
What Does a Protein Surface Add? Ablations Cannot Tell You

Odysseus Kaloidas
UC Berkeley

Abstract

1 When preprocessing computes a model input from the prediction target, removing
2 that input cannot distinguish target information supplied by preprocessing from a
3 useful geometric representation. We study this ambiguity in surface-conditioned
4 protein inverse folding. Fixed-coordinate substitutions show that chemical sur-
5 face channels in two released pipelines depend on native residue identity rather
6 than geometry alone. On SurfPro’s released checkpoint, changing the channel
7 code redirects predictions toward specified counterfactual sequences: across two
8 tested code-table permutations, agreement is 35.84 percentage points [34.89, 36.74]
9 higher with the intended alternative than with the other alternative. This targeted
10 preference supports code-specific reliance beyond an explanation based only on
11 generic degradation, while not ruling out every distribution-shift mechanism. Con-
12 trolled analyses separate whether preprocessing reads the target from whether fixed
13 weights use the resulting channel. Further comparisons ask whether the channel
14 adds information beyond the backbone and helps a specified model under a fixed
15 training budget. We therefore do not infer that genuinely backbone-derived sur-
16 faces cannot help. We provide a provenance-aware audit of input construction,
17 channel-only predictability and downstream reliance: a diagnostic standard for
18 what surface ablations can and cannot establish about representation utility.

19 1 Introduction

20 Inverse folding predicts a sequence for a given protein backbone (Kuhlman et al., 2003; Ingraham
21 et al., 2019; Dauparas et al., 2022; Hsu et al., 2022). Surface-conditioned methods add a molecular
22 surface and often assess its value by removing that input (Song et al., 2024; Wu et al., 2026). The
23 surface, however, is computed: its geometry may come from backbone coordinates, while a chemical
24 feature may look up the native residue name. For a new backbone, that name is the unknown
25 prediction target. A surface ablation changes both kinds of input at once.

26 Our clearest test changes neither the backbone nor the trained weights. We replace SurfPro’s chemical
27 code table with a fixed permutation. The model then predicts the particular alternative sequence
28 specified by that table, rather than merely losing native-sequence accuracy. This observation motivates
29 the distinction between source dependence and useful representation in Figure 1.

30 1.1 What the ablation combines

31 The SurfPro example separates two steps that an ablation merges. First, preprocessing reads a residue
32 label and converts it into a chemical code. Second, the trained model uses that code: replacing the
33 code table redirects the prediction toward the corresponding alternative sequence. These observations
34 establish construction dependence and fixed-model reliance, respectively. They do not by themselves

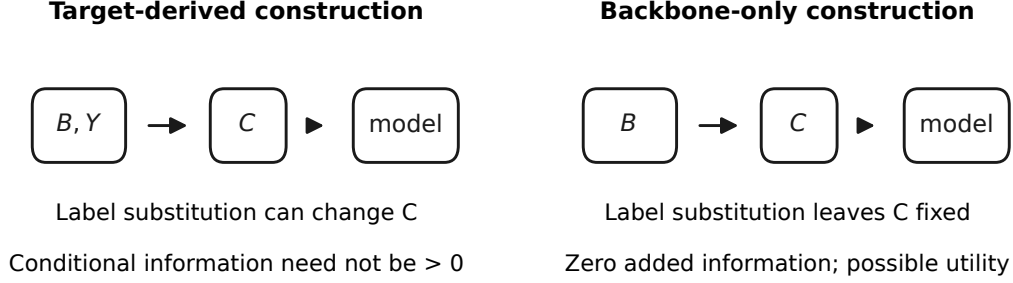


Figure 1: Different construction paths can yield an ablation gain. A target-derived channel depends on the example’s target under a controlled source substitution; that fact alone does not imply positive conditional information or model reliance. A channel computed only from the admissible backbone has zero conditional information but may benefit a finite learner.

35 establish how much information the channel adds beyond the backbone or how much an admissible
36 channel would help a newly trained model.

37 We use four terms for these questions. Let B be the admissible backbone input, Y the target sequence,
38 and C a computed channel. Construction provenance asks whether the code generating C reads this
39 example’s Y ; a labels-only intervention substitutes Y while holding the other inputs and randomness
40 fixed. Model reliance asks whether changing C changes a fixed predictor’s outputs in a specified way.
41 Conditional information is the population quantity $I(Y; C|B)$: how much C reveals about Y after B is
42 known. Finite-learner utility is the performance change under a stated model class, training budget
43 and input contract. The four questions require different tests.

44 A design-valid channel $C=f(B)$ satisfies

$$I(Y; f(B) | B) = 0$$

45 but may help a finite learner through representation or optimisation (Route B). A target-derived
46 channel may supply information unavailable from B (Route A), but dependence on Y alone does not
47 prove strictly positive conditional information. In this paper, “target-derived” describes construction;
48 “Route A” refers to added target information where supported by the construction or controls.

49 1.2 Contributions and scope

50 We make three contributions. First, we assign source interventions, simple decoders, fixed-model
51 controls and joint training to the questions they can answer. Second, we diagnose two released
52 surface pipelines and connect one source dependency to targeted prediction changes on a released
53 checkpoint. Third, matched-decoding and clean-training comparisons constrain the interpretation
54 without claiming that surface representations are generally unhelpful. These operations draw on
55 leakage audits, partial-input baselines and variable importance. The added contribution is a verified
56 input-to-prediction mechanism and a reusable conservative audit, not a new information-theoretic
57 identity or a minimal universal protocol.

58 1.3 Why matching an ablation or a ceiling is insufficient

59 The ordinary contrast

$$\Delta_{\text{ablation}} = M(B, C) - M(B, \emptyset)$$

60 is a performance difference under a particular learner. It does not reveal how C was constructed.
61 Appendix F constructs target-derived and backbone-derived channels with ablation gaps matched to
62 0.07 points across five seeds, despite different source dependence and conditional information.

63 A smaller exact example shows why marginal matching is also insufficient. Let Y be uniform on
 64 $\{0, 1, 2\}$ and $B=1[Y=0]$. Table 1 compares three constructions. Every channel has two classes with
 65 the same class frequencies and standalone Bayes accuracy $2/3$, yet their conditional utility differs.

Table 1: Exact three-state example. Accuracy columns use the population Bayes rule. The last column enumerates all replacement labels at each original state with B clamped; it is a software-input intervention, including combinations outside observational support.

Construction	Alone	With B	$I(Y;C$ given B), bits	Outputs changed / 9
$C=B$, reads backbone	$2/3$	$2/3$	0	0/9
$C=1[Y=0]$, reads label	$2/3$	$2/3$	0	4/9
$C=1[Y=1]$, reads label	$2/3$	1	$2/3$	4/9

66 The first two channels have identical observational joint laws but different construction dependence.
 67 The third reveals which of states 1 and 2 occurred when $B=0$. Matching cardinality and marginal
 68 accuracy therefore does not match complementarity. Conversely, reading the target does not by itself
 69 prove added conditional information. These are exact properties of this example, not estimates for
 70 proteins (Appendix H).

71 2 Related work

72 Partial-input NLI baselines reveal unexpected predictive information (Gururangan et al., 2018; Poliak
 73 et al., 2018), and leakage taxonomies cover illegitimate feature construction (Kaufman et al., 2011;
 74 Kapoor & Narayanan, 2023). Variable importance distinguishes ablation utility from conditional
 75 information (Williamson et al., 2023; Watson & Wright, 2021); usable information explains benefits
 76 from computing $f(B)$ for restricted learners (Xu et al., 2020).

77 SurfDesign (Wu et al., 2026) is the closest audited construction; SurfPro (Song et al., 2024) supplies
 78 the released-checkpoint test. ProBayes removes a related side-chain shortcut through model redesign
 79 (Wu et al., 2025). Surface fingerprints also support interaction design (Gainza et al., 2019, 2023;
 80 Sverrisson et al., 2021), whose input contract can legitimately include native chemistry. Our contribu-
 81 tion connects two source audits to targeted prediction changes on one released checkpoint. Broader
 82 contemporaneous work and distinctions from split/evaluator audits are discussed in Appendix H.

83 3 Methods

84 3.1 Operations and the claims they support

85 Table 2 separates the operational checks. Source substitution is conditional on fixed coordinates,
 86 preprocessing inputs, correspondence and random state. It is not a physical mutation experiment.
 87 Changed values establish dependence under that contract; unchanged values under one substitution
 88 do not certify independence. A substitution can preserve a non-injective code’s classes.

Table 2: Operations and their distinct readouts. Together, the rows separate input construction, fixed-model use and utility under stated learners. They neither numerically decompose an ablation gap nor imply that every audit needs all seven.

Operation	Supported readout
1. Fixed-coordinate label substitution	Construction dependence under fixed controls
2. Channel-only prediction	Marginal accuracy; analytic bound if justified
3. Similar standalone accuracy	Payoff comparison with remaining confounds
4. Counterfactual and neutralisation controls	Fixed-model reliance and target specificity

Operation	Supported readout
5. Train-fitted fusion of frozen predictions	Utility under that finite fusion rule
6. Joint training with admissible inputs	Representation benefit in the tested regime
7. Remove, restore and repair a pathway	Internal reliance and repair cost

89 3.2 Systems and populations

90 We inspect SurfDesign’s released pipeline at commit 9ce3039, including its unchanged preprocessing
91 and E(n)-GNN encoder (Satorras et al., 2021). Its paper reports 74.13% median CATH recovery
92 and describes surface generation as not using residue identities. The released hydropathy function
93 instead indexes a twenty-entry table by residue name. We test this implementation difference without
94 inferring intent. With no published checkpoint, our controlled SurfDesign training arms use its
95 encoder with a matched decoder, data and budget, and no protein language model. They do not
96 reproduce its published full system.

97 SurfPro uses its official repository, hash-verified CATH checkpoint and data, and official scorer; only
98 device placement changes. Frozen bridges are ProteinMPNN v_48_020 (1.66M parameters), ESM-
99 IF1 esm_if1_gvp4_t16_142M_UR50 (141.66M), and the official PiFold/ProDesign checkpoint.
100 Different readouts use different populations:

- 101 • Source-dependence audit: 38 chains. Channel-only analytic bound: 195,631 residues
102 (P-CEIL).
- 103 • Released-checkpoint test: SurfPro’s official 887-chain CATH test split.
- 104 • Joint-training and cached analyses: 593 train and 197 test chains (P-FROZEN).
- 105 • Largest frozen-model comparison: 928 test chains / 125,401 residues (P-FROZEN-FULL).
- 106 • External transfer panel, RCSB 2023+: 68 chains / 9,540 residues for ProteinMPNN; a shared
107 23 / 3,230 for ESM-IF1 and PiFold.

108 Our training pool for that encoder contains 3,946/790 train/test chains; Appendix A records the
109 intersections used in each trained comparison. TS50/TS500 are benchmark transfers, not independent
110 replications.

111 CATH family-disjoint splits are used as distributed. Our controlled selection requires canonical
112 complete backbones of 40–260 residues; SurfPro retains its own official split. We have three
113 named benchmarks (CATH, TS50, TS500), one primary data authority, and no independent full-
114 protocol external-data replication. TS50/TS500 overlap CATH; the external panel tests transport of
115 selected frozen comparisons, with modest three-model coverage. All claims are within-population
116 comparisons of sequence recovery, not fold quality or function.

117 3.3 Statistics and analysis status

118 Frozen-model intervals resample chains in paired comparisons. Trained-arm intervals resample seeds
119 and chains on the intersection scored by every seed. Two ten-seed clean contrasts support marginal
120 95% intervals; two three-seed contrasts are directional checks without calibrated tail claims. The
121 rows make no simultaneous-coverage claim. The +1-point practical threshold was fixed before those
122 runs. Training budgets, failures and constituents are in Appendices A–C.

123 Lookup rules and fusion weights are fitted on train only; train-estimated lookup accuracy is not
124 generally a population Bayes ceiling. Micro residue averages and macro chain averages are never
125 mixed within a contrast. The specificity and complementarity analyses are retrospective, but their
126 rules were recorded before the reported calculations. Their intervals condition on the existing model,
127 splits and chosen channels; they do not cover model or permutation selection. Appendix H gives
128 denominators, checks and retained artifacts.

129 3.4 A conservative source audit

130 The reusable audit consumes paired chain IDs, channel values and geometry. It requires unique
131 matched chains, equal channel lengths, nonempty geometry for every chain and zero geometry change.

Any channel change then supports dependence, conditional on the caller changing only labels while fixing other inputs and randomness. Constructing the fixture may require expensive preprocessing; comparing it needs no model.

Invariance cannot certify design validity: exchanging D and E leaves a code that merges those labels unchanged. Conversely, one changed value among 200 still demonstrates dependence. A movement threshold would therefore be unsafe. The audit treats incomplete geometry controls as inconclusive and an unchanged channel as “no dependence detected,” not as proof of independence.

A 27-case test suite covers released target-derived and known-clean fixtures, rare movement, invariant substitutions, missing, partial or changed geometry, empty inputs and duplicate IDs. Every case returns its expected verdict. This checks the audit against those specified edge cases; the caller remains responsible for fixture completeness. Observational association is a separate screening mode without a provenance verdict (Appendix H).

4 Results

4.1 Source dependence and channel decoding

SurfDesign’s hp channel assigns each point its nearest residue’s Kyte–Doolittle hydropathy (Kyte & Doolittle, 1982). Permuting residue identities with coordinates fixed changes 89.12% of values across 38 chains; checked geometry changes 0.00%. The remaining invariances follow hydropathy collisions. Source inspection and substitution together establish that this chemical channel reads the native sequence.

The table maps 20 residues to 17 values, with the sole collision {D,E,N,Q}. Across the 195,631-residue analytic population (P-CEIL), its channel-only accuracy bound is 85.89%, versus an 8.86% frequency floor. Surface points reach 99.99% of residues and 99.94% of buried residues. Among 741,285 possible 17-class tables, 82% have a higher bound and the median exceeds it by 2.61 points. This comparison concerns identity partitions, not biochemical usefulness (Appendix D).

4.2 Released weights follow a specified alternative target

SurfPro’s official recovery reproduces at 57.79%, compared with the paper’s 57.78%, on 887 chains. Its hydropathy/charge table resolves 20 residues into 18 classes. On 3,076,319 vertices, fixed-coordinate substitution changes 90.12% of chemical values and 0.00% of checked geometry.

A class lookup predicts the most frequent training residue per chemical code, without a neural model, coordinates or test-label fitting. Because output lengths can differ, we report two denominators and apply each scorer to both methods. The coverage score divides the maximum ordered match count by the native-sequence length; the official score divides the same count by the chosen alignment length. Under the coverage score, the lookup reaches 69.72% [68.93, 70.49] and the model 70.33% [69.53, 71.10], a model margin of +0.61 points [−0.05, +1.28]. Under the official score, the same comparison is 54.67% versus 57.79%, with a +3.12-point margin [+2.46, +3.78]. The model adds value under that scorer; mixing the two margins would give a misleading account.

A code-table derangement fixed before decoding changes residue a’s code to that of $\pi(a)$. Native recovery falls from 57.79% to 18.36%, while agreement with $\pi(Y)$ rises from 17.75% to 56.65%: +38.90 points [+38.03, +39.76]. A second prespecified derangement yields 52.45% agreement with its target and a +34.87-point gain [+34.05, +35.68]. These gains support targeted redirection, rather than a claim based only on loss of native accuracy.

To test whether each output follows its own code table rather than an alternative sequence that is generically easier to match, we score every counterfactual output against both alternatives. Let $A_i(j)$ denote official-scorer agreement under code-table permutation i with alternative target j . We measure

$$S = \frac{1}{2} [(A_1(1) - A_1(2)) + (A_2(2) - A_2(1))].$$

Swapping the target preference in the second term removes an explanation based only on one reference sequence being easier to match. Intended-minus-other agreement is +38.05 and +33.62 points; it is positive on 883/887 and 879/887 chains, respectively. Averaging the contrasts gives +35.84 points

[34.89, 36.74]. The interval averages within-arm 97.5% bootstrap bounds without assuming cross-arm chain alignment. Figure 2a shows the opposing target preferences. This is a retrospective control on the same outputs, not a second model replication. It supports code-specific reliance under the two tested transformations, without ruling out every distribution-shift mechanism.

Three-seed retraining also separates value meanings from identity partitioning. Scrambling values over the same 18 classes changes recovery by at most 0.342 points, whereas a random 18-class partition improves by +3.347 [+2.393, +4.301] against the native table. The channel-versus-zero contrast is 47.82 points. Small semantic effects remain, and equal cardinality does not equalise the partitions’ decodability. These findings concern sequence recovery, not enzyme or binder function (Appendix B).

4.3 Similar standalone accuracy, different frozen-model payoff

We ask whether similarly decodable channels yield similar gains when added to frozen inverse-folding models. A binary target recoding and a sixteen-class quantisation of backbone geometry each have train-estimated standalone accuracy close to 17%. The backbone code clusters ten geometric descriptors after train-only standardisation. For each code, we fit a probability distribution over the twenty residues with Laplace smoothing of 0.5.

Fusion adds $\lambda \log P(y|\text{code})$ to the frozen model’s logits. We choose λ from $\{0, 0.25, 0.5, 1, 2, 4, 8, 16, 32, 64\}$ by training recovery, taking the first maximum. Thus $\lambda=0$ means that the fitted rule declines the channel; it is not an estimate of conditional information, and train-based tuning can favour the fitted distribution. No external-panel labels refit the rule. Table 3 reports the original comparisons and unrefitted external transfer. These are artificial comparison channels, not the released channel with an 85.89% bound.

Table 3: Native-recovery gains in percentage points, paired chain-bootstrap 95% intervals. T/B denotes train-fit target/backbone decoding accuracy, not conditional information. Zero is the result of selecting $\lambda=0$. Denominators are in §3.2; codebook, fitting and scorer details are in Appendix A.

Model	Population (train-fit T/B, %)	Target-derived gain	Backbone-derived gain
ProteinMPNN	CATH (17.13/17.00)	+12.49 [+12.29, +12.68]	+0.00
ESM-IF1	CATH (17.13/17.00)	+11.07 [+10.88, +11.27]	+0.00
ProteinMPNN	RCSB 2023+ (17.13/16.76)	+12.00 [+11.32, +12.71]	+0.00
ESM-IF1	RCSB 2023+ (17.13/16.51)	+10.80 [+9.76, +11.90]	+0.00
PiFold	RCSB 2023+ (17.40/16.56)	+12.29 [+11.11, +13.59]	+0.00

Cardinality, complementary errors and learnability remain unmatched; adding backbone classes does not remove those confounds. A retained-output analysis on 197 chains makes complementarity visible. One target recoding and the sixteen-class backbone code have 16.62% and 16.46% held-out lookup accuracy, but their argmax predictions are correct while the backbone is wrong at 6.85% and 4.39% of all residue positions: a 2.46-point difference [2.03, 2.92]. The denominator is all test residues. The two-class backbone control reaches only 10.83% standalone accuracy, matching cardinality but not decoding accuracy. Appendix H reports all three target recodings and both backbone controls; none estimates conditional information or isolates provenance. The exact example in §1.3 shows why even stronger marginal matching would not suffice.

Across the original 48 channel-model pairs, target-derived gains span +9.75 to +48.30 points and backbone-derived gains −0.92 to +0.00. A learned nineteen-feature geometry predictor reaches

212 21.43% standalone accuracy, but its trained nonlinear fusion gains are -0.23 $[-0.36, -0.09]$ and -0.73
213 $[-0.87, -0.58]$ on ProteinMPNN and ESM-IF1. These are failures of specified finite fusion rules,
214 not proofs of conditional redundancy. The descriptive capacity–payoff interpolation and its failed
215 predictive precursors remain in Appendix D.

216 4.4 Joint training leaves strong clean-surface utility open

217 Frozen fusion does not let the backbone adapt to a new representation. We therefore jointly train
218 backbone-only (J0), geometry-only surface (J1), predicted-chemistry (J2) and native-chemistry (J3)
219 arms on two encoders. The second, Simple2, is a 0.37M-parameter two-layer message-passing
220 encoder without equivariant layers; it tests dependence on the released equivariant architecture.
221 Matched data and update budgets control exposure, although adding a branch still changes capacity.
222 The key J3–J2 contrast asks whether native rather than predicted chemistry changes recovery after
223 both arms use a surface branch. It is $+45.01$ points $[+33.07, +60.43]$ on the released encoder and
224 $+64.14$ $[+63.38, +64.92]$ on Simple2. The first magnitude is seed-unstable. These controlled decoders
225 are not full-system reproductions.

226 Four clean contrasts have point estimates below the predeclared $+1$ -point threshold (Figure 2b). The
227 two ten-seed marginal 95% intervals have a largest upper endpoint of $+0.22$; the two three-seed rows
228 are directional only. The models are weak ($\approx 22\%$ backbone recovery) and use Shrake–Rupley clouds
229 at about 23 points per residue rather than the substrate’s solvent-excluded mesh at 53.3. A competitive
230 clean mesh/encoder experiment remains unrun. These results support bounded non-detection in the
231 tested regime, not absence of useful surface representations.

232 The finite-head positive control adds backbone-derived features while retaining the backbone input:
233 $W1-W0$ is $+2.05$ points $[+1.57, +2.53]$ over ten seeds. Its shuffled-feature arm also gains $+1.05$ $[+0.58,$
234 $+1.52]$; the aligned-minus-shuffled gain is $+1.00$ $[+0.57, +1.45]$. Thus only the latter contrast isolates
235 the alignment advantage against this capacity control. The unrestricted-head version did not clear
236 $+1$ point. This demonstrates a possible design-valid benefit under a restricted learner, not the power
237 of our surface instruments to detect every useful representation. The clean contrasts and calibration
238 share one axis in Figure 2b.

239 4.5 Removing, restoring and repairing reliance

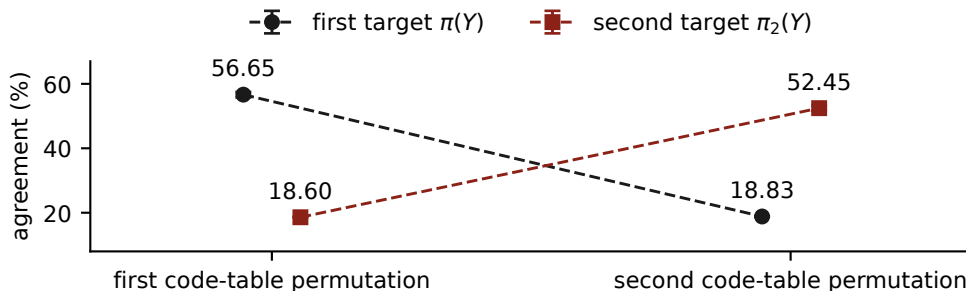
240 The controlled SurfDesign encoders allow a further question: where is the chemical-channel re-
241 liance expressed, and can the model recover when that channel is removed? With weights fixed,
242 neutralisation drops all six checkpoints below the 8.85% frequency prior. That result alone would
243 conflate reliance with an unfamiliar input distribution. Activation restoration narrows the explanation:
244 injecting the native run’s layer-0 features into the channel-free forward pass restores 73–90% of the
245 lost recovery on the released encoder and all of it on the three Simple2 seeds. Reliance is expressed
246 predominantly at the first layer under this intervention, without implying that every downstream
247 causal pathway is located there.

248 Short retraining then reaches parity with fresh training in about 1,000 steps. The operational
249 consequence is that removing an inadmissible input need not require abandoning the architecture,
250 although this repair establishes recovery only in the controlled regime. It does not reproduce a
251 competitive target-free SurfDesign system. Together, removal, restoration and repair distinguish
252 a fixed model’s dependence from a permanent inability to learn a backbone-based solution. The
253 per-seed values and intervention details remain in Appendix E.

254 5 Discussion

255 The actionable diagnosis is that two released chemical channels read the target, and one released
256 checkpoint follows a specified alternative target when that code changes. A surface-vs-no-surface gain
257 alone would not expose this mechanism. Evaluation should state the input contract, audit construction,
258 compare a simple decoder and model under one scorer, and test admissible representations with a
259 competitive learner. Similar standalone accuracy is useful context, but cannot make provenance the
260 only changing factor.

(a) Each intervention follows its own alternative target



(b) Clean joint-training effects and finite-head controls

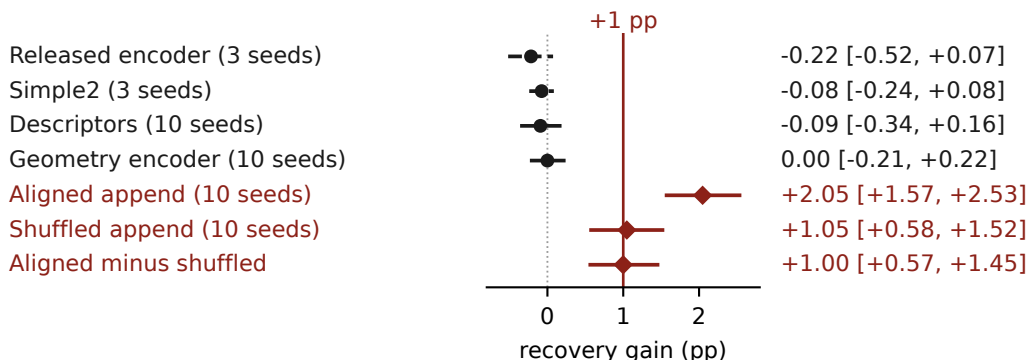


Figure 2: (a) Each intervention prefers its specified target over the other alternative (887 chains; official scorer); Section 4.2 gives the conservative mean interval. (b) Joint-training effects and finite-head controls. Solid intervals use ten seeds; dashed intervals use three and are descriptive. Aligned versus shuffled separates total feature gain from alignment. The line marks +1 point, not a universal absence bound.

Five predictions failed (Appendix C): a standalone-accuracy formula exceeded its error tolerance (5.43 versus 4 points); error overlap misclassified all six ESM-IF1 backbone channels; structured placement missed random coverage by 3.50 points against 1.5; the exposed-residue advantage reversed (17.69% versus 11.61% on sparse versus dense surfaces); and judges disagreed on fold quality, prompting retraction.

Remaining gaps are competitive design-valid surface tests and independent replication. SurfPro lacks matched-channel and clean joint-training tests; the external panel shares only 23 chains across three models. Full-atom meshing adds +2.44 decodability points but not checkpoint-use evidence. We make no prevalence, biological-function or intent claim; better clean results would complement the diagnosis.

6 Conclusion

Changing SurfPro’s input code redirects a released model to the corresponding alternative target; neither an ablation gap nor a matched standalone ceiling establishes this. Separating construction, reliance, information and finite-learner utility leaves competitive backbone-derived surface benefits open.

LLM-use disclosure

Language models and coding agents assisted this work; the authors remain responsible for its content.

References

- Bai, P., Miljković, F., Liu, X., et al. *Mask-prior-guided denoising diffusion improves inverse protein folding*. Nature Machine Intelligence 7(6):876–888, 2025.
- Bernett, J., Blumenthal, D. B., List, M. *Cracking the black box of deep sequence-based protein–protein interaction prediction*. Briefings in Bioinformatics 25(2), bbae076, 2024.
- Bushuiev, A., Bushuiev, R., Sedlar, J., Pluskal, T., Damborsky, J., Mazurenko, S., Sivic, J. *Revealing Data Leakage in Protein Interaction Benchmarks*. GEM workshop, ICLR 2024. arXiv:2404.10457.
- Connolly, M. L. *Analytical molecular surface calculation*. Journal of Applied Crystallography 16:548–558, 1983.
- Dauparas, J., et al. *Robust deep learning-based protein sequence design using ProteinMPNN*. Science 378(6615):49–56, 2022.
- Dillon, L., Maiwald, A., Crook, O. *Decoding the physicochemical basis of taxonomy preferences in protein design models*. bioRxiv 10.1101/2025.10.21.683350 v3, 2026.
- Efron, B. *Bootstrap Methods: Another Look at the Jackknife*. The Annals of Statistics 7(1):1–26, 1979.
- Feng, S., Wallace, E., Boyd-Graber, J. *Misleading Failures of Partial-input Baselines*. ACL 2019, pp. 5533–5538 (ACL Anthology P19-1554).
- Gainza, P., et al. *Deciphering interaction fingerprints from protein molecular surfaces using geometric deep learning*. Nature Methods 17:184–192, 2019 (online).
- Gainza, P., Wehrle, S., Van Hall-Beauvais, A., et al. *De novo design of protein interactions with learned surface fingerprints*. Nature 617:176–184, 2023. DOI 10.1038/s41586-023-05993-x.
- Gao, Z., Tan, C., Li, S. Z. *PiFold: Toward effective and efficient protein inverse folding*. ICLR 2023.
- Geirhos, R., et al. *Shortcut learning in deep neural networks*. Nature Machine Intelligence 2:665–673, 2020.
- Gow, L., Li, J., Tan, X., Li, L. *FORGE audits residue-level information encoded in RNA tertiary structure geometry*. bioRxiv 10.64898/2026.07.05.736550 v2, 2026.
- Gururangan, S., Swamydipta, S., Levy, O., Schwartz, R., Bowman, S. R., Smith, N. A. *Annotation Artifacts in Natural Language Inference Data*. NAACL 2018 (ACL Anthology N18-2017).
- Hsu, C., et al. *Learning inverse folding from millions of predicted structures*. ICML 2022.
- Ingraham, J., Garg, V., Barzilay, R., Jaakkola, T. *Generative models for graph-based protein design*. Advances in Neural Information Processing Systems 32:15794–15805, 2019.
- Jumper, J., et al. *Highly accurate protein structure prediction with AlphaFold*. Nature 596:583–589, 2021.
- Kapoor, S., Narayanan, A. *Leakage and the reproducibility crisis in machine-learning-based science*. Patterns 4(9):100804, 2023.
- Kaufman, S., Rosset, S., Perlich, C. *Leakage in Data Mining: Formulation, Detection, and Avoidance*. KDD 2011, 556–563.
- Korbeld, K. T., Viliuga, V., Fürst, M. J. L. J. *Limitations of the refolding pipeline for de novo protein design*. Protein Science 35(6):e70613, 2026. DOI 10.1002/pro.70613.
- Kuhlman, B., et al. *Design of a Novel Globular Protein Fold with Atomic-Level Accuracy*. Science 302:1364–1368, 2003.
- Kyte, J., Doolittle, R. F. *A simple method for displaying the hydropathic character of a protein*. Journal of Molecular Biology 157:105–132, 1982.
- Li, D., Shen, L., Song, M., Li, D., Liu, J., Liu, X. *SurfFold: a unified model for protein inverse folding by integrating surface and structural information*. Bioinformatics 42(1), btaf666, 2026.
- Lin, Z., Liu, F. *Position: Mechanistic Interpretability Must Disclose Identification Assumptions for Causal Claims*. arXiv:2605.08012, 2026.
- O’Connell, J., et al. *SPIN2: Predicting sequence profiles from protein structures using deep neural networks*. Proteins: Structure, Function, and Bioinformatics 86:629–633, 2018.
- Orengo, C. A., et al. *CATH — a hierarchic classification of protein domain structures*. Structure 5:1093–1109, 1997.

326 Poliak, A., Naradowsky, J., Haldar, A., Rudinger, R., Van Durme, B. *Hypothesis Only Baselines in Natural*
327 *Language Inference*. Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics
328 (*SEM), pp. 180–191, 2018.

329 Pubal, K., Kushnir, K., Spiwok, V., Louzecka, K., Setnicka, V., Lipovova, P. *Toward a Minimal Amino Acid*
330 *Alphabet for Protein Design*. bioRxiv 10.64898/2026.03.06.710107 v2, 2026.

331 Sanner, M. F., Olson, A. J., Spehner, J.-C. *Reduced surface: an efficient way to compute molecular surfaces*.
332 *Biopolymers* 38:305–320, 1996.

333 Satorras, V. G., Hooeboom, E., Welling, M. *E(n) equivariant graph neural networks*. ICML 2021, PMLR,
334 pp. 9323–9332.

335 Shrake, A., Rupley, J. A. *Environment and exposure to solvent of protein atoms. Lysozyme and insulin*. *Journal*
336 *of Molecular Biology* 79:351–371, 1973.

337 Song, Z., Huang, T., Li, L., Jin, W. *SurfPro: Functional Protein Design Based on Continuous Surface*. ICML
338 2024, PMLR 235:46074–46088. arXiv:2405.06693.

339 Surana, S., Grinsztajn, N., Atkinson, T., Duckworth, P., Barrett, T. D. *Overconfident Oracles: Limitations of In Sil-*
340 *ico Sequence Design Benchmarking*. ICML 2024 Workshop on AI for Science (spotlight). arXiv:2502.17246.

341 Sverrisson, F., Feydy, J., Correia, B., Bronstein, M. *Fast End-to-End Learning on Protein Surfaces*. CVPR 2021.

342 Torralba, A., Efros, A. A. *Unbiased look at dataset bias*. CVPR 2011, pp. 1521–1528.

343 Watson, D. S., Wright, M. N. *Testing conditional independence in supervised learning algorithms*. *Machine*
344 *Learning* 110:2107–2129, 2021.

345 Wei, J., et al. *HonestAffinity: Leak-Aware Evaluation of Protein and Pocket Priors for Binding Affinity Prediction*.
346 arXiv:2606.03422, 2026.

347 Williamson, B. D., Gilbert, P. B., Simon, N. R., Carone, M. *A general framework for inference on algorithm-*
348 *agnostic variable importance*. *Journal of the American Statistical Association* 118(543):1645–1658, 2023.

349 Wu, F., Jin, S., Tang, X., Gerstein, M., Zeng, X., Leskovec, J., Choi, Y., Xu, J. *SurfDesign: Effective Protein*
350 *Design on Molecular Surfaces*. Proc. 32nd ACM SIGKDD Conf. on Knowledge Discovery and Data Mining
351 V.2, pp. 12396–12407, 2026. DOI 10.1145/3770855.3818827.

352 Wu, H., Song, Y., Zhang, Zhe, Zhang, Zhilong, Zhou, H., Ma, W.-Y., Liu, J. *Rationalized All-Atom Protein*
353 *Design with Unified Multi-Modal Bayesian Flow*. *Advances in Neural Information Processing Systems* 38,
354 2025.

355 Xu, Y., Zhao, S., Song, J., Stewart, R., Ermon, S. *A Theory of Usable Information Under Computational*
356 *Constraints*. ICLR 2020.

357 Zheng, Z., Deng, Y., Xue, D., Zhou, Y., Ye, F., Gu, Q. *Structure-informed language models are protein designers*.
358 ICML 2023, PMLR 202:42317–42338.

359

A Experimental details

A.1 Released-path arms for the subject pipeline

Four arms differ only in what the surface branch carries, at matched budget: A0 the released target-derived chemistry, A2 that channel set to a constant, A3 its values shuffled within the chain (marginal preserved, alignment destroyed), A5 no surface branch. They retrain the released encoder under our decoder; they are not published-checkpoint behaviour — none exists for that pipeline — and do not reproduce the subject paper’s recoveries.

Alignment, not magnitude, is what they show. $A2 - A3 = +0.32$ pp [$+0.12, +0.56$]: shuffling channel values within a chain makes the arm slightly *worse* than a constant, while the same values aligned to residues ($A0 - A2$) recover a large gap under this retrain ($+53.17$ pp, two-level bootstrap [$+44.60, +63.78$], $n=3$; the seed range itself is [$+44.48, +63.84$]). Same values, wrong residues, no payoff. Neutralising the channel leaves the surface branch slightly worse than having none ($A2 - A5 = -0.51$ pp [$-0.76, -0.27$]). All three are two-level bootstrap intervals at $n=3$ and are read as descriptive under the rule stated in Table 8’s caption — direction robust, magnitude not (per-arm recoveries: Table 6). §4.2 runs the same alignment control on a *released* checkpoint, where it costs -46.69 pp.

This appendix expands the population, benchmark and arm-level detail behind §§3.2 and 4.2; Appendix B carries the seed-level constituents.

Matched-decodability panel detail (Table 3). The external panel sizes are 68 chains / 9,540 residues for ProteinMPNN, with ESM-IF1 and PiFold on a shared 23-chain / 3,230-residue core so those two models are read on identical structures. ProteinMPNN has a larger panel; base recoveries there are 52.31%, 60.84% and 53.50%. Two of the three seeded 2-cluster target draws tied exactly on ceiling (17.126%); the alternative pays $+11.60$ pp on ProteinMPNN and was not present in the primary ESM-IF1 artifact set. PiFold was not run on the CATH panel, which is why that cell is empty.

Base-rate reconciliation (§3.2). On our 928-chain population frozen ProteinMPNN recovers 46.07% and frozen ESM-IF1 57.53%. The published figures are 52.4% on native backbones (Dauparas et al., 2022) and 51.6% on the CATH 4.3 “All” split (Hsu et al., 2022); PiFold reports 51.66% on CATH 4.2 and LM-Design 55.65% on the same split. The differences come from population and decoding rule, not from any reimplementations: we run the released weights unmodified, on a subset of the CATH test split, with our own decoding. Our ProteinMPNN number on TS500 is 52.80%; no published ProteinMPNN TS500 figure exists to calibrate it against. §3.2 states why none of this reaches a claim. These comparisons bound our frozen-model results to the stated populations and decoding rules; they are not reproductions of every published benchmark score.

Named populations (§3.2). The body states the rule; this table states its per-population consequences.

Table 4: Named populations (Table 4). P-FROZEN-FULL is a strict superset of P-FROZEN’s 197 test chains; the counts differ because the released pipeline evaluates on its own subset of the benchmark split. No chain was re-split, reassigned or reweighted. Applying §3.2’s rule keeps 928 of the 1,120 test chains (83% by chain, 64% by residue) and drops everything longer than 258 residues, which is what bounds the frozen-model and joint-training results to that window; TS50/TS500 carry no upper bound, so the benchmark-transfer analysis does reach longer chains. P-FROZEN additionally requires at least 100 surface points per chain, caps each cloud at a seeded random 4,000 points, and takes a seeded 593/197 subsample. P-CEIL is unfiltered, which is why its frequency floor (8.86%) differs from the 8.85% used on the joint-training population.

Label	Definition
P-RELEASED	3,946 train / 790 test family-disjoint CATH 4.2 chains, 3 seeds
P-FROZEN	593 train / 197 test chains; 86,419 / 26,289 residues
P-FROZEN-FULL	928 test chains / 125,401 residues (83% of the CATH 4.2 test split by chain, 64% by residue)

Label	Definition
P-CEIL	195,631 test residues — analytic decodability bounds only
P-TS50 / P-TS500	50 / 500 chains from the subject paper’s other two benchmarks (O’Connell et al., 2018)

393 Benchmark transfer, per benchmark (§4.3). Codebook statistics and λ fitted on CATH, applied
394 unrefitted, before and after removing every chain that overlaps either CATH population. TS50 shares
395 3 exact sequences with CATH test and 1 with CATH train; TS500 shares 10 and 9.

Table 5: Benchmark transfer before and after decontamination. CATH values are Table 3’s and are not repeated; CATH is not decontaminated against itself. Removal uses an exact-identity criterion, not a similarity-clustered one, with Retained the surviving chain count. Both interval columns come from the same bootstrap run, so the before/after comparison is not confounded by resampling noise. Removing every overlapping chain shifts the contrasts by +0.02 and −0.02 pp.

Benchmark	n	Contrast	95% CI	kept	Contrast (decont.)	95% CI (decont.)
TS50	50	+12.05 pp	[+11.13, +12.96]	46	+12.08 pp	[+11.14, +13.05]
TS500	500	+12.20 pp	[+11.97, +12.43]	481	+12.18 pp	[+11.95, +12.41]

396 Released-encoder arm recoveries. The contrasts quoted in §4.2 are two-level bootstrap estimates
397 over seeds and chains, as §3.3 requires for trained arms, and differ slightly from differences of this
398 column.

Table 6: Released-encoder arms under matched training (P-RELEASED, 3 seeds). Arm labels are the released experiment’s own; A1 and A4 are internal variants that carry no claim here. Micro recovery is residue-weighted; the contrasts in the text are paired per-chain bootstrap estimates and differ slightly from differences of this column.

Arm	Channel	Micro recovery
A0	native hp	74.76% (sd 9.28)
A2	constant	21.50%
A3	shuffled within chain	21.18%
A5	surface branch off	22.07%

399 Worked decode of one chain. 1a1x.A, the first chain of the test split (108 residues), decoded from
400 the channel value alone with no model. Eight of the first ten residues are recovered exactly by table
401 lookup; the two exceptions fall in the single collision class.

Table 7: The codebook applied to one real chain, 1a1x.A. Chain-level, 80 of 108 residues (74.07%) are uniquely resolved, against the 78.65% population figure.

position	native residue	channel value	decodes to
1	G	−0.4	G uniquely
2	S	−0.8	S uniquely
3	A	+1.8	A uniquely
4	G	−0.4	G uniquely
5	E	−3.5	one of {D, E, N, Q}

position	native residue	channel value	decodes to
6	D	-3.5	one of {D, E, N, Q}
7	V	+4.2	V uniquely
8	G	-0.4	G uniquely
9	A	+1.8	A uniquely
10	P	-1.6	P uniquely

Populations. Table 4 defines them; this appendix adds only provenance. The two split files are hash-pinned — `chain_set.jsonl sha256 1944d21b...`, `chain_set_splits.json sha256 8e9a587a...` — and no chain was re-split, reassigned or reweighted by us; the selection rule that produces each evaluated population from those files is stated in Table 4.

Models. §3.2 names the four systems and their parameter counts. Nothing was modified in the two frozen models: published weights, published tokenisers, inference only.

Training. Every trained arm uses AdamW (lr 1e-3, weight_decay 1e-4) with a OneCycleLR schedule at max_lr 1e-3, gradient-norm clipping at 1.0, one chain sampled per step, and a budget of 3,000 steps. All trained arms use seeds 0, 1, 2. The controlled head is 128 hidden units with three message-passing rounds. *Matched budget* means the same optimiser, schedule, step count, clipping and seed set across the arms of a comparison; no arm received a hyperparameter search the others did not.

Compute accounting. The primary released-encoder and frozen-model work used rented A40 GPUs where the retained logs record the device; the SurfPro Stage-2 seed expansion used 24 GB RTX 4090 pods. Controlled joint-training arms also ran locally on Apple MPS, and analytic ceilings, audits and re-analyses were CPU-only. The recorded rented-GPU spend was \$9.68: \$1.93 through the first complete study freeze and \$7.75 for the prespecified extra-work round; all pods were stopped after outputs were downloaded and hash-verified. This is a cost ledger, not a hardware-normalised compute estimate; exact commands, update counts, seeds and environment versions are retained in the evidence manifests.

Uncertainty and injection. The two bootstrap procedures and the one-parameter injection rule are stated where they are used, in §3.3 and §4.3. The remaining details: the λ grid is given in §4.3 and contains 0, so a fitted $\lambda = 0$ leaves the base logits untouched and yields a gain of exactly zero rather than an estimate of one; the trained-MLP robustness variant is fitted with the same optimiser and budget as the arms it checks, and λ is refitted independently for every channel rather than carried across.

Cold replay begins without cached outputs: it fetches and hash-verifies the dataset, clones the published model, recomputes logits and reproduces the three central objects and twelve headline values. The dataset (`chain_set.jsonl sha256 1944d21b...`) and the released encoder source are pinned. The retained package contains the manuscript source and PDF, six figures and their generator, the claim contract, evidence manifest, numerical checks and two replay scripts, with checksums for every member. It excludes the historical experiment runners, which contain machine-specific paths; the reproducibility claim is limited to the portable replay scripts.

Reference verification. The bibliography is checked against two trails: the comparator papers against the project’s comparator registry, and sixteen methodological and dataset sources against Crossref — thirteen resolved by bibliographic query, three adjudicated against a named local source because conference proceedings and Nature’s AlphaFold record do not resolve cleanly. The audit trail is `Comparators/reference_verification.json`. The two trails do not cover every bibliography entry; items added after a trail was built carry their own source note instead.

B Per-seed results

An interval does not show how much a trained contrast varies across seeds. Tables 8 and 9 therefore report the seed-level constituents of every trained contrast.

The four design-valid contrasts of §4.4, which Figure 2 plots against the predeclared threshold:

Table 8: The four qualifying design-valid contrasts under joint training on P-FROZEN (197 test chains), with two-level bootstrap intervals (Table 8). Both outside-J experiments also ran a shuffled-descriptor control (B2, G2); neither carries a claim. Rows 1, 2 and 4 use the declared 4,000-point construction; preliminary 1,200-point-cache runs are excluded. Row 3 does not consume the point cloud. Row 4 uses ten seeds: its preliminary three-seed estimate was -0.23 pp $[-0.45, -0.00]$, the only Route-B row whose descriptive interval excluded zero, whereas the ten-seed result is -0.00 pp $[-0.21, +0.22]$. The rule is applied symmetrically: three seeds cannot calibrate a tail, so every three-seed two-level interval in this paper is read for direction, not for coverage — including the three that exclude zero in our favour (J3 – J2 in §4.4, A2 – A3 and A2 – A5 in Appendix A). Across ten seeds the point estimate moved to zero and the interval widened — the upper bound rose from -0.00 to $+0.22$ pp, away from zero rather than toward it — so the family still clears $+1$ pp by a factor of four and rests on no zero-excluding Route-B row.

#	Test	Contrast	Δ re- covery (pp)	95% CI	n_seed
1	released encoder, geometry-only surface	J1 – J0	-0.22 pp	$[-0.52, +0.07]$	3
2	Simple2, geometry-only surface	J1 – J0	-0.08 pp	$[-0.24, +0.08]$	3
3	rich backbone descriptors fed straight to the head	B1 – B0	-0.09 pp	$[-0.34, +0.16]$	10
4	learned encoder over rich surface geometry	G1 – G0	-0.00 pp	$[-0.21, +0.22]$	10

Table 9: Per-seed trained contrasts, in percentage points. The first eight rows are the J-series on the two encoders; the last two are the design-valid pairs that sit outside it — head only is B0/B1, descriptors fed to the decoder head with no surface encoder at all, and learned is G0/G1, a jointly trained encoder over surface geometry — whose per-seed values are the arm means differenced seed by seed. The n column is how many seeds the row’s mean and sd are computed over; the three seed columns always show seeds 0-2, so for a ten-seed row they are a window rather than the whole row. seed sd is the population standard deviation over those n seeds, not the sample estimate — at three points the two differ by a factor of $\sqrt{3/2}$, so a reader recomputing the column should know which convention it uses. It describes spread across seeds and is not an inferential interval; the chain-level bootstrap intervals for the four design-valid contrasts are in Table 8. The released encoder’s J3 – J2 spans 32.73–60.50 pp across three seeds; that range, not any single seed, is what should be quoted. The design-valid contrast J1 – J0 is seed-stable (sd well under 0.1 pp) on both encoders, even though its two-level bootstrap interval on the released encoder now includes zero — seed-stability and interval width are different properties, and the interval’s width is dominated by chain-resampling noise, not seed variance.

encoder	contrast	seed 0	seed 1	seed 2	mean	seed sd	n
released	J1 – J0	-0.32	-0.15	-0.18	-0.22	0.08	3
released	J2 – J1	-0.11	-0.34	-0.05	-0.17	0.12	3
released	J3 – J2	$+41.80$	$+32.73$	$+60.50$	$+45.01$	11.57	3
released	J3 – J0	$+41.37$	$+32.24$	$+60.27$	$+44.63$	11.67	3
Simple2	J1 – J0	-0.09	-0.14	-0.00	-0.08	0.05	3

encoder	contrast	seed 0	seed 1	seed 2	mean	seed sd	n
Simple2	J2 - J1	-0.04	-0.21	+0.11	-0.05	0.13	3
Simple2	J3 - J2	+64.21	+64.38	+63.81	+64.14	0.24	3
Simple2	J3 - J0	+64.08	+64.04	+63.92	+64.01	0.07	3
head only	B1 - B0	-0.11	-0.16	+0.08	-0.09	0.12	10
learned	G1 - G0	-0.12	-0.10	-0.35	-0.00	0.17	10

Table 10: Per-seed mechanism probes, the constituents of the ranges quoted in §4.5.

encoder	seed	A0 recovery	A2, channel removed	layer-0 restored
released	0	64.82%	7.7599%	80.79%
released	1	55.62%	7.7713%	89.97%
released	2	84.48%	6.7899%	73.20%
Simple2	0	86.20%	6.7823%	100.0000%
Simple2	1	86.47%	6.7861%	100.0000%
Simple2	2	86.15%	6.7823%	100.0000%

446 SurfPro matched retraining, per seed (§4.2). Nine arm/seed cells, 887 chains each, on SurfPro’s own
447 official scorer; the code table is the only thing that changes between arms, and data order and update
448 budget are paired by seed. Arm means are Table 11 and the contrasts they produce are Table 12.
449 The seed sd column of Table 11 is the *sample* standard deviation ($n - 1$), the convention of the
450 EXP-45 artifact these rows are quoted from — unlike Tables 8 and 9 above, which use the population
451 convention. At three seeds the two differ by $\sqrt{3/2}$, so the convention is stated per table rather than
452 assumed.

Table 11: Per-seed SurfPro Stage-2 arm means (Table 11), in % recovery.
seed sd is the sample ($n - 1$) standard deviation, matching the artifact.

arm	seed 1	seed 2	seed 3	mean	seed sd
native (released table)	57.3335	57.1449	56.8405	57.1063	0.2487
scram	56.9508	56.9499	56.5934	56.8314	0.2061
scram2	57.4244	56.9926	57.1035	57.1735	0.2243
part18 (random partition)	60.9145	60.0491	60.3971	60.4536	0.4355

Table 12: SurfPro Stage-2 paired contrasts (Table 12). The paired treatment is primary (data order and budget matched by seed); the unpaired treatment is reported as the conservative alternative, and every semantics interval sits inside the predeclared 1 pp band in both directions under both. Read for direction, not coverage, under the same three-seed rule applied above. *scram* - *native* is the one semantics contrast whose paired interval excludes zero and whose sign holds at all three seeds: a real but small table-specific effect, which is why the paper says the specific assignment may matter at the few-tenths level and not that semantics are inert. *scram2* - *scram* is sign-consistent at all three seeds but 3-of-3 carries *p* of about 0.25 and its paired interval includes zero, so it is reported as plausible, not established.

contrast	per seed (pp)	mean	paired 95% CI	unpaired 95% CI	% of the 47.82 pp gap
<i>scram</i> - <i>native</i>	-0.383, -0.195, -0.247	-0.275	[-0.516, -0.034]	[-0.707, +0.157]	0.575%
<i>scram2</i> - <i>native</i>	+0.091, -0.152, +0.263	+0.067	[-0.451, +0.586]	[-0.366, +0.500]	0.141%
<i>scram2</i> - <i>scram</i>	+0.474, +0.043, +0.510	+0.342	[-0.304, +0.988]	[-0.088, +0.772]	0.715%
<i>part18</i> - <i>native</i>	+3.581, +2.904, +3.557	+3.347	[+2.393, +4.301]	[+2.780, +3.915]	7.00%

The eleven-arm capacity family. All 11 pre-specified arms are scored — zero, *part2*, *part3*, *part5*, *part8*, *part12*, *native*, *scram*, *scram2*, *part18*, *ident*. Recovery is strongly rank-associated with the channel’s standalone capacity (Spearman 0.9636 on the coverage scorer, 0.9909 on the official one; grouped-chain interval lower bounds 0.9636 and 0.9727, both clearing the registered 0.90). No functional form survives: the pre-registered *linear* law failed (*part5* missed by +4.86 pp against a 3.7 pp tolerance, and the tolerance was not widened), and the fitted curvature changes *sign* between the two scorers, so neither “linear” nor “concave” is scorer-invariant. Only the rank association is claimed.

Reproducibility limit, stated plainly. The nine Stage-2 checkpoints (≈ 73 MB each) are on stopped pod volumes and are not in the local package. Their sha256 values are recorded, so a recovered checkpoint can be verified, but the scores, per-chain outputs, commands and logs are what is retained: this arm is not part of the advertised cold replay, and we do not describe it as one.

The two tables come from different experiments and their recovery columns should not be compared row by row: the first is the four-arm joint-training comparison on P-FROZEN, the second is the fixed-weight pathway probe on checkpoints trained separately for it. Both use seeds 0-2, and both show a related but not identical pattern — the released encoder’s *recovery* is seed-unstable (spanning 32.73–60.50 pp on the *J3* - *J2* contrast and 55.62–84.48% here) while the *necessity* probe on it is not (0.98 pp across seeds); the *localisation* probe is intermediate — seed-variable (73.20–89.97%) but far less so than recovery itself, which is the released encoder’s version of the layer-0 remainder described in §4.5.

473

474 C Complete predeclared hypotheses

475 §5 states the five outcomes in prose. The ledger is here, followed by each hypothesis in full: what
476 was predicted, how it was tested, and what came back.

Table 13: Predeclared hypotheses that failed (Table 13). Each threshold or direction was fixed before the run that tested it; all are retained because they bound the contribution.

Hypothesis	Predeclared test	Outcome
Channel payoff follows an independence law $1 - \text{acc} = (1 - \text{base})(1 - \text{ceil})$	MAE < 4 pp	REJECTED, MAE 5.43 pp
Error-overlap lift gives a transferable provenance detector	threshold transfers across models	REJECTED — misclassifies 6/6 backbone channels on the second model
Payoff depends only on the <i>count</i> of labelled residues	structured vs random placement within 1.5 pp	REJECTED — structured placement deviates 3.50 pp
The channel improves designs, not just recovery	two external judges agree	RETRACTED — the two judges disagree in sign on the same designs; no fold-quality claim is made
Exposed residues gain most from a full-atom surface	predicted direction	INVERTED — the gain concentrates where the backbone surface is <i>sparsest</i> (17.69% vs 11.61%)

1. An independence law relating base rate and channel ceiling. Predicted that a channel's payoff satisfies $1 - \text{acc} = (1 - \text{base})(1 - \text{ceil})$, i.e. that knowing a model's base recovery and a channel's standalone ceiling determines the gain. Predeclared acceptance: mean absolute error below 4 pp across the codebook sweep. Observed 5.43 pp. Rejected, and it stays rejected: a later relation of ours ($\text{gain}(1) = (1 - \text{base}) \times \text{ceil}$) is this same law at full coverage and was withdrawn for that reason rather than developed.
 2. Error-overlap lift as a transferable provenance detector. Predicted that the ratio of a channel's error overlap with the base model separates target-derived from backbone-derived channels, with a threshold that carries across models. On ProteinMPNN the two families are disjoint — target-derived [1.02, 1.49], backbone-derived [1.57, 1.61] — which is why it looked like a detector. On ESM-IF1 the ranges shift: target [1.00, 1.33], backbone [1.35, 1.38]. The threshold fitted on the first model misclassifies 6/6 backbone channels on the second. Rejected as a detector; retained only as a mechanistic observation about redundancy.
 3. Payoff depends only on how many residues are labelled. Predicted that $\text{gain}(f) = f \times \text{gain}(1)$ holds for any placement of labels, not just random placement. Predeclared tolerance: structured placements within 1.5 pp of the random-coverage line. Observed maximum deviation 3.50 pp. Rejected, and the surviving relation now carries an explicit random-coverage qualifier wherever it appears.
 4. The channel improves designs, not merely recovery. Predicted that two independent structural judges would agree that channel-conditioned designs are better. They disagreed: normalised excess of -0.0019 on one judge against $+0.3666$ on the other, over the same designs. The disagreement has a mechanical explanation — the first judge saturates near 70% identity (design 2.4901 against control 2.5320) — but the point is that we had no basis to pick the favourable judge. Retracted, and the entire fold-quality claim family was deleted rather than re-tested with the judge that agreed with us.
 5. Exposed residues gain most from a full-atom surface. Predicted direction: side chains reach the surface only on exposed residues, so the identity information a full-atom mesh adds should concentrate there. Inverted. The gain concentrates where the *backbone surface is sparsest* — $+6.08$ pp there (17.69% against 11.61%) versus $+0.53$ pp where it is densest. The stratifier that matters is local surface density, not burial. A related exposure-gradient prediction failed in sign the same way (exposed $+2.50$ / $+3.50$ pp, buried -2.58 / -3.50 pp) and that claim family was withdrawn.
- Two of these — 1 and 3 — are the numerically comparable pair plotted in Figure 4. Hypotheses 4 and 5 failed in kind rather than in magnitude and cannot be placed on a ratio axis; 2 failed as a *transfer* claim rather than against a numeric threshold.

513 D Threats to identification and claim boundaries

514 D.1 Identification threats and corresponding controls

515 These are the identification threats named in §5; each control is stated only to the evidentiary ceiling
516 it supports.

517 Weak clean-surface models limit interpretation. Frozen-model fusion measures utility for a particular
518 rule at published weights, while joint training measures representation effects at the controlled models’
519 budgets. Neither establishes the effect of a competitive clean mesh encoder trained at published scale.
520 That study remains unrun.

521 Marginal decoding accuracy is not conditional information or an isolated provenance contrast. The
522 original matched pair differs in cardinality, complementary errors and learnability. The exact example
523 in §1.3 and the retained-output analysis in Appendix H expose this limitation directly. The payoff
524 gap remains a descriptive observation about the tested channels and learners.

525 A fitted lambda of zero need not exclude a nonlinear benefit. The tested 256-256 MLP also returns
526 negative backbone gains on the matched and learned-predictor channels. This extends non-detection
527 to that finite model class; it does not rule out all useful interactions.

528 The scope of recurrence remains limited. We verify construction dependence in two implementations
529 and targeted reliance on one released checkpoint. Source inspection of a backbone-derived method
530 provides a contrasting construction, not a third full experimental replication. The audit workflow can
531 return inconclusive; reuse on other systems is a prospective test.

532 Input-capacity boundary. Full-atom meshing buys +2.44 pp decodability over backbone-only meshing
533 across 100 chains, far below the chemistry channel’s 85.89% ceiling: the geometry channel is a
534 capacity boundary, not a second provenance finding.

535 D.2 Released-checkpoint error fingerprint

536 The 20x2 code table of §4.2 resolves the 20 residues into 18 classes, colliding exactly {D,E} and
537 {N,Q}. Over 137,346 aligned residue pairs, the released checkpoint’s four largest confusions are
538 exactly the four ordered pairs inside those two classes — Q → N 32.58%, D → E 32.34%, E →
539 D 21.43%, N → Q 17.86% — and the fifth largest, C → L, is 2.4 times smaller at 7.53%. The
540 control that separates “reads the table” from “knows chemistry” is I/L/V: as similar biochemically
541 as D/E, but given three distinct values by the table, and separated by the model (I → L 7.21%). The
542 alignment used here charges an explicit mismatch and gap cost and is used only to read confusions,
543 never to compute a recovery number.

544 D.3 Claim boundaries

545 These eleven boundaries define the scope of the paper’s claims:

- 546 • the contribution is a released-pipeline diagnosis and a workflow for interpreting interven-
547 tions;
- 548 • $I(Y; f(B) \mid B) = 0$ is motivation, not novelty, and not a theorem;
- 549 • Route B is 4 qualifying joint-training contrasts; the 2 frozen-model falsifiers are reported
550 separately;
- 551 • only the two ten-seed rows support marginal interval statements below +1 pp; the three-seed
552 rows are directional, and no familywise absence statement is made;
- 553 • say 3 named benchmarks, 1 primary data authority, 0 independent external-data replications;
- 554 • say benchmark transfer, not independent-data generalisation;
- 555 • do not generalise from the tested label-free arm to all molecular surfaces;
- 556 • do not claim behaviour of unavailable published checkpoints;

- do not claim fold quality, biological function, field prevalence, intent or misconduct;
- full-atom meshing is an input-capacity boundary (capacity gain stated in §5), not checkpoint use;
- random-coverage and fixed-codebook relations are secondary.

561

562 The subject pipeline’s own channel ablation. The released paper states its surface features are
 563 “computed from local atomic geometry and element types, not from residue identity labels” and that
 564 the pipeline does “not use: (i) residue identities ... during the surface generation pipeline”. Its
 565 experimental appendix takes three channels from MaSIF — Poisson–Boltzmann electrostatics via
 566 APBS, hydrophobicity, and free electrons/protons — and reports: “*After a further ablation study, we*
 567 *discover that the hydrophobicity and the charge are pivot to the performance improvement while the*
 568 *electrostatics is not necessary.*”

569 Of those three, the two named as pivotal are the two computed by table lookup on the residue name,
 570 and the one named as unnecessary is the one solved from atomic geometry. §4.1 measures the first of
 571 them moving 89.12% under fixed-coordinate permutation against 0.00% for the geometry channel.
 572 We report the discrepancy between a stated scope and a released implementation, and make no claim
 573 about how it arose; a removal ablation of the kind quoted is exactly the comparison §1.3 shows cannot
 574 separate the two routes, so it does not settle the question either way.

575 Two nearby audits, and what they do not supply. An RNA audit framework quantifies which residue-
 576 level annotations a coarse geometry representation supports, isolating an information source with an
 577 ablated seven-atom versus six-atom control (Gow et al., 2026) — the same audit spirit, on backbone
 578 geometry rather than a derived chemistry channel, with no determinism claim and no code table to
 579 permute. A separate enumeration searches subsets of a ten-letter amino-acid pool for de novo design
 580 (Pubal et al., 2026); our enumeration is over partitions of the full twenty-letter alphabet and scores a
 581 released model’s codebook, which is a different object and a different question.

582 The matched ceiling is a frontier over our constructions, and we tested where it stops. §4.3 matches
 583 the two families at a $\approx 17\%$ marginal ceiling, and the natural objection is that this is far below the
 584 released channel’s 85.89%. Across the retained 48-pair sweep no backbone-derived channel reaches
 585 a train-estimated marginal ceiling above 21.50% on either frozen model, attained by bbkmeans1024.
 586 Two objections to reading that as a frontier are real, and we ran the test rather than answering them in
 587 prose.

588 *First, the estimator.* That 21.50% fits the per-cluster argmax on the same train residues it scores, and
 589 with ≈ 84 points per cluster the optimism grows with k . Refitting on train and scoring on held out
 590 test moves bbkmeans1024 from 21.50% to 18.60% — a +2.95 pp bias. On target-derived codebooks
 591 of at most 18 classes the same bias is +0.14 to +1.37 pp, which *falsified* a predeclared prediction that
 592 it would stay under 1 pp: the target side carries a small optimism too, about half a point larger than
 593 we predicted, and an order of magnitude smaller than the backbone side’s.

594 *Second, the stopping point.* $k = 1024$ was where the original sweep stopped, not a limit. Pushing the
 595 same recipe to 4,096, 16,384 and 65,536 clusters makes the train-estimated ceiling keep climbing —
 596 25.73%, 36.51%, 74.05% — while the held-out ceiling turns over and collapses: 18.60%, 17.02%,
 597 14.39%, 12.58%. At 1.3 train points per cluster the train estimate is optimistic by 61.47 pp: it was
 598 measuring memorisation, not capacity. Five clustering seeds at $k = 1024$ and 4,096 and three at
 599 16,384 put every held-out spread below 0.41 pp.

600 So the honest frontier over our constructions is 18.60% held out, and it is saturated in the sense that
 601 spending more clusters now *lowers* it. Two independent constructions land in the same region: the
 602 learned residue predictor over 19 backbone-only geometric features reaches 21.43% train-estimated,
 603 and full-atom meshing buys only +2.44 pp of decodability over backbone-only meshing. A three-class
 604 recoding of the native residue already exceeds the whole backbone family at 23.79%.

605 What this does and does not license. The matched contrast sits at $\approx 17\%$ because that is the top of the
 606 range in which *we* could build a design-valid channel. This is a bound over one k -means descriptor
 607 family on this population, and no amount of saturating it can prove that no design-valid learned
 608 representation exceeds it — a better one would raise the matched point, and §4.4 says so. The claim
 609 is that the 85.89% contrast is un-run for want of a construction, never that it is impossible.

610 Capacity buys recovery only when the capacity is about the target. The same operation — more
611 classes — has opposite signs in the two families (Table 14).

Table 14: The ladder (Table 14), in pp of native sequence recovery on frozen ProteinMPNN. Target-derived payoff is strongly rank-associated with ceiling (Spearman +0.9779; +0.9448 on ESM-IF1); backbone-derived payoff is rank-associated *negatively* (−0.7638; −0.9411). The registered test was Spearman > 0.90 and it passed, but the relation is not literally monotone: sorted by ceiling, target payoff *decreases* at 7 of 26 adjacent ProteinMPNN steps and 6 of 17 on ESM-IF1, the largest single drop being −7.78 pp. Raising a design-valid codebook from the smallest to the largest buys 10.47 pp of ceiling and costs 0.92 pp of recovery. Inside the shared ceiling band no target-derived channel pays less than any backbone-derived one: the separation is +11.54 pp on ProteinMPNN and +9.75 pp on ESM-IF1. Point estimates over the sweep, not a distributional claim; all four predictions here were predeclared and confirmed.

family	channel	ceiling	Δ
backbone	burial3	11.03%	+0.00
backbone	bbkmeans4	13.71%	+0.00
backbone	bbkmeans16	17.00%	+0.00
backbone	bbkmeans64	18.41%	+0.00
backbone	bbkmeans256	19.73%	−0.16
backbone	bbkmeans1024	21.50%	−0.92
target	rand_k2_s0	16.66%	+11.54
target	rand_k3_s0	23.79%	+19.32
target	rand_k4_s0	29.30%	+24.89
target	rand_k8_s0	48.32%	+34.57
target	rand_k16_s1	71.94%	+44.51
target	kd+charge	90.45%	+48.30

612 The provenance shortfall. Table 3 pairs one channel from each family at a matched ceiling, and
613 a reviewer can fairly ask whether that single pair generalises. Fit the capacity–payoff relation on
614 target-derived channels only, read it at each backbone channel’s own ceiling, and report the gap.
615 Channels whose ceiling falls outside the target range are excluded rather than extrapolated.

616 This is an estimator, and we say which one. Our first version justified the interpolation by calling
617 the target relation monotone; it is not (Table 14), so that justification is withdrawn — piecewise-
618 linear interpolation *is* a local functional-form assumption, and this paper has already had one such
619 assumption rejected (§5’s independence law). The shortfall is therefore re-run under four estimators
620 fixed in advance and reported under all of them: linear interpolation, isotonic regression, cardinality-
621 averaged local linear, and the monotone lower envelope, the smallest prediction any non-decreasing
622 reading of the target curve permits. We take the *minimum*, not the largest.

623 Two further repairs. The original leave-one-out left out one row, but the random recodings come in
624 three seeds per cardinality, so a held-out row was usually predicted from two near-duplicate siblings;
625 the control is now a group holdout removing an entire cardinality at once (2.26 pp ProteinMPNN,
626 3.33 pp ESM-IF1, against the same bar). And the estimate now carries an interval: a joint chain
627 bootstrap resamples train chains (moving every ceiling) and test chains (moving every gain) in the
628 same replicate and reruns the whole estimator inside it, $B = 1000$.

629 So qualified, the estimator predicts its own family well and the backbone family very badly, always
630 in the same direction:

Table 15: The provenance shortfall (Table 15), in pp of native sequence recovery, ProteinMPNN / ESM-IF1. Predictions come from the target-derived relation alone; the design-valid channels were never used to fit it. The shortfall column is the minimum over four predeclared estimators (linear interpolation, isotonic regression, cardinality-averaged local linear, monotone lower envelope), not the largest; the last column is the worst 2.5th percentile over all four under a joint chain bootstrap ($B = 1000$, train and test chains resampled together, the whole estimator rerun inside each replicate). The linear estimator alone gives +11.95 / +13.50 / +15.09 / +17.76 and +10.23 / +11.67 / +12.94 / +15.26. The shortfall grows with capacity on both models, and the smallest lower bound anywhere in the table, +9.65 pp, is still three times the grouped-holdout error of 2.26 / 3.33 pp inside the family the relation was fitted on. Two predeclared glosses failed here and are corrected rather than dropped: “an order of magnitude” (measured $5.1\times$ to $7.5\times$), and “monotone”. The bbkmeans1024 bootstrap mean sits *above* its own point estimate because resampling train chains inflates a 1024-cluster codebook’s train-estimated ceiling — the same optimism measured at +2.95 pp above — so the lower bound, not the mean, is the operative number. A descriptive interpolation over a fixed sweep on two frozen models, not a law.

channel	ceiling	measured	min shortfall	bootstrap 2.5th pct
bbkmeans16	17.00%	+0.00 / +0.00	+11.54 / +10.04	+11.12 / +9.65
bbkmeans64	18.41%	+0.00 / -0.16	+12.11 / +10.46	+11.68 / +10.11
bbkmeans256	19.73%	-0.16 / -0.17	+12.28 / +10.48	+11.84 / +10.14
bbkmeans1024	21.50%	-0.92 / -0.83	+13.03 / +11.14	+12.71 / +10.94

631 The channel’s ceiling is not a property of the chemistry. §4.1’s 85.89% ceiling is the Bayes-optimal
632 recovery of one particular partition of 20 residues into 17 classes — the one the Kyte–Doolittle scale
633 happens to induce. There are 741,285 such partitions ($= S(20,17)$), each with a closed-form ceiling
634 $\sum_b \max_{y \in b} p(y)$, so whether this codebook is unusually leaky is decidable by enumeration rather
635 than by argument. We enumerated all of them on P-CEIL from residue frequencies alone: no model,
636 no coordinates, no sampling. 82.04% have a strictly higher ceiling. The biochemical table ranks
637 608,152nd, at the 17.96th percentile; the best attainable reaches 95.12%, the median 88.50%, the
638 worst 77.99%. A table drawn at random therefore leaks 2.61 pp more than the released one at the
639 median.

640 The mechanism is arithmetic, not chemistry. Merging a block costs everything but its most common
641 member, so a merge is cheap exactly when the mass it destroys is small. Kyte–Doolittle merges
642 {D, E, N, Q} — the residues sharing hydropathy -3.5 — which carry 21.35 pp of mass and keep only
643 E (7.24 pp), destroying 14.11 pp. The cheapest table merges {C, K, M, W}: the three rarest residues
644 together with a common one, K (6.12 pp), which becomes the class’s prediction and so costs nothing
645 — 4.88 pp in total. Chemical similarity and informational cheapness are unrelated, so a codebook
646 designed on chemistry merges whatever is chemically alike, and chemically alike residues are not
647 rare.

648 Controls fixed before the run reproduce §4.1 exactly — the 8.86% floor, and the Kyte–Doolittle
649 ceiling on all three splits (85.89% test, 86.39% train, 86.44% validation), each against a separately
650 stored value — and the ranking holds on all three (17.96th, 20.46th, 22.41st percentile). A predeclared
651 prediction failed: we expected the best table to merge only residues from the five rarest, and it merges
652 K, the sixth *most* common. The optimum is not “collide the rarest” but “collide the rarest with a
653 common one”. It is retained, like the five in §5.

654 This bounds a parameter-free lookup, not a trained model, and it is a statement about table design, not
655 about anyone’s intent: the Kyte–Doolittle scale is a hydropathy measurement and was never chosen
656 to maximise residue recovery. What it settles is that the provenance problem here is structural — it
657 follows from partitioning residue identity into 17 classes at all, not from which partition was chosen
658 — which is why §5’s burden asks for a capacity bound rather than for a defence of the scale.

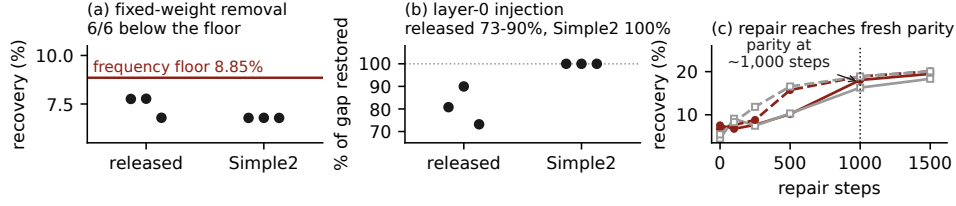


Figure 3: The mechanism chain. (a) With weights held fixed, neutralising the channel drops all six checkpoints below the frequency floor. (b) Layer-0 injection restores 73-90% of the gap on the released encoder and 100% on Simple2, so the commitment is made at layer 0 in both, completely in one. (c) Repair reaches parity with fresh training in about 1,000 steps; dark = repair, light = fresh, solid = released, dashed = Simple2.

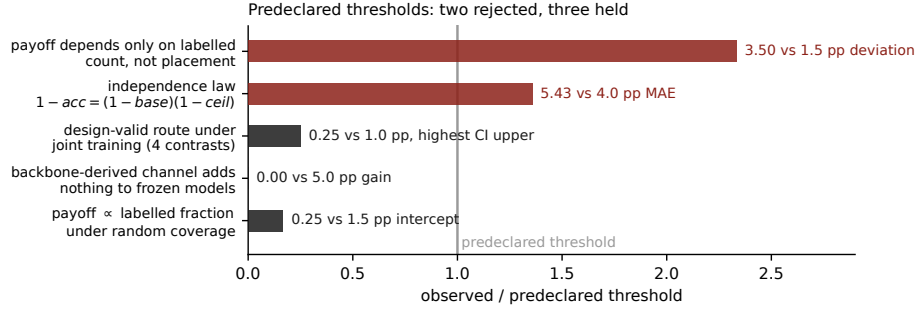


Figure 4: How the predeclarations actually performed. Five predictions whose thresholds were fixed before the runs that tested them, plotted as observed over threshold with the decision line at 1.0. Two were rejected. Three further predictions are not ratio-comparable and are listed in Table 13, so the five plotted here and the five in that table are different sets that overlap in two rows.

659 E Additional figures and figure provenance

660 Table 8 carries the four qualifying design-valid contrasts of §4.4 with their intervals and per-seed
661 constituents; the body plots them against the predeclared +1 pp threshold alongside the positive
662 control, with a separate finite-head positive calibration; this does not establish power for every surface
663 representation.

664 The controlled-model mechanism analysis has three stages. *Necessity*: with weights fixed, neutralising
665 the channel at inference drops recovery below the 8.85% frequency floor of P-FROZEN in 6/6
666 checkpoints. *Localisation*: injecting the native run’s layer-0 features into a channel-free forward
667 pass restores 73–90% of the gap on the released encoder and 100% (3/3 seeds) on Simple2 —
668 predominantly first-layer, not exclusively so. *Repair*: from below the floor, about 1,000 steps reach
669 parity with fresh training. The three stages are plotted together in Figure 3.

670 Figure 4 is placed here rather than in the body because Table 13 is the falsification record and this
671 is a secondary view of a different five-item subset: the three predeclarations that held, and the two
672 failures that are ratio-comparable with them.

673 The codebook panel illustrates the finding stated in §4.1 (17 channel values, sole collision {D, E, N, Q},
674 85.89% against 8.86%), so it is explanatory rather than load-bearing. The released channel’s codebook
675 is plotted in Figure 5.

676 Every figure is captioned where it appears; this appendix records only how each was produced, which
677 the captions do not say.

678 EXP-49’s re-estimation of the design-valid frontier is the one place where an estimator and the
679 quantity it estimates part company visibly, so the comparison of training against held-out backbone
680 lookup accuracy is Figure 6.

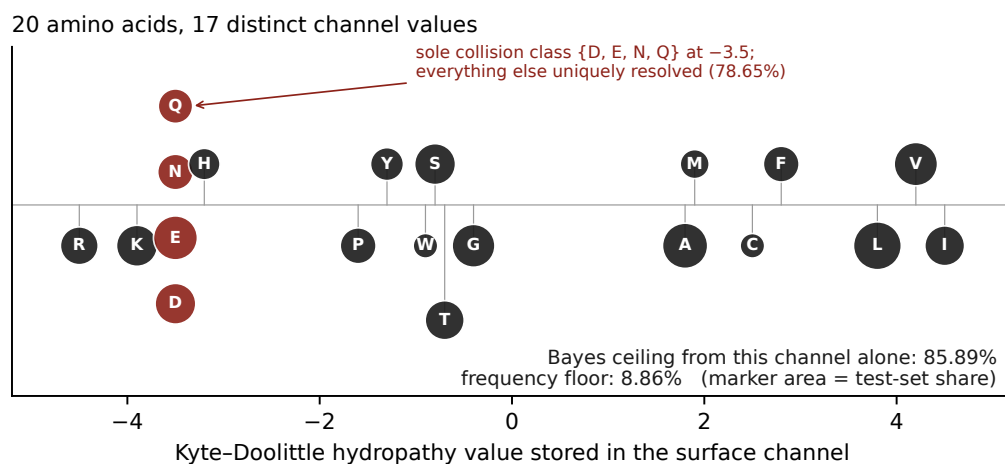


Figure 5: The artifact statistic. All 20 amino acids placed on the hydropathy axis used by the released channel; marker area is the residue's share of the test population. Twenty amino acids map to seventeen distinct values, with a single collision class, which is what puts the Bayes ceiling of the channel alone at 85.89% against an 8.86% frequency floor.

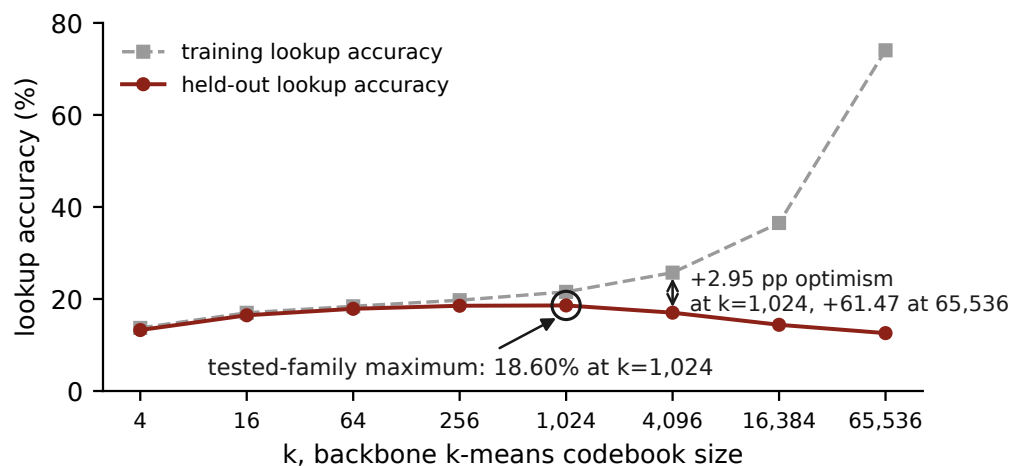


Figure 6: Training accuracy rises while held-out lookup accuracy declines. The per-cluster argmax is fitted on train; its same-sample accuracy reaches 74.05%, whereas held-out accuracy peaks at 18.60% at k=1,024 and then declines. These are accuracies of a fitted decoder, not population Bayes bounds. Shading shows the observed range over clustering seeds.

681 All six are generated by one command, `make_figures.py`. A separate panel for the matched-
682 decodability contrast is omitted because Table 3 carries the same result with strictly more information.
683 Five are built directly from the retained artifacts and the script prints every plotted value at render
684 time, to expose the plotted values for checking; the frequency floor and Bayes ceiling drawn as
685 reference lines are read from `EXP1_hp_decodability/exp1_results.json` rather than typed in.
686 One figure (`fig1_underidentification`) carries no measured value and reports that when it runs.

Table 16: The six rendered figures (Table 16), one schematic and five derived from retained artifacts. The build refuses to render a figure whose source artifact is absent, and the current build checks this list against the placed figures.

#	file	kind
1	<code>fig1_underidentification</code>	schematic
2	<code>fig4_route_b_bound</code>	artifact-derived
3	<code>fig5_mechanism_chain</code>	artifact-derived
4	<code>fig6_predeclared</code>	artifact-derived
5	<code>fig7_codebook</code>	artifact-derived
6	<code>fig8_frontier_saturation</code>	artifact-derived

687 The build refuses to proceed if a hand-written “Figure N” label disagrees with document order, if a
688 figure anchor falls mid-paragraph, or if this table and the placed figure set disagree in either direction.

689

690 F Constructing matched ablation gaps

691 §1.3 argues that Δ_{ablation} cannot separate the routes. That argument is an identification claim,
692 so it can be *built* rather than only asserted, and a construction is only worth reporting if its Route-A
693 channel genuinely depends on the target — otherwise it proves equivalence by collapsing Route A
694 into Route B.

695 A finite learner may condition on at most 64 cells. The backbone takes 4,096 states, so the learner
696 cannot afford the raw backbone and must coarsen it. Two channels of identical cardinality (8) are
697 offered to it:

- 698 • Route B — C = $g(B)$, a partition of the backbone states built by clustering them on the
699 *population* conditional $P(Y|B)$, an oracle construction using the population law, rather than
700 an implementable finite-data training procedure. It never reads an instance’s Y .
- 701 • Route A — C = $a(Y, \text{noise})$, which names Y ’s class with probability p and is otherwise
702 independent noise. p is calibrated on a disjoint sample so the match is never fitted to the
703 evaluation set.

704 Over five seeds at 200,000 evaluation points each, with the same learner, baseline and estimator
705 throughout:

Table 17: The two constructed pipelines of Appendix F (Table 17). Their ablation gaps agree to 0.07 pp across five seeds and their provenance does not: under an exact $\text{do}(Y)$ intervention the Route-A channel moves 31.25% of its values and the Route-B channel moves none. This is §1.3’s argument built rather than argued.

quantity	Route A	Route B
ablation gap Δ	11.65 pp	11.67 pp
largest per-seed gap difference	0.07 pp	0.07 pp
95% CI of the gap difference	[−0.06, +0.01]	[−0.06, +0.01]

quantity	Route A	Route B
channel moved under exact do(Y)	31.25%	0.00%
conditional information on the target	0.96 bits	0.00 bits

The matched scalar gap in this construction does not reveal the source; its specified intervention does. Three properties make the construction non-degenerate rather than rhetorical. The Route-A channel carries 0.96 bits about the target given the *full* backbone state, so it is not a disguised function of B. A positive control — the same Route A with a higher leak rate — separates the two gaps by 9.72 pp, so the matching is a property of the construction and not of a blunt estimator. And the intervention recovers the true leak fraction to a mean absolute error of 0.0036, which is the same instrument reading 90.12% on the released system in §4.2.

The construction is deterministic: re-running it reproduces the result file bit-for-bit.

G Evidence governance

Scope of the following audit record. The counts and replay claims below describe the earlier frozen rev268 package. They do not certify this revised manuscript or a newly uploaded supplement. The rev271 implementation record supplies a new input/output hash manifest, scoped numerical checks, exact-construction assertions, audit-tool regression results and a rendered-page inspection. No independent full replay or human reader study was run in this revision. Original experiments and frozen packages remain unchanged.

The two audits, and what each certifies. The looser one asks whether a manuscript decimal exists anywhere in the evidence base; it reports 0 unbound, and the earlier manuscript claimed only that. The stricter one asks whether a value sits *in* the artifact its own contract row names. Of 828 contract decimals, 743 bind there directly — exactly, rounded, or as a truncated quote of a longer stored decimal — and 70 are reproduced from that same file by arithmetic: a pairwise difference in the artifact’s unit or in percentage points, a sum, an interval midpoint, or a ratio. The remaining 15 are declared non-measurements, listed one at a time with a reason in the contract’s own section NC: a published constant (the Kyte–Doolittle value of the collision class), a dataset version, a predeclared threshold, and figures a row itself labels superseded. 0 are mis-scoped, and the check exits non-zero if one is.

An earlier adjudicator pattern matched only an ASCII hyphen, so every negative contract value was read as positive and charged against an artifact storing its negation. That inflated the miss count and hid five correctly scoped rows among the residue. Accepting the typographic minus this paper actually uses moved 134 values from miss to bound; the check was then mutation-tested — a falsified interval bound, a falsified point estimate and a falsified ceiling are each still caught.

Why the stricter audit is stated separately. A guard that verifies a number exists *somewhere* in a large evidence base verifies very little: the pooled set of artifact decimals here covers essentially the whole two-decimal grid over [0,100), so membership is nearly free. Row scoping restores discrimination by cutting the search space from that pool to the few dozen values in one named file. The declaration list is the escape hatch, and its size is printed on every run so it cannot grow unnoticed.

Entry point. README.md -> checks/archive_self_audit.py re-hashes every member, rebinds every decimal and confirms the headline values still render using nothing outside the archive. The $\approx$ 100 historical runners are not part of that advertised replay, and we do not claim they are.

H Identification analyses and diagnostic-tool validation

H.1 Exact counterexample

Let Y be uniform on three states and B indicate $Y=0$. A channel computed as $C=B$, one computed as $C=1[Y=0]$, and one computed as $C=1[Y=1]$ all have class probabilities (2/3,1/3) and standalone

750 Bayes accuracy $2/3$. B alone distinguishes state 0 from $\{1,2\}$. The first two channels add no further
751 distinction, while the third separates 1 from 2 and hence makes (B,C) identify Y exactly. Thus
752 $H(Y|B)=2/3$ bit and $H(Y|B,C)$ is respectively $2/3$, $2/3$ and 0. The conditional information follows by
753 subtraction.

754 Clamping B and enumerating all three replacement labels at each original state produces nine software
755 evaluations. The three constructions change $0/9$, $4/9$ and $4/9$ outputs, respectively. The first two
756 have identical observational joint laws but different preprocessing dependence. The intervention
757 includes combinations outside observational support; it identifies code dependence under that software
758 contract, not a physical causal mutation effect. This textbook finite example is explanatory, not a new
759 theorem. The retained enumeration script (`EXP50_identification_examples/run.py`) checks
760 the stated fractions and information values exactly up to floating-point logarithms. No random sample
761 or fitted parameter is involved.

762 H.2 Counterfactual specificity on released SurfPro outputs

763 We retrospectively analyze retained official-scorer arrays on 887 chains (EXP52). For the first
764 permutation, intended-target agreement is 56.65%; for the second it is 52.45%. The corresponding
765 intended-minus-other differences are 38.05 and 33.62 points. Define S as half the sum of these
766 two differences. $S=35.84$ points, with interval $[34.89,36.74]$. This uses 10,000 within-arm chain-
767 bootstrap draws (RNG 271); each constituent interval has nominal 97.5% coverage and their averaged
768 endpoints form a nominal at-least-95% Bonferroni interval, subject to bootstrap approximation.
769 Positive differences occur on 883 and 879 of 887 chains; medians are 39.04 and 34.25 points. These
770 comparisons counter the explanation that one alternative is generically easier for both outputs.

771 The retained scoring code evaluates both targets for each prediction in one loop. Array lengths,
772 finite ranges and all saved means are checked. Retained cross-arm score arrays omit chain IDs;
773 raw Cpi/Cpi2 target order was not independently recoverable. Consequently the primary S interval
774 requires no cross-arm ordering. Paired cross-arm differences and difference-in-differences are
775 retained in results.json only as sensitivity analyses conditional on the historical common ordering.
776 They are not stronger headline evidence. Uncertainty covers chains, not new checkpoints or random
777 permutations. Native, pi and pi2 target scores for every arm, quantiles and all analysis code are
778 retained.

779 H.3 Error complementarity and cardinality sensitivity

780 The complementarity analysis (EXP51) uses the unchanged P-FROZEN cache: 593 training chains /
781 86,419 residues and 197 test chains / 26,289 residues. The fixed ProteinMPNN baseline is 46.52%
782 micro recovery. All configured binary target recodings (seeds 200,201,202) and two backbone K-
783 means codebooks ($K=2,16$) are reported. K-means and feature standardisation use training descriptors
784 only; `random_state=0`, `n_init=3` and `batch_size=4096` reproduce the original recipe. The per-class
785 lookup uses the same training data and Laplace smoothing as the original fusion. Selecting lambda
786 on train is preserved; this is not a held-out calibration experiment.

Table 18: Retrospective descriptor/fusion comparison. Fractions use all test residues as denominator; intervals resample 197 chains with 10,000 paired draws (RNG 271). Rows are descriptive, with no familywise hypothesis-rejection claim. Different fusion gains can occur with identical argmax accuracy because the full probability tables differ. These rows are not the 928-chain headline population.

Channel	Train lookup (%)	Test lookup (%)	Channel right, backbone wrong (%)	Fusion gain (pp)
Target binary, seed 200	16.66	16.18	6.29 [5.94,6.67]	+11.54
Target binary, seed 201	17.13	16.62	6.85 [6.44,7.28]	+12.55

Channel	Train lookup (%)	Test lookup (%)	Channel right, backbone wrong (%)	Fusion gain (pp)
Target binary, seed 202	17.13	16.62	6.85 [6.44,7.28]	+11.68
Backbone, 2 classes	11.27	10.83	4.60 [4.30,4.89]	+0.00
Backbone, 16 classes	17.00	16.46	4.39 [4.14,4.65]	+0.00

The seed-201 minus sixteen-class difference in complementary correct predictions is 2.46 points [2.03,2.92]. Among positions where the backbone is wrong, the respective lookup accuracies are 12.81% and 8.20%. The union of correct argmax predictions is not an accuracy upper bound for distribution fusion: combining distributions can correct positions where neither individual argmax is right. No error-overlap provenance detector is proposed; the previously failed cross-model detector remains rejected in Appendix C. Historical seed-200 accuracy and fusion gain are reproduced numerically. Train/test code assignments, lookup probability tables, cluster centres, normalisation constants, labels and chain indices are retained in a compressed NumPy artifact.

H.4 Conservative diagnostic utility

The revised `provenance_probe` (0.3.0) compares supplied values and returns one of four verdicts: target dependence detected, no dependence detected, unverified control, or invalid intervention. Any changed channel value can support dependence if every chain’s supplied geometry is unchanged and the caller satisfies the labels-only contract. No unchanged substitution certifies $C=f(B)$. Nonempty geometry is required for each chain; the tool can check only supplied inputs and cannot establish fixture completeness.

Regression cases include rare dependence (one of 200 values), an invariant class-preserving substitution, missing and partial geometry, changed geometry, duplicate IDs and empty data, alongside the retained target-derived and clean fixtures. They correct two defects in v0.2: a 1% threshold that falsely certified small changes, and a geometry check that could certify partial coverage. This validation is local instrument calibration, not evidence of broad uptake. The observational screen remains explicitly non-identifying. Its legacy `standalone_ceiling` key is documented as train-fitted lookup accuracy, not a population Bayes bound.

H.5 Additional related work

Partial-input baselines. Hypothesis-only NLI classifiers (Gururangan et al., 2018; Poliak et al., 2018) and benchmark-artifact work (Torralba & Efron, 2011; Geirhos et al., 2020) diagnose *datasets*, and are themselves one-directional (Feng et al., 2019); we diagnose a *derived input channel inside a method*, at matched marginal decodability (§4.3).

Ablation as an estimand. That a with/without contrast is algorithm-specific, and that its population analogue vanishes whenever the feature is conditionally redundant given the rest, is settled in the variable-importance literature (Williamson et al., 2023; Watson & Wright, 2021); a concurrent position paper makes the parallel argument for interpretability (Lin & Liu, 2026). Route B is then the creation of *usable* information by computation (Xu et al., 2020), which $I(Y; f(B) \mid B) = 0$ permits because Shannon information is unchanged. Those concepts motivate inspecting whether a method computes its input from the target; our source interventions test that question for the audited implementations. Leakage taxonomies (Kaufman et al., 2011; Kapoor & Narayanan, 2023) already include illegitimate feature construction; our per-example substitutions measure dependence in specific implementations. Protein-ML work on split leakage and evaluator validity (Bushuiev et al., 2024; Bernett et al., 2024; Surana et al., 2024; Korbeld et al., 2026; Wei et al., 2026) questions evaluation with inputs fixed; we do the reverse. Nearest in spirit is a concurrent RNA audit of residue-level information in tertiary geometry (Gow et al., 2026) and a concurrent enumeration of amino-acid *subsets* for de novo design (Pubal et al., 2026); ours enumerates *partitions* of a shipped codebook. Closest in framing (Dillon et al., 2026): one score, several information routes, models grouped by what is available at scoring time — but that work decomposes variance by *controlling*

830 observed confounders where we intervene, and its grouping presumes one already knows what each
831 model reads, which is the assumption §4.1 breaks.

832 Surface-conditioned design. Surface fingerprints and their use in interaction design (Gainza et al.,
833 2019, 2023; Sverrisson et al., 2021) and surface-conditioned designers (Song et al., 2024; Li et al.,
834 2026; Wu et al., 2026) are the substrate; backbone-only inverse folders (Dauparas et al., 2022; Hsu et
835 al., 2022; Gao et al., 2023; Zheng et al., 2023) are bridges, and their ablations are the underidentified
836 comparison of §1.3.

837 SurfDesign is the closest substrate: its surface ablation does not separate target-derived chemistry from
838 a backbone-only surface control, which §§4.3–§4.4 investigate under specified, limited instruments.
839 Wu et al. (2025) remove a related side-chain-to-sequence shortcut by redesigning a generator; we
840 instead identify provenance in released pipelines. Surface-fingerprint interaction design (Gainza et al.,
841 2023) and refolding-evaluator limitations (Korbeld et al., 2026) establish that surfaces can be useful
842 and evaluators unreliable, but neither resolves provenance. The general limitations of ablation and
843 leakage are established prior ideas; our particular source-to-prediction diagnosis combines controls
844 on released pipelines. Other systems also differ: one construction is backbone-derived (Bai et al.,
845 2025), whereas several surface pipelines use residue identity. We make no prevalence claim; this
846 variation motivates a protocol, not a family-wide generalisation.

847 **I Additional analyses and disclosures**

848 **I.1 Reproducibility, disclosure and ethics**

849 Retained artifacts include outputs for the exact enumeration, retrospective analyses and audit stress
850 tests. Appendix H records input hashes, protocols and limitations; Appendix G separates cold replay
851 from later checks. We claim neither public availability nor human-reader validation.

852 The implementation discrepancy concerns code and measured behaviour, not intent. An input contract
853 may legitimately differ for interaction or binder design. A private disclosure package exists but has
854 not been sent; no transmission is claimed.