
Supplementary information

Function-preserving watermarking of AI-generated proteins

In the format provided by the
authors and unedited

Appendix A SynthIDBio-sequence

As in the main paper, throughout the supplementary material we will refer to “resequencing” as the application of ProteinMPNN to any backbone to generate new designs.

A.1 Mechanism of tournament sampling

SynthIDBio-sequence employs tournament sampling, adapted from SynthID-text [1], to embed watermarks during the autoregressive generation of protein sequences by ProteinMPNN. This process leverages the entropy in the predicted amino acid distribution from ProteinMPNN and biases towards amino acid residues that score highly according to secret, key-dependent scoring functions (g -functions), while aiming to preserve the structure of the designed protein.

Figure S1, adapted from [1], illustrates the workflow. At each generation step t , a random seed is generated based on the preceding sequence context (H -gram) and a secret watermarking key. This seed determines the binary scores assigned by L distinct g -functions ($g_1, \dots, g_L$) to every possible next residue. Tournament sampling proceeds by initially drawing 2^L candidate residues from the ProteinMPNN probability distribution $p(x_t|x_{<t})$. These candidates undergo a series of L elimination rounds. In layer l , candidates are paired, and the one with the higher score according to g_l advances (ties broken randomly). The final remaining candidate is selected as the next residue x_t . This procedure ensures that the generated sequence exhibits a statistically detectable elevation in average g -value when analyzed with the correct secret key.

A.2 Protein design filters and detectability thresholds

The integration of SynthIDBio-sequence into a protein design pipeline necessitates the use of filters to ensure the designs are likely to be binders and the watermark is detectable above a calibrated threshold. To select high-quality binder candidates, we employed a comprehensive set of structural metrics derived from AlphaFold 3 (AF 3) predictions, applied to both the monomeric binder and the binder-target complex. Table S1 details the specific metrics and the thresholds used for both the *in silico* analyses and the *in vitro* validation. Thresholds for *in vitro* experiments were calibrated based on historical AlphaProteo campaign data to maximize the likelihood of predicting binders.

To ensure reliable detection while minimizing false positives, we introduced an additional filter based on the average g -value of the designed sequence. The thresholds for this filter were calibrated using a custom dataset of non-watermarked sequences (approximately 3.5 million), comprising non-watermarked binders and natural sequences from PDB and UniRef50 (see extended methods). This calibration allows us to set a specific target false positive rate (FPR). Table S2 presents the calibrated g -value thresholds for different FPR levels across the two primary watermarking settings used in the *in vitro* validation. To contextualize our watermarking parameters against the broader protein universe, we further derived FPR thresholds using $n = 46.4\text{M}$ sequences from the UniRef50 datasets. These values are detailed in Table S3. A comparison of distributions (Figure S2 and Figure S3) reveals that our

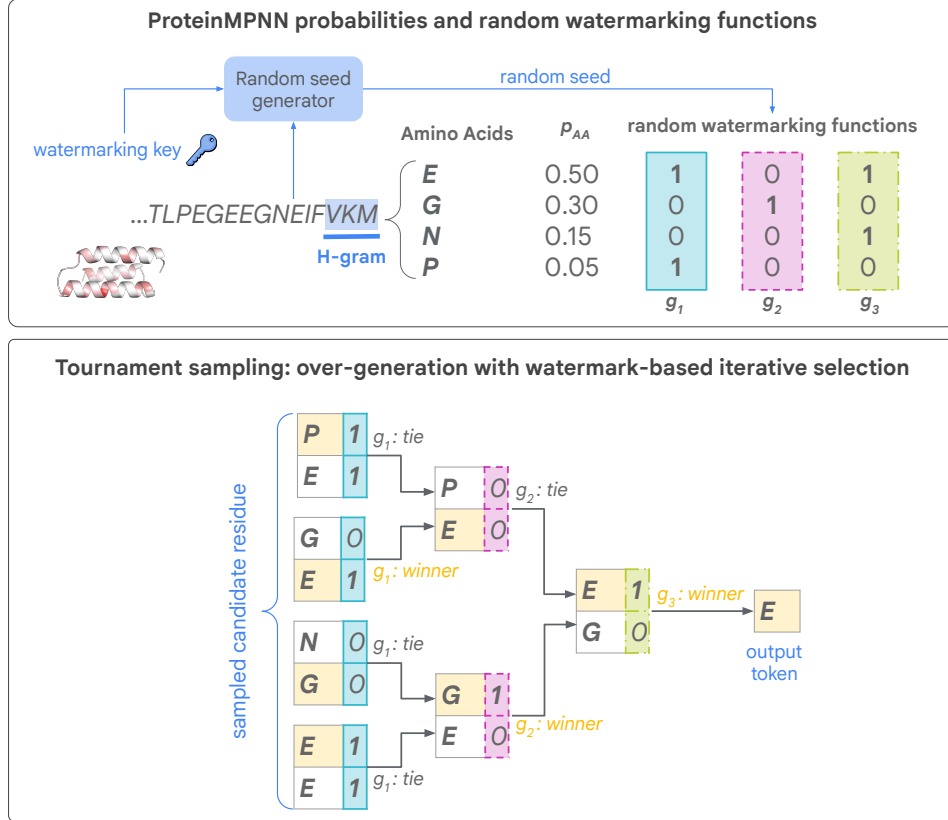


Fig. S1: Top: To generate a new amino acid residue x_t , we first score each residue in the vocabulary using multiple (in this case, $L = 3$) random watermarking functions $g_1, \dots, g_m$. These functions assign random values using a random seed, which is generated based on both the recent protein sequence context (H -gram) and a watermarking key. Bottom: Then, we choose the next amino acid residue using a tournament process. First, we sample $2^m = 8$ (possibly non-unique) candidate residues from $p_{AA}(\cdot|x_{<t})$. These are split into pairs of competing residues; in each pair, the highest-scoring one (based on g_1) is chosen, breaking ties randomly. The resulting winners compete in the next layer, where winners are chosen based on g_2 , until in the last tournament layer the final winner is selected based on g_m : this becomes the next generated amino acid residue x_t . Adapted from [2].

experimental thresholds were significantly more conservative. We partially attribute this to the substantially shorter length of the custom dataset (184 residues on average) compared to the UniRef50 dataset (325 residues on average). As shown in Figure S38, the reciprocal relationship between design length and variance likely necessitated stricter cutoffs to maintain FPR stability when using the custom dataset. To assess

Metric	<i>In vitro</i> threshold	<i>In silico</i> threshold
complex gdt ha	≥ 79.49152542305517	–
complex hotspot accuracy	$= 1$	$= 1$
complex interface lddt	≥ 79.4354838372515	–
complex iptm	≥ 0.8786113995624512	–
complex non template ptm	≥ 0.7999093541201046	≥ 0.8
complex num hbonds interface	≥ 1	–
complex pae template interface min	≤ 1.31038162763581	≤ 3
complex template ptm	≥ 0.7794860677513266	≥ 0.7
complex tm score	≥ 96.01487437201173	≥ 90
monomer buried unsat hbonds	≤ 10	–
monomer core percentage	≥ 0.05	–
monomer hpatch score	≤ 40	–
monomer hydrophobic ratio	$\geq 0.30 \ \& \ \leq 0.60$	–
monomer isolated residue percentage	≤ 0.1864406779661017	–
monomer lddt	≥ 89.84444442447901	–
monomer ptm	≥ 0.7696069045226787	≥ 0.7
monomer residue surface percentage	≤ 0.70	–
monomer rmsd	≤ 1.8833156021953987	–
monomer tm score	≥ 90.72174734301662	≥ 90
monomer topology string length	≥ 3	≥ 3
monomer very buried unsat hbonds	≤ 1	–
sequence length	≥ 1	≥ 1
sequence max consecutive identical aatypes	≤ 5	≤ 5
sequence max single aatype fraction	≤ 0.25	≤ 0.25
sequence num blocklisted motifs	$= 0$	$= 0$
sequence unknown residue count	$= 0$	$= 0$

Table S1: Filtering thresholds for *in vitro* and *in silico* experiments. Blocklisted motifs are defined as Strep-tag, c-Myc-tag, TEV protease, 3C protease cleavage site, HA-tag, His-tag, FLAG-tag, V5-tag, and thrombin cleavage. Designs must pass all thresholds to pass the filter. We define a monomer as a metric where the binder’s structure is predicted on its own, and complex metrics when the binder is in complex with the target. For all VEGF-A backbones and SC2RBD backbone 3, we relaxed two metrics to increase the number of resequences per setting that passed thresholds. For these, the cutoff value for *complex pae template interface min* was increased to 1.8107376472057974 and for *complex iptm* was decreased to 0.85. These metrics were selected from experimental data due to their predictive power distinguishing binder designs from non-binders.

functional retention under these strict regimes, we retrospectively filtered our synthesized designs using the UniRef50 thresholds. As shown in Figure S4, binder hit rates remained robust even at extreme stringencies. Notably, we successfully recovered functional designs at thresholds corresponding to a UniRef50 0.0005% FPR (one false positive per 200,000 sequences) for distortionary watermarking, confirming that high detectability is compatible with functional activity. Crucially, these extremely low FPR binders were recovered without explicit optimization for minimal FPRs; we simply applied a standard 0.1% *g*-value filter (Table S2) and prioritized candidates based solely on passing the metric filtering thresholds (Table S1). We also show the

TPR, though due to g -value filtering we can always trade compute for 100% TPR (by discarding designs lower than our desired FPR).

Target FPR	Temperature 0.5 Non-distortionary	Temperature 0.1 Distortionary
5%	0.519	0.589
1%	0.531	0.649
0.5%	0.536	0.676
0.1%	0.545	0.722
0.05%	0.549	0.734
0.01%	0.558	0.778

Table S2: FPR based thresholds on average g -value used for filtering. Values derived from a custom dataset of 2.8M wild-type proteins and 0.7M non-watermarked binder designs as described in the extended methods. We used 0.1% as the FPR for all *in vitro* experiments.

Target UniRef50 FPR	Temperature 0.5 Non-distortionary	Temperature 0.1 Distortionary
5%	0.513	0.566
1%	0.521	0.603
0.5%	0.524	0.619
0.1%	0.531	0.653
0.05%	0.534	0.667
0.01%	0.541	0.703
0.005%	0.545	0.722
0.001%	0.564	0.800
0.0005%	0.585	0.833
0.0001%	0.607	1.0

Table S3: FPR based thresholds calculated using $n = 46.4$ M wild-type proteins from the UniRef50 dataset. The table compares cutoffs for non-distortionary and distortionary watermarking. Note, these thresholds were used for computational analysis only and not for *in vitro* experiments.

A.3 Analysis of replicate variability

To assess the reproducibility of our *in vitro* measurements, we calculated the coefficient of variation (CV) for binding affinity (K_d) across two technical replicates for all design strategies. As illustrated in Figure S5, the distribution of measurement variability is

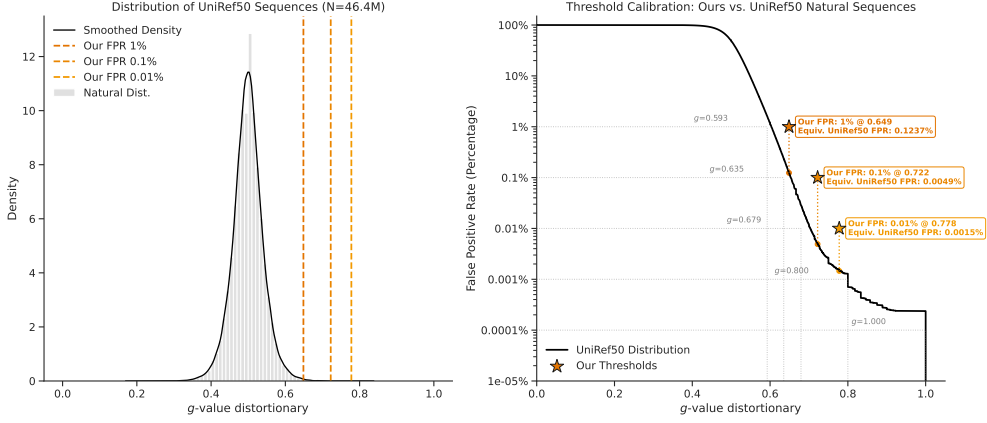


Fig. S2: Left: Distribution of g -values for natural UniRef50 sequences ($N = 46.4M$). FPR thresholds derived from the custom dataset for distortionary watermarking shown as vertical dashed lines. Right: FPR curve for UniRef50 sequences compared to our calibrated thresholds (orange stars). The thresholds derived from the custom dataset (Table S2) are significantly stricter (higher g -value) than those derived solely from UniRef50 (Table S3). For example, our 0.1% FPR threshold ($g = 0.649$) corresponds to a $\sim 0.0049\%$ FPR on the UniRef50 distribution. This conservative calibration accounts for the higher g -value variance observed in shorter designed sequences.

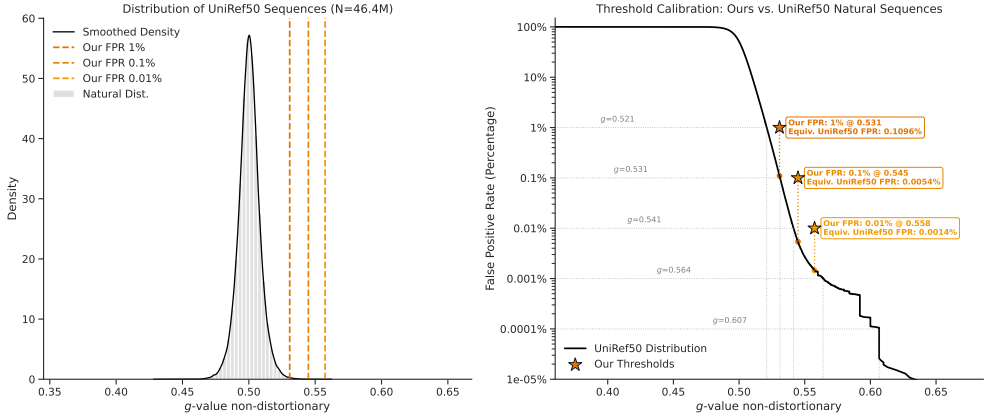


Fig. S3: Left: Distribution of g -values for natural UniRef50 sequences ($N = 46.4M$). FPR thresholds derived from the custom dataset for non-distortionary watermarking shown as vertical dashed lines. Right: FPR curve for UniRef50 sequences compared to our calibrated thresholds (orange stars). Consistent with Figure S2, the thresholds used in our experiments are significantly stricter.

uniform across non-watermarked baselines, the parent sequence positive controls, and

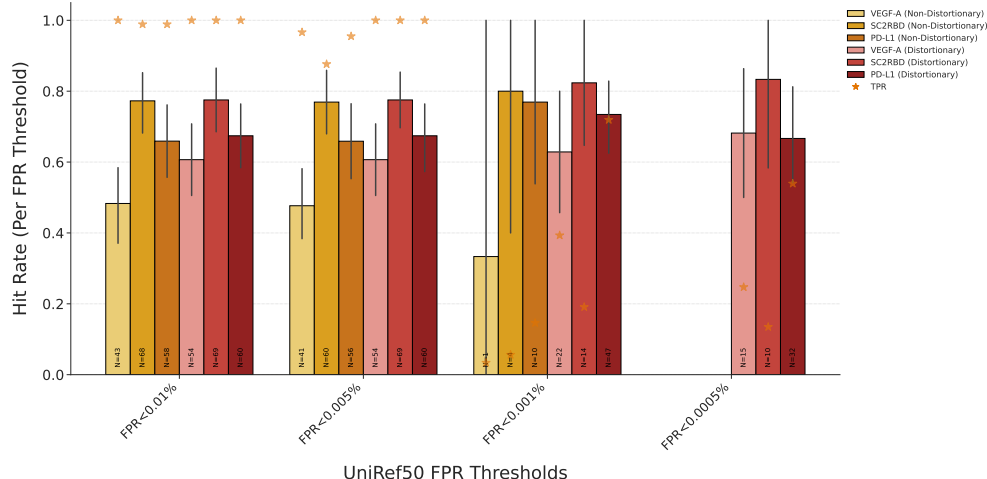


Fig. S4: We show the *in vitro* binder hit rates for designs generated with distortionary and non-distortionary watermarking after filtering with g -value thresholds calibrated to specific FPR on the UniRef50 dataset (see Table S3). The x-axis indicates the target FPR and the specific g -value threshold used. The y-axis shows the hit rate, defined as the fraction of synthesized designs that displayed binding activity. N values inside the bars indicate the number of designs remaining after filtering. Error bars indicate the standard error of the proportion. While the number of available designs (N) decreases with stricter thresholds (lower FPR), the hit rate remains robust across valid sample sizes for all three targets (VEGF-A, SC2RBD, PD-L1). Notably, we successfully recovered functional designs at Uniref50 distortionary thresholds corresponding to a 0.001% FPR or lower, confirming that high detectability is compatible with functional activity. The plateau in N at lower thresholds reflects the initial experimental inclusion criterion (g -value ≥ 0.722); deviations are only observed once the retrospective UniRef50 threshold exceeds this value.

both watermarking regimes. The mean CV is 21.9% for distortionary, 19.6% for non-distortionary, 20.4% for non-watermarked, and 19.4% for the parent sequences. Designs where replicates were inconsistent in detecting binding were excluded ($n = 2$). This highlights that watermarking does not adversely affect the precision or reproducibility of *in vitro* binding measurements.

A.4 Analysis of sequence diversity

A concern in watermarking generated protein sequences is the potential collapse of sequence diversity due to the watermark and/or the g -value threshold. To assess this risk, we performed a comprehensive analysis of sequence diversity across the *in vitro* sequences per backbone, as well as across the full population of un-filtered sequences per backbone. Our analysis reveals that watermarking significantly expands the sampled sequence space, driving diversity rather than limiting it. These results are further

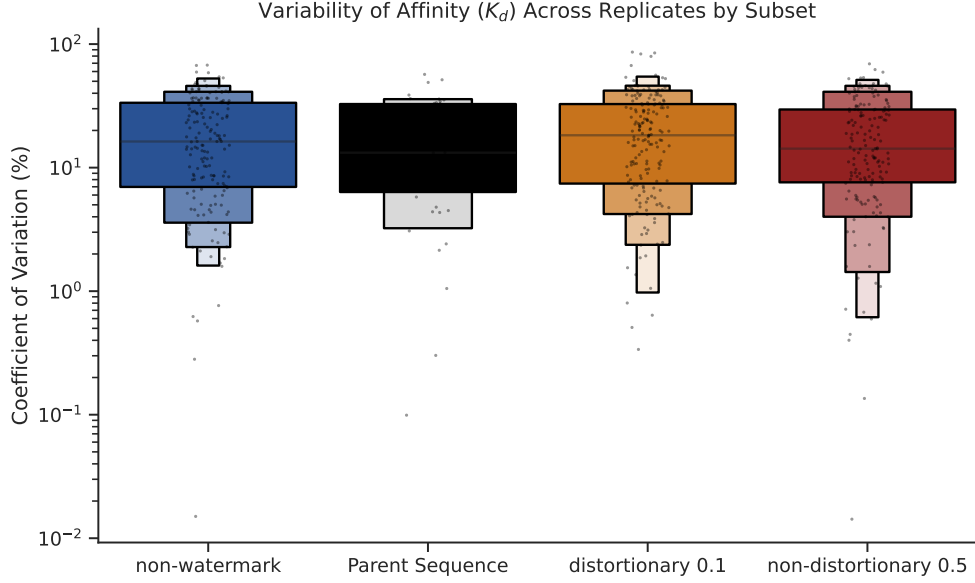


Fig. S5: We show quantile distributions of the Coefficient of Variation (CV) for binding affinity (K_d) derived from technical replicates. We compare experimental variability for non-watermarked designs, the parent sequences, and designs generated via distortionary and non-distortionary watermarking. The mean CV is 21.9% for distortionary, 19.6% for non-distortionary, 20.4% for non-watermarked and 19.4% for the parent sequences. There were two designs which we excluded from the analysis due to their replicates disagreed between whether the design was a binder. The consistent mean and median CV and distribution spread across all quantiles indicate that watermarking does not adversely affect the precision or reproducibility of *in vitro* binding measurements.

supported by Figure S38C, which demonstrates that higher ProteinMPNN entropy correlates with higher g -values.

A.4.1 Global expansion of sequence space

We first quantified global diversity using Shannon entropy and sequence identity metrics. As shown in Figure S7, both distortionary and non-distortionary watermarking regimes exhibit significantly higher mean Shannon entropy per-backbone after filtering and plate selection, 0.61 and 0.67 respectively, compared to non-watermarked baselines of 0.30. Similarly before any filtering or thresholding is applied Figure S6 shows distortionary and non-distortionary watermarking regimes exhibit significantly higher mean Shannon entropy per-backbone, 0.92 and 0.95 respectively, compared to non-watermarked baselines of 0.42. All watermarked entropy results are statistically significant compared to the non-watermarked case, with a $p < 0.0001$ (as per a Wilcoxon rank-sum test).

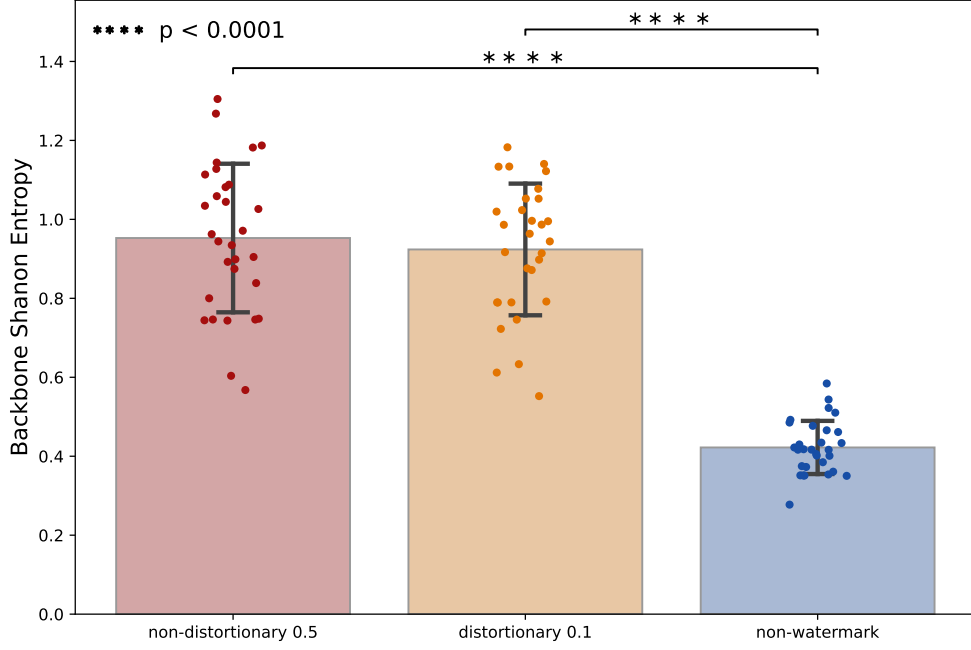


Fig. S6: Distribution of Shannon entropy scores for non-watermarked ($N = 178,941$), distortionary ($N = 178,343$), and non-distortionary ($N = 178,271$) designs per backbone before any filtering is applied. Watermarking significantly increases entropy ($p \leq 0.0001$, Wilcoxon rank-sum test), indicating a higher information content and broader sampling of the available sequence space compared to the non-watermarked baseline. Distortionary and non-distortionary watermarking have mean Shannon entropy per backbone of 0.92 and 0.95 respectively, compared to non-watermarked baselines of 0.42.

This expansion is further corroborated by sequence identity analysis. Figure S8 shows that watermarked sequences display a marked reduction in sequence identity relative to the parent sequence, shifting the mean identity from 68.7% in non-watermarked designs to 50% and 49.7% in distortionary and non-distortionary populations. Crucially, this divergence does not imply a loss of structural integrity and function but rather a successful exploration of alternative sequence solutions that satisfy the backbone constraints. Furthermore, the internal diversity of the generated populations is higher in watermarked groups, as evidenced by a lower backbone sequence conservation shown in Figure S9, where non-watermarked has 89.6% sequence conservation, distortionary and non-distortionary having 79.9% and 78.1%.

Overall, we find that watermarking does not negatively impact sequence diversity. In contrast, it seems to have a positive effect on the metrics investigated above.

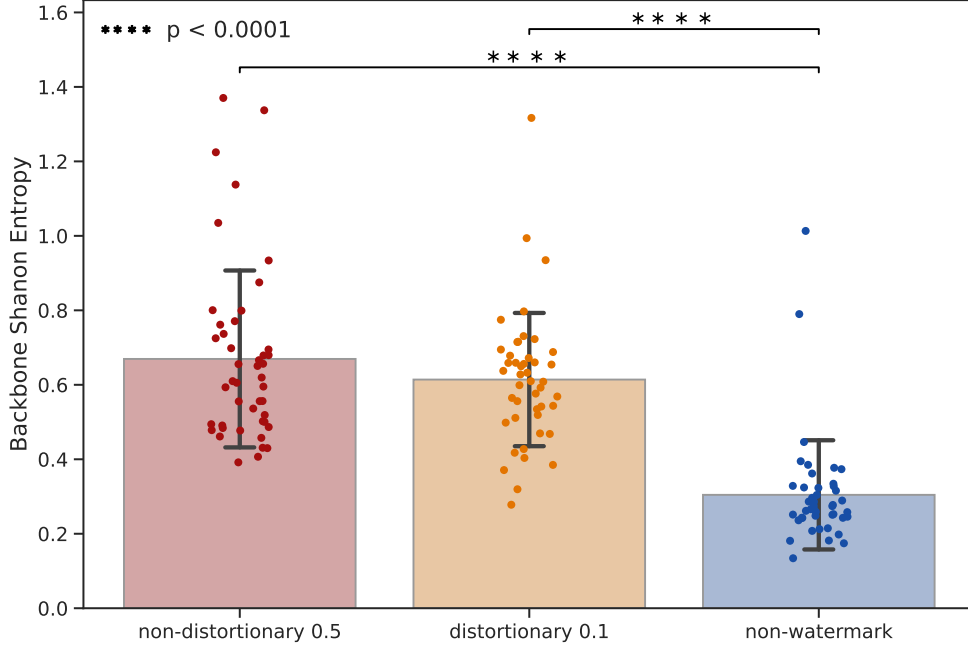


Fig. S7: Distribution of Shannon entropy scores for non-watermarked ($N = 222$), distortionary ($N = 267$), and non-distortionary ($N = 267$) designs per backbone, after filtering and plate selection. Watermarking significantly increases entropy ($p \leq 0.0001$, Wilcoxon rank-sum test), indicating a higher information content and broader sampling of the available sequence space compared to the non-watermarked baseline. Distortionary and non-distortionary watermarking have mean Shannon entropy per backbone of 0.61 and 0.67 respectively, compared to non-watermarked baselines of 0.30

A.4.2 Residue-level conservation analysis

The results in Section A.4.1 suggest that watermarking introduces a shift to higher global diversity. In order to measure the impact of watermarking on local diversity, we calculated per-position conservation relative to the original parent sequence (Figure S10) or the within-group consensus sequence (Figure S11) at various conservation thresholds. Consistent with the global values, across all thresholds measured the watermarked sequences demonstrate lower conservation to their parent sequence and to each other than non-watermarked sequences. At the highest level of conservation (100% of observations at that position), 68.0% of non-watermarked positions are entirely internally conserved and 54.6% of positions are conserved with the parent sequence. Under both distortionary 0.1 and non-distortionary 0.5 watermarking regimes, this rate of strict 100% conservation is significantly reduced ($p < 0.0001$, Chi-Squared Test). Relative to the parent sequence, strict conservation drops to 30.8% and 29.6%, respectively. Similarly, global internal conservation is reduced to 43.2% and 42.4%.

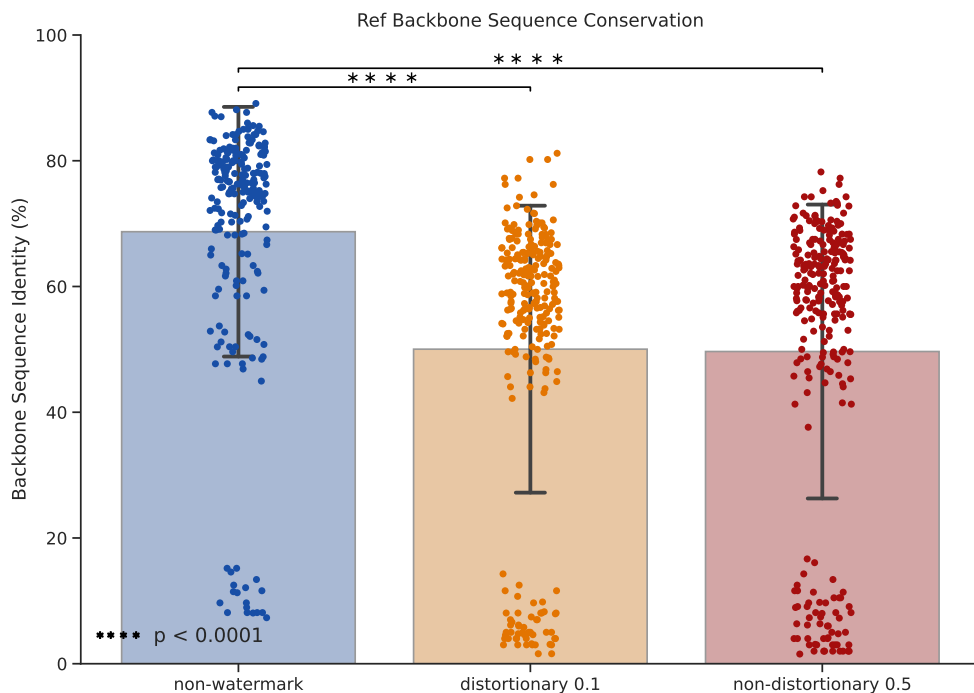


Fig. S8: Comparison of backbone sequence identity relative to the source parent sequence after filtering and plate selection. Watermarked populations ($N = 267$) show significantly lower identity to the parent compared to non-watermarked designs ($N = 222$); on average, with 68.7% in non-watermarked designs compared to 50% and 49.7% in distortionary and non-distortionary populations.

We further explored this phenomenon by exploring the most and least conserved sequences across all positions within each watermarking setting relative to their parent sequence (Figure S12) and consensus sequence (Figure S13). While displaying values for a limited number of samples, this analysis fits the overall trend of reduced conservation for watermarked sequences. Specifically, the highest mean values for distortionary 0.1 and non-distortionary 0.5 watermarked sequences relative to their parent sequence are reduced from 85.3% in the baseline to 78.7% and 75.7%, respectively. Similarly, the lowest mean is reduced to 3.7% and 3.2% compared to the non-watermarked baseline of 8.3%. The trend is maintained when considering within-group consensus sequences. Overall, these plots support significant diversity in conservation across generated sequences, but with a consistent signal that watermarked sequences are more diverse than non-watermarked.

A.5 Computational cost modeling

To evaluate the economic feasibility of watermarking in high-throughput protein design pipelines, we developed a cost model estimating the financial expenditure required to

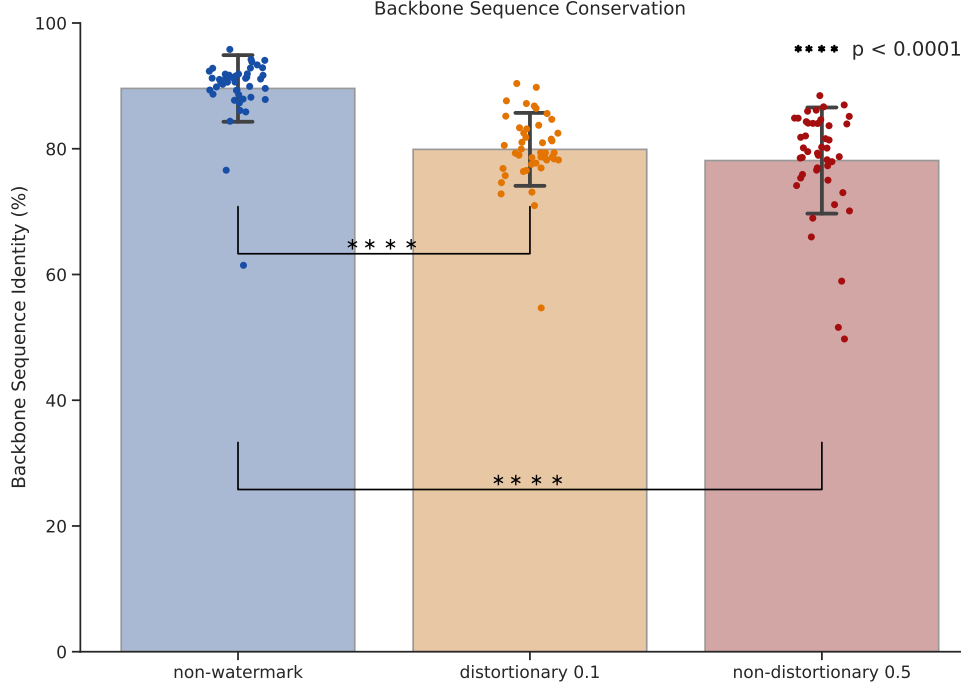


Fig. S9: Internal Population Diversity. Global backbone sequence conservation per backbone after filtering and plate selection. Lower conservation percentages in watermarked groups reflect higher internal diversity, preventing the redundant generation of near-identical sequences observed in the non-watermarked baseline. Where non-watermarked has 89.6% sequence conservation, distortionary and non-distortionary having 79.9% and 78.1%. The analysis comprises $n = 222$ non-watermarked, $n = 267$ distortionary, and $n = 267$ non-distortionary sequences.

generate a single successful *in silico* design, denoted *Cost per Success*. Importantly, we define a successful design, for a given FPR threshold, as one which passes g -value filters as well as passing the stringent design cuts in Table S1. The reason we define success as such, is so that we can study the large scale impact watermarking has on acquiring designs up to the final end point of *in silico* experimentation. Importantly, this is the only analysis on pass rate in this paper that uses the most stringent filters we used for *in vitro* experimentation, to best resemble our selection procedure. Its worth noting that all watermarking settings discussed have 100% TPR due to g -value filtering. We utilized on-demand pricing from Google Cloud Platform (GCP) as of December 11, 2025, alongside internal benchmarks for inference latency on representative hardware. We assumed a fully GPU-accelerated pipeline, utilizing NVIDIA A100 (40GB) instances for backbone generation with RFDiffusion, sequence design with Protein-MPNN, and structure prediction with AF3. We utilized the GCP `a2-highgpu-1g` instance type. The on-demand cost was set to $P_{\text{GPU}} = \$3.6748/\text{hr}$ ($\approx \$0.00102/\text{sec}$) [3].

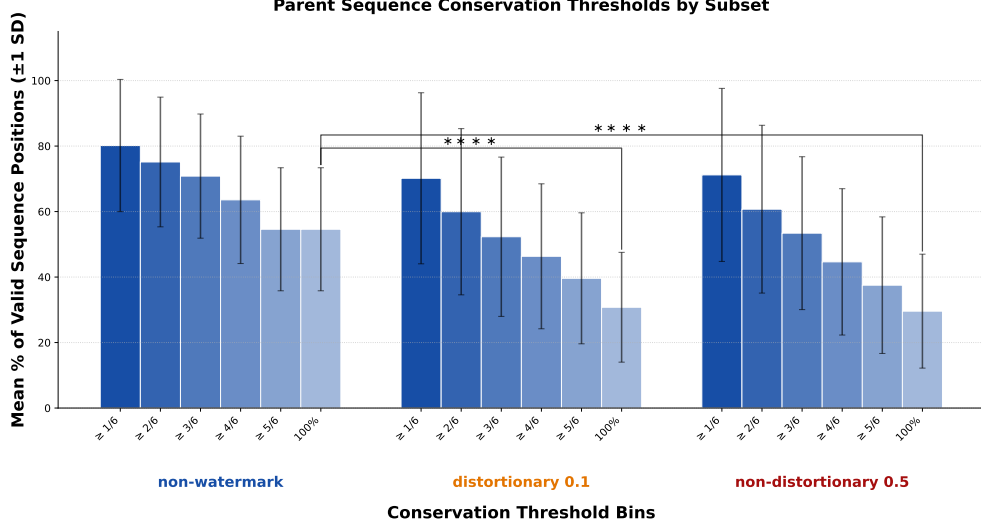


Fig. S10: Conservation relative to the parent sequence across binned thresholds. The bar plot illustrates the mean percentage of valid residue positions, aggregated per backbone, that meet or exceed specific conservation thresholds (from ≥ 0 to 100%) compared to parent sequence. Watermarking reduces the proportion of rigidly conserved positions (the 100% threshold) from 54.6% in the non-watermarked baseline down to approximately 30%. The sequence conservation is smoothly redistributed across lower fractional bins, demonstrating a controlled increase in mutational diversity. Statistical significance computed with chi-square statistic and * * * represents $p\text{-value} \leq 0.0001$.

A.5.1 Unit cost estimation

We estimated inference costs for RFDiffusion based on NVIDIA NIM benchmarks [4], which report a generation throughput of 1 step per second for a 507-residue protein on an A100 GPU (the closest available benchmark results to 512 residues). Using a standard trajectory of $S = 50$ diffusion steps (as used in the original publication [5]), the total inference time is $t_{\text{RF}} = 50$ seconds:

$$C_{\text{RF}} = 50 \text{ s} \times P_{\text{GPU}} \approx \$0.0510 \quad (\text{A1})$$

We internally benchmarked ProteinMPNN on the target A100 GPU. For a 512-residue chain, the average forward pass inference latency was observed to be $t_{\text{MPNN}} = 16 \text{ ms}$ (0.016 s):

$$C_{\text{MPNN}} = 0.016 \text{ s} \times P_{\text{GPU}} \approx \$0.000016 \quad (\text{A2})$$

For AF3 validation within a *de novo* design pipeline, we leverage the fact that binders are generated without evolutionary history, obviating the need for Multiple

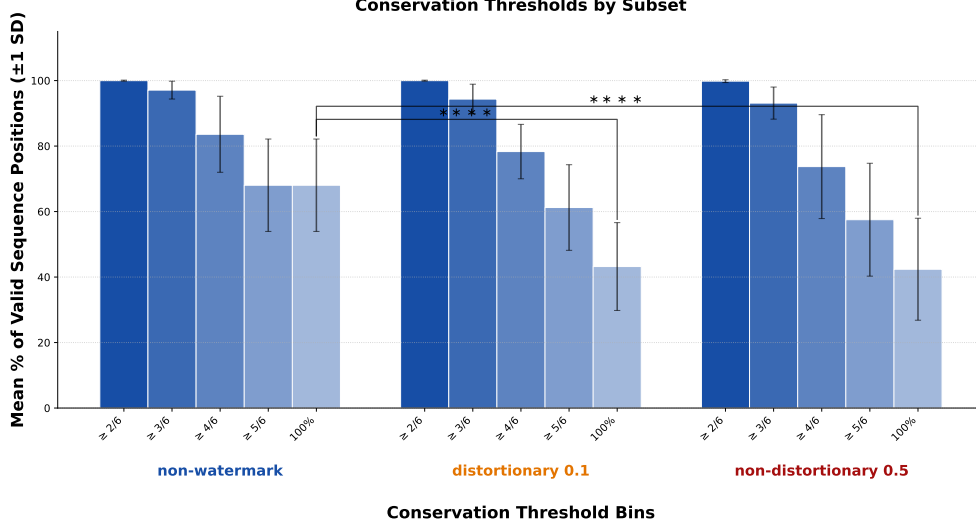


Fig. S11: Internal conservation of generated sequence ensembles across binned thresholds. The plot displays the mean percentage of residue positions, aggregated per backbone, that are conserved. Similar to the parent-relative conservation, while the non-watermarked baseline retains 68.0% of positions at strict 100% conservation, both distortionary and non-distortionary watermarking regimes reduce this strict conservation to roughly 42% to 43%. This demonstrates that the watermarking algorithm successfully redistributes sequence variability across moderately conserved thresholds while preserving overall structural properties and function. Statistical significance computed with chi-square statistic and **** represents $p\text{-value} \leq 0.0001$.

Sequence Alignment (MSA) construction. We rely solely on the target template. Internal benchmarks indicate that a 512-residue complex can be inferred in $t_{\text{AF3}} = 20$ seconds on an A100 GPU under these conditions:

$$C_{\text{AF3}} = 20 \text{ s} \times P_{\text{GPU}} \approx \$0.0204 \quad (\text{A3})$$

Note that ProteinMPNN has $\sim 222\times$ fewer parameters than AF3. The speed-up, however, is significantly larger, $\sim 1250\times$. This discrepancy arises because AF3’s inference time is dominated by the $O(N^3)$ cubic scaling of its pairwise attention modules (specifically triangular multiplication), whereas ProteinMPNN relies on a more efficient encoder-decoder architecture.

A.5.2 Pipeline metrics and population data

To determine the *Cost per Success*, we modeled the design process as a funnel defined by two probability rates calculated from our population data. Our estimates are derived from a large-scale analysis comprising approximately $\approx 180,000$ sequences per

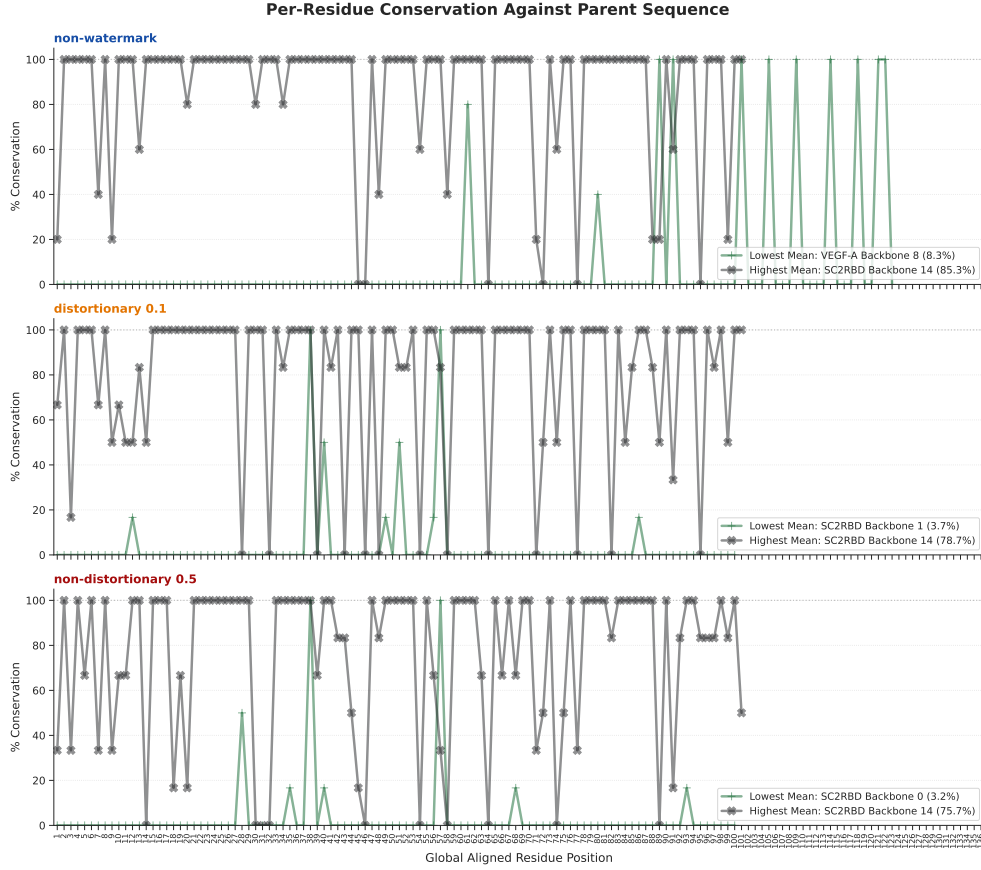


Fig. S12: Per-residue conservation profiles after filtering and plate selection showing most and least conserved backbones w.r.t parent sequence, across settings. The y-axis denotes the percentage of generated sequences retaining the parent identity at each position. Non-watermarked designs show high adherence to the parent residue. Watermarked designs exhibit variable conservation. The analysis comprises all sequences evaluated *in vitro*: $n = 5$ non-watermarked, $n = 6$ distortionary, and $n = 6$ non-distortionary sequences per backbone, across all 45 backbones.

design strategy. We generated $\sim 6,000$ sequences using ProteinMPNN for each of 30 distinct backbones, 15 derived from PD-L1 and 15 from SC2RBD, to rigorously approximate these cost tradeoffs.

1. **Generation rate ($R_{\text{gen}}^{\text{FPR}}$):** The fraction of ProteinMPNN sequences that satisfy the watermarking g -value filter (Filter Pass Rate) at a given FPR threshold (e.g. 0.1%). For non-watermarked baselines, $R_{\text{gen}}^{\text{FPR}} = 1.0$ for all FPR thresholds i.e no designs are filtered based on the g -value. Importantly, this measures the amount of work ProteinMPNN needs to do to generate a sequence to satisfy a given FPR constraint.

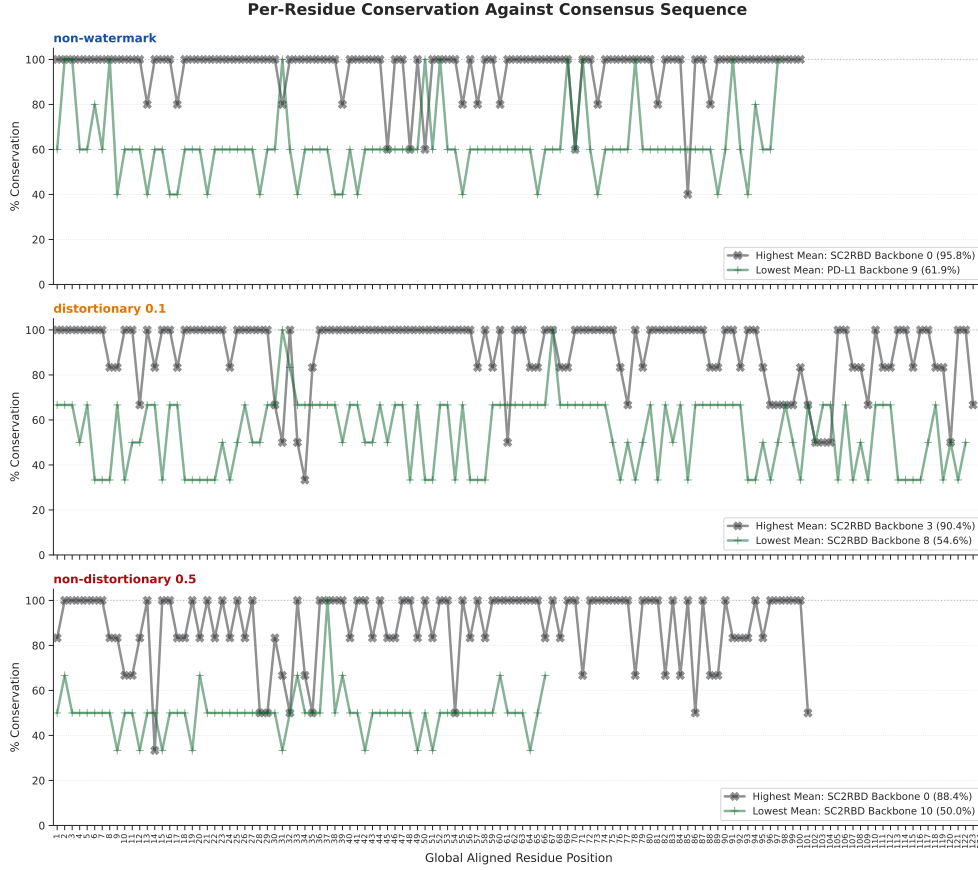


Fig. S13: Conservation frequency of the consensus amino acid at each position after filtering and plate selection for the most and least conserved backbones, across settings. The lower overall conservation traces for distortionary and non-distortionary groups confirm that watermarking introduces site-specific variability, effectively “softening” the constraints at mutable positions compared to the rigid consensus of non-watermarked designs. The analysis comprises all sequences evaluated *in vitro*: $n = 5$ non-watermarked, $n = 6$ distortionary, and $n = 6$ non-distortionary sequences per backbone, across all 45 backbones.

2. **Validation rate ($R_{\text{val}}^{\text{FPR}}$):** The fraction of sequences passing the design filters S1 from those that have already passed the watermarking g -value filter for a given FPR threshold. Note $R_{\text{val}}^{\text{FPR}}$ is constant for non-watermarked, as there is no change in population at various FPR thresholds as we do not apply g -value filtering to these designs. This measures the amount of work AF 3 must do in order to find a high quality design for a given backbone from a pool of sequences which all satisfy a given FPR constraint.

The total expected cost per successful design is the sum of the fixed backbone cost plus the variable sampling costs required to achieve one passing sequence:

$$\text{Cost}_{\text{success}} = C_{\text{RF}} + \left(\frac{1}{R_{\text{gen}}^{\text{FPR}} \cdot R_{\text{val}}^{\text{FPR}}} \times C_{\text{MPNN}} \right) + \left(\frac{1}{R_{\text{val}}^{\text{FPR}}} \times C_{\text{AF3}} \right) \quad (\text{A4})$$

For the non-watermarked baseline, we observed a validation rate of $R_{\text{val}}^{\text{FPR}} = 31.4\%$ and a generation rate of $R_{\text{gen}}^{\text{FPR}} = 100\%$. The baseline cost is derived as follows:

$$\text{Cost}_{\text{base}} = \$0.051 + \left(\frac{1}{1.0 \cdot 0.314} \times \$0.000016 \right) + \left(\frac{1}{0.314} \times \$0.0204 \right) \quad (\text{A5})$$

$$\text{Cost}_{\text{base}} = \$0.051 + \$0.00005 + \$0.065 \approx \$0.12 \quad (\text{A6})$$

This baseline figure of \$0.12 serves as the reference point for evaluating the economic impact of watermarking strategies.

For the non-distortionary watermarking scheme at the $< 0.1\%$ FPR threshold, we observed a reduced generation rate of $R_{\text{gen}}^{0.1\%} = 67.4\%$ and a validation rate on the designs that passed the g -value filter of $R_{\text{val}}^{0.1\%} = 21.3\%$. Despite needing ~ 7 ProteinMPNN inferences per success, the costs of AF 3 far outweigh the costs of MPNN. Because watermarking does not incur additional structure validation, this means that the cost increases only slightly:

$$\text{Cost}_{\text{ND}} = \$0.051 + \left(\frac{1}{0.674 \cdot 0.213} \times \$0.000016 \right) + \left(\frac{1}{0.213} \times \$0.0204 \right) \quad (\text{A7})$$

$$\text{Cost}_{\text{ND}} = \$0.051 + \$0.00011 + \$0.096 \approx \$0.147 \quad (\text{A8})$$

A.6 Cost-benefit analysis of SynthIDBio-sequence

We evaluated whether the strict sampling requirements of watermarking impose a prohibitive economic burden and found that the computational cost of oversampling sequences to satisfy strict watermarking filters is negligible. Specifically, the cost is dominated by AF 3 validation, which is not affected by watermarking.

A.6.1 Stability of Cost per Success

As shown in Figure S14, the introduction of watermarking increases the Cost per Success from the baseline \$0.12 to approximately \$0.15 (non-distortionary) or \$0.18 (distortionary). This cost increase is driven entirely by the drop in base folding yield from 31.4% (non-watermarked) to 21.5% (non-distortionary) and 16.4% (distortionary). Because the validation rate $R_{\text{val}}^{\text{FPR}}$ determines how often the expensive AF 3 model must be run, these lower yields inherently raise the floor price of the pipeline.

However, once a watermarking scheme is selected, increasing the stringency of the filter has virtually no economic impact. For example, achieving a $< 0.01\%$ FPR with non-distortionary watermarking requires $\sim 6.8\times$ more ProteinMPNN inferences than

the non-watermarked case ($R_{\text{gen}}^{0.01\%}$ drops from 99.8% to 22.9%). Because ProteinMPNN inference costs $\approx \$0.000016$, this massive increase in sampling adds less than \$0.0001 to the total cost per success. As a result, the cost curve is nearly flat across all FPR thresholds.

For the specific threshold used in this study (0.1% FPR), the cost remains stable at \$0.15 for non-distortionary methods – an increase of only 26.8% over baseline. Similarly, distortionary methods at $< 0.1\%$ FPR incur a cost of \$0.18 (+55.9%), driven by the stricter constraints on the backbone. Overall, across all schemes and thresholds, the cost per success spans a narrow range from \$0.12 to \$0.18 which in absolute terms highlights the economic feasibility of watermarking at high detectability.

To operationalize these findings, our experiments indicate that for a non-distortionary strategy targeting a $< 0.5\%$ FPR (at 100% TPR), the initial sequence pool must be scaled by a factor of ≈ 1.6 . In this pipeline, generated sequences are first screened against the watermark (g -value) filter; only those meeting the threshold proceed to computationally expensive structural validation. This process supports two selection modalities: an efficient “early-stopping” approach, where sampling for a backbone halts immediately upon identifying the first candidate to pass both watermark and structural filters, or a “best-in-class” approach, where multiple dual-passing candidates are ranked solely by structural quality metrics to select the optimal binder. Users should calibrate the initial sequence pool size based on the difficulty of the specific design task to ensure a sufficient number of candidates survive the watermark filter to proceed to structural validation. In practice, this calibration can be automated by implementing the dynamic early-stopping criteria described above, ensuring computational resources are allocated efficiently regardless of the target complexity.

A.6.2 Cost independence of validation quality

Crucially, we observed that the structural quality of the designs, measured by the AF3 validation rate, is independent of the watermark stringency within a given scheme. As illustrated by the “Validation Folding Pass Rate” in Figures S14 and S15, the pass rate for downstream structural filters remains effectively constant regardless of whether the watermark filter is set to a lenient 5% FPR or a strict 0.01% FPR. The independence of $R_{\text{val}}^{\text{FPR}}$ from filter stringency (FPR filtering threshold), combined with the negligible cost of ProteinMPNN sampling, implies that pipelines can adopt extremely strict thresholds (e.g., $< 0.001\%$ FPR) to maximize detectability without any economic tradeoff. While this study did not explicitly optimize for minimizing the yield gap between watermarked and baseline designs, the distinct cost asymmetry between generation and validation offers a clear pathway for optimization.

A.7 Windowed Screening Analysis

In the proposed biosecurity implementation described in section 5.1, nucleic acid synthesis companies would screen orders for the presence of a watermark. Within the existing screening regimes for sequences of concerns, algorithms typically evaluate overlapping subsequences (“windows”) against a reference database. If a similar fragmentation approach were performed for watermark detection, there is a risk that the

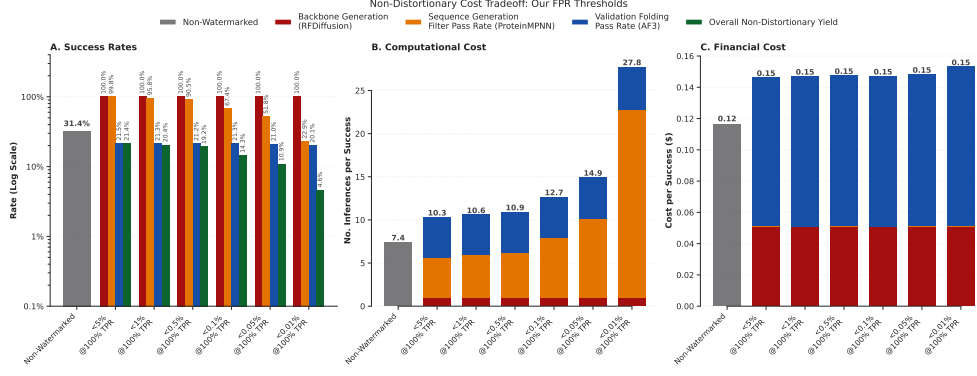


Fig. S14: Cost-benefit analysis for non-distortionary watermarking: **A.** Success rates for generation (filter pass rate) and validation (AF3 folding rate) across decreasing FPR thresholds. Note that while the generation rate drops as filters tighten, the validation rate (pink) remains constant. **B.** Computational cost required per successful design. The increase in work is driven solely by cheap ProteinMPNN inferences (orange), while expensive AF3 inferences (pink) remain stable. **C.** The estimated financial cost per successful design remains effectively flat ($\sim \$0.15$) despite strict filtering, as ProteinMPNN costs are negligible compared to AF3.

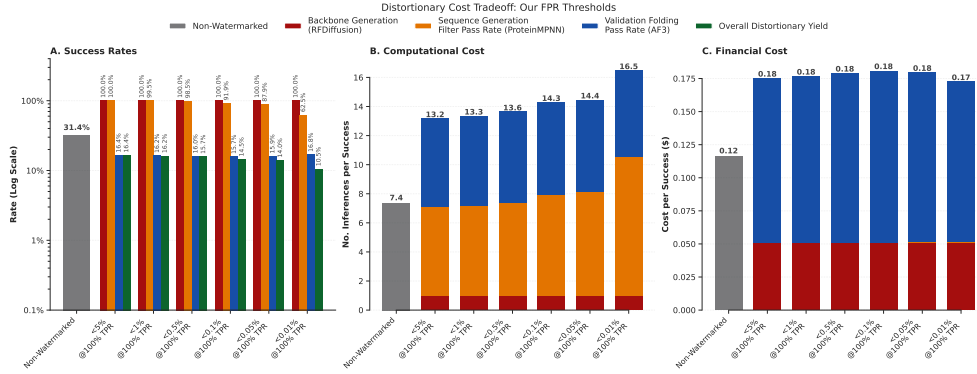


Fig. S15: Cost-benefit analysis for distortionary watermarking: Similar to the non-distortionary case, the cost per success (c) remains stable ($\sim \$0.18$) even at extreme stringencies. The higher cost compared to non-distortionary designs reflects the lower initial folding rate of the distortionary scheme, not the cost of the watermarking filtering process itself.

calibrated FPRs displayed in Figures S2 and S3 will underestimate the effective FPR for natural sequence orders due to (a) the multiple analyses per input sequence and (b) the higher variance in g -value for shorter sequences (see Figure S38).

To evaluate g -value computation under conditions more representative of real-world synthesis screening, we implemented a scanning window analysis on both natural and generated sequences. To quantify the impact of this variance on the FPR, we established the following evaluation framework:

1. Subsequence extraction and evaluation: We extracted length-60 subsequences using a tiled 30-residue overlap across a subsample of the UniRef50 natural sequences. This approach yielded 2M subsequences.
2. Empirical FPR assessment: We applied our pre-established thresholds used in the main paper (“Ours”), as well as full-sequence UniRef50 g -value thresholds to this windowed dataset to determine the empirical FPRs (Tables S4 and S5). Here, we naively apply a windowed approach with no altering of our calibrated thresholds. In practice, we would expect users to calibrate FPRs appropriately based on their detection pipeline.
3. Threshold recalibration: We calculated the necessary threshold shifts required to restore the target FPRs on the windowed distributions for both distortionary and non-distortionary watermarking schemes (Figures S16 and S17), and subsequently quantified the impact of these shifts on sequence retention.

As expected, applying UniRef50 full-sequence thresholds (Table S3) directly to windowed g -value dataset systematically inflates the empirical FPR. For example, applying the standard 0.1 % non-distortionary threshold ($g = 0.531$) to windowed UniRef50 sequences yields an inflated empirical FPR of 1.0% (Table S5). Consequently, thresholds must be recalibrated slightly upward to maintain statistical confidence. To achieve a true 0.1% FPR on UniRef50 using a sliding window, the non-distortionary threshold shifts from 0.531 to 0.543, and the distortionary threshold shifts from 0.635 to 0.679 (Figure S17).

However, applying the thresholds used in the paper (Table S2) directly to the windowed g -value dataset results in negligible differences in empirical FPR. The only substantial correction required was for distortionary watermarking at the most stringent rate (0.01%), which required a threshold shift from 0.679 to 1.000. However, this need for increased values is not consistent across all FPRs, with a 0.1% FPR being achieved with a lower threshold (0.679 versus 0.722). Additionally, as we know threshold calibration is computationally lightweight from Figures S14 and S15, calibrating on the window-based approach preserves both real-world usability and the expected true positive rate (TPR) compared to our utilized thresholds.

There are several technical and implementation approaches through which a window-based approach could be operationalized. For example, positive detection could be defined as the maximum g -value for any window within an open reading frame (ORF). However, deploying this max-window approach at scale requires careful recalibration of the background null distribution to account for the multiple hypothesis testing inherent in evaluating numerous windows per sequence. Alternatively, one could take the mean value for all windows within an ORF to limit the risk of a single window stochastically having a score about the threshold. To limit false positives, individual ORFs could also be screened using standard alignment-based tools prior to g -value analysis. Those with high homology to known sequences could be diverted

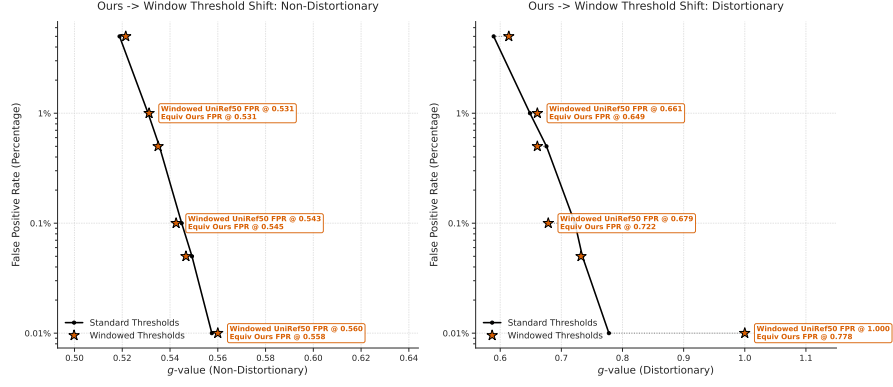


Fig. S16: Threshold calibration shifts for our pre-established thresholds (Table S2). Comparison of standard thresholds used in the main text (black line) versus re-calibrated windowed thresholds (orange stars) required to achieve target false positive rates (FPRs) on 60-residue subsequences. For both non-distortional (left) and distortional (right) watermarking, recalibration requirements are largely negligible. Substantial threshold adjustments are only necessary at the most stringent FPR targets (0.1% and 0.01% for distortional designs) to compensate for the variance of short evaluation windows.

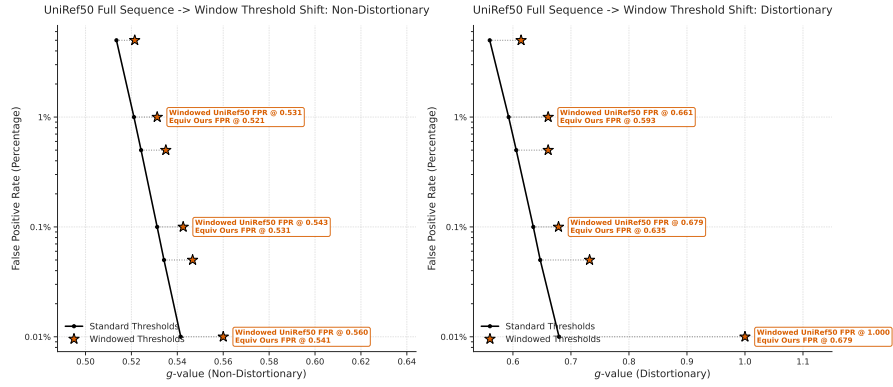


Fig. S17: Threshold calibration shifts for the UniRef50 full-sequence FPR thresholds (Table S3). Necessary g -value threshold adjustments when transitioning from UniRef50 full-sequence evaluations (black line) to a 60-residue scanning window (orange stars). Due to the broader variance inherent to shorter sequences, retaining the full-sequence thresholds systematically inflates the empirical false positive rate (FPR). Shifting the thresholds upward for both non-distortional (left) and distortional (right) watermarking successfully restores the target FPRs and maintains necessary statistical confidence.

from watermark detection as they are highly unlikely to be the product of an AI

	Method	Target FPR (%)	Threshold (g)	Empirical FPR (%)
0	Non-Distortionary	5.0	0.519	7.5
1	Non-Distortionary	1.0	0.531	1.1
2	Non-Distortionary	0.5	0.536	0.44
3	Non-Distortionary	0.1	0.545	0.067
4	Non-Distortionary	0.05	0.549	0.033
5	Non-Distortionary	0.01	0.558	0.014
6	Distortionary	5.0	0.589	9.9
7	Distortionary	1.0	0.649	1.6
8	Distortionary	0.5	0.676	0.54
9	Distortionary	0.1	0.722	0.12
10	Distortionary	0.05	0.734	0.097
11	Distortionary	0.01	0.778	0.066

Table S4: Empirical false positive rates using our pre-established thresholds on windowed sequences. Application of standard full-sequence g -value thresholds (Ours) to length-60 scanning windows. With the exception of highly stringent distortionary targets (e.g., 0.01% FPR), applying the original thresholds to windowed evaluations yields empirical FPRs that closely track the target rates.

	Method	Target FPR (%)	Threshold (g)	Empirical FPR (%)
0	Non-Distortionary	5.0	0.513	15.0
1	Non-Distortionary	1.0	0.521	5.4
2	Non-Distortionary	0.5	0.524	3.5
3	Non-Distortionary	0.1	0.531	1.0
4	Non-Distortionary	0.05	0.534	0.57
5	Non-Distortionary	0.01	0.541	0.12
6	Distortionary	5.0	0.560	20.0
7	Distortionary	1.0	0.593	8.7
8	Distortionary	0.5	0.606	6.1
9	Distortionary	0.1	0.635	2.2
10	Distortionary	0.05	0.647	1.6
11	Distortionary	0.01	0.679	0.45

Table S5: Empirical false positive rates using UniRef50 full-sequence calibrated thresholds on windowed sequences. Application of baseline UniRef50 g -value thresholds to length-60 scanning windows. The inherently broader variance of shorter subsequences systematically inflates the empirical FPRs relative to the target FPRs. For example, applying the full-sequence 0.1% non-distortionary threshold yields a 1.0% empirical FPR, demonstrating the necessity of upward threshold recalibration if using the UniRef50 thresholds when considering windowed screening.

design tool. Further, synthesis companies could accomplish a similar goal by requiring submitters to provide coordinates demarcating the boundaries where a watermark

should be detectable. Given the litany of approaches that depend on specific implementations, the formal optimization and integration of this localized watermarking detection strategy into synthesis screening algorithms is left for future work.

A.8 *In vitro* results and analysis

A.8.1 Impact of resequencing on binding affinity

The experimental design utilized known binder backbones, which were resequenced using ProteinMPNN with or without watermarking (Figure S18). The parent sequences were similarly outputs from resequencing a given backbone and had been selected for *in vitro* validation in a previous AlphaProteo campaign. This enabled analysis of the impact of resequencing itself on binding affinity. Within this analysis, one interesting backbone was VEGF-A backbone 13. Here, resequencing resulted in a binder with binding affinity two orders of magnitude higher than the parent sequence. In contrast, for VEGF-A backbone 12 the best resequenced design was two orders of magnitude lower than the parent sequence. This highlights the variability across backbones on success.

Overall, resequencing generated a higher affinity binder than the parent sequence for 74% ($n = 31/42$) of backbones, which is significantly lower than would be expected by chance if all designs were equivalently likely to be the best binder. We chose 14 designs for the statistical test as that is the lower bound per backbone ($p = 1e-8$, exact binomial test).

Furthermore, we analyzed the relationship between the number of resequencing attempts and the improvement in binding affinity independent of watermarking. Figure S19 shows a bootstrapped analysis of the best K_D improvement ratio achieved as a function of the number of designs synthesized per backbone. The trend clearly indicates that increasing the sampling effort during resequencing enhances the probability of identifying a higher-affinity binder, suggesting that ProteinMPNN effectively explores the sequence space around the provided backbone to optimize interactions when coupled with our filters.

A.8.2 Detailed analysis of backbones

We investigated the distribution of watermark signal across the protein structure and the impact of backbone structure and watermarking setting on the structural variance.

Figure S20 presents a detailed view of all designs derived from PD-L1 backbone 0. While the g -values within a specific watermarking setting show relatively low variability for this backbone, the binding affinities vary by nearly four orders of magnitude.

Figures S21, S22, and S23 display the highest-affinity binder identified for each of the 15 backbones per target, illustrating the diversity of structures and the corresponding distribution of g -values. Crucially, the selected cohort of backbones ($n = 45$) encompassed diverse secondary structure topologies, comprising 38 predominantly α -helical structures, 6 β -sheet structures, and one backbone exhibiting a mixed α/β topology.

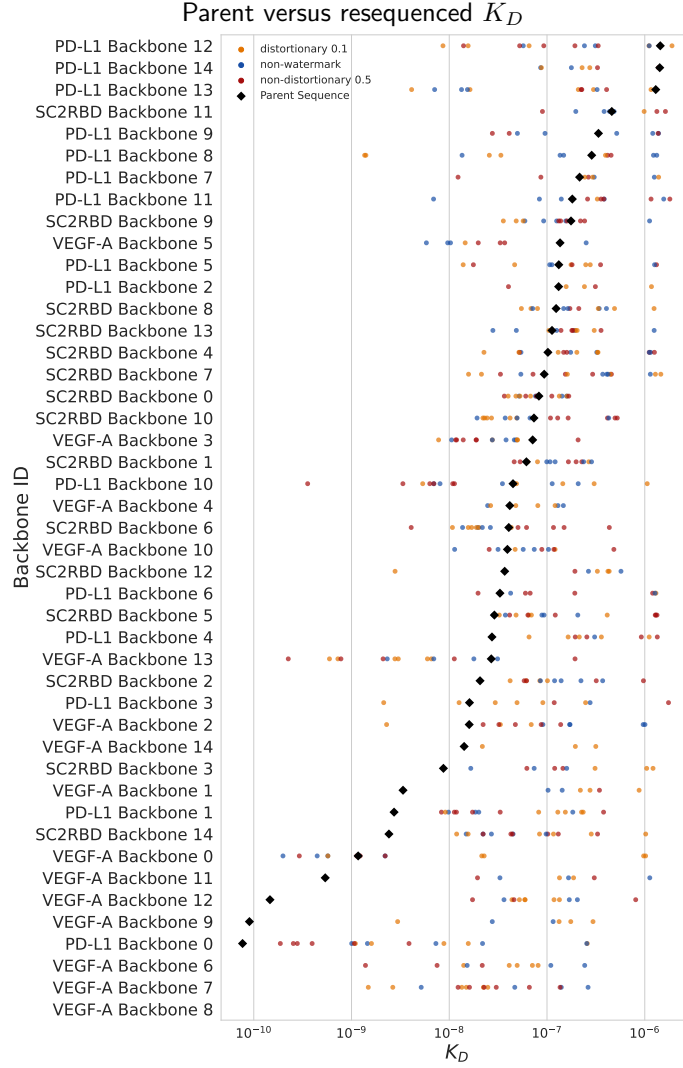


Fig. S18: We show K_D values from all resequenced sequences versus their corresponding parent sequence (black diamond). There are three missing backbones as the parent sequence did not bind in the SPR assay (bottom three). In total 74% (31/42) of parent sequences had their K_D improved from resequencing. Designs had 17 resequences per backbone apart from PD-L1 backbone 5, SC2RBD backbone 14, and VEGF-A backbone 11, which had 14 each (due to negative controls). VEGF-A backbone 8 had no hits across all settings. After analysing the original BLI affinity characterisation of the parent sequence, we determined this to be explained due to the original sequence being a very low confidence binder (poor binding curve fit), as well as being a weak binder.

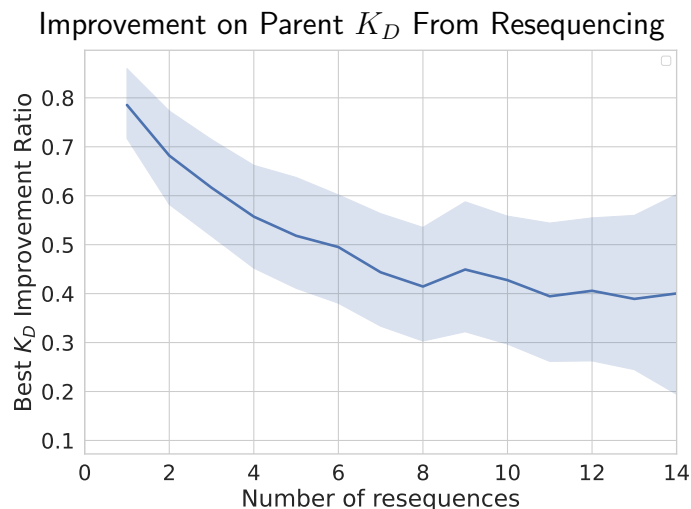


Fig. S19: Best K_D improvement from all resequences across watermarked and non-watermarked settings versus the corresponding backbones parent sequence via bootstrapping. The number of samples is limited to 14 rather than 17 sequences as there are three backbones with 14 sequences due to negative and positive controls. Overall, we see a clear trend of improvement as the number of resequences increases. K_D improvement is computed by resequenced K_D divided by parent K_D . Where we give any sequence a K_D improvement score of 1.0 if it demonstrates lower affinity than the parent sequence, as we fail to improve above the original parent sequence affinity.

Figures S24 and S25 provide complete multiple sequence alignments colored by per-residue g -values for VEGF-A backbone 1, further illustrating the differences in signal distribution between non-distortionary 0.5 and distortionary 0.1 settings.

A.8.3 Analysis of g -value distributions across backbones

The efficacy of watermarking is dependent on the underlying entropy of the sequence design task, which can vary significantly depending on the structural constraints imposed by the backbone. We analyzed the distribution of g -values across all 15 backbones for each target, both before and after applying the protein design filters from Table S1 and detectability filters from Table S2.

Figures S26 through S31 present the g -value distributions for the populations of designs that passed all filters (structural quality and g -value threshold at 0.1% FPR). These represent the pool of designs eligible for synthesis. Even after stringent filtering, a wide range of g -values above the threshold can be observed, indicating that designs with very high detectability can be readily selected.

Figures S32 through S37 show the g -value distributions before any filtering. These plots highlight the significant variability, emphasizing the necessity of the g -value filter to ensure consistent detectability across diverse design tasks and backbones.

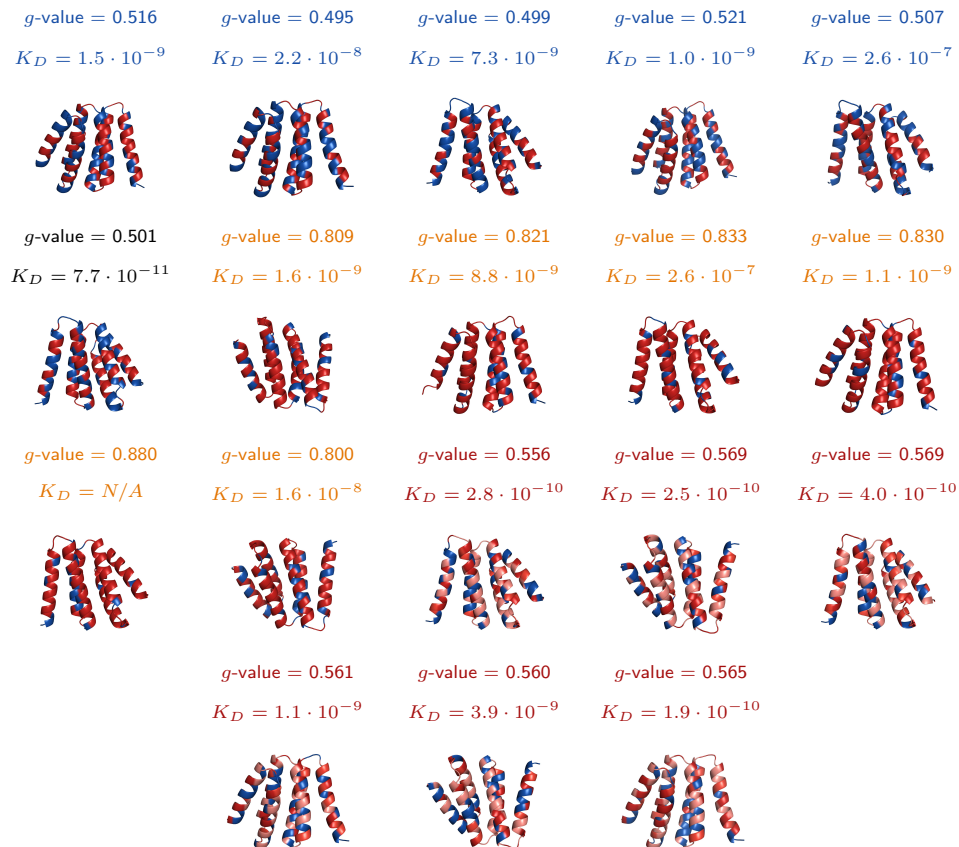


Fig. S20: All resequencences produced from PD-L1 backbone 0. Titles indicate the watermark setting with distortionary at temperature 0.1 being **orange**, non-distortionary at temperature 0.5 being **red**. Black titles indicate the parent sequence. The respective g -values are highlighted as in the main paper with a color map ranging from **blue** to **red**. We color non-watermarked designs according to the non-distortionary g -values.

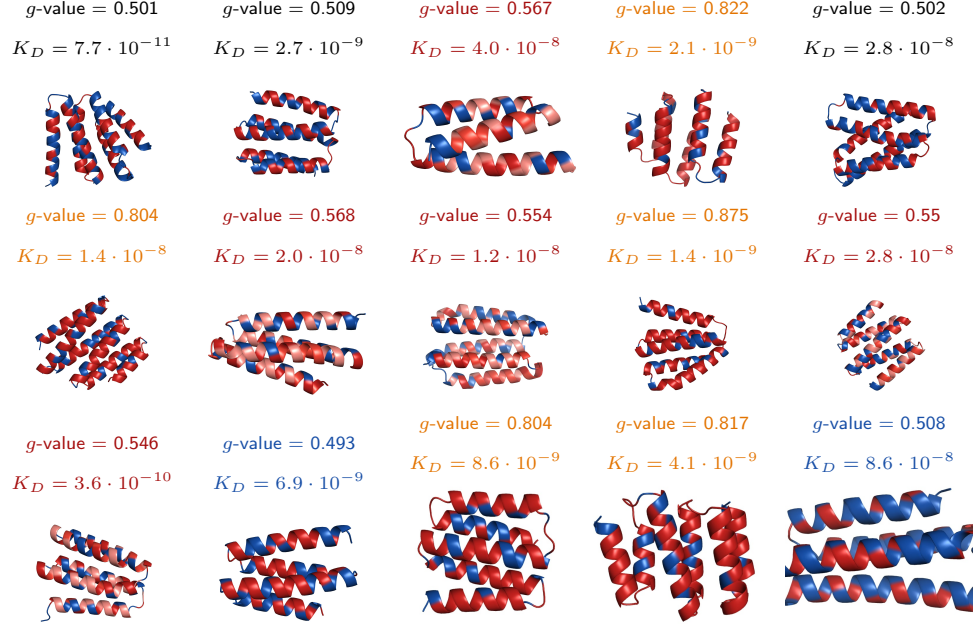


Fig. S21: Highest-affinity binder found from resequencing PD-L1 backbones, with backbone ID increasing from left to right. Top backbones 0-4, middle backbones 5-9, and bottom backbones 10-14. Titles indicate the watermark setting with distortionary at temperature 0.1 being **orange**, non-distortionary at temperature 0.5 being **red**. Black titles indicate the parent sequence. The respective g -values are highlighted as in the main paper with a color map ranging from **blue** to **red**. We color non-watermarked designs according to the non-distortionary g -values.

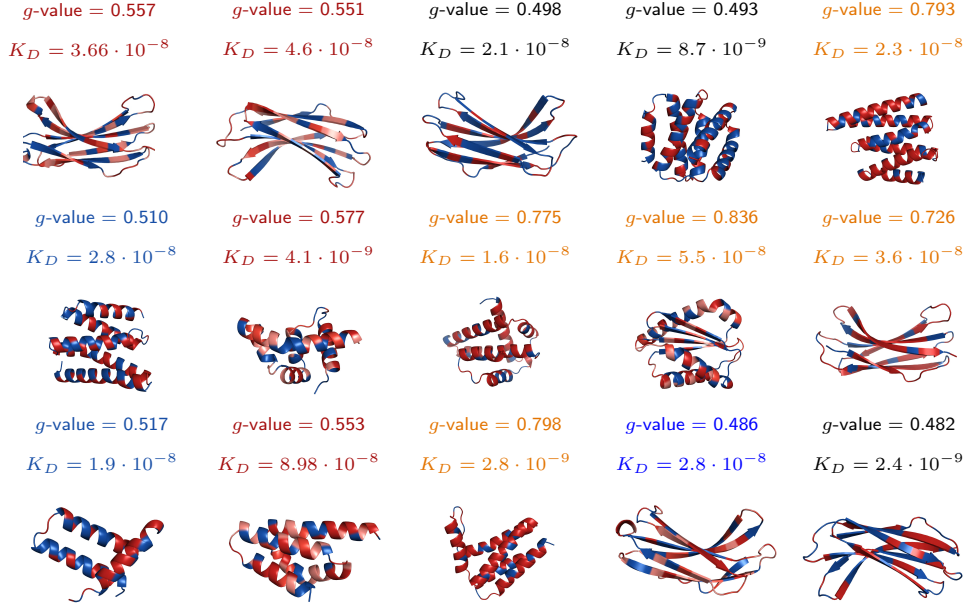


Fig. S22: Highest-affinity binder found from resequencing SC2RBD backbones, with backbone ID increasing from left to right. Top backbones 0-4, middle backbones 5-9, and bottom backbones 10-14. Titles indicate the watermark setting with distortionary at temperature 0.1 being **orange**, non-distortionary at temperature 0.5 being **red**. Black titles indicate the parent sequence. The respective g -values are highlighted as in the main paper with a color map ranging from **blue** to **red**. We color non-watermarked designs according to the non-distortionary g -values.

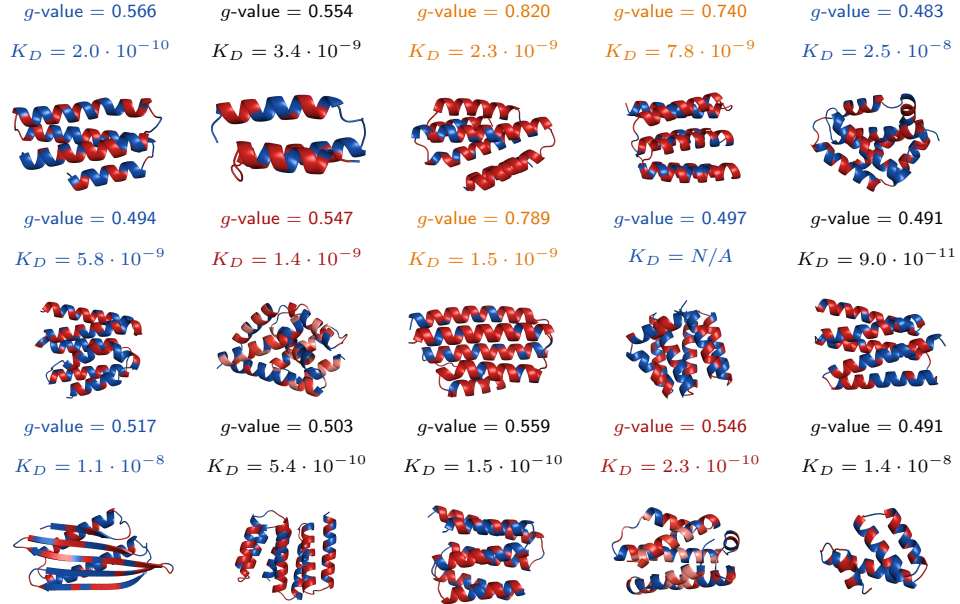


Fig. S23: Highest-affinity binder found from resequencing VEGF-A backbones, with backbone ID increasing from left to right. Top backbones 0-4, middle backbones 5-9, and bottom backbones 10-14. Titles indicate the watermark setting with distortionary at temperature 0.1 being **orange**, non-distortionary at temperature 0.5 being **red**. Black titles indicate the parent sequence. The respective g -values are highlighted as in the main paper with a color map ranging from **blue** to **red**. We color non-watermarked designs according to the non-distortionary g -values.

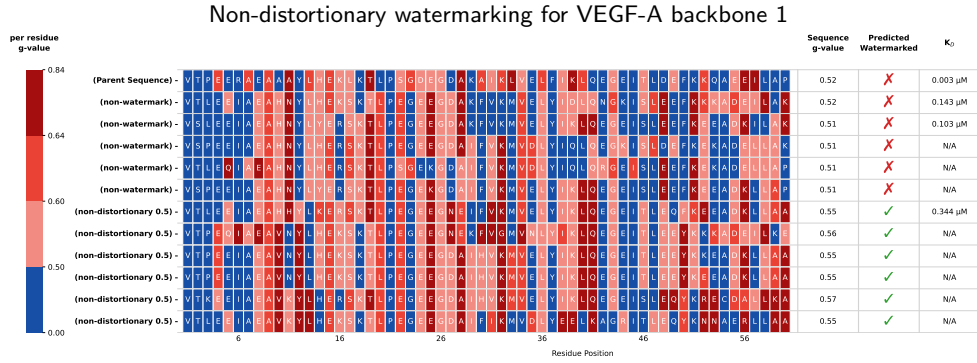


Fig. S24: Representative multiple sequence alignment of non-watermarked parent sequence (previously validated design corresponding to a selected backbone), all non-watermarked, and all non-distortionary 0.5 watermarked binders targeting VEGF-A backbone 1. Color indicates g -values. Watermarked sequences are detected based on a sequence-level, average g -value threshold of 0.545 for a 0.1% FPR.

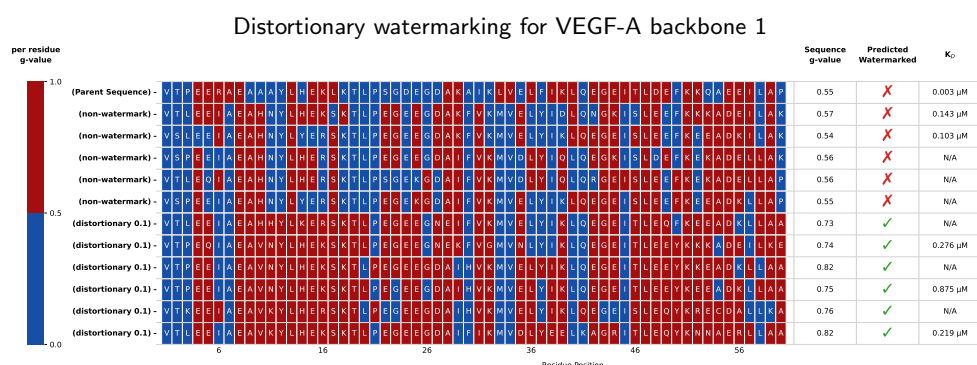


Fig. S25: Representative multiple sequence alignment of non-watermarked parent sequence (previously validated design corresponding to a selected backbone), all non-watermarked, and all distortionary 0.1 watermarked binders targeting VEGF-A backbone 1. Color indicates g -values. Watermarked sequences are detected based on a sequence-level, average g -value threshold of 0.722 for a 0.1% FPR.

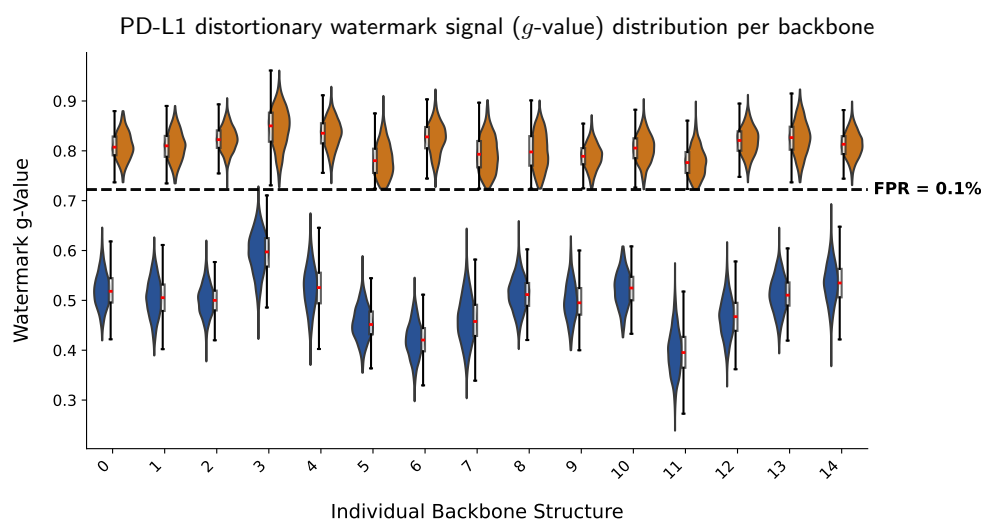


Fig. S26: PD-L1 watermark signal (g -value) distribution by backbone distortionary 0.1 (orange) vs non-watermark (blue) after g -value filtering and protein design filters from Table S1.

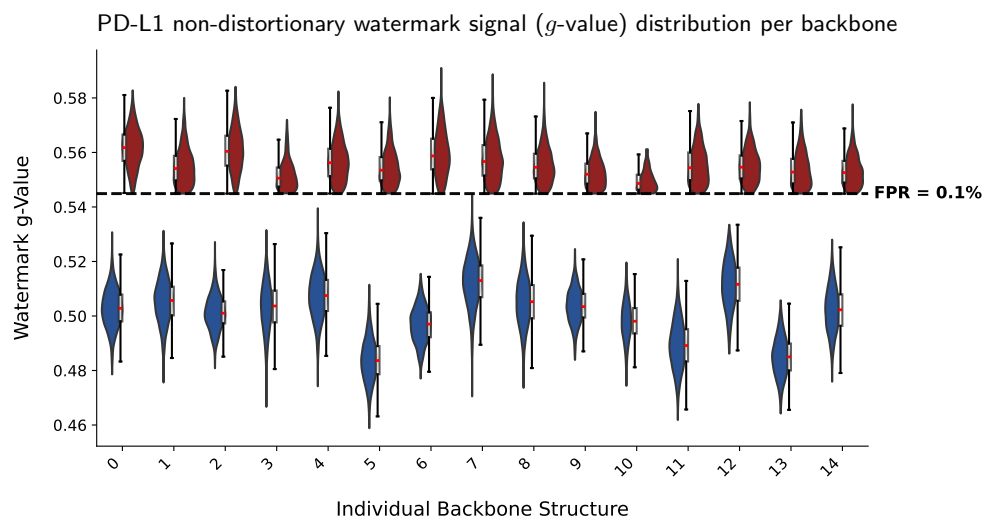


Fig. S27: PD-L1 watermark signal (g -value) distribution by backbone non-distortionary 0.5 (red) vs non-watermark (blue) after g -value filtering and protein design filters from Table S1.

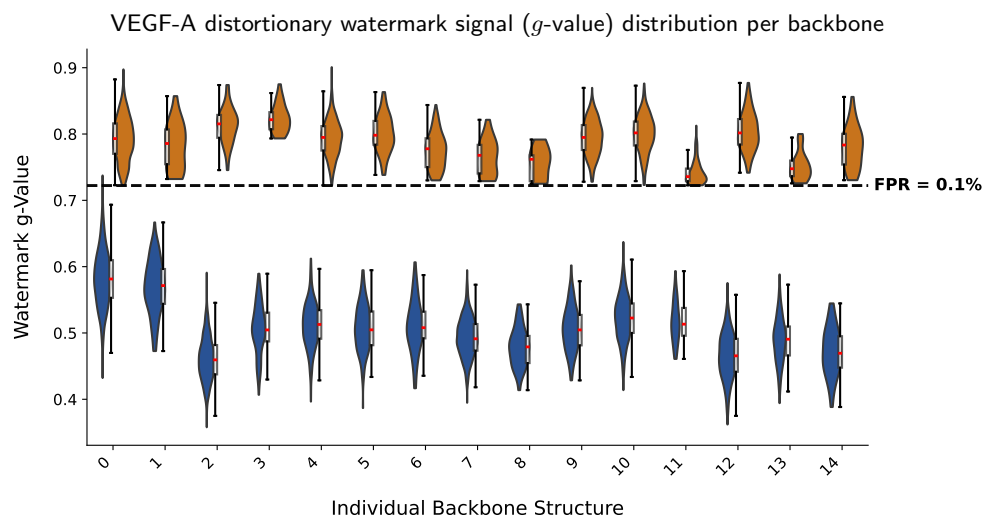


Fig. S28: VEGF-A watermark signal (g -value) distribution by backbone distortionary 0.1 (orange) vs non-watermark (blue) after g -value filtering and protein design filters from Table S1.

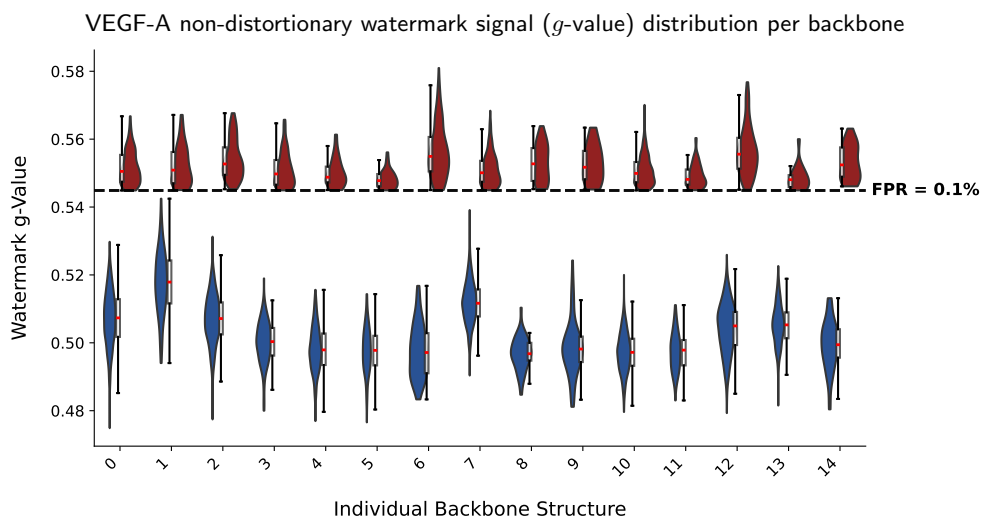


Fig. S29: VEGF-A watermark signal (g -value) distribution by backbone non-distortionary 0.5 (red) vs non-watermark (blue) after g -value filtering and protein design filters from Table S1.

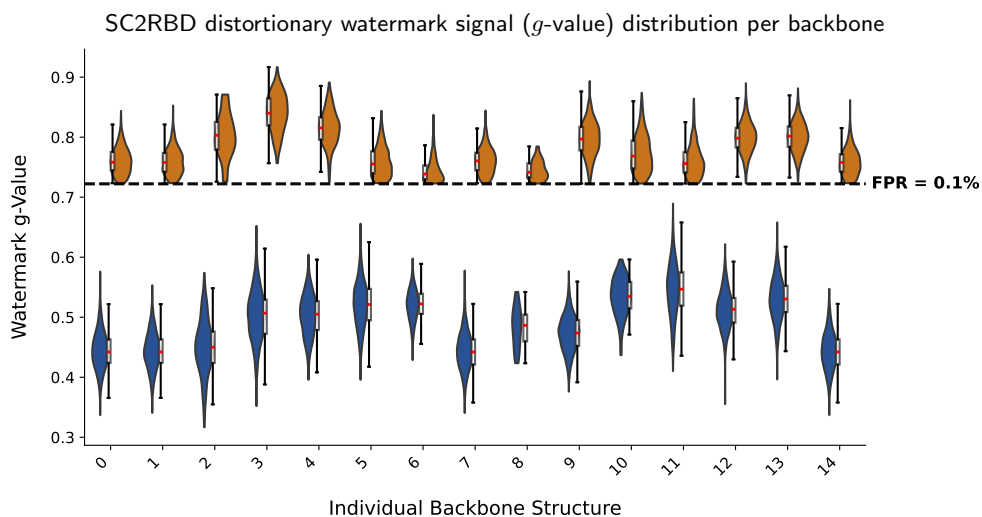


Fig. S30: SC2RBD watermark signal (g -value) distribution by backbone distortionary 0.1 (orange) vs non-watermark (blue) after g -value filtering and protein design filters from Table S1.

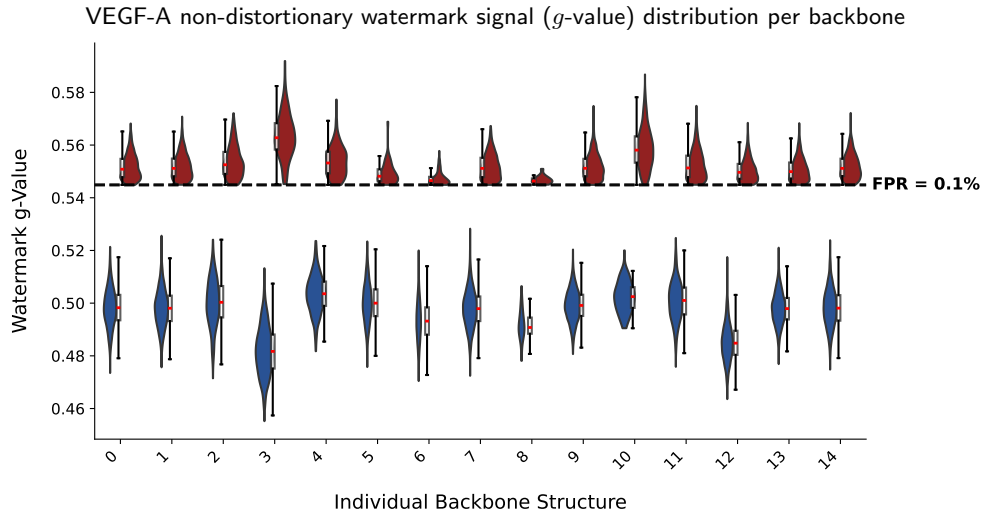


Fig. S31: VEGF-A watermark signal (g -value) distribution by backbone non-distortionary 0.5 (red) vs non-watermark (blue) after g -value filtering and protein design filters from Table S1.

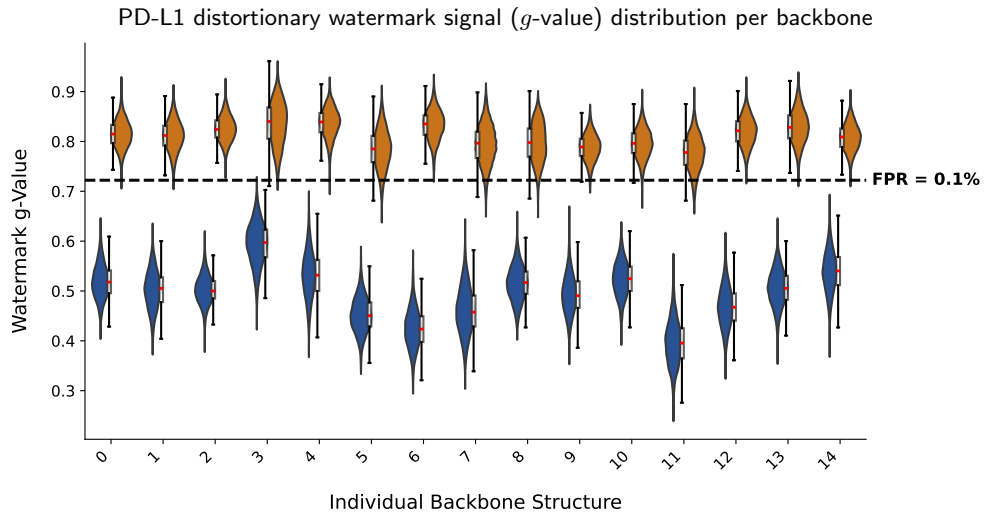


Fig. S32: PD-L1 watermark signal (g -value) distribution by backbone distortionary 0.1 (orange) vs non-watermark (blue) before any filtering is applied to the resequenced population.

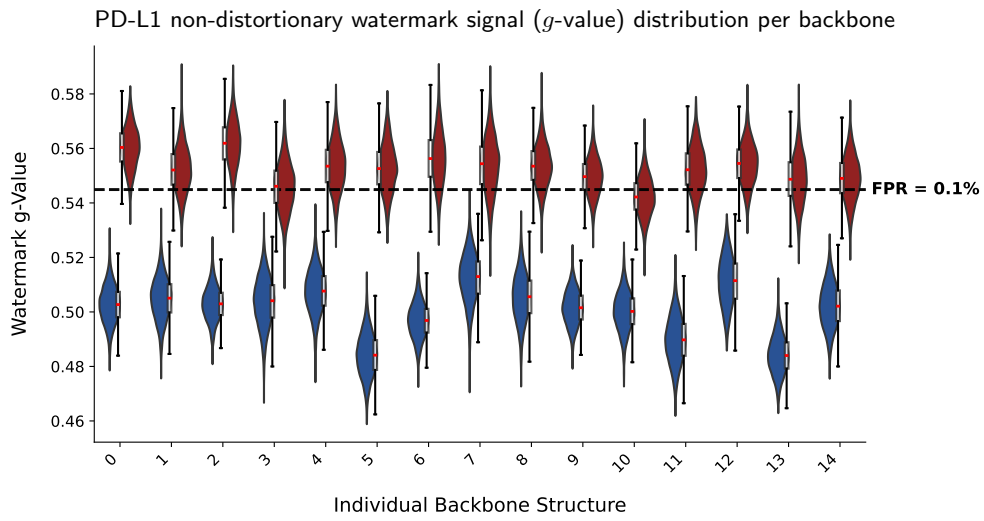


Fig. S33: PD-L1 watermark signal (g -value) distribution by backbone non-distortionary 0.5 (red) vs non-watermark (blue) before any filtering is applied to the resequenced population.

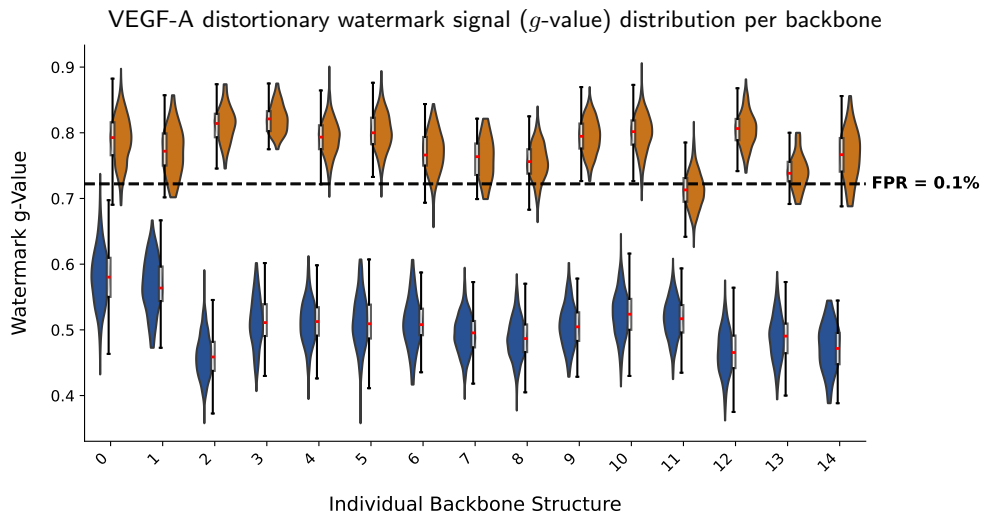


Fig. S34: VEGF-A watermark signal (g -value) distribution by backbone distortionary 0.1 (orange) vs non-watermark (blue) before any filtering is applied to the resequenced population.

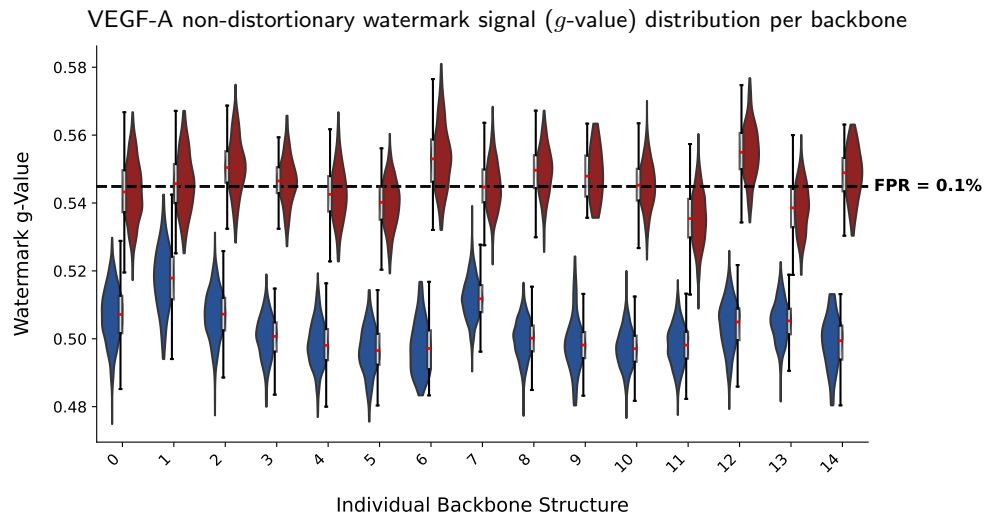


Fig. S35: VEGF-A watermark signal (g -value) distribution by backbone non-distortionary 0.5 (red) vs non-watermark (blue) before any filtering is applied to the resequenced population.

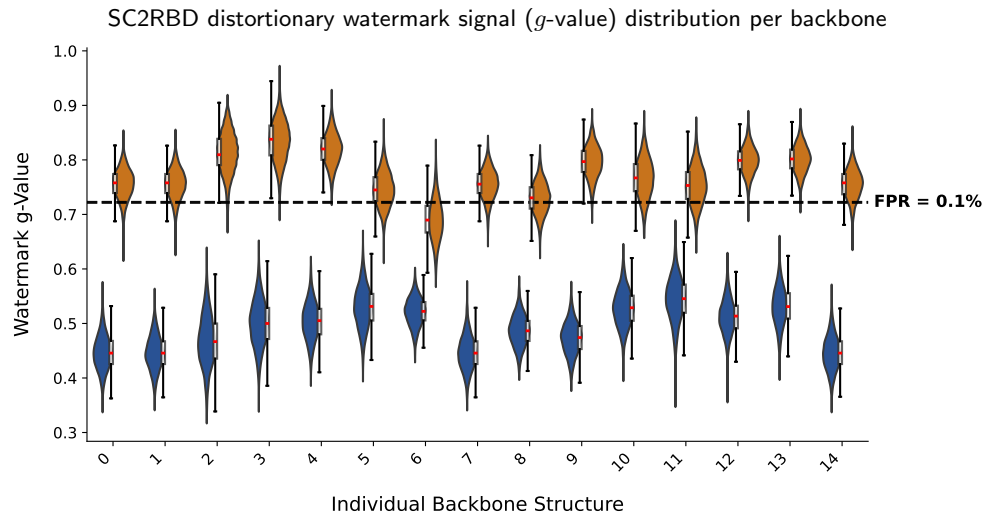


Fig. S36: SC2RBD watermark signal (g -value) distribution by backbone distortionary 0.1 (orange) vs non-watermark (blue) before any filtering is applied to the resequenced population.

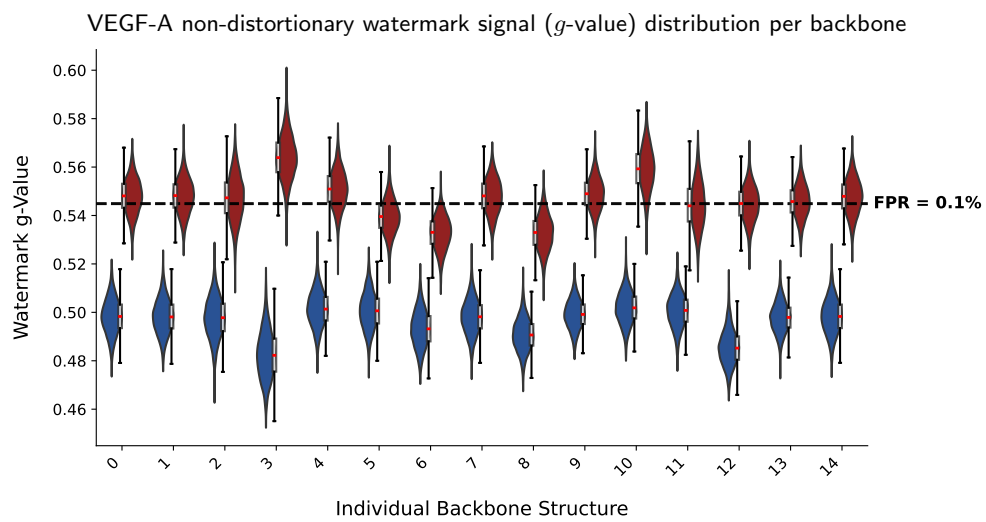


Fig. S37: SC2RBD watermark signal (g -value) distribution by backbone non-distortionary 0.5 (red) vs non-watermark (blue) before any filtering is applied to the resequenced population.

A.9 *In silico* results and analysis

We conducted *in silico* experiments across 23 targets, as detailed in Table S6. We considered 10k candidate designs for each target.

A.9.1 Impact on structural metric distributions

We assessed the impact of watermarking on the distributions of key structural metrics used for design filtering. Figure S38A shows the distributions of predicted template modeling score (pTM) and TM-score across the different watermarking settings. Consistent with the main paper (Figure 3C), the impact of watermarking on these quality metrics is negligible, except for a slight downward shift in the distortionary setting.

A.9.2 Relationship between sequence length, entropy, and detectability

The detectability of SynthIDBio-sequence is influenced by the sequence length and the entropy of the generation process. Figure S38B examines the relationship between sequence length and average g -value. Across all 23 targets we evaluated, a slight negative correlation can be seen in terms of average g -value, while the spread of g -values across designs also reduces. This is in line with expectations from [1]. In protein design, longer sequences may impose more structural constraints, potentially reducing the per-residue entropy available for watermarking.

Figure S38C investigates the relationship between entropy and detectability directly, plotting average g -values against the average log-probability of the sequences, assigned by ProteinMPNN. The results clearly show that higher entropy (lower log-probability) correlates with higher average g -values for the watermarked groups. The separation between watermarked and non-watermarked distributions increases with temperature (0.1 to 0.5), and the distortionary setting achieves the highest separation.

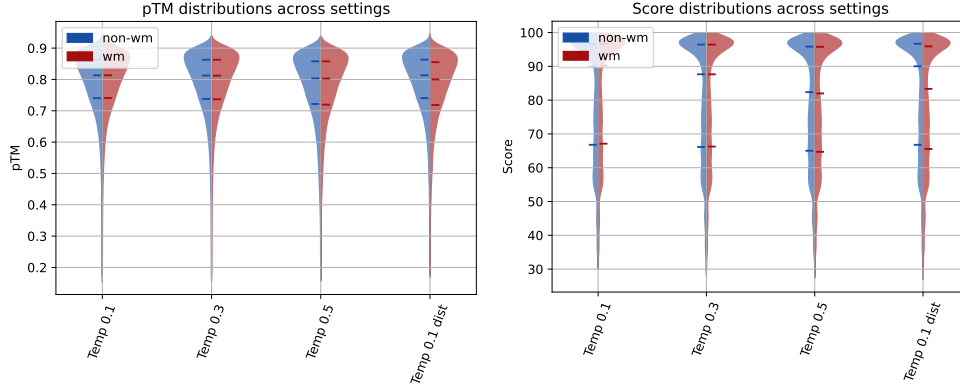
A.9.3 Target specific performance

We analyzed the variability in watermarking performance across the 23 different binding targets (Table S6). Figure S40 details the change in pass rates and the true positive rate (TPR) for the non-distortionary 0.5 setting. Performance varies significantly across targets, likely reflecting differences in the entropy associated with designing binders for different targets. Pass rate varies widely by target: pass rates may increase or decrease significantly, without (top) or with (middle) additional g -value filtering. The TPR at 1%FPR (bottom) remains high for most targets with >90%, with only the TPR for Barnase dropping below 90%.

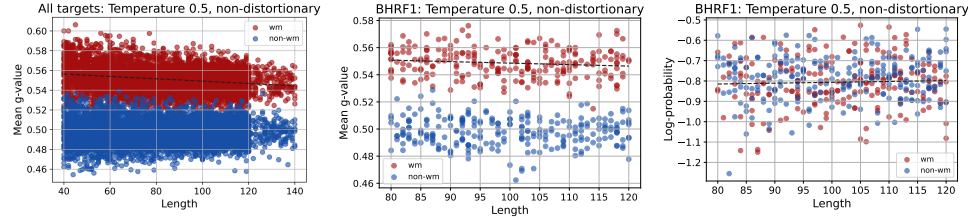
A.9.4 Impact of reduced pass rate

In Figure 3 in the main paper, we show how watermarking and g -value thresholding affects *in silico* pass rates. Figure S39 shows the same plot translated to a multiple of candidates needed to obtain the same number of passed designs as without SynthIDBio-sequence. For example, considering non-distortionary 0.5 watermarking, with a 0.1% FPR threshold, the pass rate reduces by 41.7% across a range of targets

A Metric distributions



B g -values by length



C g -values by ProteinMPNN average log-probability

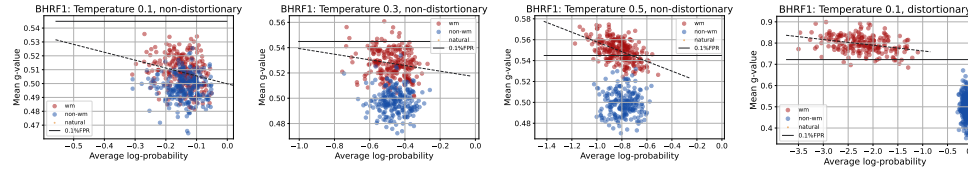


Fig. S38: A Additional metric distributions including predicted template modeling (pTM, left) and TM score (right), supporting the minimal impact of watermarking on filtering when not employing an additional g -value based filter. **B** Mean g -value plotted against sequence length for temperature 0.5, non-distortionary, across all 23 targets (left). As expected from SynthID-text, average g -value goes down for longer sequences. This can be explained by less entropy for longer sequences as shown for BHRF1 (middle and right). **C** Mean g -values plotted against average log-probabilities of the predicted sequences as measured by ProteinMPNN. We plot watermarked and non-watermarked sequences, indicating that lower log-probabilities, i.e., higher entropy, results in larger average g -values for the watermarked group. It can also be seen that distortionary watermarking artificially increases entropy to increase watermarking signal.

Protein Name	PDB Code	Target Motif	Target Chain Hotspot Residues
Anaplastic lymphoma kinase (ALK)	7NWZ	B678-986	B972
BHRF1	2WH6	A2-158	A65,A74,A77,A82,A85,A93
Barnase	1XIU	A1-110	A73
Human CD3-e/d (CD3d)	1XIW	B1-74	B38
CD86	1I85	B1-109	B31
Complement receptor 2 (CR2)	1GHQ	B69-129/C70-134	B86/C131
Influenza A hemagglutinin (H1)	5VLI	A5-48,A281-325/B501-559,B594-670	B521,B545,B552
Interleukin 10 receptor alpha (IL10Ra)	1LQS	R2-208	R43,R190
Interleukin 17 alpha (IL17a)	4HSA	A41-131/B19-127	A95,B43,B67
Interleukin-2 receptor alpha chain (IL2Ra)	2ERJ	A1-165	A42
Cytokine receptor common gamma chain (IL2Rb_gc)	2ERJ	B6-205/C35-224	B134,C208
Interleukin 7 receptor alpha (IL7Ra)	3DI3	B17-209	B58,B80,B139
Human Insulin Receptor (InsulinR)	4ZXB	E6-155	E64,E88,E96
Leukocyte tyrosine kinase (LTK)	7NX0	B66-378	B366
PD-L1	5O45	A17-132	A56,A115,A123
Platelet-derived growth factor receptor (PDGFR)	3MJG	X122-312	X185,X242,X268
Receptor Tyrosine Kinase-Like Orphan Receptor 1 (ROR1)	6BA5	M311-391	M330,M383
SC2RBD	6MOJ	E333-526	E485,E489,E494,E500,E505
Transforming growth factor beta-1 (TGFb1)	3KFD	A1-112/B1-112	A92,B6,B64
TEK receptor tyrosine kinase (TIE2)	2GY7	B122-212	B164
Tropomyosin Receptor Kinase A (TrkA)	1WWV	X282-382	X294,X296,X333
VEGF-A	1BJ1	V14-107/W14-107	W81,W83,W91

Table S6: The 23 structures used as targets for binders for *in silico* assessment of SynthIDBio-sequence. For crystal structures with bound complexes, *Protein Name* defines the motif used as target. Hotspot residues are those which we require the target protein to have in contact with a design (see hotspot accuracy in Table S1).

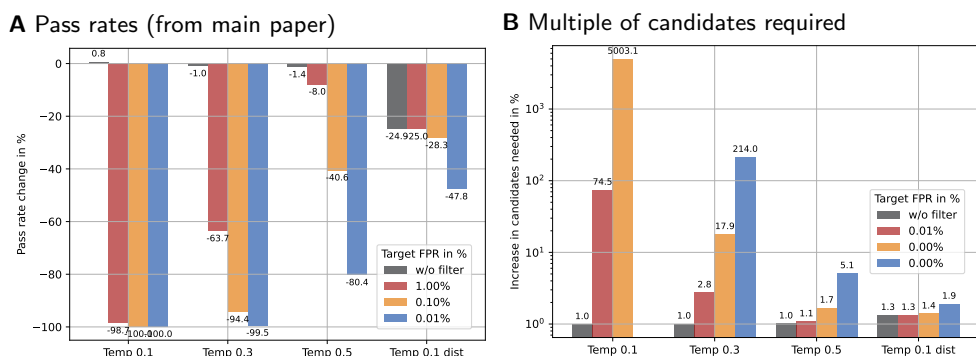


Fig. S39: A Impact of SynthIDBio-sequence on pass rate; this is the same plot as in Figure 3 in the main paper. **B** These pass rates can be translated to a multiple. This multiple indicates how many more candidates SynthIDBio-sequence needs to consider to obtain the same number of passing candidates as without watermarking.

(see Table S3). This means that we need to generate $100\% / (100\% - 41.7\%) = 1.715$ times the candidates to obtain the same number of passed designs. While we refer to Section A.5 for a detailed cost analysis showing that this multiple does not translate to significant cost increases, this gives an idea of how many additional candidates are needed depending on the target FPR. Note that the thresholds and sequences evaluated in Section A.5 differ from Figure S39 such that numbers are not fully comparable.

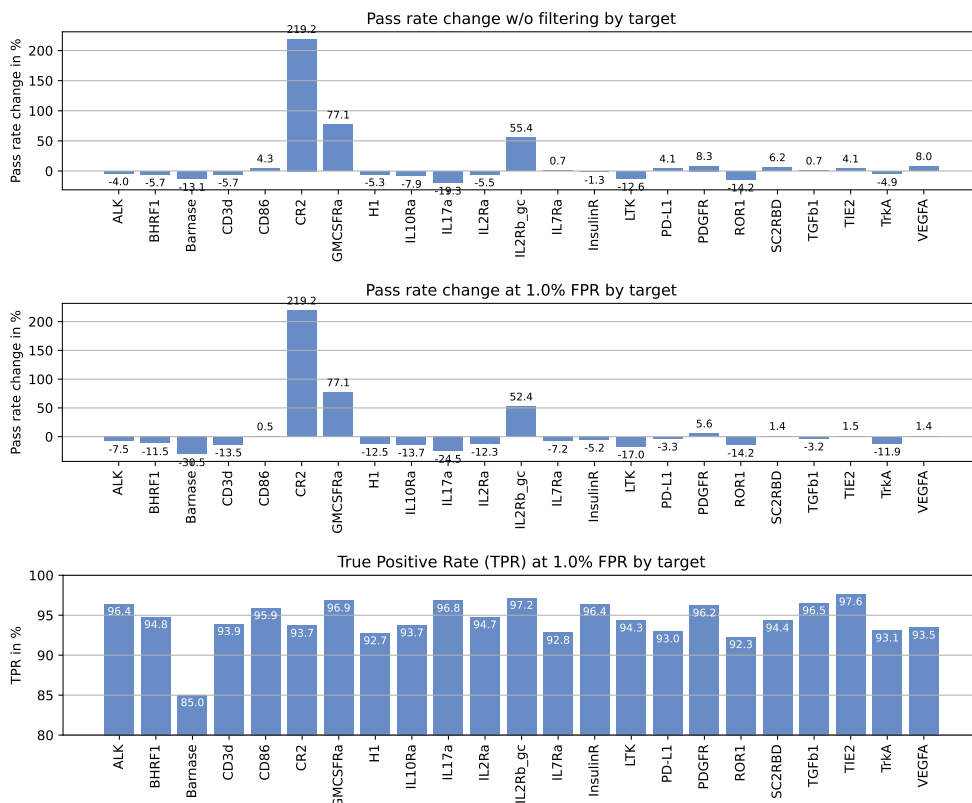


Fig. S40: Pass rate change and TPR by target for temperature 0.5, non-distortionary, with and without g -value filtering for 1% FPR. Also, TPRs per target for 1% FPR.

A.10 Robustness evaluation

In the context of provenance for protein design, we define a threat model [6] by the attacker's knowledge and resources. The former range from no generative AI and protein design knowledge to the ability to independently run state-of-the-art protein design pipelines. For the latter, we distinguish between no resources, reasonable compute resources, and access to a wet lab. We considered a few examples of concrete threat models:

- **Random substitutions** of residues within the designed sequence requires no specific knowledge about AI for protein design and negligible resources.
- **Informed substitutions** requires some knowledge about protein design but does not require familiarity with AI for protein design; if knowledge about systems like AlphaProteo exist, such attacks can be complemented with a proxy of AF 3-based metric filters. These informed substitutions can be based on amino acid replacement matrices and/or known H -gram lengths.

- **Resequencing** requires the ability to run the ProteinMPNN resequencer, which is applied one or multiple times to remove the watermark while expecting that protein binding remains unaffected. Knowledge of the used AF 3 metric filters, or a proxy thereof, can be helpful, but no wet lab access for validation may be available.

We performed random and informed substitutions attacks on a population of BHRF1 binders that passed both the *in vitro* thresholds (Table S1) and an FPR threshold of 0.1% (Table S2). Random substitutions, as implied, replace a residue with a randomly selected replacement. For informed substitutions, we used Grantham’s distance [7] and experimental exchangeability [8] as substitution matrices. We utilized the substitution matrices by inserting the next best scoring substitution until a target percentage of the sequence was modified, with ties broken randomly. As Grantham’s distance is symmetric, residues could only be substituted a single time. For the results described here, we considered the entire sequence equivalently. However, more complex approaches could limit the number of substitutions within an *H*-gram to one, fix specific residues (e.g., hotspot interface residues), or consider alternative substitution matrices better fit to specific proteins.

For these attacks to be successful, an attacker would need to be confident that their substitutions removed the watermark and preserved protein function. While the latter could be predicted with AF 3 structural metrics, the former will be, at best, a probabilistic estimate. As shown in Figures S41 and S42, the three substitutions schemes were virtually identical in terms of their impact on detectability. The detectability percentages effectively represent the likelihood that a given attacked design would remain detectable with access to the secret watermarking key. For our assessments, we considered a high confidence level (e.g., >90%) as an appropriate target. Even at the more stringent FPR detection thresholds, this would require at least 15% of the protein residues to be substituted.

Across all three attack schemes, even a small number of substitutions dramatically decreased *in silico* pass rates, with pass rates from as few as 5% of residues being substituted falling to a range from 18.8% (random, non-distortionary watermarking) to 35.2% (Grantham’s distance, distortionary watermarking). In contrast to detectability, the three substitution schemes were not equivalent in their impact on *in silico* pass rates. Substitutions based on Grantham’s distance were far superior, with experimental exchangeability only slightly outperforming random substitutions. Further, substitution attacks, as expected, were substantially less effective at removing the watermark in distortionary watermarked sequences compared to non-distortionary as more positions in the sequence provide watermarking signal.

All together, these results demonstrate that simplistic substitution methods are likely to be an ineffective attack. Even with the use of a Grantham’s distance-based substitution matrix, the scale of substitutions needed to reliably remove the watermark ($\geq 15\%$ of the residues) would reduce the population of structure filter-passing designs to 12.0% and 14.7% for non-distortionary and distortionary watermarking, respectively.

In the main paper, we consider a threat model for watermark removal utilizing ProteinMPNN to resequence watermarked designs. Turning attention to this attack,

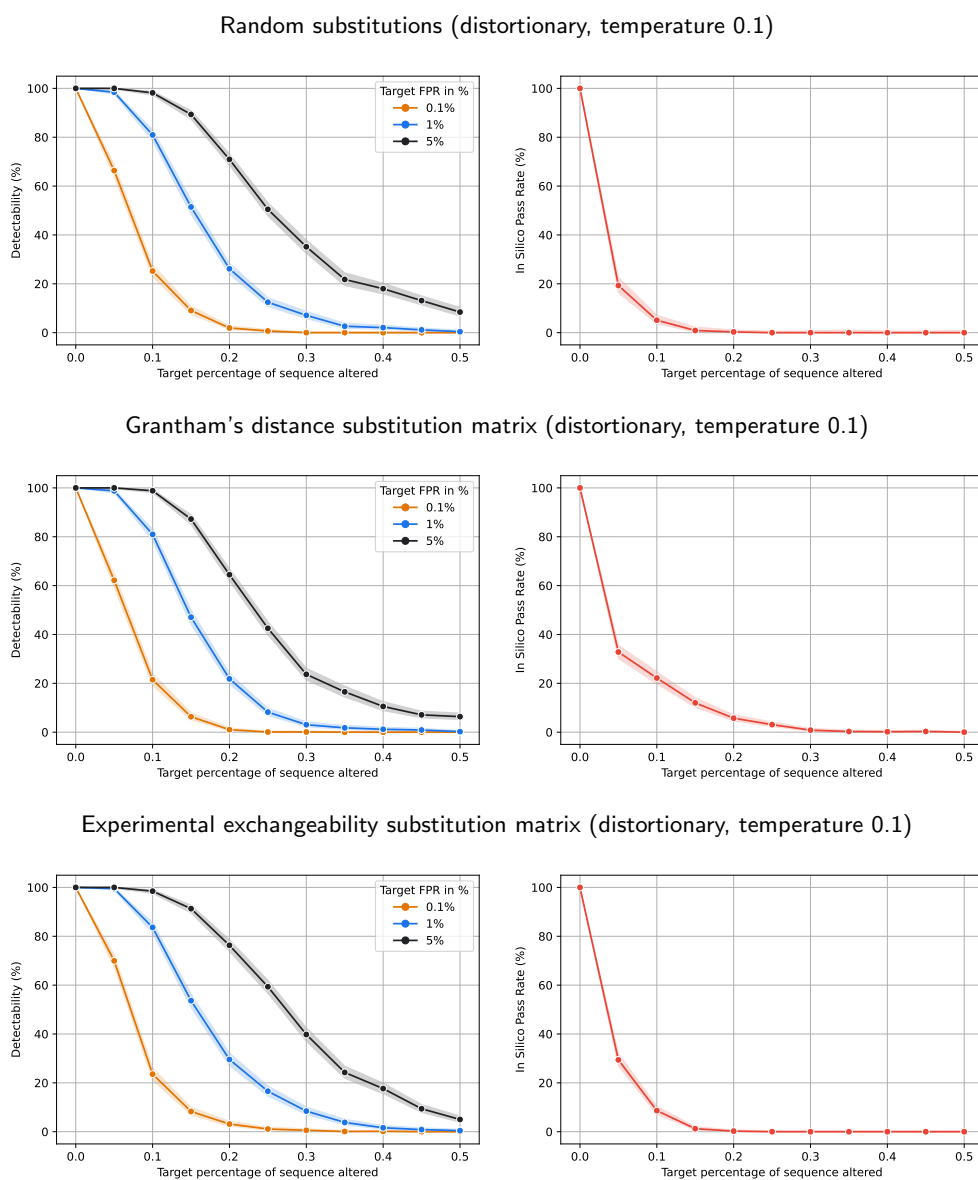
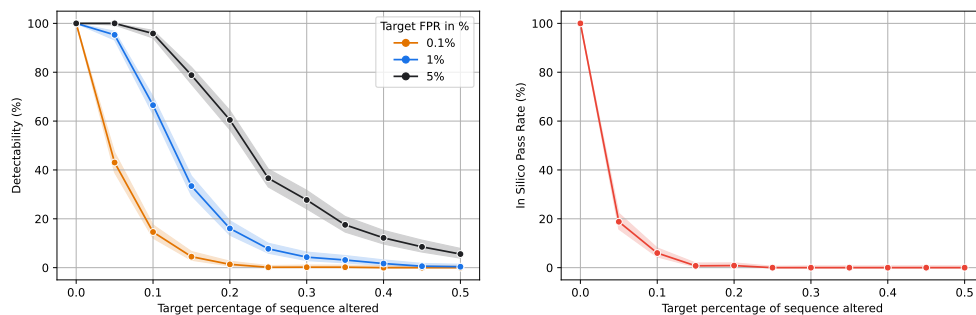
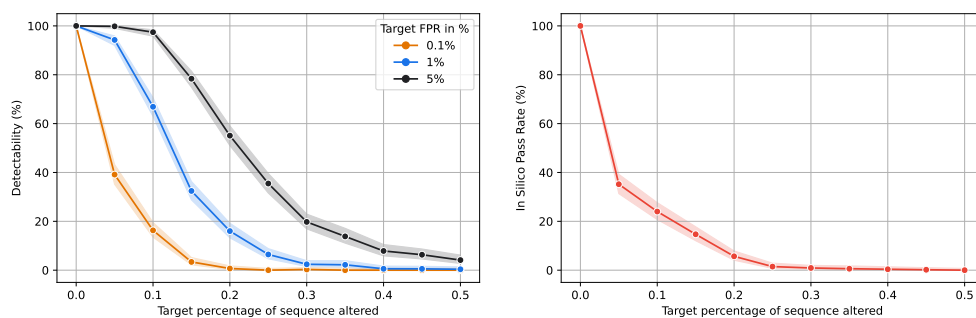


Fig. S41: Detectability and *in silico* pass rates for different substitution attacks on distortionary watermarked BHRF1 binders ($n = 33752$). Within a substitution attack, each design is attacked at most once to ensure independence. Data is depicted with 95% Wilson confidence intervals.

Random substitutions (non-distortionary, temperature 0.5)



Grantham's distance substitution matrix (non-distortionary, temperature 0.5)



Experimental exchangeability substitution matrix (non-distortionary, temperature 0.5)

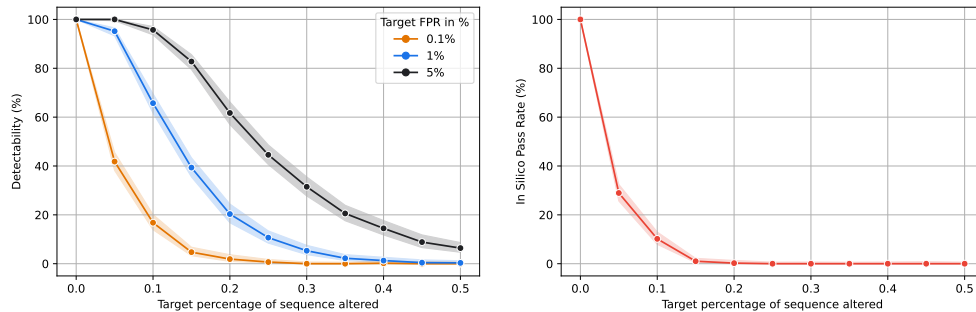


Fig. S42: Detectability and *in silico* pass rates for different substitution attacks on non-distortionary watermarked BHRF1 binders ($n = 65756$). Within a substitution attack, each design is attacked at most once to ensure independence. Data is depicted with 95% Wilson confidence intervals.

Table S7 shows a full table of resequencing hit rates used to estimate the efficacy of resequencing for watermark removal. Here, we define:

- **Initial binder hit rate:** The rate at which designs selected through AlphaProteo are binders *in vitro*. These values are derived from historical AlphaProteo campaigns (data not shown).
- ***In silico* pass rate:** The rate at which resequenced sequences pass the AlphaProteo filters.
- **Resequencer hit rate:** The rate at which resequenced sequences **that passed our design filters** produced binders *in vitro* based on our *in vitro* campaign data (Figure 2).

And consider the attacker either having access to the AF3 structural filters (“w/ filters”) or not (“w/o filters”). The former corresponds to the strongest threat model we considered, although other approaches, such as only resequencing non-interface residues, may lead to higher hit rates. A malicious actor able to run our attack with filters has the majority of tools and resources to enable protein design from scratch. Within a punitive implementation of watermarking, we assess that this significantly reduces the motivation of such an attacker to use a watermarked protein design pipeline in the first instance.

Initial Conditions	Target	Initial binder hit rate	Resequencer hit rate	In silico pass rate	w/o filters hit rate post attack	w/ filters hit rate post attack
Known binder	PD-L1	100%	70.3%	28%	19.7%	70.3%
Known binder	VEGF-A	100%	66%	15%	10.5%	66%
Known binder	SC2RBD	100%	97%	34%	33.0%	97%
Unknown binder	PD-L1	15% (AP)	70.3%	28%	3%	10.5%
Unknown binder	VEGF-A	33% (AP)	66%	15%	3.3%	21.8%
Unknown binder	SC2RBD	12% (AP)	97%	34%	4.0%	11.6%

Table S7: Estimated binding hit rates for known and unknown binders against various targets after the resequencing-based attack.

Figure S43 shows additional *in silico* results for the resequencer-based attack. Figure S43A highlights the estimated binder hit rate starting from an *unknown* binder, complementing Figure 3D in the main paper. Figure S43B shows the effect of this attack on a sample of 38,396 distortionary 0.5 sequences, after the attack only 15 have *g*-values above the 0.1% FPR threshold.

A.11 Application to RFdiffusion

In Figure S44, we show complementary results for Figure 3A-B based on RFdiffusion backbones. To this end, we downloaded backbones from the official repository¹ and sampled 35 designs per backbone, totalling 9975 designs. We follow the same analysis as for Figure 3A-B except for higher FPR due to the lower sample size and see that our

¹<https://figshare.com/s/439fdd59488215753bc3?file=40226533>

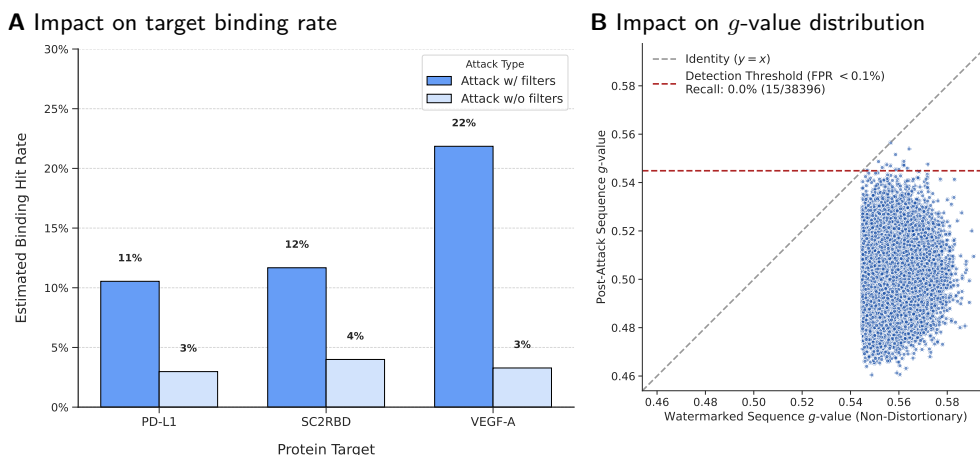


Fig. S43: **A** Complementary to Figure 3D, we show resequencing impact on binder hit rate assuming starting from an unknown binder. **B** We show the impact of resequencing on the g -value distribution of watermarked 0.5 non-distortionary designs against PD-L1. Only 15/38396 sequences continue to be above the g -value threshold.

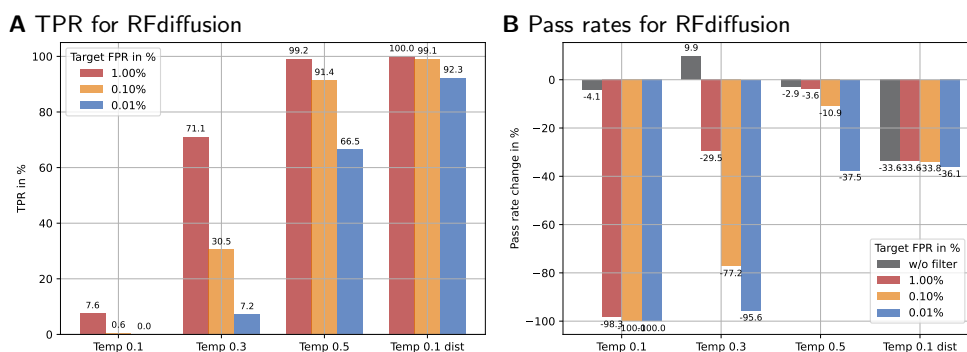


Fig. S44: Results for SynthIDBio-sequence using RFdiffusion, complementary to results for Figure 3A-B. We show **A** TPRs and **B** pass rate change for various FPRs across watermarking settings.

results generalize very well, especially for distortionary 0.1 and non-distortionary 0.5. In contrast to the analysis for SynthIDBio-sequence in Figure 3A-B, for RFdiffusion we sample a new backbone for each design rather than a fixed number of backbones. As we argue in our *in vitro* validation, fixing backbones can have a significant impact on the distribution of average g -values for detectability.

Appendix B SynthIDBio-structure

B.1 Watermark diffusion loss in AF 3 notation

We restate our watermarking loss in terms of the notation of [9, Supplementary material, Section 3.7.1]. To this end, we first consider their Equation (6), which represents the diffusion loss for AF 3’s diffusion module:

$$\mathcal{L}_{\text{diffusion}} = (\hat{t}^2 + \sigma_{\text{data}}^2)/(\hat{t} + \sigma_{\text{data}})^2 \cdot (\mathcal{L}_{\text{MSE}} + \alpha_{\text{bond}} \cdot \mathcal{L}_{\text{bond}}) + \mathcal{L}_{\text{smooth_lddt}} \quad (\text{B9})$$

At its core is a mean squared error (MSE) loss $\mathcal{L}_{\text{MSE}}$ computed after the denoised structure $\vec{x}$ (corresponding to $D(x + \epsilon, \sigma)$ in our extended methods) has been aligned to the ground truth structure $\vec{x}^{\text{GT}}$ (corresponding to x in our extended methods). It includes an additional loss $\mathcal{L}_{\text{bond}}$ to ensure that bonded ligands have the correct length, and a smooth IDDT loss $\mathcal{L}_{\text{smooth_lddt}}$. The loss is weighted by a term based on the sampled noise level $\hat{t}$ (corresponding to σ in our extended methods) and a fixed data variance σ_{data} . In our extended methods, this term, $(\hat{t}^2 + \sigma_{\text{data}}^2)/(\hat{t} + \sigma_{\text{data}})^2$, represents the noise dependent weight $\lambda(\sigma)$.

In this notation, our watermarked diffusion loss is

$$\mathcal{L}_{\text{diffusion}} = (\hat{t}^2 + \sigma_{\text{data}}^2)/(\hat{t} + \sigma_{\text{data}})^2 \cdot (\mathcal{L}_{\text{MSE}} + \alpha_{\text{bond}} \cdot \mathcal{L}_{\text{bond}}) + \mathcal{L}_{\text{smooth_lddt}} + \mathcal{L}_{\text{wm}} \quad (\text{B10})$$

To define the loss $\mathcal{L}_{\text{wm}}$ we introduce an additional denoised structure $\vec{x}^{\text{Ref}}$ called the reference denoised structure that we obtained from running the original AF 3 diffusion module. That is, in the notation of our extended methods, $\vec{x}$ is obtained from $D_{\hat{\theta}}$ (the denoiser fine-tuned with watermark loss) and $\vec{x}^{\text{Ref}}$ is obtained from D_{θ} (the original, fixed AF 3 denoiser). With this, we can define:

$$\mathcal{L}_{\text{wm}}(\vec{x}, \vec{x}^{\text{Ref}}, \vec{x}^{\text{GT-aligned}}) = \text{CE}(\vec{x}^{\text{Ref}}, \vec{x}) + \text{CE}(\vec{x}^{\text{GT-aligned}}, \vec{x}) \quad (\text{B11})$$

using

$$\text{CE}(y, \tilde{y}) = \log(1 - d(T(y))) - \log d(T(\tilde{y})). \quad (\text{B12})$$

as defined in our extended methods, with T being a transformation that adds uniform noise to the positions of y and d being the watermark detector.

For fine-tuning, we follow the description for stage “Fine-tuning 3” in the AF 3 supplementary material: [9], supplementary Table 6). We use a reduced sampling weight for disorder PDB distillation data (from 0.02 to 0.01).

We include simplified pseudocode of our detector architecture in Listing 1.

```
1 def extract_distances(positions, atom_mask):
2     """Consecutive-atom distances within each residue."""
3     d = safe_sqrt(sum((positions[:, 1:, :] - positions[:, :-1, :]) ** 2, axis=-1))
4     mask = correlate_1d(atom_mask, [1, 1]) == 2 # both atoms present
5     return d * mask
6
7 def extract_angles(positions, atom_mask):
8     """Torsion angles over every 4 consecutive atoms in each residue."""
9     a = positions[:, 1:21, :] - positions[:, 0:20, :] # 1->2
```

```

10     b = positions[:, 2:22, :] - positions[:, 1:21, :] # 2->3
11     c = positions[:, 3:23, :] - positions[:, 2:22, :] # 3->4
12     sin_arg, cos_arg = compute_torsion(a, b, c)
13     mask = correlate_1d(atom_mask, [1, 1, 1, 1]) == 4
14     return concat([sin_arg * mask, cos_arg * mask], axis=-1)
15
16 def get_features(positions, atom_mask):
17     """Concatenate selected features; pad/clip to 2048 residues."""
18     feat = concat([
19         extract_distances(positions, atom_mask),
20         extract_angles(positions, atom_mask),
21     ], axis=-1)
22     return pad_or_clip(feat, max_residues=2048)
23
24
25 def watermark_detector(positions, atom_mask, is_training):
26     """PointNet watermark detector head."""
27     # positions is [N_res, 24, 3], atom_mask is [N_res, 3] for each structure
28     if is_training:
29         positions += Uniform(-0.01, 0.01, shape=positions.shape)
30         positions *= atom_mask[..., None]
31
32     residue_mask = any(atom_mask, axis=-1, keepdims=True)
33     residue_mask = pad_or_clip(residue_mask, 2048)
34     residue_mask = residue_mask[None, :, :]
35
36     for _ in range(5):
37         x = Conv1D(out_channels=288, kernel_size=11, padding='SAME')(x)
38         x = relu(x)
39         x = residue_mask * x
40
41     N_actual = sum(residue_mask)
42     x = mean(x, axis=1)
43     x = x * (2048 / N_actual)
44
45     logit = Linear(out=1)(x)
46     return squeeze(logit)

```

Listing 1: Python and JAX inspired pseudocode for our watermark detector architecture.

B.2 Detailed detection performance analysis

We investigated the impact of the noise parameter s (uniform noise scale applied during detector training) on detection performance. As noted in the main paper, a minimum of $s = 0.001$, called WM-0.001, is usually required for common file formats such as CIF that save atom coordinates with three decimal points. Larger $s > 0.001$ are beneficial to improve robustness to perturbations of the coordinates after folding. In Figure S45A+B, while all models achieve >99% TPR at 0.1% FPR on unperturbed structures, their robustness to post-generation noise varies. Training with stronger noise usually improves robustness while also slightly worsening RMSD, IDDT or TM score with respect to the ground truth structures. Figure S45C also shows the impact of this trade-off on detectability of shorter structures. Figure S45D+F shows the ROC curves when structures are perturbed with Gaussian noise (standard deviation 0.1 Angstrom). Higher values of s during training lead to significantly more robust watermarks. ROC curves without perturbations are not informative as ROC AUC is close to 100%. It is also clear that longer structures are easier to detect.

B.3 Analysis of watermark imperceptibility

To confirm that the watermark does not introduce biologically implausible distortions, we conducted a detailed analysis of structural feature distributions, including atom-atom distances, bond angles, and torsion angles. Figures S46 to S48 present representative examples of these distributions for $s = 0.001, 0.01$, and 0.1 , referred to as WM-0.001, WM-0.01, and WM-0.1 in figures. In contrast to the main paper, where we show hand-picked features that exhibit significant shifts for $s = 0.001$, the vast majority of features show negligible deviation from the baseline AF 3 distributions. This is particularly true for angles, which are less affected than atom-atom distances. This highlights that the watermark does not affect the majority of structural features. We can also see that the watermark heavily relies on few features, particularly the $C-C_\alpha$ atom distances. Increasing s typically leads to larger shifts in these feature distributions with $s = 0.1$ exhibiting very significant shifts in atom-atom distances, in particular. To summarize these results, we computed the Wasserstein distances between these histograms. For atom-atom distance distributions, we find that in 72.5% of cases (i.e., amino-acid, atom-atom pairs), the distance between SynthIDBio-structure vs. ground truth and AF 3 vs. ground truth is not statistically significant. For torsion angles, this is the case for 96.9%; for bond angles it is 47.2%. Moreover, in a bootstrap analysis, computing the standard error of mean, we find that the overall change is within one standard deviation for torsion and bond angles and within two standard deviations for atom-atom distances.

We quantify these changes more systematically in Figure S45D which analyzes the average change in intra-residue atom-atom distances relative to the error between the distance predicted by AF 3 and the distance in the ground truth structure across different values of s . The impact remains minimal unless s is increased substantially to $s = 0.1$. We note that these are changes relative to the *error* of AF 3. If a particular atom-atom distance predicted by AF 3 is very close to the ground truth, even a high percentage change in Figure S45D may be small in absolute terms.

Finally, Figure S50 additionally shows distributions of RMSD and IDDT, comparing SynthIDBio-structure and AF 3 to the ground truth as well as SynthIDBio-structure to AF 3.

B.4 Interference of watermarks

In Figure S51 we evaluated the detector of SynthIDBio-structure for $s = 0.001$ and $s = 0.1$ on watermarked structures predicted for all $s = 0.001, 0.01$, and 0.1 . We found that the $s = 0.001$ detector picks up on the $s = 0.01$ watermarks. This is not too surprising as these models were trained exactly the same way except for a different s ; this includes architecture, weight initialization, training data ordering, etc. This means models will likely learn the same watermark patterns. We hypothesize that this can be fixed by injecting additional negative structures with a different watermark during training.

B.5 Structure Watermarking Example Predictions

For each PDB, we show the top ranked prediction for $s = 0.001$ (orange), ground truth (grey) and AF 3 (blue), with the left image showing AF 3 overlay with ground truth (GT), the middle image showing $s = 0.001$ overlay with GT and the right image showing AF 3 overlay with $s = 0.001$ in Figures S52, S53, S54, S55, S56, S57, S58, S59 and S60. Across selected PDBs, we show a range of predicted biomolecules: protein, ions, ligands, RNA and PTMs. All selected PDBs were deposited after the AF 3 training cut off. We selected PDBs of varying structural accuracy and similarity between the watermarked ($s = 0.001$) and non-watermarked (AF 3) predictions as detailed in Table S8. For each PDB, we visualize the top ranked sample and we report the corresponding IDDTs between $s = 0.001$ vs AF 3, $s = 0.001$ vs GT and AF 3 vs GT for each prediction.

Table S8: Comparison of IDDT scores between $WM - 0.001$ and AF 3 against Ground Truth (GT) across performance buckets.

IDDT $s = 0.001$ vs. AF 3 Bucket	Fraction of structures in bucket	Target	IDDT $s = 0.001$ vs. AF 3	IDDT $s = 0.001$ vs. GT	IDDT AF 3 vs. GT
99 - 95	45.84%	7PPA (S52)	99.00	88.85	88.82
99 - 95		7X7A (S53)	96.30	71.52	71.10
95 - 90	5.17%	7P25 (S54)	95.00	89.61	91.26
95 - 90		7XSP (S55)	94.53	71.45	70.58
90 - 85	0.86%	7SJN (S56)	89.92	77.45	77.27
90 - 85		7Y81 (S57)	88.13	67.70	68.06
85 - 80	0.33%	7UZP (S58)	84.77	67.55	70.34
85 - 80		7OIX (S59,S60)	84.63	68.35	69.07

Appendix C Extended related work

C.1 Biosecurity for generative biology

AI models promise to transform biotechnology research and application. However, there are legitimate concerns that models could be intentionally misused by malicious actors, including for the production of harmful biological agents [18]. To address this challenge, recent research focuses on various aspects of biosecurity in the context of generative AI in biology. This ranges from work on governance challenges [19, 20], to technical discussions and approaches [21], and concrete recommendations from the academic and industrial community [22, 23]. A common distinction [24] in these discussions is between general-purpose foundation models such as LLMs and specialist

models such as AlphaProteo [25], RFDiffusion [5], or ProteinMPNN [26]. For the former, various studies have evaluated to what extent and with which requirements user can leverage LLMs to design and synthesize dangerous agents [27–31]. Moreover, frontier AI labs have produced research evaluating the biosecurity risk of their products, often subsumed within the term dangerous capabilities [32–34]. There is considerably less work regarding specialist models. However, there is some agreement that “safeguard[ing] the digital-to-physical frontier” [23] is a critical aspect and motivates additional technical development in and new policy for nucleic acid synthesis screening specifically.

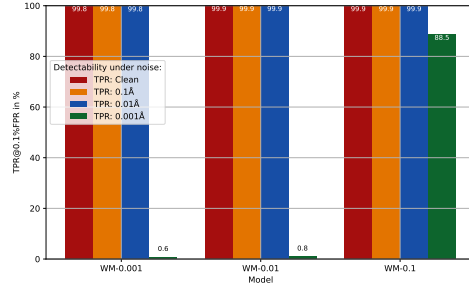
C.2 Provenance and screening for generative biology

Recent work demonstrates the feasibility of designing and synthesizing dangerous agents [35] and highlights the critical role of synthesis screening. SecureDNA [36, 37] proposes a privacy-preserving way of checking submitted sequences against an up-to-date hazard database. Beyond hazard screening, [18] particularly emphasizes the importance of establishing provenance, i.e., the origin, of sequences submitted for synthesis. Its proposal involves a central database, similar to SecureDNA, against which synthesis providers can check submitted sequences. There are also lines of work on post-hoc detection of sequence origin, both in the context of generative biology [38] and lab of origin detection [39, 40].

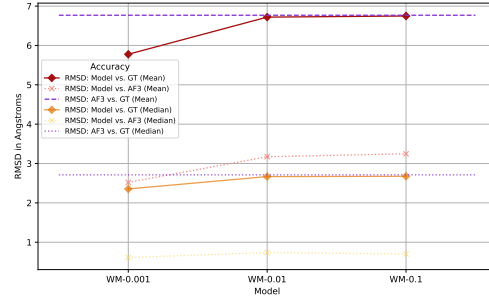
C.3 Provenance for generative text and multimedia

In the context of general-purpose generative models, watermarking has received considerable attention across modalities. For text, SynthID-text [1] has been deployed to watermark Google’s flagship model Gemini. Such approaches are typically based on biasing the next-token distribution in a detectable manner [41, 41–44], see [45] for a comprehensive review. Alternative approaches can also rely on poisoning [46]. All of these approaches introduce a trade-off between detectability and distortion of the original, non-watermarked text distribution. For multimedia, including image, video, and audio, approaches follow either a post-hoc approach where models are trained to add and then detect an invisible/inaudible watermark [47–61] or directly bias the sampling distribution of the underlying model [62–64], see [65] for a review. While solutions based on databases and retrieval have been explored across modalities [41, 66–70], watermarking has found much wider adoption.

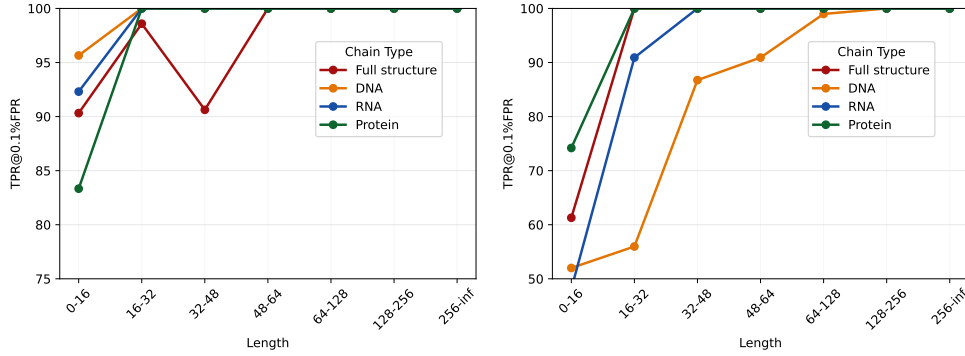
A Detectability



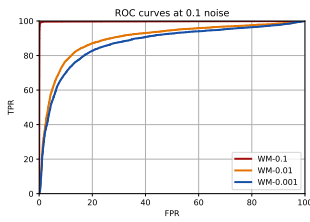
B Accuracy



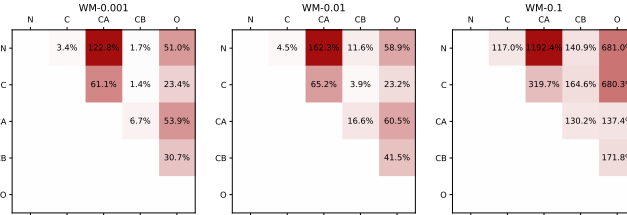
C Results by biomolecule WM-0.01 (left) and WM-0.1 (right)



D ROC curves



E Comparison of atom-atom distance increases by s



F ROC curves by structure length, evaluated after adding Gaussian noise with $\sigma = 0.1$

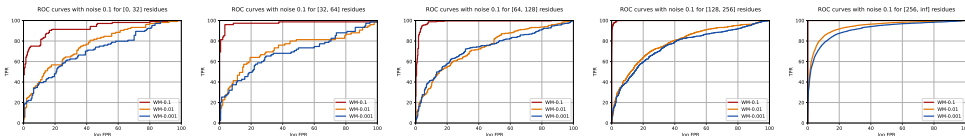


Fig. S45: Supplementary results for SynthIDBio-structure, complementing Figure 4; see text for detailed discussion. **A** Results for all three $s \in \{0.001, 0.01, 0.1\}$, called WM-0.001, WM-0.01, and WM-0.1, in terms of TPR without and with Gaussian noise applied at test time. **B** We also show impact on average RMSD. **C** Detectability by biomolecule and structure length for WM-0.01 and WM-0.1. **D** ROC curves for all models evaluated on perturbed structures with Gaussian noise of standard deviation 0.1 Angstrom. We excluded ROC curves for Gaussian noise with standard deviation < 0.1 as the area under the ROC curve is $> 99.99\%$. **E** Effect of watermarking on average atom-atom distances across amino acids, relative to the change in mean distance between AF3 predictions and ground truth structures. **F** Additional ROC curves by structure length, with Gaussian noise of standard deviation 0.1 Angstrom.

Representative examples of feature distributions for $s = 0.001$

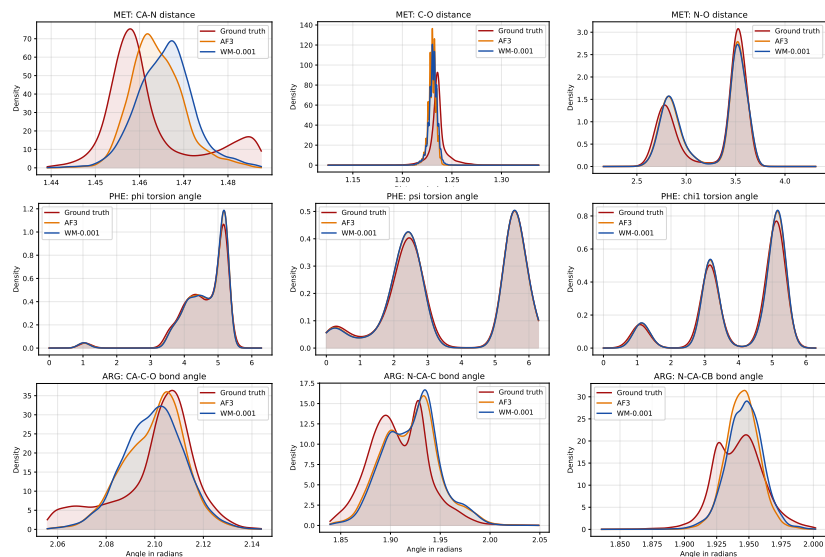


Fig. S46: Additional atom-atom distance, torsion and bond angle distributions for selected amino acids on 1000 short structures for SynthIDBio-structure trained with $s = 0.001$.

Representative examples of feature distributions for $s = 0.01$

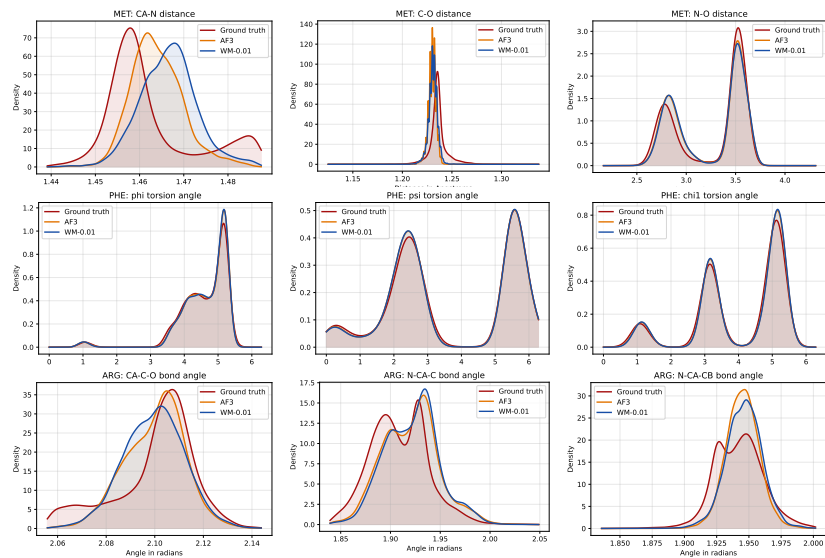


Fig. S47: Additional atom-atom distance, torsion and bond angle distributions for selected amino acids on 1000 short structures for SynthIDBio-structure trained with $s = 0.01$.

Representative examples of feature distributions for $s = 0.1$

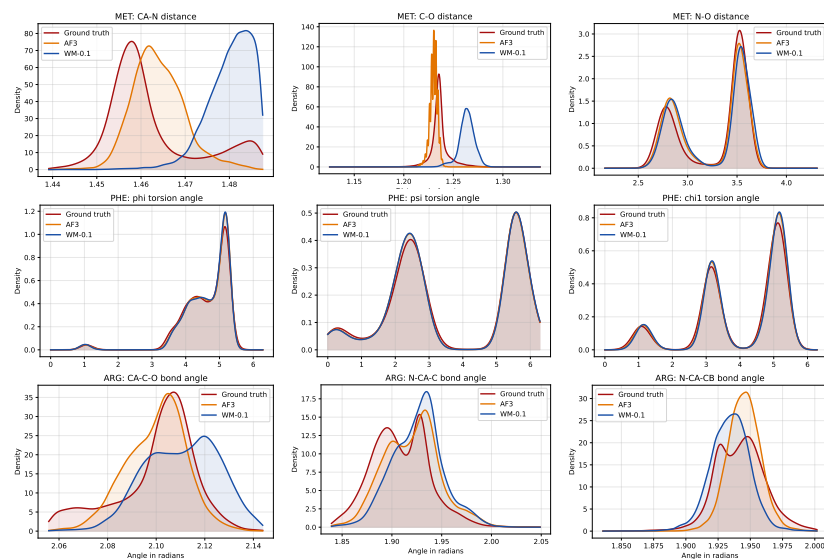


Fig. S48: Additional atom-atom distance, torsion and bond angle distributions for selected amino acids on 1000 short structures for SynthIDBio-structure trained with $s = 0.1$. Compared to $s < 0.1$, we see significant shifts in feature distributions of atom-atom distances and bond angles.

Atom-atom distance distributions for ALA, $s = 0.001$

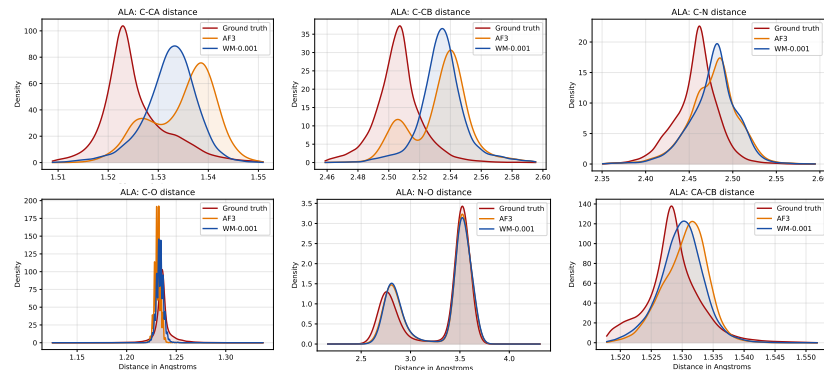


Fig. S49: Additional atom-atom distance distributions for ALA using SynthIDBio-structure with $s = 0.001$ compared to AF3. We show that only few atom-atom pairs are affected noticeably.

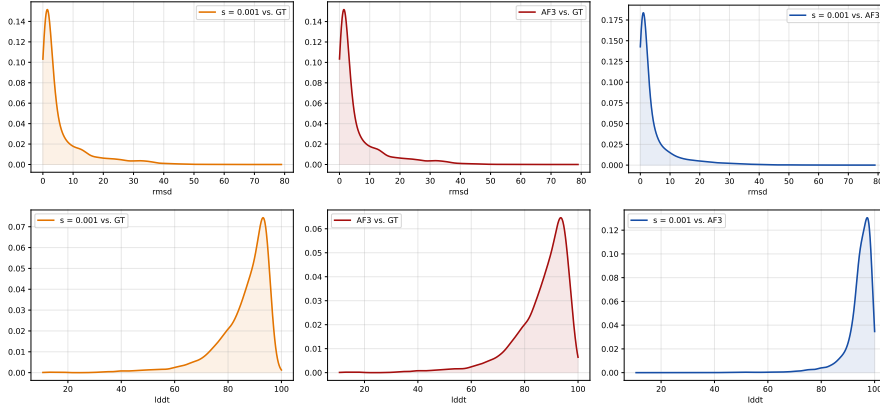


Fig. S50: Top: RMSD distribution for SynthIDBio-structure with $s = 0.001$ measured against ground truth (left); AF 3 vs. ground truth (middle); and SynthIDBio-structure measured against AF 3. Bottom: Distributions for IDDT. As demonstrated in Figure 4 in the main paper and Figure S45, we do not see a significant impact of watermarking in terms RMSD or IDDT measured against the ground truth. Nevertheless, the watermark may introduce a change relative to AF 3. However, we find that this correlates with the prediction quality of AF 3. In fact, the Spearman rank coefficient between SynthIDBio-structure and AF 3 IDDT vs. ground truth is 0.9780.

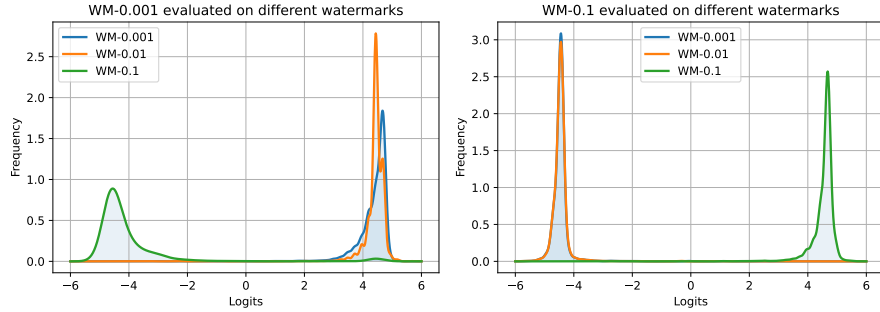


Fig. S51: Logits of the SynthIDBio-structure detector trained with $s = 0.001$ (left) or $s = 0.1$ (right) evaluated on watermarked structures predicted by all three settings for $s = 0.001, 0.01$, and 0.1 .

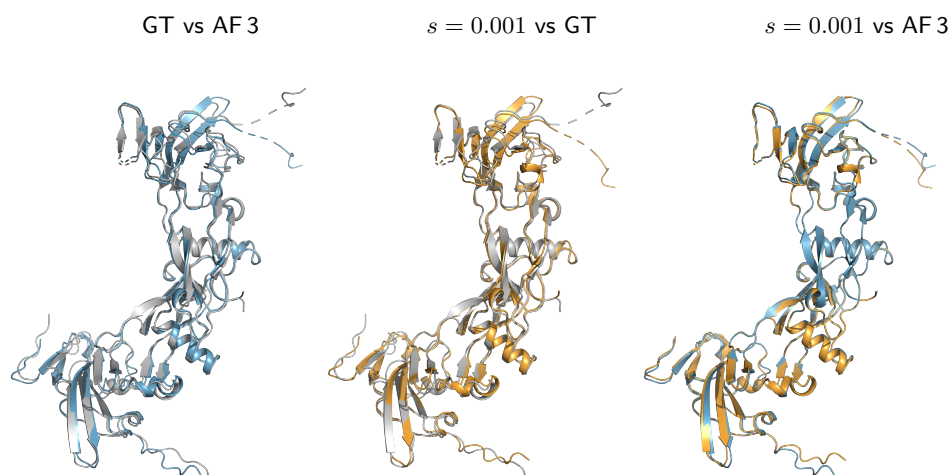


Fig. S52: Bone morphogenetic protein receptor type II (BMPRII) extracellular domain in complex with BMP10 (PDB 7PPA [10]; IDDT GT vs AF 3 88.82, IDDT $s = 0.001$ vs GT 88.85, IDDT $s = 0.001$ vs AF 3 99.00). Left AF 3 (blue) overlay with GT (grey), middle $s = 0.001$ (orange) overlay with GT (grey) and right AF 3 (blue) overlay with $s = 0.001$ (orange).

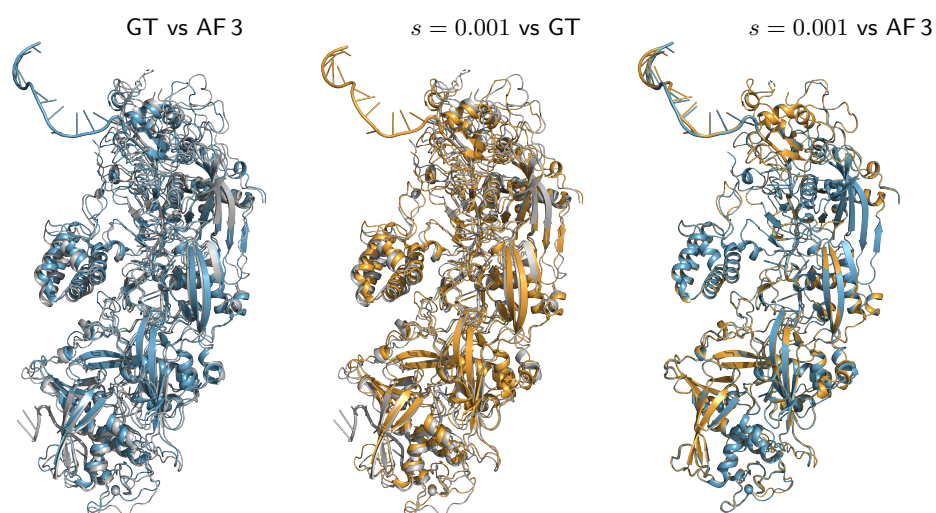


Fig. S53: SbCas7-11 in complex with crRNA and target RNA (PDB 7X7A [11]; IDDT GT vs AF 3 71.10, IDDT $s = 0.001$ vs GT 71.52, IDDT $s = 0.001$ vs AF 3 96.30). Left AF 3 (blue) overlay with GT (grey), middle $s = 0.001$ (orange) overlay with GT (grey) and right AF 3 (blue) overlay with $s = 0.001$ (orange).

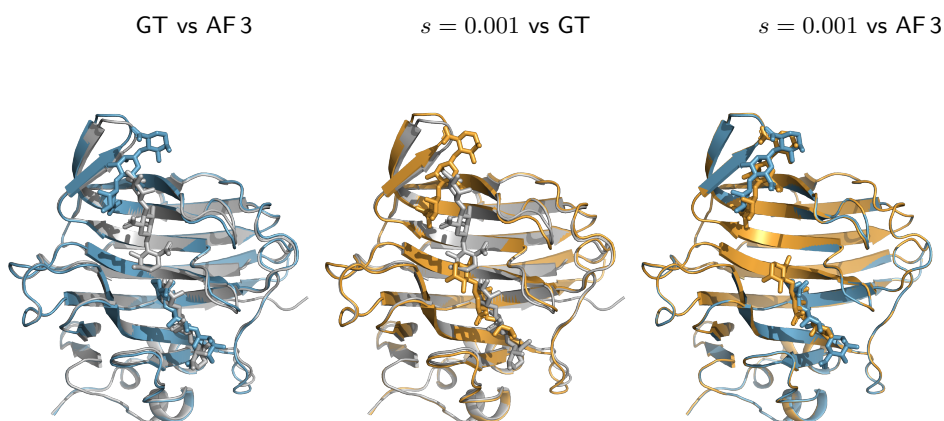


Fig. S54: *Paradendryphiella salina* PL7A alginate lyase in complex with hexamannuronic acid products (PDB 7P25 [12]; IDDT GT vs AF 3 91.26, IDDT $s = 0.001$ vs GT 89.61, IDDT $s = 0.001$ vs AF 3 95.00). Left AF 3 (blue) overlay with GT (grey), middle $s = 0.001$ (orange) overlay with GT (grey) and right AF 3 (blue) overlay with $s = 0.001$ (orange).

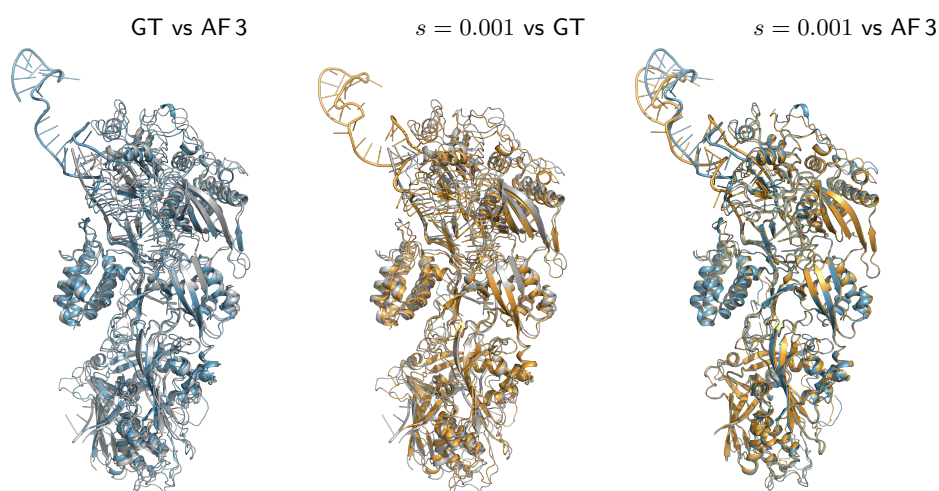


Fig. S55: gRAMP-target RNA (PDB 7XSP [13]; IDDT GT vs AF 3 70.58, IDDT $s = 0.001$ vs GT 71.45, IDDT $s = 0.001$ vs AF 3 94.53). Left AF 3 (blue) overlay with GT (grey), middle $s = 0.001$ (orange) overlay with GT (grey) and right AF 3 (blue) overlay with $s = 0.001$ (orange).

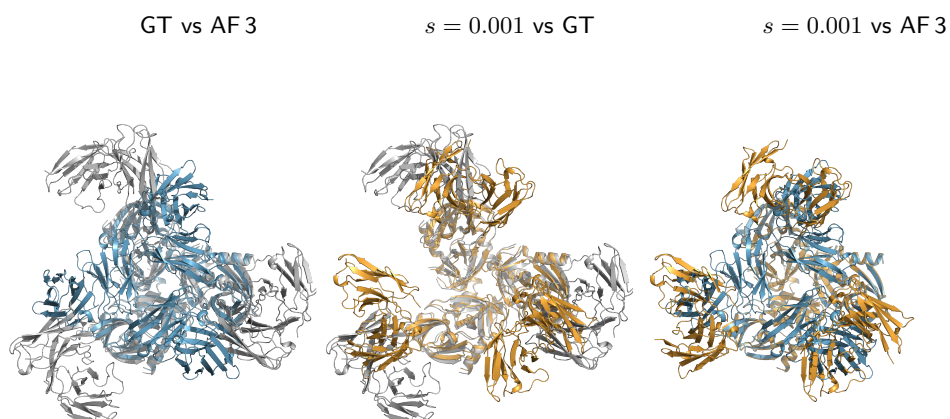


Fig. S56: HtrA1:Fab15H6.v4 complex (PDB 7SJN [14]; IDDT GT vs AF 3 77.27, IDDT $s = 0.001$ vs GT 77.45, IDDT $s = 0.001$ vs AF 3 89.92). Left AF 3 (blue) overlay with GT (grey), middle $s = 0.001$ (orange) overlay with GT (grey) and right AF 3 (blue) overlay with $s = 0.001$ (orange).

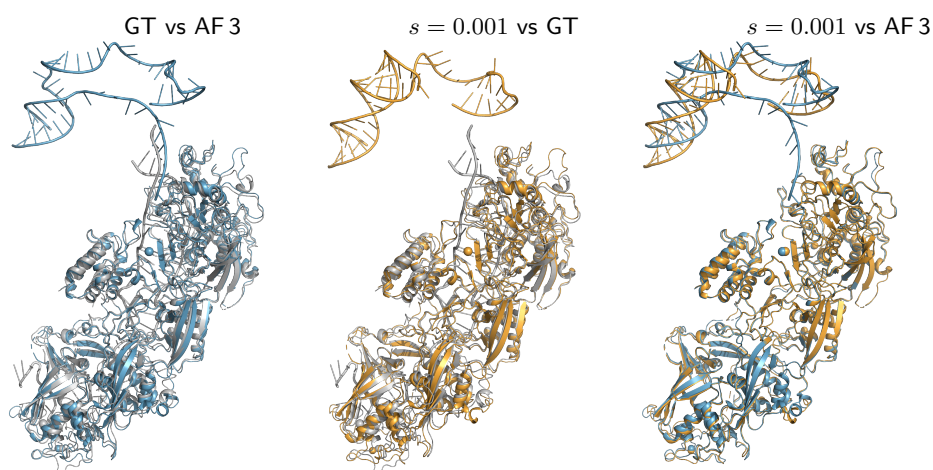


Fig. S57: Type III-E CRISPR Caspase gRAMP-crRNA complex bound to non-self RNA target (PDB 7Y81 [15]; IDDT GT vs AF 3 68.06, IDDT $s = 0.001$ vs GT 67.70, IDDT $s = 0.001$ vs AF 3 88.13). Left AF 3 (blue) overlay with GT (grey), middle $s = 0.001$ (orange) overlay with GT (grey) and right AF 3 (blue) overlay with $s = 0.001$ (orange).

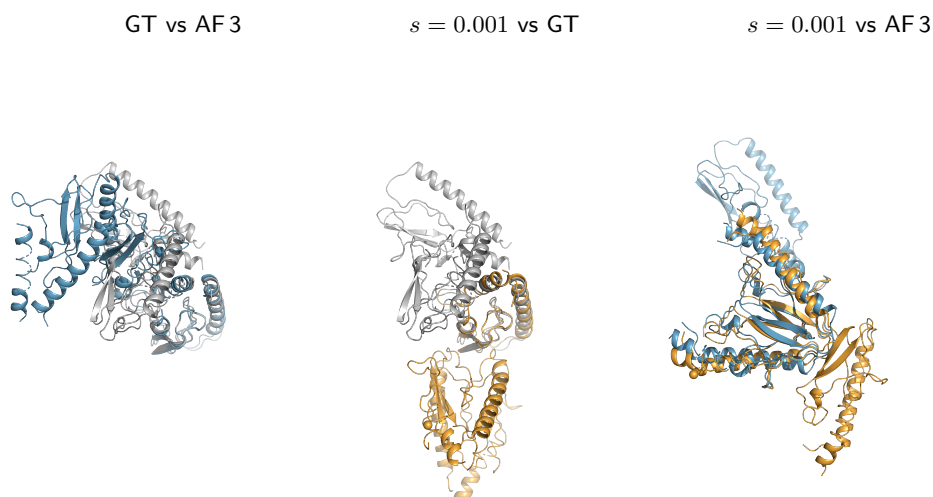


Fig. S58: Parathyroid hormone 1 receptor extracellular domain complexed with a peptide ligand containing three beta-amino acids (PDB 7UZP [16]; lDDT GT vs AF 3 70.34, lDDT $s = 0.001$ vs GT 67.55, lDDT $s = 0.001$ vs AF 3 84.77). Left AF 3 (blue) overlay with GT (grey), middle $s = 0.001$ (orange) overlay with GT (grey) and right AF 3 (blue) overlay with $s = 0.001$ (orange).

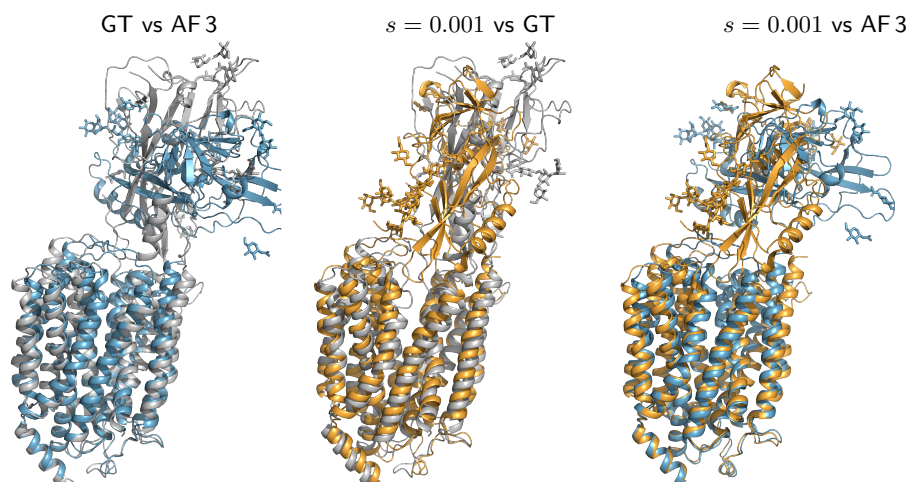


Fig. S59: Thermostable human MFSD2A in complex with thermostable human Sync2 (PDB 7OIX [17]; lDDT GT vs AF 3 69.07, lDDT $s = 0.001$ vs GT 68.35, lDDT $s = 0.001$ vs AF 3 84.63). Left AF 3 (blue) overlay with GT (grey), middle $s = 0.001$ (orange) overlay with GT (grey) and right AF 3 (blue) overlay with $s = 0.001$ (orange).

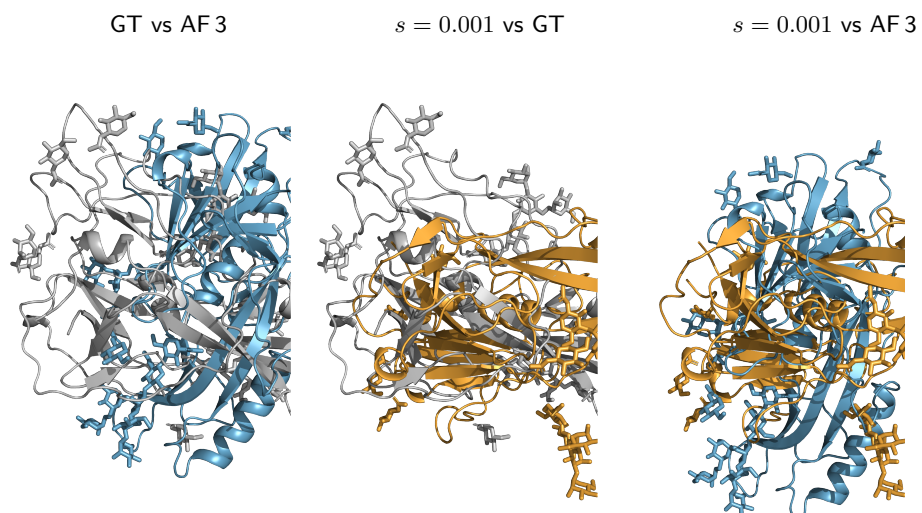


Fig. S60: Thermostable human MFSD2A in complex with thermostable human Sync2 zoomed into the high divergence region (PDB 7OIX [17]; lDDT GT vs AF 3 69.07, lDDT $s = 0.001$ vs GT 68.35, lDDT $s = 0.001$ vs AF 3 84.63). Left AF 3 (blue) overlay with GT (grey), middle $s = 0.001$ (orange) overlay with GT (grey) and right AF 3 (blue) overlay with $s = 0.001$ (orange).

References

- [1] Dathathri, S. *et al.* Scalable watermarking for identifying large language model outputs. *Nature* **634**, 818 – 823 (2024). URL <https://doi.org/10.1038/s41586-024-08025-4>.
- [2] Synthid: A tool to watermark and identify content generated through AI. <https://deepmind.google/science/synthid/>. Accessed 2025-08-26.
- [3] Google Cloud. Google cloud pricing calculator (2025). URL <https://cloud.google.com/products/calculator>. Accessed: 2025-12-11.
- [4] NVIDIA Corporation. Rfdiffusion nim: Benchmarking (2025). URL <https://docs.nvidia.com/nim/bionemo/rfdiffusion/latest/benchmarking.html>. Accessed: 2025-12-11.
- [5] Watson, J. L. *et al.* De novo design of protein structure and function with RFdiffusion. *Nature* **620**, 1089–1100 (2023). URL <https://doi.org/10.1038/s41586-023-06415-8>.
- [6] Biggio, B. & Roli, F. Wild patterns: Ten years after the rise of adversarial machine learning. *Pattern Recognition* **84**, 317–331 (2018). URL <https://doi.org/10.1016/j.patcog.2018.07.023>.
- [7] Grantham, R. Amino acid difference formula to help explain protein evolution. *Science* **185**, 862–864 (1974). URL <https://doi.org/10.1126/science.185.4154.862>.
- [8] Yampolsky, L. Y. & Stoltzfus, A. The exchangeability of amino acids in proteins. *Genetics* **170**, 1459 – 1472 (2005). URL <https://api.semanticscholar.org/CorpusID:27008401>.
- [9] Abramson, J. *et al.* Accurate structure prediction of biomolecular interactions with AlphaFold 3. *Nature* **630**, 493 – 500 (2024). URL <https://doi.org/https://doi.org/10.1038/s41586-024-07487-w>.
- [10] Guo, J. *et al.* Crystal structures of bmprii extracellular domain in binary and ternary receptor complexes with bmp10. *Nature Communications* **13**, 2395 (2022).
- [11] Yu, G. *et al.* Structure and function of a bacterial type iii-e crispr-cas7-11 complex. *Nature microbiology* **7**, 2078–2088 (2022).
- [12] Rivas-Fernández, J. P. *et al.* Unraveling the molecular mechanism of polysaccharide lyases for efficient alginate degradation. *Nature Communications* **16**, 2670 (2025).
- [13] Liu, X. *et al.* Target rna activates the protease activity of craspase to confer antiviral defense. *Molecular Cell* **82**, 4503–4518 (2022).

- [14] Gerhardy, S. *et al.* Allosteric inhibition of htra1 activity by a conformational lock mechanism to treat age-related macular degeneration. *Nature communications* **13**, 5222 (2022).
- [15] Cui, N. *et al.* Structural basis for the non-self rna-activated protease activity of the type iii-e crispr nuclease-protease craspase. *Nature Communications* **13**, 7549 (2022).
- [16] Liu, S. *et al.* Altered signaling at the pth receptor via modified agonist contacts with the extracellular domain provides a path to prolonged agonism in vivo. *Proceedings of the National Academy of Sciences* **119**, e2212736119 (2022).
- [17] Martinez-Molledo, M., Nji, E. & Reyes, N. Structural insights into the lysophospholipid brain uptake mechanism and its inhibition by syncytin-2. *Nature structural & molecular biology* **29**, 604–612 (2022).
- [18] Baker, D. & Church, G. Protein design meets biosecurity. *Science* **383**, 349–349 (2024). URL <https://doi.org/10.1126/science.adol671>.
- [19] Groff-Vindman, C. S. *et al.* The convergence of AI and synthetic biology: the looming deluge. *npj Biomedical Innovations* **2**, 20 (2025). URL <https://doi.org/10.1038/s44385-025-00021-1>.
- [20] Bloomfield, D. *et al.* AI and biosecurity: The need for governance. *Science* **385**, 831–833 (2024). URL <https://doi.org/10.1126/science.adq1977>.
- [21] Wheeler, N. E. Responsible AI in biotechnology: balancing discovery, innovation and biosecurity risks. *Frontiers in Bioengineering and Biotechnology* **Volume 13 - 2025** (2025). URL <https://doi.org/10.3389/fbioe.2025.1537471>.
- [22] Carter, S. R. *et al.* Responsible AI x biodesign statement. <https://responsiblebiodesign.ai/#commitments> (2024). Accessed: 2025-05-09.
- [23] Helena project executive summary. <https://helena.org/projects/helena-biosecurity>. Accessed: 2025-05-09.
- [24] Sandbrink, J. B. Artificial intelligence and biological misuse: Differentiating risks of language models and biological design tools. *arXiv* (2023). URL <https://doi.org/10.48550/arXiv.2306.13952>.
- [25] Zambaldi, V. *et al.* De novo design of high-affinity protein binders with AlphaProteo. *arXiv* (2024). URL <https://doi.org/10.48550/arXiv.2409.08022>.
- [26] Dauparas, J. *et al.* Robust deep learning-based protein sequence design using ProteinMPNN. *Science* **378**, 49–56 (2022). URL <https://doi.org/10.1126/science.add2187>.

- [27] Pannu, J. *et al.* Prioritizing high-consequence biological capabilities in evaluations of artificial intelligence models. *ArXiv abs/2407.13059* (2024). URL <https://api.semanticscholar.org/CorpusID:271270120>.
- [28] Mouton, C. A., Lucas, C. & Guest, E. The operational risks of AI in large-scale biological attacks a red-team approach. Tech. Rep., RAND (2024). URL <https://api.semanticscholar.org/CorpusID:265036157>.
- [29] Gopal, A. *et al.* Will releasing the weights of future large language models grant widespread access to pandemic agents? *arXiv* (2023). URL <https://doi.org/10.48550/arXiv.2310.18233>.
- [30] Soice, E. H., Rocha, R. H. S., Cordova, K., Specter, M. A. & Esvelt, K. M. Can large language models democratize access to dual-use biotechnology? *arXiv* (2023). URL <https://doi.org/10.48550/arXiv.2306.03809>.
- [31] Yassif, J. M., Korol, S. & Kane, A. Guarding against catastrophic biological risks: Preventing state biological weapon development and use by shaping intentions. *Health Security* **21**, 258 – 265 (2023). URL <https://doi.org/10.1089/hs.2022.0145>.
- [32] Patwardhan, T. *et al.* Building an early warning system for LLM-aided biological threat creation. <https://openai.com/index/building-an-early-warning-system-for-llm-aided-biological-threat-creation/> (2024). Accessed: 2025-05-09.
- [33] Anthropic’s responsible scaling policy. <https://www.anthropic.com/news/anthropics-responsible-scaling-policy> (2023). Accessed: 2025-08-04.
- [34] Shevlane, T. *et al.* Model evaluation for extreme risks. *arXiv* (2023). URL <https://doi.org/10.48550/arXiv.2305.15324>.
- [35] Wittmann, B. J. *et al.* Toward AI-resilient screening of nucleic acid synthesis orders: Process, results, and recommendations. *bioRxiv* 2024–12 (2024). URL <https://doi.org/10.1101/2024.12.02.626439>.
- [36] Baum, C. *et al.* A system capable of verifiably and privately screening global DNA synthesis. *arXiv* (2024). URL <https://doi.org/10.48550/arXiv.2403.14023>.
- [37] Gretton, D. W. *et al.* Random adversarial threshold search enables automated DNA screening. *bioRxiv* (2024). URL <https://doi.org/10.1101/2024.03.20.585782>.
- [38] Ratcliff, J. Transformer model generated bacteriophage genomes are compositionally distinct from natural sequences. *NAR Genomics and Bioinformatics* **6** (2024). URL <https://doi.org/10.1093/nargab/lqae129>.

- [39] Nielsen, A. A. K. & Voigt, C. A. Deep learning to predict the lab-of-origin of engineered DNA. *Nature Communications* **9** (2018). URL <https://doi.org/10.1038/s41467-018-05378-z>.
- [40] Mo, W., Vaiana, C. A. & Myers, C. J. The need for adaptability in detection, characterization, and attribution of biosecurity threats. *Nature Communications* **15** (2024). URL <https://doi.org/10.1038/s41467-024-55436-y>.
- [41] Kuditipudi, R., Thickstun, J., Hashimoto, T. & Liang, P. Robust distortion-free watermarks for language models. *Transactions on Machine Learning Research* (2024). URL <https://openreview.net/forum?id=FpaCL1MO2C>.
- [42] Aaronson, S. & Kirchner, H. Watermarking of large language models (2022). URL <https://www.scottaaronson.com/talks/watermark.ppt>. Accessed: 2025-05-09.
- [43] Kirchenbauer, J. *et al.* A watermark for large language models. *arXiv* (2024). URL <https://arxiv.org/abs/2301.10226>.
- [44] Christ, M., Gunn, S. & Zamir, O. Undetectable watermarks for language models. *IACR Cryptol. ePrint Arch.* **2023**, 763 (2023). URL <https://api.semanticscholar.org/CorpusID:259092330>.
- [45] Kamaruddin, N. S., Kamsin, A., Por, L. Y. & Rahman, H. A review of text watermarking: Theory, methods, and applications. *IEEE Access* **6**, 8011–8028 (2018). URL <https://doi.org/10.1109/ACCESS.2018.2796585>.
- [46] Gu, C., Huang, C., Zheng, X., Chang, K.-W. & Hsieh, C.-J. Watermarking pre-trained language models with backdoor. *arXiv* (2022). URL <https://doi.org/10.48550/arXiv.2210.07543>.
- [47] Van Schyndel, R. G., Tirkel, A. Z. & Osborne, C. F. A digital watermark (1994).
- [48] Braudaway, G. Protecting publicly-available images with an invisible image watermark **1**, 524–527 vol.1 (1997). URL <https://doi.org/10.1109/ICIP.1997.647965>.
- [49] Pan, M. *et al.* AnchMark: Anchor-contrastive watermarking vs genai-based image modifications (2023).
- [50] Mun, S., Nam, S., Jang, H., Kim, D. & Lee, H. A robust blind watermarking using convolutional neural network. *arXiv* (2017). URL <https://doi.org/10.1016/j.neucom.2019.01.067>.
- [51] Zhu, J., Kaplan, R., Johnson, J. & Fei-Fei, L. HiDDeN: Hiding data with deep networks. *arXiv* (2018). URL <https://doi.org/10.48550/arXiv.1807.09937>.
- [52] Wen, B. & Aydınoğlu, S. ROMark: A robust watermarking system using adversarial training. *arXiv* (2019). URL <https://doi.org/10.48550/arXiv.1910.01221>.

- [53] Hayes, J. & Danezis, G. Generating steganographic images via adversarial training. *arXiv* (2017). URL <https://doi.org/10.48550/arXiv.1703.00371>.
- [54] Hayes, J. *et al.* Towards transformation-resilient provenance detection of digital media. *arXiv* (2020). URL <https://doi.org/10.48550/arXiv.2011.07355>.
- [55] Tancik, M., Mildenhall, B. & Ng, R. StegaStamp: Invisible hyperlinks in physical photographs. *arXiv* (2020). URL <https://doi.org/10.48550/arXiv.1904.05343>.
- [56] Luo, X., Zhan, R., Chang, H., Yang, F. & Milanfar, P. Distortion agnostic deep watermarking. *arXiv* (2020). URL <https://doi.org/10.48550/arXiv.2001.04580>.
- [57] Bui, T., Agarwal, S. & Collomosse, J. P. TrustMark: Universal watermarking for arbitrary resolution images. *arXiv* (2023). URL <https://doi.org/10.48550/arXiv.2311.18297>.
- [58] Bui, T., Agarwal, S., Yu, N. & Collomosse, J. P. RoSteALS: Robust steganography using autoencoder latent space. *arXiv* (2023). URL <https://doi.org/10.48550/arXiv.2304.03400>.
- [59] Pan, M. *et al.* JIGMARK: A black-box approach for enhancing image watermarks against diffusion model edits. *arXiv* (2024). URL <https://doi.org/10.48550/arXiv.2406.03720>.
- [60] Xu, R. *et al.* InvisMark: Invisible and robust watermarking for AI-generated image provenance. *arXiv* (2024). URL <https://doi.org/10.48550/arXiv.2411.07795>.
- [61] Sander, T., Fernandez, P., Durmus, A., Furon, T. & Douze, M. Watermark anything with localized messages. *arXiv* (2024). URL <https://doi.org/10.48550/arXiv.2411.07231>.
- [62] Fernandez, P., Couairon, G., Jégou, H., Douze, M. & Furon, T. The stable signature: Rooting watermarks in latent diffusion models. *arXiv* (2023). URL <https://doi.org/10.48550/arXiv.2303.15435>.
- [63] Wen, Y., Kirchenbauer, J., Geiping, J. & Goldstein, T. Tree-ring watermarks: Fingerprints for diffusion images that are invisible and robust. *arXiv* (2023). URL <https://doi.org/10.48550/arXiv.2305.20030>.
- [64] Yang, Z. *et al.* Gaussian shading: Provable performance-lossless image watermarking for diffusion models. *arXiv* (2024). URL <https://doi.org/10.48550/arXiv.2404.04956>.
- [65] Wan, W. *et al.* A comprehensive survey on robust image watermarking. *Neurocomputing* **488** (2022). URL <https://doi.org/10.1016/j.neucom.2022.02.083>.

- [66] Saberi, M., Sadasivan, V. S., Zarei, A., Mahdavifar, H. & Feizi, S. DREW : Towards robust data provenance by leveraging error-controlled watermarking. *arXiv* (2024). URL <https://doi.org/10.48550/arXiv.2406.02836>.
- [67] Balan, K. *et al.* EKILA: synthetic media provenance and attribution for generative art. *arXiv* (2023). URL <https://doi.org/10.48550/arXiv.2304.04639>.
- [68] Bharati, A. *et al.* Transformation-aware embeddings for image provenance. *IEEE Transactions on Information Forensics and Security* **16**, 2493–2507 (2021). URL <https://doi.org/10.1109/TIFS.2021.3050061>.
- [69] Zhang, X., Sun, Z. H., Karaman, S. & Chang, S. Discovering image manipulation history by pairwise relation and forensics tools. *IEEE Journal of Selected Topics in Signal Processing* **14**, 1012–1023 (2020).
- [70] Nguyen, E., Bui, T., Swaminathan, V. & Collomosse, J. OSCAR-Net: Object-centric scene graph attention for image attribution. *arXiv* (2021). URL <https://doi.org/10.48550/arXiv.2108.03541>.