilay. Boltz-2: Towards accurate and efficient binding affinity prediction. *bioRxiv*, pp. 2025.06.14.659707, 2025. doi: 10.1101/2025.06.14.659707. 5, 21

Jian Peng and Jinbo Xu. Raptorx: exploiting structure information for protein alignment by statistical inference. *Proteins: Structure, Function, and Bioinformatics*, 79(S10):161–171, 2011. doi: 10.1002/prot.23175. 16

Martin L Puterman. *Markov decision processes: Discrete stochastic dynamic programming*. John Wiley & Sons, 2014. URL https://uat.store.wiley.com/en-us/markov-decision-processes-discrete-stochastic-dynamic-programming-p-9780470316887. 3, 34

Carol A Rohl, Charlie EM Strauss, Kira MS Misura, and David Baker. Protein structure prediction using rosetta. In *Methods in enzymology*, volume 383, pp. 66–93. Elsevier, 2004. URL https://forum.rosettacommons.org/node/9346. 2, 16

John SantaLucia, Jr. A unified view of polymer, dumbbell, and oligonucleotide dna nearest-neighbor thermodynamics. *Proceedings of the National Academy of Sciences*, 95(4):1460–1465, 1998. doi: 10.1073/pnas.95.4.1460. 20, 21, 24

David E Shaw, Paul Maragakis, Kresten Lindorff-Larsen, Stefano Piana, Ron O Dror, Michael P Eastwood, Joseph A Bank, John M Jumper, John K Salmon, Yibing Shan, and Willy Wriggers. Atomic-level characterization of the structural dynamics of proteins. *Science*, 330(6002):341–346, 2010. doi: 10.1126/science.1187409. 2, 16

Alexis Vallée-Bélisle, Francesco Ricci, and Kevin W Plaxco. Thermodynamic basis for the optimization of binding-induced biomolecular switches and structure-switching biosensors. *Proceedings of the National Academy of Sciences*, 106(33):13802–13807, 2009. doi: 10.1073/pnas.0904005106. 20

Joseph L Watson, David Juergens, Nathaniel R Bennett, Brian L Trippe, Jason Yim, Helen E Eisenach, Woody Ahern, Andrew J Borst, Robert J Ragotte, Lukas F Milles, Basile I M Wicky, Nikita Hanikel, Samuel J Pellock, Alexis Courbet, William Sheffler, Jue Wang, Preetham Venkatesh, Isaac Sappington, Susana Vázquez Torres, Anna Lauko, Valentin De Bortoli, Emile Mathieu, Sergey Ovchinnikov, Regina Barzilay, Tommi S Jaakkola, Frank DiMaio, Minkyung Baek, and David Baker. De novo design of protein structure and function with RFdiffusion. *Nature*, 620(7976):1089–1100, 2023. doi: 10.1038/s41586-023-06415-8. 1, 3, 16

Odin Zhang, Xujun Zhang, Haitao Lin, Cheng Tan, Qinghan Wang, Yuanle Mo, Qiantai Feng, Gang Du, Yuntao Yu, Zichang Jin, et al. ODesign: A world model for biomolecular interaction design. *arXiv preprint arXiv:2510.22304*, 2025. URL https://arxiv.org/abs/2510.22304. 6, 16

A PER-TARGET MAIN-BENCHMARK RESULTS

We report the per-target values underlying Sec. 4. Search-time and held-out evaluations are shown separately; the dual-expert table also distinguishes unequal retained-output counts.

Target	RNA Len	ODesign@100	RNA-FrameFlow@100	MCTH
7WKP_A→B	22	0.883	0.932	**0.941**
7YEW_A→B	5	0.928	0.935	**0.960**
7YGL_A→B	4	0.918	0.924	**0.963**
7YGL_A→C	4	0.928	0.927	**0.953**
8TG4_A→C	5	0.924	0.925	**0.963**
8GXB_C→B	61	0.835	**0.880**	0.879
9C7A_A→B	8	0.841	0.940	**0.960**
Mean	–	0.894	0.923	**0.946**

(a) Matched-budget Boltz-2 evaluation.

Target	ODesign	RNA-FrameFlow	MCTH
7WKP_A→B	0.800	0.730	**0.940**
7YEW_A→B	0.880	0.890	**0.900**
7YGL_A→B	0.780	0.780	**0.880**
7YGL_A→C	0.780	0.800	**0.870**
8TG4_A→C	**0.930**	0.870	**0.930**
8GXB_C→B	0.770	0.680	**0.850**
9C7A_A→B	**0.830**	0.810	0.820
Mean	0.824	0.794	**0.884**

(b) Held-out AlphaFold3 evaluation.

Table 4: Protein–RNA aptamer design. **(a):** matched 100-candidate evaluation under the Boltz-2 search-time oracle. **(b):** independent AlphaFold3 re-scoring of final designs. Values are ipTM; higher is better.

RNA candidate-pool quality. Table 5 reports average ipTM within each 100-candidate baseline pool, complementing the best-of-pool comparison in Table 4a. Pool averages and selected-design scores measure different properties and are not directly compared as design yield.

	ODesign		RNA-FrameFlow + NA-MPNN	
Target	Pool mean	Best@100	Pool mean	Best@100
7WKP_A→B	.745	.883	.853	.932
7YEW_A→B	.871	.928	.887	.935
7YGL_A→B	.897	.918	.898	.924
7YGL_A→C	.897	.928	.897	.927
8TG4_A→C	.868	.924	.876	.925
8GXB_C→B	.635	.835	.637	.880
9C7A_A→B	.755	.841	.899	.940
Mean	.810	.894	.850	.923

Table 5: RNA baseline pool means and best-of-100 Boltz-2 ipTM, under common refolding.

Target	ODesign@100	MCTH
7XVN_A→N	0.936	**0.952**
7XVN_A→P	**0.957**	0.950
7YSF_A→B	0.890	**0.898**
7YSF_A→C	0.869	**0.876**
7YUK_A→C	0.904	**0.906**
7YUK_A→D	0.901	**0.912**
8PMF_A→B	0.914	**0.956**
8PMF_A→D	0.895	**0.947**
Mean	0.908	**0.925**

(a) Matched-budget Boltz-2 evaluation on the 8 non-degenerate chain-A targets.

Target	BoltzDesign	ODesign	MCTH
7XVN_A→N	0.770	**0.780**	0.570
7XVN_A→P	0.740	0.570	**0.850**
7XVN_B→N	**0.810**	0.560	0.680
7XVN_B→P	0.440	**0.740**	0.670
7YSF_A→B	0.700	0.630	**0.750**
7YSF_A→C	0.410	0.470	**0.680**
7YUK_A→C	**0.810**	0.780	0.630
7YUK_A→D	0.630	0.780	**0.820**
8PMF_A→B	0.700	0.570	**0.800**
8PMF_A→D	0.640	0.560	**0.760**
Mean	0.665	0.644	**0.721**

(b) Held-out AlphaFold3 evaluation on all 10 targets.

Table 6: Protein–DNA aptamer design. **(a):** matched 100-candidate evaluation under the Boltz-2 search-time oracle; the two 7XVN chain-B cases are excluded because Boltz-2 collapses for all compared designs. **(b):** independent AlphaFold3 re-scoring on all 10 targets, under which the two chain-B cases are non-degenerate. Values are ipTM; higher is better. Mean AF3 pLDDT across the same ten targets is 86.0 for MCTH, 83.8 for BoltzDesign, and 82.7 for ODesign.

Target	BoltzDesign1@100	ODesign@100	Complexa@100	MCTH @100
1DJS	0.758	**0.944**	0.907	0.933
1MOX	0.291	0.901	0.845	**0.929**
1XIW	0.884	0.940	**0.946**	**0.946**
2GY7	0.189	0.672	**0.930**	0.862
2IFG	0.527	0.730	0.356	**0.806**
3DI3	0.508	0.901	0.926	**0.947**
3FKD	0.588	0.556	**0.958**	0.686
3MJG	0.697	0.878	0.888	**0.904**
3ZTJ	0.568	0.872	**0.941**	0.914
4O3V	0.778	0.944	0.852	**0.953**
4OGA	0.136	**0.916**	0.904	0.912
5U8R	–	0.532	**0.783**	0.576
Mean	0.539†	0.816	0.853	**0.864**

Table 7: Matched-budget protein-binder optimization on CaoData. All methods use up to 100 folding/evaluator calls per target. ODesign and Complexa candidates are evaluated under the common Boltz-2 refolding protocol, while BoltzDesign1 uses its native Boltz evaluation during optimization. †BoltzDesign1 mean is over 11 targets because 5U8R is unavailable. Values are best ipTM per target; higher is better.

Target	Single-expert MCTH			Dual-expert MCTH			RFdiffusion		
	B2	AF2	Chai	B2	AF2	Chai	B2	AF2	Chai
1DJS	0.89	0.19	0.16	0.86	0.79	**0.73**	0.39	0.28	0.59
1MOX	0.70	0.14	0.15	0.23	0.48	0.15	0.27	0.24	**0.50**
1XIW	0.60	0.39	0.34	**0.95**	**0.86**	**0.84**	0.77	0.84	0.81
2GY7	0.31	0.21	0.17	**0.90**	**0.68**	**0.76**	0.37	0.67	0.73
2IFG	0.72	0.18	0.10	**0.76**	0.15	0.59	0.30	**0.78**	**0.70**
3DI3	**0.90**	0.12	**0.34**	0.76	**0.74**	0.22	0.58	0.62	0.26
3FKD	**0.52**	0.23	0.14	0.43	0.17	0.19	0.32	**0.88**	**0.31**
3MJG	0.50	0.13	0.12	**0.82**	**0.61**	**0.64**	0.47	0.16	0.53
3ZTJ	**0.70**	0.19	0.13	0.67	**0.80**	**0.74**	0.35	0.56	0.66
4O3V	0.93	0.16	0.26	**0.94**	**0.83**	**0.82**	0.68	0.75	0.78
4OGA	0.77	0.11	0.09	**0.87**	0.62	0.63	0.71	**0.66**	**0.69**
Mean	0.69	0.19	0.18	**0.74**	**0.61**	0.57	0.47	0.58	**0.60**

Table 8: Cross-predictor evaluation on the 11 shared CaoData protein-binder targets. Single-expert MCTH uses Boltz-2 during search, whereas dual-expert MCTH uses both Boltz-2 and AlphaFold2; Chai-1 is held out from search. For MCTH, each evaluator column reports the best ipTM over three retained designs; RFdiffusion reports the best over ten designs. Higher is better.

B Related works

Physics-Based and Knowledge-Driven Biomolecular Design. Early approaches to biomolecular design were dominated by physics-based energy modeling and sampling, exemplified by frameworks such as Rosetta and its modern all-atom energy functions (Rohl et al., 2004; Leaver-Fay et al., 2011; Alford et al., 2017). These methods provide explicit physical interpretability and fine-grained energetic control, but suffer from high computational cost and limited scalability when exploring large sequence spaces or multimolecular complexes. Molecular dynamics (MD) provides an alternative physics-first paradigm for atomistic refinement and thermodynamic reasoning (Shaw et al., 2010; Lindorff-Larsen et al., 2011), yet remains computationally costly for design-time exploration at scale, especially for long sequences and heterogeneous complexes. While hybrid approaches combining coarse-grained sampling with heuristic pruning have improved tractability, global exploration of the coupled sequence–structure landscape remains challenging.

Learning-Based Structure Prediction and Generative Design. Learning-based structure modeling (Peng & Xu, 2011; Lin et al., 2023) has fundamentally transformed biomolecular modeling. AlphaFold2 and AlphaFold-Multimer established strong protein(-complex) priors (Jumper et al., 2021; Evans et al., 2021) and RoseTTAFold provided a complementary open predictor (Baek et al., 2021). More recently, generalized all-atom models extend prediction to assemblies containing proteins, nucleic acids, ligands/ions, and covalent modifications, including RoseTTAFold All-Atom (Krishna et al., 2024) and AlphaFold3 (Abramson et al., 2024). Concurrently, ODesign proposes an AlphaFold3-like all-atom "world model" for all-to-all biomolecular interaction design across proteins, nucleic acids, and small molecules, with multi-level controllability via masking and conditional generation (Zhang et al., 2025). These models provide powerful learned structural priors and intrinsic confidence signals, but are primarily trained for prediction rather than optimization; repurposing them for controllable design still requires effective exploration and a principled mechanism for incorporating external constraints.

Generative design with diffusion and inverse folding. Diffusion and flow-matching models enable direct backbone/structure generation, with RFdiffusion providing a general framework for protein generation and binder/motif-conditioned design (Watson et al., 2023). Sequence design models such as ProteinMPNN provide efficient inverse folding for fixed backbones and are widely used as downstream redesign modules (Dauparas et al., 2022). For nucleic acids, emerging geometric generators and inverse-design pipelines include gRNAde (3D RNA inverse design) and RNA-FrameFlow (RNA backbone generation) (Joshi et al., 2024; Anand et al., 2025), highlighting both the promise and the current fragmentation of modality-specific tooling. For protein–ligand settings, diffusion has also

been used to model ligand poses and docking distributions (e.g., DiffDock) (Corso et al., 2023). A common limitation across these approaches is that design quality often depends on large candidate pools plus refolding-based verification, and controllable global exploration remains challenging when objectives are multi-constraint or when the model is used out-of-distribution.

Gradient-based and iterative hallucination methods. A complementary line of work turns pretrained predictors into optimizers via backpropagation through confidence surrogates or via iterative cycling between structure prediction and sequence redesign. BindCraft demonstrates strong empirical performance for protein binder design via AF2-based gradient optimization and filtering (Pacesa et al., 2024), while BoltzDesign1 inverts an all atom predictor (Boltz-1, AF3-like) to optimize distograms for generalized binder design (Cho et al., 2025a). Cycling-based "hallucination" pipelines that repeatedly refold and redesign can avoid fine-tuning and yield designable structures in practice (Anishchenko et al., 2021; Cho et al., 2025b; Fang et al., 2025). However, these methods are typically local: they rely on greedy acceptance, fixed schedules, or initialization-sensitive continuous optimization, and can be computationally expensive or unstable for long sequences and heterogeneous complexes. Moreover, constraint handling is often implicit (through heuristics) rather than explicitly controllable at the algorithmic level.

Search and planning for molecular generation. Search-based methods are well studied in Reinforcement Learning (RL) and combinatorial optimization, but are less explored for continuous, multimodal biomolecular design due to large branching factors, delayed rewards, and expensive evaluations. ERP integrates MCTS with language-model generation for ligands, improving exploration over greedy decoding (Liu et al., 2024a), while MCTD-ME couples masked diffusion with MCTS for protein *sequence* planning and long-range revision (Liu et al., 2025). More broadly, RL-style generative frameworks (e.g., GFlowNets) aim to sample diverse high-reward objects under complex objectives (Bengio et al., 2021). Despite these advances, prior planning-based methods are typically limited to a single modality (often sequences or SMILES) and do not unify all-atom sequence–structure co-design across proteins, nucleic acids, and ligand-containing complexes.

Positioning of MCTH. Relative to (i) physics-first energy optimization pipelines (e.g., Rosetta/MD), (ii) backbone and all-atom generators (e.g., RFdiffusion and ODesign), and (iii) gradient-based or fixed-schedule hallucination/cycling methods (e.g., BindCraft, BoltzDesign), MCTH occupies a distinct point in the design stack: it is a *planning layer* that wraps pretrained folding and inverse-folding experts together with plug-in biophysical models (e.g., energy/geometry feasibility and clash/packing/contact penalties) and allocates expensive inference via MCTS. Our key conceptual distinction is *co-hallucination*: learned experts provide expressive structural proposals, while plug-in biophysical feasibility/energy cues enter the *same* decision loop (selection/expansion/proposal reweighting) as controllable guidance signals, rather than as purely post hoc filters. This separation of concerns—experts for proposal, planning for compute allocation, and physics for controllable feasibility—enables a unified, fine-tuning-free workflow that remains applicable in data-sparse regimes such as nucleic-acid and ligand-conditioned design.

B.1 Positioning Summary

Table 9 summarizes the high-level positioning of MCTH relative to representative method families. The goal of this comparison is not to claim that any individual ingredient is new in isolation, but to clarify the specific combination emphasized by MCTH: planning over explicit sequence–structure states, broad modality coverage, biophysical feasibility inside the decision loop, and no task-specific fine-tuning.

C Additional Methodological Details

C.1 Uncertainty-Aware MCTS over Sequence–Structure States

Following Sec. 2, we represent a biomolecular design hypothesis by a state $s = (S, X)$, where S denotes the current (possibly underspecified) sequence(s) of all designable components and X denotes the corresponding all-atom complex structure under task conditioning C. MCTH casts design as a finite-budget planning problem in which the budget is dominated by expensive folding calls.

Table 9: High-level comparison with related method families. A check mark indicates an explicit/core component, △ indicates partial or task-specific support, and × indicates absence.

Method family / example	Plans over seq.–struct. states	Multi-modality	Biophysics in-loop	Fine-tuning-free
Rosetta / MD-style optimization	×	Limited	✓	✓
RFdiffusion / ODesign	×	△	×	✓
ProteinMPNN / inverse folding	×	Limited	×	✓
BindCraft / BoltzDesign	△	Limited	△	✓
MCTD-ME	✓	Protein only	×	✓
MCTH	✓	✓	✓	✓

Actions and transitions. From state s, an action $a \in \mathcal{A}(s)$ proposes a successor by: (i) proposing an updated design sequence S' for the designable component (e.g., via inverse folding, optionally with lightweight mutations), followed by (ii) invoking a folding/hallucination expert to obtain a corresponding complex structure under task conditioning C (fixed receptor, ligand specification, motif constraints, etc.), yielding $s' = (S', X')$. When multiple folding/hallucination experts are available, we instantiate an ensemble $\{X'^{(e)}\}_{e=1}^{E}$ and use cross-expert agreement as an explicit uncertainty signal.

Node value (reward) from model confidence. Each expanded state is evaluated by a task-aware value function

$$V(s) \;=\; f\big(m_1(s), \ldots, m_J(s)\big), \tag{6}$$

where $\{m_j\}$ are intrinsic signals produced by pretrained predictors, such as global fold confidence (e.g., pLDDT/pTM), interface confidence (e.g., ipTM for binding tasks), and optional geometric or contact-based terms when available. The aggregation function f maps these raw predictor outputs to a scalar state value used for MCTS selection, evaluation, and backup. In our experiments, unless otherwise stated, f is instantiated as a fixed linear aggregation of confidence and interface-quality signals, with weights reported in Appendix C.6. The aggregation is task-aware but not target-tuned: the same weights are used across benchmarks unless a task-specific term, such as a motif constraint, is absent and therefore set to zero. Importantly, $V(s)$ is computed using only forward inference outputs; no gradients, fine-tuning, or backpropagation through structure predictors is used.

Search procedure (uncertainty-aware CUPUCT-MCTS). MCTH performs Monte Carlo Tree Search rollouts under a fixed compute budget (e.g., a maximum number of expensive folding calls). Each rollout consists of four stages: selection, expansion, evaluation, and backpropagation.

Selection. Starting at the root, we recursively select child edges to traverse until reaching a leaf node. We introduce a *consensus–uncertainty–physics* PUCT-style score, C$\mathcal{U}$-PUCT, where the exploration term is modulated by (i) a *consensus prior* $\pi_{\mathrm{cons},\tau}(a \mid s_t)$ and (ii) an *uncertainty–feasibility* factor from inverse/forward-fold experts, with an optional biophysical penalty $E_{\mathrm{phys}}(s_t, a)$. We compute

$$\begin{aligned} C\mathcal{U}\text{-PUCT}((s_t,a)) = Q(s_t,a) + c_p\,\frac{\sqrt{\log N(s_t)}}{1+N(s_t,a)}\cdot \pi_{\mathrm{cons},\tau}(a \mid s_t) \\ \cdot \Big(w_{\mathrm{inv}}\,U_{\mathrm{inv}}(s_t,a) \;+\; w_{\mathrm{fold}}\,U_{\mathrm{fold}}(s_t,a)\Big)\cdot \exp\Big(-\;w_{\mathrm{phys}}\,E_{\mathrm{phys}}(s_t,a)\Big), \end{aligned}$$

and select $a^*(s_t) \;=\; \arg\max_{a\in\mathcal{A}(s_t)} C\mathcal{U}\text{-PUCT}((s_t,a))$. Here Q is the empirical action-value estimate, $N(s_t)$ and $N(s_t,a)$ are node/edge visit counts, and c_p controls the exploration–exploitation trade-off. The terms $\pi_{\mathrm{cons},\tau}$, U_{inv}, U_{fold}, and E_{phys} are computed once at expansion and cached, keeping selection lightweight.

Consensus prior. We define $\pi_{\mathrm{cons},\tau}(a \mid s_t)$ to favor actions whose transition is supported by multiple experts. For a candidate child produced by inverse folding S' and an ensemble $\{X'^{(e)}\}_{e=1}^{E}$, we form an unnormalized consensus score

$$\tilde{\pi}_{\text{cons}}(s_t, a) = \exp\Big(\frac{1}{E}\sum_{e=1}^{E} \log q_\phi(S' \mid X'^{(e)}, C)\Big) \cdot \exp\big(-\kappa\, \mathcal{D}(\{X'^{(e)}\})\big),$$

where q_ϕ is the inverse-folding likelihood and $\mathcal{D}(\{X'^{(e)}\})$ is a cross-expert disagreement measure (e.g., backbone/interface dispersion). We then apply temperature-τ normalization

$$\pi_{\text{cons},\tau}(a \mid s_t) = \frac{\tilde{\pi}_{\text{cons}}(s_t, a)^{1/\tau}}{\sum_{a'} \tilde{\pi}_{\text{cons}}(s_t, a')^{1/\tau}}. \tag{7}$$

This yields an action prior that interpolates between uniform exploration (large τ) and sharp consensus-driven focus (small τ).

Uncertainty terms. We decompose uncertainty into two sources. First, inverse-folding uncertainty

$$U_{\text{inv}}(s_t, a) = \frac{1}{|M|}\sum_{i \in M} \mathsf{H}(q_\phi(\cdot \mid X, C)_i)\,, \tag{8}$$

where $\mathsf{H}(\cdot)$ is categorical entropy and M is the set of updated positions (when masking is disabled, M is the full designable set). Second, folding uncertainty is defined either from confidence-derived uncertainty or from cross-expert structural disagreement:

$$U_{\text{fold}}(s_t, a) = g\big(\text{pLDDT}, \text{pTM}, \text{ipTM}, \dots\big), U_{\text{fold}}(s_t, a) = \mathcal{D}(\{X'^{(e)}\}). \tag{9}$$

Here, $g(\cdot)$ maps folding confidence outputs to an uncertainty bonus. In practice, g is implemented as a monotone decreasing transformation of confidence, so that lower confidence or poorer interface resolution yields larger U_{fold} and encourages exploration of under-resolved states. The alternative form $\mathcal{D}(\{X'^{(e)}\})$ captures epistemic disagreement across folding experts, where larger dispersion among predicted structures indicates greater folding uncertainty.

Finally, we include an optional biophysical term E_{phys} to discourage nonphysical hypotheses. Overall, CUPUCT-MCTS provides an uncertainty-aware planning layer: $\pi_{\text{cons},\tau}$ stabilizes search toward expert-agreed transitions, while U_{inv} and U_{fold} direct exploration toward uncertain or under-resolved regions of the coupled sequence–structure landscape.

Biophysical feasibility term. Finally, we incorporate an optional biophysical feasibility/energy factor $E_{\text{phys}}(s_t, a)$ to down-weight proposals that are high-confidence under learned priors but physically implausible (e.g., clashes, strained geometry, infeasible pockets). We define a modular biophysical score $E_{\text{phys}}(\cdot) \in [0, 1]$ (near 0 if feasible; near 1 if implausible) and set $E_{\text{phys}}(s_t, a) \triangleq E_{\text{phys}}(S', X')$ for the child produced by a, computed from the proposed sequence S' and/or predicted coordinates X' *without additional folding calls*. This term enters C$\mathcal{U}$-PUCT multiplicatively as $\exp(-w_{\text{phys}}\, E_{\text{phys}})$ and is cached at expansion time.

Expansion. Upon reaching a leaf, we expand it by sampling up to K candidate actions (e.g., K sequence proposals) and adding the resulting successor states as children. Each child is constructed by running the inverse-folding expert (to propose S') and then the folding/hallucination expert (to obtain X'), producing a concrete complex hypothesis. At expansion time, we compute and cache $\pi_{\text{cons},\tau}(a \mid s_t)$, $U_{\text{inv}}(s_t, a)$, $U_{\text{fold}}(s_t, a)$, and (when available) $E_{\text{phys}}(s_t, a)$, which are then reused by the selection policy.

Evaluation. Newly added children are scored by computing $V(s')$ from the folding expert's confidence outputs (and any task-specific geometric checks). In our implementation, this evaluation is performed immediately after expansion (i.e., the expensive folding call doubles as the rollout evaluation), keeping the planner modular and plug-and-play.

Backpropagation. The evaluated value is propagated back along the selected path to update visit counts and value statistics. Concretely, we update $N(\cdot)$ and maintain $Q(\cdot)$ via a standard backup rule (running-mean backup; Table 11), enabling the tree policy to increasingly favor high-value branches over repeated simulations.

C.2 Biophysical Penalty

The confidence signals $V(s)$ (such as pLDDT and ipTM) measure how consistent a predicted structure is with the folding expert's learned priors, but they do not capture all physical constraints relevant to

real-world binding or function. Different biomolecular design tasks impose different task-specific physical requirements—thermodynamic stability windows for nucleic acid aptamers, steric feasibility for protein–ligand interfaces, geometric packing quality for de novo protein design—that are invisible to a folding expert evaluating only the bound complex.

In addition to the ML confidence signal, we introduce a modular *biophysical energy term* $E_{\text{phys}}(S') \in [0, 1]$ that encodes task-specific physical feasibility. E_{phys} is near zero for physically plausible candidates and approaches one for implausible ones; it enters the CU-PUCT selection rule as a multiplicative factor $\exp(-w_{\text{phys}}\, E_{\text{phys}})$, acting as a soft preference ($w_{\text{phys}} = 0.1$) rather than a hard filter. The specific form of E_{phys} is defined per task and can be computed from the sequence alone or from predicted coordinates, without requiring an additional structure prediction call.

Example: DNA aptamer design. As a concrete example, consider DNA aptamer design. Self-folding stability can affect the balance between conformational pre-organization and the cost of rearrangement upon binding (Alkhamis et al., 2025; Vallée-Bélisle et al., 2009). We use this trade-off to motivate a soft stability window as a design heuristic, rather than treating either reference as evidence for universal optimal bounds in protein-binding aptamers. Neither failure mode is visible to the folding expert, which only evaluates the bound complex.

For a candidate aptamer sequence S', we define E_{phys} as a soft window penalty on the nearest-neighbour minimum free energy:

$$E_{\text{phys}}(S') \;=\; 1 - \sigma\big(k_l\,(\Delta G_{\text{self}}(S') - \Delta G_l)\big) \\ \sigma\big(-\,k_u\,(\Delta G_{\text{self}}(S') - \Delta G_u)\big). \tag{10}$$

where $\Delta G_{\text{self}}(S')$ is the minimum free energy of the aptamer's intramolecular self-folding, estimated using nearest-neighbour thermodynamic parameters (SantaLucia, 1998) at 37 °C, and $\sigma(x) = 1/(1 + e^{-x})$ is the logistic sigmoid. The product of the two sigmoids forms a soft window over ΔG_{self}: when $\Delta G_l < \Delta G_{\text{self}} < \Delta G_u$, both sigmoids are close to 1 and $E_{\text{phys}} \approx 0$; outside this range one sigmoid drops toward zero, pushing E_{phys} toward 1. The parameters ΔG_l and ΔG_u define the boundaries of the favourable region, and k_l, k_u control how sharply the penalty rises on each side.

The boundary parameters ($\Delta G_l = -10$ kcal mol^{-1}, $\Delta G_u = -2$ kcal mol^{-1}) and steepness ($k_l = k_u = 0.75$) are derived from the empirical distribution of nearest-neighbour self-folding free energies across random DNA sequences (Appendix C.3).

Integration into CU-PUCT selection. This penalty enters the CU-PUCT selection rule as a multiplicative factor $\exp\big(-w_{\text{phys}}\, E_{\text{phys}}\big)$, which reduces the UCB score of candidates at either extreme. The weight $w_{\text{phys}} = 0.1$ makes the penalty a soft preference rather than a hard filter. Like the consensus prior and uncertainty bonuses, E_{phys} is computed once when a node is expanded and cached for reuse during selection, adding no extra cost to the selection phase. ΔG_{self} is computed from the sequence alone, without requiring an additional structure prediction call.

For aptamer types where nearest-neighbour thermodynamic models do not apply (e.g., G-quadruplex sequences governed by Hoogsteen base-pairing), we use the constant $E_{\text{phys}} = 0.5$. This introduces no sequence-dependent thermodynamic preference within those cases, although it rescales the exploration bonus by $\exp(-0.5 w_{\text{phys}})$.

Overall, the exploration term is large for actions that are (i) plausible under expert consensus, (ii) informative due to residual sequence/structure uncertainty, and (iii) physically reasonable under task-specific biophysical constraints, while being down-weighted for candidates at either biophysical extreme.

C.3 Parameter derivation.

We use the same calibration principle for all parameters: each sigmoid should reach its sensitive region (input $\approx \pm 3$) at the most extreme ΔG_{self} values observed in practice. To ground this in data, we computed ΔG_{self} for a sample of random DNA sequences across the 20–80 nt range using nearest-neighbour thermodynamics (`seqfold`, SantaLucia 1998, 37 °C). The observed distribution spans approximately -14 to $+2$ kcal mol^{-1}, with a median near -2.5 kcal mol^{-1}.

Setting	Designable component	Inverse-folding expert	Typical biophysics cue
Protein/peptide design	protein/peptide sequence	ProteinMPNN	clash/packing/contact
Ligand-conditioned protein design	pocketed protein sequence	LigandMPNN	sterics / pose geometry
RNA/DNA aptamer design	NA sequence	NA-MPNN	ΔG_{self}, base-pair potentials
Hybrid complexes	per-chain (as applicable)	dispatched	task-specific

Table 10: **Modality-aware inverse-folding dispatch.** The folding expert (Boltz-2) is shared across settings via a unified all-atom representation; modality differences enter primarily through the inverse-folding backend and optional plug-in biophysical feasibility cues.

For the upper boundary, we set $\Delta G_u = -2\ \text{kcal mol}^{-1}$. This is a heuristic boundary for the chosen calibration distribution, not a universal minimum energy needed to maintain secondary structure. Nearest-neighbor parameters depend on sequence and physical conditions (SantaLucia, 1998).

For the lower boundary, we set $\Delta G_l = -10\ \text{kcal mol}^{-1}$, corresponding to the 1st–5th percentile of the observed distribution. The lower boundary discourages strongly self-stabilized sequences in this calibration set; it is not a measured affinity threshold.

The steepness parameters follow from the same calibration. On the over-stability side, the observed extreme $\Delta G_{\min} \approx -14\ \text{kcal mol}^{-1}$ lies $4\ \text{kcal mol}^{-1}$ below ΔG_l, giving $k_l = 3\,/\,4 = 0.75$. On the under-structure side, the observed extreme $\Delta G_{\max} \approx +2\ \text{kcal mol}^{-1}$ lies $4\ \text{kcal mol}^{-1}$ above ΔG_u, giving $k_u = 3\,/\,4 = 0.75$. The two sides yield the same steepness, so we use a single $k_l = k_u = 0.75$; the transition width of $4.4\,/\,0.75 \approx 6\ \text{kcal mol}^{-1}$ is intentionally gradual, reflecting uncertainty in the exact threshold values.

C.4 Additional details: multi-modality representation and expert dispatch

MCTH operates across diverse biomolecular modalities—proteins, peptides, DNA/RNA (single- and double-stranded), small molecules, ions, and post-translational modifications—without modality-specific fine-tuning or architectural changes. This generality follows from two complementary design choices: *(i)* a unified all-atom representation in the folding expert, and *(ii)* a modular, modality-aware dispatch in the inverse-folding step. Across all settings, the planner state remains $s = (S, X)$ under conditioning C (partners, ligands, motifs, etc.), and the MCTS logic (selection/expansion/caching/backup) is unchanged.

Unified all-atom tokenization. We use Boltz-2 as the folding/refolding expert, whose unified all-atom representation supports proteins, nucleic acids, small molecules, ions, and covalent modifications within a single framework (Passaro et al., 2025). As a result, MCTH can reuse this representation across tasks, but usable design transitions still require compatible inverse-folding experts and task-specific scoring or feasibility terms.

Modality-aware expert dispatch. Sequence redesign is implemented as a lightweight dispatch based on the design target type. Concretely, the planner invokes an appropriate inverse-folding expert for the currently designable component (e.g., ProteinMPNN for proteins and peptides (Dauparas et al., 2022), NA-MPNN for nucleic acids (Kubaney et al., 2025), and LigandMPNN for ligand-pocket-aware protein redesign). The state representation, tree policy, and value computation remain fixed; only the inverse-folding call differs. Table 10 summarizes the dispatch and typical plug-in biophysical cues used for controllability.

C.5 Inference-time structural guidance for nucleic acids

Sequence-level biophysical signals—such as self-folding free energy or base-pair assignments—capture important physical constraints, but they do not directly influence the structure generation process. A more direct form of biophysical control is to inject physical constraints *inside* the structure generation itself, steering the diffusion model's denoising trajectory toward conformations that are consistent with the sequence's known secondary-structure preferences.

For a candidate sequence S' with self-folding stability in the thermodynamically favourable range ($\Delta G_l \leq \Delta G_{\text{self}}(S') \leq \Delta G_u$; boundaries derived in Appendix C.3), we derive an inference-time structural guidance signal from the `seqfold` computation.

`seqfold` outputs. The `seqfold` computation for a candidate sequence S' produces two outputs from a sequence-level computation:

1. $\Delta G_{\text{self}}(S')$ — the minimum free energy of intramolecular self-folding.
2. $P(S')$ — the MFE base-pair assignment, i.e., the set of intramolecular base pairs predicted to form in the free aptamer.

This section uses $P(S')$ as a *contact guidance potential* inside Boltz-2's structure prediction (described below). The base-pair assignment reuses the same `seqfold` computation that produces ΔG_{self}; evaluating and applying the contact potential may still add overhead during folding.

Inference-time structural guidance via Boltz-2 potentials. We activate Boltz-2's inference-time potential-energy steering (`use_potentials = True`), which steers the denoising trajectory via gradient-based $\hat{x}_0$ correction and Feynman-Kac particle resampling (Del Moral et al., 2006) entirely within the forward pass—no backpropagation through the diffusion model is required.

Within this steering framework, we inject $P(S')$ as a *contact guidance potential*: aptamer positions I that form contacts with the protein interface in the current hallucinated complex X' are identified, and base pairs in $P(S')$ whose both endpoints lie *outside* I (non-interface pairs) are used as soft structural constraints. This encourages the aptamer scaffold to retain its pre-organized secondary structure away from the binding site, while bases at the interface ($\in I$) are left unconstrained so they remain available for protein contact. The contact weight is higher during the high-noise structural assembly phase and disabled near convergence, following Boltz-2's default schedule.

Formally, let $\mathcal{P}_{\text{free}} = \{(i,j) \in P(S') : i \notin I,\, j \notin I\}$ denote the set of non-interface base pairs predicted by `seqfold`. The guidance potential is

$$U_{\text{contact}}(X) \;=\; \sum_{(i,j)\,\in\,\mathcal{P}_{\text{free}}} w_{ij}\,\big(d_{ij}(X) - d_0\big)^2, \tag{11}$$

where $d_{ij}(X)$ is the distance between the heavy atoms of positions i and j in structure X, d_0 is the target contact distance for a Watson-Crick base pair, and w_{ij} is a weight that decays with diffusion noise level. This potential is incorporated into the Boltz-2 denoising step via the $\hat{x}_0$-correction mechanism, yielding a guided reverse transition.

C.6 Reproducibility and Hyperparameters

Table 11 summarizes the audited configuration used for the main experiments. Unless otherwise stated, the same configuration is used across RNA aptamer, DNA aptamer, protein–protein binder, and ligand-conditioned design tasks. We do not tune search hyperparameters per dataset; modality differences enter only through the inverse-folding backend and the task conditioning passed to the folding expert.

For each target, we report the design with the highest ipTM among all states evaluated during search, including both internal and leaf states. This candidate-selection rule is fixed across all reported MCTH runs.

C.7 Sequence-Level Biophysical Refinement

Biophysical signals in MCTH can influence search not only through the state value E_{phys} in Sec. 3.2, but also through optional refinement operators that modify candidate states before subsequent refolding. These refinements are task-specific and may act on sequence, structure, or both, while leaving the underlying MCTS procedure unchanged. For DNA aptamer design, we instantiate sequence-level refinement using nearest-neighbour self-folding thermodynamics:

Table 11: Main hyperparameters and configuration.

Component	Parameter	Value
Tree search	Fold-call budget T	100
	Expansion width K	4
	Maximum tree depth $D_{\max}$	5
	Maximum MCTS iterations	50
	Fold calls per iteration	8
	Early-stop patience	10 iterations without improvement
	Exploration constant c_p	$\sqrt{2} \approx 1.414$
	Backup rule	sum-backup, $Q = \texttt{total_reward}/\texttt{visits}$
	Main-table seed	42
	Ablation seeds	$\{11, 23, 47\}$
Value function	w_{conf}	1.0
	w_{iface}	0.5
	w_{motif}	5.0 when motif constraints are specified; 0 otherwise
Folding expert	Model	Boltz-2
	Recycling steps	3
	Diffusion denoising steps	200
	Sampling steps	200
	Diffusion samples per call	1
Inverse-folding experts	Protein / peptide	ProteinMPNN
	Nucleic acid (RNA / DNA)	NA-MPNN
	Ligand-pocket-aware protein	LigandMPNN
	Sampling temperature τ	0.1
	Samples per inverse-fold call	4
Compute	GPU	NVIDIA A100 80GB
	Per target / seed	1 GPU

Identifying the failure mode. For each candidate (S', X') output by MCTS, we compute $\Delta G_{\text{self}}(S')$ using `seqfold` and classify it into three regimes:

- **Over-stable** ($\Delta G_{\text{self}} < \Delta G_l$): strong self-structure implies an unfolding penalty before binding.
- **Favourable** ($\Delta G_l \leq \Delta G_{\text{self}} \leq \Delta G_u$): no refinement needed.
- **Under-structured** ($\Delta G_{\text{self}} > \Delta G_u$): weak pre-organisation implies a high entropic cost upon binding.

Only over-stable or under-structured candidates are refined. The boundaries $(\Delta G_l, \Delta G_u)$ are calibrated from an empirical distribution of random DNA sequences (Appendix C.3).

Interface-aware mutation strategy. To avoid disrupting the binding surface, mutations are restricted to *non-interface* positions. We define an interface set $I \subset \{1, \ldots, |S'|\}$ as positions whose predicted contacts with the target in X' exceed a threshold (e.g., heavy-atom distance < 8 Å), and restrict edits to $j \notin I$.

Over-stable correction. We compute an MFE base-pair assignment $P(S')$ from `seqfold`, select a non-interface position j^* with high base-pairing propensity, and apply a point mutation $S'_{j^*} \to \tilde{s}$ chosen to locally destabilize the stem (e.g., introduce a mismatch/bulge). We iterate until $\Delta G_{\text{self}} \geq \Delta G_l$ or a budget of $T_{\max}$ mutations is reached.

Under-structured correction. We identify a contiguous non-interface window (default length $w = 6$) with low base-pairing propensity and propose a stabilizing substitution that forms a short hairpin (e.g., a GC-rich stem with a stable tetraloop motif (SantaLucia, 1998)), accepting the edit if it achieves $\Delta G_{\text{self}} \leq \Delta G_u$.

Acceptance criterion. We accept a refined sequence S'' iff it reduces the biophysical penalty:

$$E_{\text{phys}}(S'') < E_{\text{phys}}(S'). \tag{12}$$

Otherwise, we retain the original candidate (S', X').

C.8 Optional Structural Guidance Interface

Beyond value shaping and sequence refinement, biophysical information can optionally enter a compatible folding model directly during inference. This provides a third control interface in MCTH: rather than only scoring or editing a candidate, structural priors can steer the generation of its corresponding conformation. This mechanism is backend-dependent and is not required by the MCTS planner.

For a candidate aptamer sequence S', we run `seqfold` to obtain $\Delta G_{\text{self}}(S')$ and an MFE base-pair set $P(S')$; when $\Delta G_l \leq \Delta G_{\text{self}}(S') \leq \Delta G_u$ (boundaries in Appendix C.3), we enable Boltz-2 inference-time potentials (`use_potentials=True`) and use $P(S')$ as a soft contact prior. Specifically, we first identify the interface set I from the current hallucinated complex X' (positions contacting the protein by a distance cutoff). We then retain only non-interface base pairs $\mathcal{P}_{\text{free}} = \{(i,j) \in P(S') : \; i \notin I, \; j \notin I\}$ and impose the quadratic contact potential

$$U_{\text{contact}}(X) \; = \sum_{(i,j)\in\mathcal{P}_{\text{free}}} w_{ij} \left(d_{ij}(X) - d_0\right)^2, \tag{13}$$

where $d_{ij}(X)$ is the heavy-atom distance for (i,j) in structure X, d_0 is a target base-pair distance, and w_{ij} can decay with diffusion noise level. Boltz-2 incorporates U_{contact} via its potential-energy steering (e.g., $\hat{x}_0$ correction / particle resampling (Del Moral et al., 2006)), requiring no backpropagation through the diffusion model. This preserves secondary structure away from the binding site while leaving interface bases unconstrained for target binding. Full implementation details (interface identification, contact scheduling, and potential weights) are provided in Appendix C.5.

D Additional Experiments

This supplement reports additional evaluations showing that the same MCTH pipeline can be instantiated across modalities and constraint types beyond the three main-text benchmarks (Sec. 4).

Target	RFDiff-AA	MCTH	ODesign
5SDV_IAI_A	0.932	**0.949**	0.691
5SDV_IAI_B	**0.959**	0.939	0.516
5SDV_IAI_C	**0.924**	0.895	0.269
5SDV_IAI_D	**0.961**	0.902	0.449
7BKC_FAD_A	0.465	**0.848**	0.200
7BKC_FAD_a	0.878	**0.919**	0.291
7BKC_FAD_E	**0.839**	0.833	0.410
7BKC_FAD_e	0.720	**0.846**	0.510
7C7M_SAM_A	0.928	**0.940**	0.847
7V11_OQO_A	0.897	**0.938**	0.413
Mean	0.850	**0.901**	0.460

Table 12: **Protein–ligand binding under Boltz-2 refold evaluation.** Best ipTM per target (higher is better). Bold indicates the better of MCTH and RFDiff-AA for each target; MCTH attains a higher mean ipTM.

For concision, we currently include only protein–ligand binding and the corresponding inverse-folding ablation (Appendix E); the inactive source archive is retained below.

D.1 Protein–Ligand Binding Design

We evaluate MCTH on protein–small-molecule binding design, where the objective is to generate a protein sequence that forms a confident complex with a specified ligand.

Benchmark. We construct a 10-target ligand-binding benchmark from four PDB complexes: 5SDV (IAI; chains A/B/C/D), 7BKC (FAD; chains A/a/E/e), 7C7M (SAM; chain A), and 7V11 (OQO; chain A). Each target specifies a fixed ligand (CCD code) and a binding protein chain.

Setup. MCTH instantiates MCTS with Boltz-2 folding in the loop and **LigandMPNN** as the inverse-folding proposal model. Starting from random sequences, the search iterates folding, scoring with an ipTM-dominant value function, and MCTS expansion with ligand-aware inverse-fold proposals. We use a fixed budget of up to 100 Boltz-2 fold calls per target.

Baselines and evaluation. We compare against (i) RFdiffusion-AA + LigandMPNN and (ii) ODesign (rigid ligand checkpoint). All designs are evaluated under a common Boltz-2 re-fold protocol. We report **ipTM** (higher is better). A specialist comparison to BoltzDesign1 is provided as a set-aside table below.

Results. MCTH attains a higher mean ipTM (0.901) than RFdiffusion-AA (0.850) and ODesign (0.460), with the main gap concentrated on the most challenging FAD cofactor targets (higher pocket-precision demands). Ablations are in Appendix E.

E Additional Ablations

E.1 Ligand-aware vs. ligand-blind inverse folding

To isolate the effect of ligand-aware sequence redesign, we compare MCTH under two inverse-folding backends: (i) **LigandMPNN** (ligand-aware; conditions on ligand atoms in the binding pocket), and (ii) **ProteinMPNN** (ligand-blind; conditions only on protein backbone atoms). All other components are held fixed: identical MCTS configuration, identical ipTM-dominant scoring for ligand tasks, and the same Boltz-2 evaluation protocol as in Appendix D.1. We report best-of-N ipTM (higher is better) and iPDE (lower is better).

	ipTM ↑		iPDE ↓	
Ligand	LigandMPNN	ProteinMPNN	LigandMPNN	ProteinMPNN
IAI (4 targets)	**0.921**	0.900	1.60	**1.18**
FAD (4 targets)	**0.862**	0.730	**5.46**	7.98
SAM (1 target)	**0.940**	0.818	**0.92**	4.55
OQO (1 target)	**0.938**	0.924	1.02	**0.84**

Table 13: **Ablation by ligand type.** Ligand-aware inverse folding yields the largest improvements on FAD and SAM, where pocket complementarity is most critical.

	ipTM ↑		iPDE ↓	
Target	LigandMPNN	ProteinMPNN	LigandMPNN	ProteinMPNN
5SDV_IAI_A	0.949	0.889	2.24	1.25
5SDV_IAI_B	0.939	0.921	1.28	1.00
5SDV_IAI_C	0.895	0.876	1.65	1.42
5SDV_IAI_D	0.902	0.914	1.23	1.04
7BKC_FAD_A	0.848	0.733	7.69	9.49
7BKC_FAD_a	0.919	0.737	5.38	8.46
7BKC_FAD_E	0.833	0.753	6.60	7.75
7BKC_FAD_e	0.846	0.695	2.17	6.22
7C7M_SAM_A	0.940	0.818	0.92	4.55
7V11_OQO_A	0.938	0.924	1.02	0.84

Table 14: **Per-target ablation results.** Ligand-aware inverse folding improves both ipTM and iPDE on most targets, with the strongest gains on FAD and SAM.

Summary. Ligand-aware inverse folding substantially improves both interface confidence and interface precision: LigandMPNN improves mean ipTM from 0.826 to 0.901 and reduces mean iPDE from 4.20 to 3.02.

Per-target results.

E.2 Evaluation Protocol and Cross-Oracle Validation

We distinguish between *search-time* and *held-out* evaluation.

Search-time oracle evaluation. For each MCTH instantiation, candidate states are scored using the folding-model signals available to the search procedure. The modality-spanning main experiments use Boltz-2 as the search-time folding oracle. For comparisons that can be re-scored consistently, including ODesign, RFdiffusion, RFdiffusion-AA, and Complexa, generated sequences are evaluated using the same Boltz-2 refolding protocol. BoltzDesign1 instead uses its native Boltz evaluation during optimization; we therefore report it separately while matching the number of folding/evaluator calls where applicable.

These scores measure optimization under a common computational proxy and should not be interpreted as oracle-independent evidence of physical binding.

Matched-budget evaluation. To separate search quality from raw access to the evaluator, we match candidate or folding-call budgets whenever possible. In the main benchmarks, ODesign, RNA-FrameFlow, Complexa, and BoltzDesign1 are compared at up to 100 candidate or folding/evaluator calls where applicable. Appendix E.3 further compares MCTH against exploration-off search, greedy cycling, and best-of-N sampling under identical initialization, proposal models, folding oracle, and evaluation budget.

Held-out predictor evaluation. We additionally evaluate designs with structure predictors that are not used to guide the corresponding search. AlphaFold3 is used to re-score the RNA and DNA

aptamer benchmarks. For the protein-binder dual-expert experiment, Boltz-2 and AlphaFold2 are used during search, while Chai-1 is held out entirely from proposal, tree selection, and output ranking. These evaluations test whether improvements transfer beyond the optimized predictor rather than merely reproducing its scoring preferences.

Oracle-independent physical evaluation. For protein binders, we additionally use Rosetta InterfaceAnalyzer as a non-search evaluation of interface geometry and energetics. We report size-normalized interface energy together with packing and buried unsatisfied hydrogen-bond statistics. These metrics provide an additional physical-plausibility check but are not treated as experimental affinity measurements.

Final-design pass rate. Best-per-target ipTM measures whether a method can discover a strong candidate but does not characterize design yield. We therefore also compare final protein-binder archives under a common held-out binder-quality filter. Candidates are selected without access to held-out metrics, and pass rates are computed only after final archive construction.

E.3 Matched-Budget Search Ablations

To isolate the effect of search-guided compute allocation, we run matched-budget ablations under the same 100 folding-call budget. For each target, all variants use identical initialization, evaluation, scoring, folding oracle, and inverse-folding backend. The only difference is the planning strategy.

We compare the following variants:

- **MCTH (full):** tree search with exploration.
- **Exploration-off** ($c_p = 0$): the same tree search without the exploration term, corresponding to pure exploitation.
- **Greedy cycling:** sequential inverse-folding $\rightarrow$ folding refinement along a single trajectory, without branching.
- **Best-of-N:** independent sampling under the same fold-call budget, without feedback or iterative refinement.

Greedy cycling isolates the effect of iterative refinement without branching, while best-of-N isolates pure sampling from strong pretrained proposal models without feedback. Together, these baselines test whether MCTH's gains come from search-guided compute allocation rather than refinement or sampling alone.

Table 15: Matched-budget ablation on RNA aptamer design. All values are mean $\pm$ standard deviation over three seeds.

Variant	7WKP A$\rightarrow$B	7YEW A$\rightarrow$B	8GXB C$\rightarrow$B	Mean	Δ vs. Full
MCTH (full)	**0.902$\pm$0.010**	**0.979$\pm$0.002**	**0.858$\pm$0.006**	**0.913**	0.000
Exploration-off ($c_p = 0$)	0.756$\pm$0.007	0.972$\pm$0.002	0.791$\pm$0.040	0.839	-0.073
Greedy cycling	0.750$\pm$0.004	0.957$\pm$0.003	0.646$\pm$0.012	0.784	-0.129
Best-of-N random	0.752$\pm$0.030	0.953$\pm$0.001	0.629$\pm$0.010	0.778	-0.135

Table 16: Matched-budget ablation on DNA aptamer design. All values are mean $\pm$ standard deviation over three seeds.

Variant	7XVN A_N	7YUK A_C	8PMF A_B	Mean	Δ vs. Full
MCTH (full)	**0.955$\pm$0.002**	**0.946$\pm$0.008**	**0.964$\pm$0.005**	**0.955**	0.000
Exploration-off ($c_p = 0$)	0.909$\pm$0.011	0.904$\pm$0.009	0.930$\pm$0.009	0.914	-0.041
Greedy cycling	0.756$\pm$0.014	0.909$\pm$0.018	0.819$\pm$0.006	0.828	-0.127
Best-of-N random	0.753$\pm$0.010	0.900$\pm$0.006	0.834$\pm$0.005	0.829	-0.126

Across all targets, MCTH achieves the highest final quality, with especially large gains on RNA and DNA tasks. This suggests that the improvement comes from search-guided compute allocation rather than simply from iterative refinement or a larger independent candidate pool.

Table 17: Matched-budget ablation on protein–protein binder design. All values are mean ± standard deviation over three seeds.

Variant	3DI3	4O3V	Mean	Δ vs. Full
MCTH (full)	**0.912±0.002**	**0.964±0.002**	**0.938**	0.000
Exploration-off ($c_p = 0$)	0.897±0.008	0.954±0.016	0.925	-0.013
Greedy cycling	0.901±0.023	0.955±0.013	0.928	-0.011
Best-of-N random	0.803±0.005	0.900±0.008	0.851	-0.087

Table 18: Summary of matched-budget ablations across all targets.

Variant	Overall mean	Protein mean	RNA mean	DNA mean
MCTH (full)	**0.935**	**0.938**	**0.913**	**0.955**
Exploration-off ($c_p = 0$)	0.889	0.925	0.839	0.914
Greedy cycling	0.836	0.928	0.784	0.828
Best-of-N random	0.815	0.851	0.778	0.829

E.4 Effective Convergence Diagnostic

Final performance alone does not reveal whether additional folding calls continue to produce useful search progress. We therefore measure the last folding call at which the incumbent best score improves by more than $\epsilon = 0.002$. Runs with no improvement above this threshold are assigned a last-improvement call of zero and counted as immediately stagnant.

MCTH continues to extract useful improvements later in the evaluation budget: every MCTH run improves beyond the threshold, with the last improvement occurring at fold call 46.6 on average. In contrast, 44–62% of the non-adaptive or reduced-search runs never make a nontrivial improvement after initialization. Together with the matched-budget results in Appendix E.3, this indicates that the advantage of MCTH is not simply additional iterative refinement; tree search changes how useful evaluations are allocated over the course of optimization.

E.5 Cross-Oracle Agreement Analysis

Held-out re-scoring tests whether improvements under the search-time oracle transfer to an independent structure predictor. We further quantify this relationship at the *design level*: given a fixed target, do Boltz-2 and the held-out predictor rank candidate designs similarly?

Design-level agreement. We compute within-target Spearman correlation between Boltz-2 and AlphaFold3 scores and bootstrap over targets to obtain 95% confidence intervals. This within-target analysis removes between-target difficulty effects that can otherwise inflate pooled correlations.

Agreement is moderate for DNA aptamers, weak but positive for protein binders, and not distinguishable from zero for RNA aptamers. Importantly, the pooled raw ipTM correlation across all designs is much larger ($\rho = 0.670$) than the within-target correlation ($\rho = 0.261$). The pooled value is therefore driven substantially by target-level difficulty that both predictors capture, rather than by agreement over which design is best for a fixed target.

Global folding versus interface ranking. The disagreement is specific to interface-level discrimination rather than global folding quality. For protein binders ($n = 255$), within-target agreement in pLDDT is $\rho = 0.762$ (95% CI $[0.64,\ 0.85]$), compared with only $\rho = 0.252$ (95% CI $[0.05,\ 0.46]$) for ipTM. The non-overlapping confidence intervals indicate that the two predictors agree substantially more on whether a complex is globally well folded than on the relative quality of its binding interface. This distinction is important for binder and aptamer design, where interface discrimination is the primary optimization objective.

Use as a screening proxy. We next ask whether Boltz-2 can serve as a practical screening proxy for selecting designs that score highly under the held-out oracle. For each target, we measure whether

Variant	Final mean	Never improve	Avg. last improvement
MCTH (full)	**0.935**	**0.0%**	**46.6**
Exploration-off ($c_p = 0$)	0.889	61.9%	29.3
Greedy cycling	0.836	53.8%	23.5
Best-of-N	0.815	44.4%	33.3

Table 19: Effective convergence at $\epsilon = 0.002$ under the matched folding-call budget. "Never improve" denotes runs with no improvement above ϵ after initialization.

Modality	n designs	Within-target ρ	95% CI
DNA aptamer	50	0.487	$[0.29,\ 0.67]$
Protein binder	255	0.252	$[0.05,\ 0.46]$
RNA aptamer	77	0.173	$[-0.06,\ 0.36]$

Table 20: Design-level agreement between Boltz-2 and AlphaFold3 interface scores. Correlations are computed within target and confidence intervals are obtained by bootstrap over targets.

the Boltz-selected top design falls in the held-out top set (top-1 hit) and the precision of the Boltz top quartile with respect to the held-out top quartile.

Boltz-2 is a useful but imperfect proxy for DNA, where both top-1 recovery and top-quartile precision are approximately three times their random baselines. For protein binders, screening performance is above random but substantially weaker. For RNA, the observed top-quartile precision does not exceed random expectation.

RNA score saturation. The weak RNA agreement is partly explained by restricted score dynamic range rather than clear opposite rankings. Across the RNA candidates, the ipTM interquartile range is only 0.080 under AlphaFold3 and 0.022 under Boltz-2; the latter has a median ipTM of approximately 0.946. Thus, many RNA designs lie near the Boltz-2 score ceiling, leaving little resolution for within-target ranking.

Method-level rankings. At the method level, the two predictors preserve the broad direction of performance but are not equivalent. Kendall's τ between method rankings is 0.333 for DNA (10 targets), 0.738 for protein binders (5 common targets), and 0.333 for RNA (7 targets). These results motivate reporting held-out evaluation alongside the search-time oracle rather than treating either predictor as a ground-truth interface metric.

Exploratory check for search-oracle preference. Finally, we test whether methods that optimize Boltz-derived feedback tend to receive relatively better rankings under Boltz-2 than under AlphaFold3. Across target–method pairs, Boltz-based systems shift by $+0.367$ rank positions on average (positive means relatively favored by Boltz-2), whereas non-Boltz systems shift by -0.353, a difference of 0.720 rank positions. A 20,000-permutation test that shuffles the Boltz/non-Boltz label within modality gives $p = 0.067$. We therefore treat this as a suggestive diagnostic rather than evidence of a systematic evaluator bias. It nevertheless reinforces the need for the held-out evaluations used throughout Sec. 4.

E.6 Dual-Expert Search and Final-Archive Yield

We provide additional implementation and archive-level evaluation details for the dual-expert protein-binder experiment in Sec. 4.2. The purpose of this experiment is to test whether the multi-expert mechanism defined in Sec. 3.2 improves robustness beyond optimization against a single folding predictor.

Dual-expert configuration. We retain the same MCTS procedure and ProteinMPNN sequence-proposal mechanism used in the single-expert protein-binder setting. The difference is that each candidate sequence is evaluated with both Boltz-2 and AlphaFold2. Their structural-quality signals

Modality	Top-1 hit	Random	Prec.@25%	Random
DNA aptamer	0.600	0.200	0.600	0.200
Protein binder	0.167	0.053	0.427	0.241
RNA aptamer	0.286	0.091	0.143	0.273

Table 21: Screening utility of Boltz-2 for selecting designs favored by the held-out predictor. Random columns give the corresponding chance-level expectation under the same target-level candidate-set sizes.

contribute to node evaluation, while cross-expert agreement, disagreement, and uncertainty enter the CU-PUCT search rule. No component model is fine-tuned or differentiated through.

Each Boltz-2 or AlphaFold2 structure-prediction invocation is counted as one folding/evaluator call. Thus, evaluating one sequence with both experts consumes two calls from the fixed inference budget. This accounting prevents the dual-expert variant from receiving uncounted additional structure-prediction compute.

Chai-1 is not used for sequence proposal, node evaluation, tree selection, backup, or archive ranking. It is run only after search and therefore serves as a held-out predictor.

Cross-predictor transfer. Across the 11 CaoData targets shared by the single-expert, dual-expert, and RFdiffusion evaluations, the original Boltz-only MCTH obtains mean best ipTM values of $0.69/0.19/0.18$ under Boltz-2/AF2/Chai-1, respectively. The dual-expert variant obtains $0.74/0.61/0.57$. Thus, introducing AF2 as a second search-time expert substantially improves performance under AF2 while maintaining Boltz-2 performance, and the improvement transfers to the held-out Chai-1 evaluator. Full per-target results are reported in Table 8.

For MCTH, the evaluator-wise values in Table 8 are maxima over three retained final designs. Because the maximum is taken independently for each evaluator, the Boltz-2, AF2, and Chai-1 values for a target need not correspond to the same sequence. RFdiffusion values are maxima over ten outputs, so we do not interpret the unequal-pool best-of-output comparison as a direct design-yield comparison.

Final-archive construction. To measure yield beyond a single best candidate, we reconstruct a reward-ranked archive from the dual-expert search logs. Within each target, candidates from the available dual-expert runs are pooled and deduplicated by exact binder amino-acid sequence identity after removing whitespace and converting sequences to uppercase. Candidate ranking uses only the joint reward recorded during search; no held-out Chai-1 quantity is used for retrospective selection.

We report the top-K unique archive for $K \in \{5, 10, 25\}$. Intermediate MCTS nodes are optimization states rather than independently submitted final designs and are therefore not used as the denominator of the final-design pass rate.

Held-out Chai-1 pass criterion. After archive construction, each candidate is independently evaluated with Chai-1. We define a design as passing when Chai-1 interface confidence satisfies $\mathrm{ipTM}_{\mathrm{Chai}} \geq 0.50$, global confidence satisfies $\mathrm{pTM}_{\mathrm{Chai}} \geq 0.50$, and its interchain-clash flag is false. These are the computational criteria used in the archive evaluation. Chai-1 is never used during sequence proposal, MCTS selection, reward computation, or archive ranking, so the pass criterion is fully out of the search loop. All designs in the reported archives have valid Chai-1 evaluations; missing or failed evaluations, if present, are excluded from the valid denominator and reported separately.

Native Complexa retained archive. On the 11 shared targets, Complexa's native retained archive contains 16 passing designs among 26 unique outputs (61.5%), with at least one pass on 8/11 targets. The reward-ranked MCTH top-10 archive contains 41/110 passes (37.3%) with the same target coverage. Complexa contributes 1–3 unique final outputs per target after native selection and exact-sequence deduplication; MCTH contributes a fixed number of unique outputs ranked by its in-loop reward. These fractions describe conditional archive precision under the respective retention policies, rather than a matched generation-yield ranking.

Archive	Passing/valid	Pass rate	Targets ≥ 1 pass
Dual-expert MCTH, top-5	27/55	49.1%	8/11
Dual-expert MCTH, top-10	41/110	37.3%	8/11
Dual-expert MCTH, top-25	83/275	30.2%	8/11
Complexa native retained archive	16/26	61.5%	8/11

Table 22: Held-out Chai-1 archive precision on 11 shared targets. MCTH uses fixed-size top-K archives ranked by in-loop joint reward; Complexa uses its native retained archive of 1–3 unique final outputs per target. Both use the stated Chai-1 pass criterion. The different retention policies define a descriptive archive-level comparison.

Method	Outputs/target	Passing/valid	Pass rate	Targets ≥ 1 pass
Dual-expert MCTH	10	41/110	**37.3%**	8/11
RFdiffusion	10	22/110	20.0%	8/11

Table 23: Matched final-design yield using ten outputs per target and the same held-out Chai-1 pass criterion.

The pass fraction decreases as progressively lower-ranked candidates enter the archive ($49.1\% \rightarrow 37.3\% \rightarrow 30.2\%$), while target coverage remains 8/11 at all three archive sizes.

Matched comparison with RFdiffusion. For a direct baseline comparison, we use exactly ten final designs per target for both methods on the same 11-target set and apply the identical held-out filter. MCTH uses its reward-ranked top-10 unique archive, while RFdiffusion uses its ten generated outputs without held-out re-ranking.

Under equal output counts, dual-expert MCTH produces 41 passing designs compared with 22 for RFdiffusion, corresponding to a 17.3 percentage-point or approximately $1.86\times$ higher pass rate. Both methods identify at least one passing design on 8/11 targets. The result therefore reflects a higher density of qualifying outputs rather than broader target coverage alone.

RFdiffusion held-out score distribution. For completeness, Table 24 reports the per-target held-out scores used in the matched baseline analysis.

Together, the cross-predictor and archive analyses address two distinct questions. The dual-expert experiment tests whether search guided by more than one structural model transfers to a predictor outside the optimization loop, while the matched pass-rate analysis tests whether this robustness extends beyond a single best candidate to the final output distribution. Neither analysis constitutes experimental binding validation; rather, they measure cross-predictor robustness and computational final-design yield under controlled evaluation protocols.

E.7 Oracle-Independent Rosetta Interface Evaluation

Structure-predictor confidence is not a physical energy and may preserve biases of the models used during search. We therefore additionally evaluate protein-binder designs with Rosetta InterfaceAnalyzer, which is not used for proposal generation, MCTS selection, archive ranking, or any other component of MCTH.

The MCTH designs in this check use the minimal configuration with the task-specific biophysical term disabled. We evaluate the six CaoData targets shared by MCTH and RFdiffusion and the five with available BindCraft designs. We report interface separation energy ΔG_{sep}, the same energy normalized by buried interface area, packing quality (packstat), and buried unsatisfied hydrogen bonds.

Raw ΔG_{sep} is sensitive to interface size, so we treat the area-normalized energy as the more interpretable comparison. Under this measure, MCTH is essentially matched with RFdiffusion (-3.12 vs. -3.13 per $100\,\text{Å}^2$), while BindCraft is somewhat more favorable (-3.40). Packing

Target	Chai pass/10	Chai mean/best ipTM	AF2 mean/best ipTM
1DJS	1/10	0.327 / 0.594	0.166 / 0.278
1MOX	0/10	0.252 / 0.498	0.148 / 0.240
1XIW	6/10	0.540 / 0.808	0.536 / 0.840
2GY7	5/10	0.519 / 0.729	0.419 / 0.669
2IFG	2/10	0.288 / 0.700	0.259 / 0.783
3DI3	0/10	0.228 / 0.258	0.233 / 0.616
3FKD	0/10	0.230 / 0.310	0.559 / 0.883
3MJG	1/10	0.248 / 0.525	0.107 / 0.159
3ZTJ	1/10	0.280 / 0.660	0.258 / 0.557
4O3V	3/10	0.451 / 0.780	0.349 / 0.751
4OGA	3/10	0.430 / 0.685	0.422 / 0.656

Table 24: Per-target RFdiffusion held-out evaluation for the matched top-10 comparison. Mean/best values are computed over the ten outputs for each target.

Method	n	ΔG_{sep}	$\Delta G/100$ Å^2	Packstat	Δunsat H-bonds
MCTH	6	-71.2	-3.12	0.577	5.7
BindCraft	5	-65.7	**-3.40**	0.600	5.2
RFdiffusion	6	-54.2	-3.13	**0.602**	6.5

Table 25: Oracle-independent Rosetta InterfaceAnalyzer evaluation of protein-binder designs. More negative interface energy and higher packstat are favorable; fewer buried unsatisfied hydrogen bonds are preferred.

and buried-unsatisfied-hydrogen-bond statistics are of comparable order across the three methods, although MCTH does not outperform the strongest baseline on packing quality.

We therefore interpret this analysis conservatively: MCTH produces interfaces with competitive physical plausibility under an evaluation that is entirely outside the search loop, but these Rosetta scores do not establish superior binding energetics or experimental affinity.

E.8 Compute and Search-Trajectory Diagnostics

MCTH explicitly trades additional structure-prediction inference for adaptive optimization. We therefore report both the realized folding-call budget and wall-clock cost, and examine how search behavior changes over the course of optimization.

Realized compute. The nominal maximum budget is 100 folding calls, but early stopping terminates a search after 10 iterations without improvement. Consequently, the realized cost differs substantially across modalities.

Protein-binder design nearly exhausts the full budget, whereas RNA and DNA aptamer searches terminate substantially earlier. This is consistent with the different optimization landscapes: the short nucleic-acid tasks frequently reach high search-time confidence early, while de novo protein-binder design begins from a much larger sequence space and continues to improve later in the budget.

Performance versus compute. For protein binders, where nearly all searches reach the full budget, the held-forward mean best ipTM improves steadily from 0.614 after the first folding evaluation to 0.724, 0.768, 0.803, 0.830, and 0.864 after 5, 10, 25, 50, and 100 calls, respectively.

The continued improvement at later checkpoints complements Appendix E.4: the protein-binder gain is not produced solely by a strong initial proposal followed by stagnant computation.

Coverage versus allocation. We additionally compare the number and diversity of evaluated sequence states with the best-of-N control. MCTH does not consistently visit more unique states than independent sampling under the same evaluation budget. Instead, the principal difference is *where* evaluations are allocated within the evolving search tree.

Modality	Targets	Mean folding calls	Mean wall-clock
Protein binder	12	98.1	∼9.0 h
RNA aptamer	7	52.4	∼0.8 h
DNA aptamer	8	66.1	∼1.1 h

Table 26: Realized compute for the main MCTH experiments. DNA reports the eight targets used in the non-degenerate Boltz-2 comparison. Wall-clock values are means from the corresponding run logs and are hardware-dependent; they are not a normalized runtime comparison against other methods.

Fold call	1	5	10	25	50	100
ODesign	0.518	0.668	0.713	0.758	0.796	0.816
MCTH	0.614	0.724	0.768	0.803	0.830	0.864

Table 27: Protein-binder performance as a function of folding-call budget. Values are mean held-forward best Boltz-2 ipTM across the 12 targets. Both pipelines use the common refolding protocol; proposal generation differs.

Protein-binder searches tend to concentrate evaluations around promising branches early and broaden exploration later when progress slows. RNA and DNA searches show the complementary pattern: they explore broadly during the initial iterations and subsequently concentrate around an elite neighborhood once high-confidence candidates emerge. These trajectories are consistent with adaptive exploration–exploitation rather than uniformly greater state-space coverage.

Combined with the matched-budget ablation and convergence diagnostic, this suggests that the primary advantage of MCTH is *adaptive compute allocation*: the tree structure determines which sequence–structure trajectories receive additional expensive folding evaluations, rather than simply increasing the number of generated candidates.

F Extended Preliminaries

F.1 Biomolecular representation

Biomolecular Representation. We consider a general biomolecular system composed of one or more interacting components, including proteins, cyclic peptides, nucleic acids (DNA/RNA), small molecules, ions, and post-translational modifications. A biomolecule is represented by a *sequence* S and an associated *all-atom structure* X. Let $S = (s_1, \ldots, s_n)$ denote a sequence of discrete residue or atom types, where each s_i may correspond to an amino acid, nucleotide, small-molecule atom, or a special token (e.g., unknown or masked). Let

$$X = \{\mathbf{x}_1, \ldots, \mathbf{x}_m\}$$

denote the set of 3D atomic coordinates in Cartesian space. For multimolecular complexes, we represent the full system as

$$\mathcal{B} = \{(S^{(k)}, X^{(k)})\}_{k=1}^{K},$$

where k indexes the biomolecular components. Interactions between components are implicitly encoded through spatial proximity and pairwise relationships modeled by the structure prediction network.

F.2 Forward folding (diffusion) and inverse folding

All-Atom Structure Prediction via Diffusion Models. Let f_θ denote a pretrained all-atom structure prediction model parameterized by θ. Given an input sequence S and optional conditioning information C (e.g., binding partner, fixed motifs, or partial structure), the model predicts an all-atom structure according to

$$X \sim p_\theta(X \mid S, C).$$

These models typically employ a diffusion-based denoising process, starting from Gaussian noise and progressively refining atomic coordinates. When the input sequence is underspecified—such as an all-X sequence or a partially defined complex—the model often produces *hallucinated* yet physically plausible structures guided by learned structural priors. In this work, we explicitly exploit this hallucination behavior as a constructive signal for biomolecular design.

Sequence Design and Inverse Folding. Given a fixed or partially specified structure X, inverse folding models aim to generate sequences compatible with that structure:

$$S \sim q_\phi(S \mid X, C),$$

where q_ϕ denotes a pretrained sequence design model. Sequence redesign improves foldability and interaction consistency while preserving structural constraints. Within our framework, inverse folding serves as a primitive action that updates sequence hypotheses conditioned on hallucinated or refined structures.

Forward and Inverse Folding as Dual Operators. We treat forward folding and inverse folding as dual operators acting on the biomolecular state:

- **Forward folding:** $S \to X$ via all-atom structure prediction.
- **Inverse folding:** $X \to S$ via sequence redesign.

Rather than enforcing a fixed alternation, our framework allows these operators to be applied adaptively and repeatedly within a search process, enabling flexible traversal of the joint sequence–structure landscape.

F.3 Confidence/value signals

Confidence and Value Signals. Modern structure prediction models provide intrinsic confidence metrics that correlate with structural correctness and interaction quality, including predicted local accuracy, global confidence, and interface-specific measures. We collectively denote these signals as

$$V(X, S) \in \mathbb{R},$$

which serve as value estimates for evaluating candidate biomolecular designs. These values are treated as learned consistency scores rather than explicit physical energies.

F.4 MDP notation and value functions

We briefly summarize standard notation for finite-horizon Markov decision processes (Puterman, 2014). A policy $\pi(a \mid s)$ specifies a distribution over actions at state s. The action-value function $Q^\pi(s, a)$ is the expected cumulative reward obtained by taking a in s and then following π, and $V^\pi(s)$ is the corresponding state-value function. In our design setting, we primarily use this formalism to motivate MCTS-style planning and compute allocation: evaluations are obtained from model-derived scores $V(s)$ (confidence and feasibility), rather than long rollouts.

F.5 Diffusion models and guidance

We include a short diffusion primer to clarify our use of guided hallucination. Let $y_{0:T} = (y_0, \ldots, y_T)$ denote a latent trajectory, where y_0 is a clean sample (e.g., structure coordinates or discrete tokens) and y_T is maximally noisy. A forward noising process $q(y_t \mid y_{t-1})$ progressively corrupts y_0 into y_T. A learned reverse model $p_\phi(y_{t-1} \mid y_t)$ generates samples by iteratively denoising from y_T to y_0.

Guidance via reweighting. When an auxiliary score $f(\cdot)$ is available (property predictor, constraint critic, feasibility score), a common guidance view reweights reverse transitions:

$$p_\phi^{\text{guide}}(y_{t-1} \mid y_t) \propto p_\phi(y_{t-1} \mid y_t)\, \exp\{\beta\, f(y_{t-1})\}\,, \tag{14}$$

where β controls guidance strength. This formulation covers both continuous and discrete variants (with appropriate definitions of p_ϕ).

How we use this perspective. Our method is *inference-only*: we do not backpropagate through the predictor. Instead, we implement guidance by *proposal resampling*: for a given parent state (or diffusion step), we sample a small candidate set $\{y'_{t-1,j}\}_{j=1}^{J}$ (e.g., by changing random seeds, experts, or noise schedules), and then select or resample according to weights proportional to $\exp\{\beta f(y'_{t-1,j})\}$. In our biomolecular setting, $f(\cdot)$ is instantiated by the same task-aware value signal used by planning, combining intrinsic confidence metrics (pLDDT/pTM/ipTM) and optional feasibility penalties (clash/geometry/energy-like terms).

Multi-expert proposals. When multiple experts are available (e.g., multiple folding/hallucination experts and/or inverse-folding experts), we can form a mixture proposal $p_{\text{mix}}(\cdot) = \sum_{e=1}^{E} \omega_e \, p^{(e)}(\cdot)$ and apply the same resampling principle on samples drawn from the mixture. This describes an optional multi-expert proposal interface with MCTS: experts provide diverse proposals, while planning allocates compute and selects among them.

F.5.1 Monte Carlo Tree Search details

We summarize a few implementation-level details used in the paper.

Backup rules. After evaluating a newly expanded child state s', its value $V(s')$ is propagated along the selected path. We maintain visit counts $N(s)$ and edge counts $N(s, a)$, and update $Q(s, a)$ by a standard backup. The reported configuration uses running-mean backup, $Q = \texttt{total_reward}/\texttt{visits}$ (Table 11). Output selection follows Appendix C.6; the dual-expert archive is defined in Appendix E.6.

Evaluation cost and "rollouts". In biomolecular design, the dominant cost is a folding/refolding forward pass. Accordingly, we often treat expansion+evaluation as a single expensive step: each new child is instantiated by calling the folding expert and then scored by $V(s')$. This keeps the planner modular and makes compute accounting explicit via a fold-call budget.

MCTS with guided proposals. Guided sampling (Appendix F.5) can be used *inside* expansion: for a fixed parent (S, X), we generate multiple candidate proposals (different experts/seeds/noise schedules), reweight them by $\exp\{\beta V(\cdot)\}$ (optionally with feasibility penalties), and then add the selected children to the tree. MCTS then performs global credit assignment across these locally guided proposals, balancing exploration and exploitation across distinct design hypotheses.