RECEIVER-QUERY DIVERSITY IN GRAPH TRANSFORMERS: FROZEN INTERVENTIONS MISESTIMATE ADAPTIVE CAPACITY

Anonymous authors
Paper under double-blind review

ABSTRACT

Full attention is often valued for its receptive field. Here I investigate a different resource: how many distinct receiver-specific summaries a model can construct from that shared context. Whereas global pooling broadcasts a single uniform representation to all nodes, full attention can tailor a report to each node. I formalize the spectrum between these extremes by parameterizing attention through M discrete receiver-query modes. Theoretically, a random-access lower bound and matching construction establish the necessity of query diversity, while controlled graph experiments demonstrate that the demand diversity of receiver tasks, rather than context size alone, governs the required number of modes. Empirically, restricting query diversity within standard Graph Transformers significantly degrades dense node-level segmentation (COCO-SP, PascalVOC-SP), confirming that query diversity represents a primary capacity bottleneck. Crucially, I show that standard frozen ablations, commonly used to assess architectural components, fundamentally misestimate the capacity recoverable through end-to-end adaptation. Across multiple benchmarks, training with constrained queries recovers substantial performance over post-hoc frozen interventions, yet this recovery gap non-monotonically reverses at higher capacities. The findings establish that receiver-query diversity is a genuine capacity axis for Graph Transformers, while demonstrating that frozen sensitivity, adaptive capacity, and routing trainability are distinct properties that standard ablation practices routinely conflate.

1 INTRODUCTION

Message-passing graph neural networks (GNNs) and graph Transformers differ in many coupled ways. The usual comparison of local versus global access asks which nodes can exchange information. It misses a second question that remains after every node is visible: how many different global reports can be tailored to different receivers? Here a receiver is simply the node whose representation is being updated. In a scientific mesh, for example, different locations may read the same global field while requiring different information from it.

Distance, over-squashing, and positional encoding remain important (Alon & Yahav, 2021; Ma et al., 2023; Sun et al., 2024), but they do not answer this question. A pooled global feature gives every receiver the same report. Full attention can give every receiver its own report. Between them, a bank of M reports can serve groups of receivers with similar demands (Fig. 1). All three systems may see exactly the same nodes; only their capacity to tailor summaries differs.

The hypothesis is that receiver-query diversity is a capacity axis distinct from global access. A second methodological question follows: can that capacity be read from an intervention on a frozen model? I test the first question with matched trained endpoints and the second with a prospectively specified trained-versus-frozen comparison. Keeping these estimands separate is essential: frozen compression measures sensitivity of one fitted solution, whereas constrained retraining measures adaptive capacity.

Here the contributions are:

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

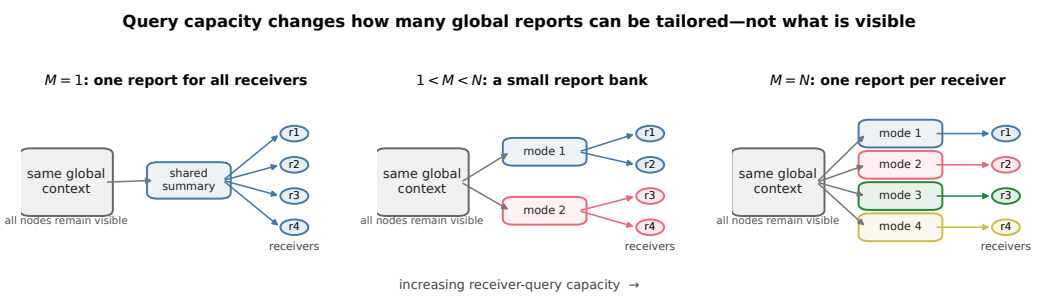

Figure 1: **Global access and query capacity are different resources.** Every variant sees the same global context. With $M = 1$, all receivers obtain one shared summary. Intermediate M provides a small bank of summaries, and full attention can tailor one to every receiver. I ask how task performance saturates along this path.

- I make query count an explicit capacity variable and prove a random-access lower bound with a matching exact-recovery construction. Controlled experiments show why effective demand diversity, rather than node count alone, governs the value of additional summaries.
- In standard tuned GPS, one shared query costs 0.0303 COCO-SP and 0.0355 PascalVOC-SP macro-F1 across three paired seeds despite unchanged global access and parameter count.
- In a prospective five-seed PascalVOC-SP test, retraining with four query modes improves macro-F1 by 0.0216 over the paired frozen intervention, with a 95% seed-block bootstrap interval of $[0.0114, 0.0289]$. Although the trained capacity curve rises from $M = 4$ to $M = 16$, the paired frozen-to-trained correction is 0.0280 larger at $M = 4$, ruling out a universal positive recovery correction based on frozen sensitivity. A same-forward straight-through control is significantly detrimental at $M = 16$, directly testing the strongest simple optimization-artifact explanation. A prospectively specified Peptides-func test replicates the estimand gap at an intermediate budget on an unrelated graph-level task: training at $M = 4$ gains 0.0346 AP over freezing a full model and imposing $M = 4$, with all three paired effects positive.

2 Why the number of summaries is a capacity

Consider K independent facts and a receiver that later asks for one of them. If the system must form one short report before learning which fact is wanted, that report must preserve information about all K possibilities. If instead the receiver can choose among M reports, each report can specialize. The following elementary random-access result makes this dependence on M explicit; its role is to isolate information ordering rather than to equate neural width with bits.

Theorem 1 (Query-mode random-access separation)**.** *Let $Z = (Z_1, \dots, Z_K)$ be independent fair bits, and let a uniformly chosen query Q request $Y = Z_Q$. An encoder forms M reports before observing Q, each containing at most b bits, and Q selects one report for decoding. If the Bayes classification error is $p \in [0, 1/2]$, then*

$$h_2(p) \geq 1 - \frac{Mb}{K}, \qquad p \geq h_2^{-1}\left(\max\{0, 1 - \frac{Mb}{K}\}\right).$$

Moreover, $\lceil K/b \rceil$ modes suffice for exact recovery.

When $Mb < K$, some error is unavoidable; when $M = \lceil K/b \rceil$, each report can store a different block of facts and exact recovery is possible. The full proof and a general task-conditioned distortion definition are in Appendix A. The bound is the classical one-way random-access phenomenon (Kushilevitz & Nisan, 1997; Cover & Thomas, 2006), expressed in the resource varied by the experiments.

The number of facts is not sufficient to predict difficulty. If receiver demands are correlated, many facts can share a low-dimensional summary. For linear squared retrieval, Appendix A.3 shows that the information lost by a B-dimensional shared report is exactly the covariance energy beyond its

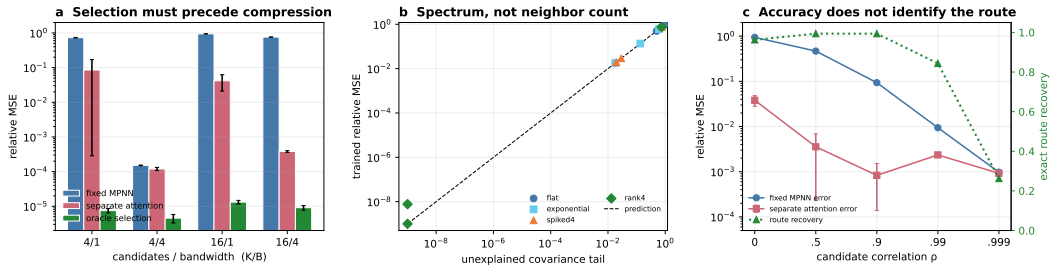

Figure 2: **Selection before compression isolates receiver demand.** (a) A shared report fails only when independent candidate information exceeds its bandwidth; an oracle route verifies that the post-selection channel is sufficient. (b) Shared-summary error follows covariance energy outside the available subspace. (c) Correlated demands can share a compact report even when exact routes are not identifiable.

leading B directions. This special case motivates measuring effective query variation in the learned GPS rather than using graph size as a proxy.

3 Controlled Mechanism Tests

I first isolate the proposed mechanism before using a standard model. Each receiver must retrieve one value from K candidates. A shared-summary graph layer compresses the candidates before seeing which value the receiver needs; an attention layer selects before applying the same bottleneck. Trainable parameter count, message and decoder networks, optimizer, training budget, and seeds are matched.

3.1 Raw Graph Layers Reproduce the Boundary

At $K = 16, B = 4$, fixed relative MSE is 0.7592, close to the independent-candidate floor 0.75, while attention reaches 3.82×10^{-4}. At $K = B = 4$, the gap closes (1.52×10^{-4} versus 1.19×10^{-4}). An oracle selector solves every cell through the same bottleneck (4.5×10^{-6}– 1.3×10^{-5}). Thus the separation appears only when the shared report is too small to preserve the independent candidate values; it is not caused by a parameter or access mismatch.

3.2 Effective Dimension, Not Raw Candidate Count

For equicorrelated candidates, $Z_j = \sqrt{\rho}C + \sqrt{1-\rho}\epsilon_j$, the exact fixed error is $(K-B)(1-\rho)/K$. At $K = 16$, trained raw MPNNs track this law with $R^2 = 0.999988$ for $B = 1$ and 0.996851 for $B = 4$. A factorized linear solver on flat, exponential, spiked, and exact-rank-four spectra is retained only as a numerical implementation check: because its data and objective instantiate the same PCA problem as the theorem, near-exact agreement is not independent hypothesis evidence. Detailed sweeps, route-recovery diagnostics, and the failed strict $10\times$ gate are reported in Appendix B.

4 Receiver-Conditioned Selection in Standard GPS

The theory predicts a graded resource, not merely a difference between the one-query and full-query endpoints. I test it on COCO-SP and PascalVOC-SP, public LRGB nodewise semantic segmentation benchmarks (Dwivedi et al., 2022). The base model is the reassessed, tuned eight-layer GPS recipe (Rampášek et al., 2022; Tönshoff et al., 2024): every layer combines a local GatedGCN branch with four-head full attention.

4.1 One Query Is a Consequential Counterfactual

Standard GPS uses the current state of each receiver to request its own global summary. The one-query counterfactual instead gives every receiver the graph-mean query and therefore the same kind

of global summary. It retains global keys and values, attention and output projections, the eight local GatedGCN branches, residual and feed-forward blocks, depth, optimizer, global access, and exact parameter count. It removes only the ability to form receiver-specific global summaries.

Table 1 shows the validation-selected test results. Standard GPS reproduces the reassessed regime: the 0.387 ± 0.006 COCO-SP and 0.455 ± 0.006 PascalVOC-SP scores are close to its reported 0.388 ± 0.006 and 0.444 ± 0.007, respectively (Tönshoff et al., 2024). All causal claims use paired arms within recipe, not comparisons with the lower original LRGB numbers. Standard GPS wins every paired seed on both datasets. Even with the same global access, one shared summary therefore loses roughly three to four macro-F1 points.

Table 1: **Receiver-specific summaries matter at fixed global access.** Means and sample standard deviations over three paired training seeds.

Dataset	standard GPS	one shared query	paired gain	wins
COCO-SP	$.387 \pm .006$	$.356 \pm .002$	$.0303 \pm .00470$	3/3
PascalVOC-SP	$.455 \pm .006$	$.420 \pm .005$	$.0355 \pm .00895$	3/3

One Pascal run ended after epoch 198; the endpoint analysis uses the common epoch window 0–198 for all arms, and every selected checkpoint occurs by epoch 157. This operational truncation and a failed prediction that Pascal would show a smaller effect are documented in Appendix C.

4.2 Frozen sensitivity is not adaptive capacity

A common way to estimate required capacity is to compress a trained model and scan the budget. That experiment is not causally matched to a model trained under the constraint. I test the difference explicitly. Receiver queries are clustered independently in every graph and layer using deterministic farthest-first initialization and eight Lloyd updates. Receivers assigned to one cluster share its mean pre-projection query. This changes no trainable parameter. I compare two estimands for the same M: (i) apply the constraint only at inference to a full-query checkpoint (frozen), or (ii) train GPS end-to-end with the constraint in every layer (trained).

The training-time comparison was fixed before any intermediate-budget outcome was inspected. PascalVOC-SP is primary; budgets are $M \in \{1, 4, 16, \text{full}\}$ with five paired seeds and a common final-epoch checkpoint rule. Existing $M = 1$ and full endpoints are reused only after exact forward-and-backward equivalence tests. I preregistered two signed contrasts: $\mathbb{E}[\Delta_4] > 0$ and $\mathbb{E}[\Delta_4 - \Delta_{16}] > 0$, where

$$\Delta_M = F_M^{\text{trained}} - F_M^{\text{frozen}}.$$

All values are recomputed with one full-precision evaluator; uncertainty resamples complete seed blocks.

The frozen scan substantially understates performance attainable after adaptation at $M = 4$: F1 rises from 0.4231 ± 0.0086 to 0.4447 ± 0.0096. The paired gain is 0.0216, with a 95% bootstrap interval $[0.0114, 0.0289]$ (Fig. 3). All five paired effects are positive. Relative to independently trained endpoints, $M = 4$ recovers 60.7% of the mean $M = 1$–full gain, although its interval is broad ($[35.8\%, 79.0\%]$).

The two metrics must not be conflated. Trained capacity rises from 0.4447 at $M = 4$ to 0.4472 at $M = 16$; the descriptive trained recoveries are likewise monotone, $T_4 = 0.607$ and $T_{16} = 0.678$. This difference is small and its post-hoc interval includes zero. By contrast, the frozen-to-trained correction depends on how strong the frozen baseline already is. Frozen $M = 16$ obtains 0.4537 ± 0.0115, whereas trained $M = 16$ obtains 0.4472 ± 0.0031, giving $\Delta_{16} = -0.00642$. Its post-hoc interval includes zero, so I do not claim population-level harm. The preregistered paired contrast is decisive: $\Delta_4 - \Delta_{16} = 0.0280$ $[0.0207, 0.0353]$. The trained $M = 16$ models use all 16 clusters and reach lower training loss than $M = 4$, arguing against codebook collapse or a failure to fit. Counter-intuitively, a full-query model compressed to 16 modes only at inference is at least as strong as the model trained under that same constraint. Thus capacity and hard-routing trainability interact; frozen sensitivity does not provide a universal nonnegative recovery correction for adaptive capacity. Per-seed results are in Appendix D; the historical weight-frozen curves and the exploratory cross-dataset predictor are in Appendices E–G.

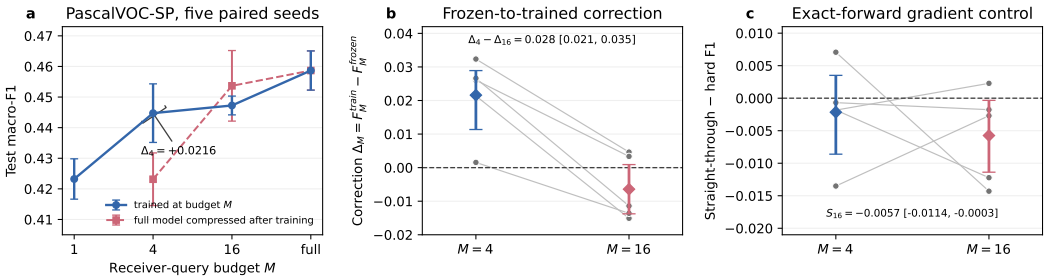

Figure 3: **Frozen sensitivity does not identify adaptive capacity.** (a) PascalVOC-SP test macro-F1 under matched trained budgets and inference-time compression of paired full-query checkpoints; error bars are sample standard deviations over five seeds. (b) Each gray line is one paired seed. Diamonds show means and 95% seed-block bootstrap intervals; the Δ_4 interval and the $\Delta_4 - \Delta_{16}$ contrast were prospectively specified, whereas the Δ_{16} interval is post hoc. Training closes a substantial frozen-to-trained gap at $M = 4$, whereas the correction is slightly negative at $M = 16$; this panel is not the trained capacity curve. (c) A parameter-matched straight-through control has the identical hard forward computation and changes only the final receiver-to-prototype assignment gradients; it is detrimental at $M = 16$.

I directly test whether missing gradients through discrete receiver-to-mode assignments explain the boundary. The control preserves the exact hard forward assignments and all 500,613 parameters, but uses a scale-normalized softmax surrogate for the final assignment backward pass. With the same five seeds, straight-through minus hard-routing F1 is -0.00216 $[-0.00861, 0.00351]$ at $M = 4$ and -0.00574 $[-0.0114, -0.00033]$ at $M = 16$ (Fig. 3c; Appendix D.1). Thus this gradient estimator is significantly detrimental at the high budget. It rules out the simplest missing-final-assignment-gradient account, while leaving fully soft or learnable routing as a distinct untested mechanism.

4.3 The same estimand gap, a different capacity conclusion

Peptides-func supplies an unrelated graph-level test of the methodological claim, while also bounding the capacity claim. Its prospective validation gate had predicted that trained full queries would outperform trained $M = 1$; that prediction failed, so I stopped the planned intermediate-budget training. I did not reinterpret that failure as evidence of low query diversity: a separately frozen post-hoc audit finds $V_4 = 0.460$, essentially COCO's 0.462, and $V_{16} = 0.812$, below its prespecified 0.85 support threshold.

I then froze a distinct prospective estimand test before training at $M = 4$. This does not reopen the failed capacity gate: it asks whether frozen compression and constrained training disagree at an intermediate, less severe intervention. On the official test split with one concatenated multilabel-AP evaluator, frozen $M = 4$ obtains 0.6273 ± 0.00949 AP and trained $M = 4$ obtains 0.6619 ± 0.0110. Thus $\Delta_4^{\text{Peptides}} = +0.0346$ AP, with all three paired effects positive $(+0.0323, +0.0441, +0.0273)$ and a prespecified seed-block interval $[0.0273, 0.0441]$. The test passes its preregistered rule that every effect be positive and the mean exceed 0.015 AP. Trained full-minus-$M = 4$ is -0.00771 on average and changes sign across seeds, so this result replicates the estimand gap without rescuing the failed capacity-superiority prediction.

An earlier same-evaluator post-hoc audit at the more severe $M = 1$ intervention gave $\Delta_1^{\text{Peptides}} = +0.0851$ AP. I use it only to examine intervention intensity, not as the independent replication. Some magnitude ordering is expected: imposing $M = 1$ is more severe than imposing $M = 4$, so $|\Delta_M|$ can naturally grow as M decreases. I do not treat the size of Δ_1 itself as the non-trivial result. The important observation is that neither intervention intensity nor the frozen curve determines the sign or scale of the correction: across the measured settings, $\Delta_1^{\text{Peptides}} = +.0851$, $\Delta_4^{\text{Peptides}} = +.0346$, $\Delta_4^{\text{Pascal}} = +.0216$, and $\Delta_{16}^{\text{Pascal}} = -.00642$, even while Pascal's trained capacity increases from $M = 4$ to $M = 16$.

Table 2: **Frozen compression misstates adaptive performance on two tasks.** All entries use final checkpoints and a common test evaluator within each row. Both rows are prospective; Pascal uses five paired seeds and Peptides uses three. Metrics differ and are not compared across rows.

Dataset	metric, budget	frozen M	trained M	trained full	correction Δ_M
PascalVOC-SP	F1, $M = 4$	.4231	.4447	.4587	+.0216
Peptides-func	AP, $M = 4$	.6273	.6619	.6542	+.0346

5 Related work

GNN and graph-Transformer biases. Over-squashing describes loss of distant information under repeated local aggregation (Alon & Yahav, 2021); graph Transformers add positional, topological, or dynamic-routing biases (Ma et al., 2023; Sun et al., 2024). GPS combines local message passing with global attention (Rampášek et al., 2022), and LRGB supplies standardized tasks where long-range structure matters (Dwivedi et al., 2022; 2023). Subsequent reassessment showed that normalization and tuning close much of the originally reported gap between graph Transformers and message passing (Tönshoff et al., 2024); the interventions therefore use that reassessed GPS recipe and matched within-recipe controls. These works primarily study access, topology, or architectural recipes. I hold access and the local branch fixed and intervene only on the diversity of receiver queries.

Conditional aggregation. Deep Sets characterize invariant sum-decomposition (Zaheer et al., 2017), while Transformers and Set Transformers introduce conditional interactions (Vaswani et al., 2017; Lee et al., 2019). Set Transformer inducing points and Perceiver latent arrays compress large inputs through fixed latent bottlenecks (Lee et al., 2019; Jaegle et al., 2021). Those architectures learn latent receivers; I instead constrain the number of distinct reports delivered back to the original receivers while holding GPS fixed. Random-access and communication-complexity results formalize why compressing before seeing a query can be costly and have also characterized Transformer expressivity (Kushilevitz & Nisan, 1997; Cover & Thomas, 2006; Sanford et al., 2023). The query-mode theorem interpolates between a shared invariant summary and fully receiver-specific attention, making the number of conditional summaries explicit; unlike the sequence constructions of Sanford et al. (2023), it isolates an explicit receiver-report budget and tests it by intervention in graph Transformers.

Query compression and efficient attention. Clustering queries has been proposed to accelerate Transformers (Vyas et al., 2020), alongside many low-rank and sparse approximations. Here the objective is different: query clustering supplies the same parameter-free capacity constraint for a frozen intervention and end-to-end retraining. This paired design exposes the estimand gap between sensitivity of one fitted model and capacity after adaptation.

6 Limitations

The theorem concerns discrete messages with explicit entropy budgets; it does not turn neural width into bits. The covariance formula is limited to linear channels and squared retrieval, and retained query variance is an empirical proxy rather than a universal task statistic. The controlled experiments are designed to identify the mechanism, not to imitate natural graph data. Appendix K collects every scope boundary in one place.

The trained one-query endpoints cover two related superpixel datasets with three seeds; the primary intermediate-budget study covers PascalVOC-SP with five paired seeds and one standard GPS recipe. Peptides-func is a scope boundary for the capacity-axis prediction: its preregistered validation gate found no trained full-query advantage over one query (mean -0.00908 AP, negative in all three seeds), and a label-free post-hoc audit rejects the tempting explanation that graph-mean readout implies low intermediate query variation ($V_4 = 0.460$, essentially COCO's 0.462). The separate prospective $M = 4$ estimand test (Section 4.3; Appendix I) replicates only the frozen-versus-adaptive gap; it neither reverses the failed capacity prediction nor establishes equivalence between trained endpoints. The cross-dataset predictor in Appendix F is an exploratory map fitted on six COCO

budgets that systematically underpredicts Pascal at low budgets; it is not a universal law. The one-query comparison is an ablation of GPS, not a new leaderboard method.

The hard assignment rule is parameter-free but non-differentiable in its cluster assignments. At $M = 16$, constrained training does not improve over the paired frozen intervention despite full cluster utilization and lower training loss than $M = 4$. I therefore do not claim that adaptive performance must follow the frozen curve, that four modes are universally sufficient, or that frozen ablations always overestimate required capacity. The trained curve itself increases from $M = 4$ to $M = 16$. An exact-forward straight-through estimator for the final assignment is detrimental at $M = 16$; fully soft clustering, learned prototypes, or a different optimizer may still behave differently. A prospective adjacent-epoch assignment audit shows the predicted lower ARI and higher churn at $M = 16$ in all three paired seeds, but five of six rerun endpoints fail its fixed reproduction tolerance. I therefore do not present routing nonstationarity as the mechanism behind the boundary.

The prototype executor lowers peak attention allocation by about 24% at $M = 16$ on Pascal but is 16% slower than the optimized full-attention kernel (Appendix H). I therefore make no deployment or efficiency claim. One earlier Pascal endpoint run ended after epoch 198; the validation-selected endpoint comparison uses the common 0–198 window. The prospective capacity study instead uses complete epoch-199 checkpoints for all four budgets and five seeds.

7 Conclusion

Global attention is not a binary property. Once access is held fixed, the number of receiver-conditioned summaries remains a consequential capacity axis. A random-access separation and controlled graphs expose the mechanism; an exact GPS intervention shows that one shared query loses three to four macro-F1 points on two public benchmarks. Most importantly, a prospective five-seed test shows that a frozen capacity ablation can misstate what a constrained model achieves after adaptation: retraining at $M = 4$ gains 0.0216 F1 over inference-only compression, while the correction changes sign at $M = 16$ even though trained capacity itself increases. An unrelated prospective Peptides-func test finds the same estimand error at $M = 4$: retraining improves over matched frozen compression by 0.0346 AP, with all three paired effects positive, even though full attention has no consistent trained advantage. A same-forward straight-through gradient control is detrimental at $M = 16$, separating this result from both a universal positive recovery correction and the simplest hard-assignment-gradient artifact. Architectural studies should distinguish what a fitted model is sensitive to from what its model family can learn under the same constraint.

Reproducibility statement

Every reported result is generated by a versioned script, machine-readable summary, and time-stamped protocol. The trained-capacity protocol was frozen before intermediate-budget outcomes, reuses endpoints only after exact forward-and-backward equivalence tests, and reports all five paired seeds with 100,000 seed-block bootstrap resamples. The repository includes tests for clustering determinism, endpoint equivalence, parameter preservation, same-forward straight-through routing, and the efficient prototype executor. The Peptides-func continuation decision uses only its frozen validation gate; the failed screen and stopped continuation are retained in the artifact map. Its matched test audit is explicitly labeled post-hoc, uses one evaluator for all three arms, and reports every paired seed.

AI use statement

I used generative AI tool for knowledge acquisition, experiment execution and manuscript language polishments. I have reviewed all AI-assisted work: the proof, implementation, result, citation. I take responsibility for the final content of this work, including text, results, and claims.

References

Uri Alon and Eran Yahav. On the bottleneck of graph neural networks and its practical implications. In *ICLR*, 2021.

Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley, 2 edition, 2006.

Vijay Prakash Dwivedi, Ladislav Rampášek, Mikhail Galkin, Ali Parviz, Guy Wolf, Anh Tuan Luu, and Dominique Beaini. Long range graph benchmark. In *NeurIPS Datasets and Benchmarks Track*, 2022.

Vijay Prakash Dwivedi, Chaitanya K. Joshi, Anh Tuan Luu, Thomas Laurent, Yoshua Bengio, and Xavier Bresson. Benchmarking graph neural networks. *Journal of Machine Learning Research*, 24 (43):1–48, 2023.

Andrew Jaegle, Felix Gimeno, Andrew Brock, Oriol Vinyals, Andrew Zisserman, and Joao Carreira. Perceiver: General perception with iterative attention. In *ICML*, 2021.

Eyal Kushilevitz and Noam Nisan. *Communication Complexity*. Cambridge University Press, 1997.

Juho Lee, Yoonho Lee, Jungtaek Kim, Adam Kosiorek, Seungjin Choi, and Yee Whye Teh. Set transformer: A framework for attention-based permutation-invariant neural networks. In *ICML*, 2019.

Liheng Ma, Chen Lin, Derek Lim, Adriana Romero-Soriano, Puneet K. Dokania, Mark Coates, Philip Torr, and Ser-Nam Lim. Graph inductive biases in transformers without message passing. In *ICML*, 2023.

Ladislav Rampášek, Mikhail Galkin, Vijay Prakash Dwivedi, Anh Tuan Luu, Guy Wolf, and Dominique Beaini. Recipe for a general, powerful, scalable graph transformer. In *NeurIPS*, 2022.

Clayton Sanford, Daniel Hsu, and Matus Telgarsky. Representational strengths and limitations of transformers. In *NeurIPS*, 2023.

Junshu Sun, Chenxue Yang, Xiangyang Ji, Qingming Huang, and Shuhui Wang. Towards dynamic message passing on graphs. In *NeurIPS*, 2024.

Jan Tönshoff, Martin Ritzert, Eran Rosenbluth, and Martin Grohe. Where did the gap go? reassessing the long-range graph benchmark. *Transactions on Machine Learning Research*, 2024.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017.

Apoorv Vyas, Angelos Katharopoulos, and François Fleuret. Fast transformers with clustered attention. In *NeurIPS*, 2020.

Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, Barnabás Póczos, Ruslan Salakhutdinov, and Alexander Smola. Deep sets. In *NeurIPS*, 2017.

A FORMAL MODEL AND PROOFS

This appendix gives the formal statements behind the report-bank intuition in the main paper. A "report" is an information-limited message formed from a common context; a "query mode" specifies which report a receiver can inspect. The finite-information result is a direct application of the classical one-way INDEX problem. Its contribution here is conceptual: it exposes the number of receiver-conditioned summaries as a resource between pooling and full attention.

A.1 TASK-CONDITIONED DISTORTION

Let context data be Z, receiver demand be Q, and target be $Y = t(Z, Q)$. For loss ℓ and declared channel class $\mathcal{C}$, define

$$D_{\text{shared}}(\mathcal{C}, \ell) = \inf_{e \in \mathcal{C}, d} \mathbb{E}\, \ell(Y, d(e(Z), Q)), \tag{1}$$

$$D_{\text{adaptive}}(\mathcal{C}, \ell) = \inf_{a \in \mathcal{C}, d} \mathbb{E}\, \ell(Y, d(a(Z, Q), Q)). \tag{2}$$

The shared encoder must summarize Z without knowing Q; the adaptive encoder may tailor its summary after seeing Q. Between them, an M-mode encoder forms $B_1(Z), \ldots, B_M(Z)$ and a fixed query gate $g(Q)$ exposes one message to the decoder. Its optimum D_M satisfies

$$D_{\text{adaptive}} \le D_M \le D_{\text{shared}}, \tag{3}$$

with $M = 1$ recovering one shared report. The channel class is essential: without an entropy, noise, regularity, quantization, or linear-rank restriction, a real scalar can pathologically encode arbitrarily many discrete states.

A.2 PROOF OF THE QUERY-MODE RANDOM-ACCESS THEOREM

Let $g(i) \in \{1, \ldots, M\}$ choose the report available to coordinate query i, and define $S_m = \{i : g(i) = m\}$. Let p_i be the minimum error in predicting Z_i from $B_{g(i)}$ when the decoder knows $Q = i$, and let $p = K^{-1} \sum_i p_i$. Because Z_i is a fair bit, binary Fano gives, for $i \in S_m$,

$$H(Z_i \mid B_m) \le h_2(p_i), \qquad I(Z_i; B_m) \ge 1 - h_2(p_i). \tag{4}$$

The coordinates are independent. For each mode, conditional-entropy subadditivity therefore gives

$$\sum_{i \in S_m} I(Z_i; B_m) \le I(Z_{S_m}; B_m) \le H(B_m) \le b. \tag{5}$$

Summing over modes yields

$$\frac{1}{K} \sum_{i=1}^{K} h_2(p_i) \ge 1 - \frac{Mb}{K}. \tag{6}$$

By concavity of h_2, $h_2(p) \ge K^{-1} \sum_i h_2(p_i)$, proving the lower bound whenever the right-hand side is positive; clipping at zero covers $Mb \ge K$.

For achievability, partition the K coordinates into $\lceil K/b \rceil$ blocks of at most b bits. Store one block per report and route query i to the block containing coordinate i. The decoder uses Q to read the requested bit. Thus $\lceil K/b \rceil$ modes suffice for exact recovery. The bound and construction assume that g depends only on the query. If the gate could also depend on Z, the gate index itself could communicate information and would need to be charged to the budget.

A.3 LINEAR SQUARED RETRIEVAL AND COVARIANCE TAIL

The finite-bit theorem does not describe continuous neural features. A second, loss-specific special case explains why effective variation matters. Let centered $Z \in \mathbb{R}^K$ have covariance Σ, let Q be uniform and independent, and let $Y = Z_Q$. A shared rank-B linear encoder A and stacked query-conditioned decoder D have average coordinate loss

$$\frac{1}{K} \sum_{q=1}^{K} \mathbb{E}[(Z_q - d_q^\top A Z)^2] = \frac{1}{K} \mathbb{E} \| Z - DAZ \|_2^2. \tag{7}$$

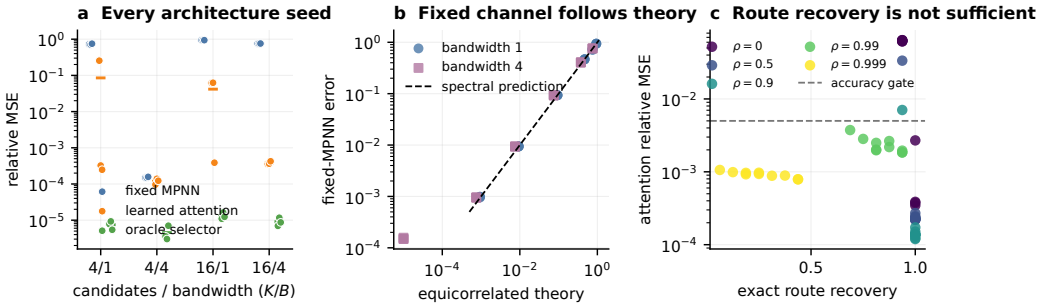

Figure 4: **Seed-level controlled audit.** All seeds are shown. Discrete learned-attention failures correspond to missed routes rather than omitted outliers. The preregistered claim that learned attention would exceed a $10\times$ gap in every hard cell is rejected: the weakest gap is $8.56\times$. The oracle and correlation conclusions are unaffected.

Writing $W = DA$, with $\mathrm{rank}(W) \le B$, gives

$$\mathbb{E}\|Z - WZ\|_2^2 = \|\Sigma^{1/2} - W\Sigma^{1/2}\|_F^2. \tag{8}$$

Eckart–Young shows that the optimum discards exactly $\sum_{j>B} \lambda_j(\Sigma)$. Dividing by total variance gives relative risk

$$R_B^\star = \frac{\sum_{j>B} \lambda_j(\Sigma)}{\mathrm{tr}\,\Sigma}. \tag{9}$$

An adaptive width-one encoder can transmit Z_Q exactly. This is a theorem only for linear channels and squared retrieval; I do not interpret it as a universal neural-width law.

B Controlled mechanism evidence

Each synthetic example is an irregular directed graph. A receiver has K candidate scalar values and a selector indicating which candidate determines the target. The shared-summary MPNN maps candidates into B dimensions and sums before the selector reaches the decoder. Edge attention uses the selector before the same B-dimensional channel. Message and decoder networks, embeddings, optimizer, update count, and trainable parameter count are matched within every comparison.

At $K = 16, B = 4$, the shared MPNN has relative MSE 0.7592, close to the independent candidate floor 0.75, while attention reaches 3.82×10^{-4}. At $K = B = 4$, both have enough bandwidth and the gap closes (1.52×10^{-4} versus 1.19×10^{-4}). The oracle selector solves all cells through the same bottleneck (4.5×10^{-6}–1.3×10^{-5}).

For equicorrelated candidates $Z_j = \sqrt{\rho}C + \sqrt{1-\rho}\epsilon_j$, the exact linear shared-summary error is $(K-B)(1-\rho)/K$. At $K = 16$, trained MPNNs track this curve with $R^2 = 0.999988$ for $B = 1$ and 0.996851 for $B = 4$. Route recovery falls to 26.3% at $\rho = .999$ while prediction remains accurate, showing that receiver demands can share a compact report when their candidate values are redundant.

The arbitrary flat, exponential, spiked, and exact-rank spectra shown in Fig. 2 are generated from the same PCA objective as the spectral theorem. Their near-exact fit is consequently a numerical solver check, not independent hypothesis evidence.

C Standard-GPS endpoint intervention

The standard arm uses the tuned eight-layer GPS configurations for COCO-SP and PascalVOC-SP. Each layer combines four-head full attention with a local GatedGCN branch. In the one-query arm, the graph-mean real-node state is repeated as every receiver query. Keys, values, all projections, local branch, residual and feed-forward blocks, depth, optimizer, schedule, and trainable parameter count are unchanged. COCO models have 499,677 parameters and Pascal models have 500,613 in both arms.

Table 3: **Per-seed trained endpoint counterfactual.**

Dataset	seed	standard GPS	one query	paired gain
COCO	0	.38900	.35586	.03314
COCO	1	.37960	.35473	.02487
COCO	2	.39179	.35890	.03289
Pascal	0	.45353	.42590	.02763
Pascal	1	.46137	.41616	.04521
Pascal	2	.45022	.41668	.03354

Checkpoint selection maximizes validation macro-F1 and reports test F1 at the selected epoch. Table 3 exposes every paired test result.

All six COCO runs complete 200 epochs. Five Pascal runs complete epoch 199; one one-query run ends after epoch 198. I therefore use the common Pascal window 0–198 for every arm. Selected epochs are 135, 157, and 154 for standard GPS and 139, 122, and 141 for one-query GPS, so the missing final epoch cannot enter the selection. The earlier frozen prediction that Pascal would show a mean absolute effect at most .015 is rejected: its observed mean is .03546, and the COCO-minus-Pascal interaction is $-.00516 \pm .01350$. I retain this failure and make the narrower replicated endpoint claim.

D PROSPECTIVE TRAINED-VERSUS-FROZEN CAPACITY STUDY

The time-stamped protocol (listed in Appendix J) was frozen on 2026-08-10 before any intermediate-budget training result was inspected. The primary dataset is PascalVOC-SP; the query budgets are $M \in \{1, 4, 16, \text{full}\}$ and the paired seeds are 0–4. All comparisons use the same eight-layer GPS architecture, optimizer, schedule, batch size, 200 epochs, and 500,613 trainable parameters. The $M = 4$ and $M = 16$ paths use deterministic farthest-first initialization and eight Lloyd updates in every graph and layer. Cluster assignment is discrete; gradients pass through the within-cluster state averages.

Only epoch-199 weights remained from the earlier endpoint study, so the protocol declares final epoch 199 for every arm before launching new runs. Existing seeds 0–2 for $M = 1$ and full GPS are reused after a double-precision fixed-batch test found exactly zero difference between the clustered $M = 1$ and graph-mean implementations in real-node outputs, input gradients, and every parameter gradient. Seeds 3–4 extend both endpoints; all five seeds are trained at $M = 4$ and $M = 16$.

For seed s, define

$$D_{M,s} = F^{\text{trained}}_{M,s} - F^{\text{trained}}_{1,s}, \tag{10}$$

$$T_M = \frac{\text{mean}_s\, D_{M,s}}{\text{mean}_s\, D_{\text{full},s}}, \tag{11}$$

$$\Delta_{M,s} = F^{\text{trained}}_{M,s} - F^{\text{frozen}}_{M,s}. \tag{12}$$

Every score is recomputed with one full-precision scikit-learn macro-F1 evaluator on the test split. The two prospectively signed hypotheses are $\text{mean}_s\, \Delta_{4,s} > 0$ and $\text{mean}_s(\Delta_{4,s} - \Delta_{16,s}) > 0$. Intervals use 100,000 resamples of complete five-seed blocks with NumPy PCG64 seed 20260810 and 2.5th/97.5th percentiles. I report the finite seed sample rather than treating the bootstrap as a substitute for replication.

The paired Δ_4 values are .03235, .02139, .02607, .02662, and .00155; all five are positive. The corresponding Δ_{16} values are .00463, $-.01508$, .00335, $-.01141$, and $-.01360$, with mean $-.00642$. A post-hoc 95% interval for that mean is $[-.01376, .00091]$, so I do not claim that $M = 16$ training is harmful in the population. The trained $M = 16 - M = 4$ difference is .00251 with post-hoc interval $[-.00575, .01317]$; the two trained budgets are not distinguishable here.

The descriptive trained recoveries are $T_4 = .607$ [.358,.790] and $T_{16} = .678$ [.574,.785]. Thus trained capacity is monotone between the two measured intermediate budgets. This is compatible with $\Delta_4 > \Delta_{16}$ because the frozen baseline itself rises from .42314 to .45367. The Δ_M panel measures a

Table 4: **Five-seed PascalVOC-SP trained-capacity results.** Values are final-epoch test macro-F1. Frozen entries apply the budget only at inference to the paired full-query checkpoint.

Setting	seed 0	seed 1	seed 2	seed 3	seed 4	mean	SD
Trained $M = 1$	.41581	.42500	.42112	.43348	.42072	.42323	.00659
Frozen $M = 4$	.41099	.43405	.42153	.42120	.42794	.42314	.00860
Trained $M = 4$	.44334	.45544	.44760	.44782	.42950	.44474	.00957
Frozen $M = 16$	.43854	.46074	.44432	.45975	.46500	.45367	.01153
Trained $M = 16$	.44317	.44566	.44767	.44833	.45139	.44725	.00307
Trained full	.45248	.46502	.45222	.45835	.46528	.45867	.00640

Table 5: **Prospectively specified frozen-to-trained correction contrasts.** Support required the lower endpoint of the two-sided 95% seed-block bootstrap interval to exceed zero.

Contrast	mean	SD	95% interval	outcome
Δ_4	.02160	.01186	[.01136,.02891]	supported
$\Delta_4 - \Delta_{16}$	.02802	.00956	[.02069,.03535]	supported

budget-dependent *frozen-to-trained correction*; it is not a capacity curve. These continuous ratios replace the unstable threshold-crossing M_{95} statistic and are descriptive rather than preregistered success gates.

M=16 boundary audit. The lower $M = 16$ adaptation is not explained by an obvious implementation failure. Saved $M = 4$ and $M = 16$ configurations differ only in the intended budget and output path; parameter counts match; endpoint, padding, and gradient tests pass. On seed 1 validation, every layer uses all 16 clusters on every graph, with normalized occupancy entropy .853–.909. The $M = 16$ models also have lower final training loss than $M = 4$ in all first three audited seeds (.195–.199 versus .214–.216) while obtaining lower validation F1. These post-hoc diagnostics rule out codebook collapse and failure to fit, but they do not establish a mechanism. I therefore test the most direct gradient-based version of the hard-boundary explanation next.

Prospective routing-nonstationarity diagnostic. We also froze a paired trajectory diagnostic before observing any assignment history. For $M \in \{4, 16\}$ and seeds 0–2, it records the first eight fixed validation graphs at every epoch and compares adjacent-epoch partitions using adjusted Rand index (ARI) and receiver churn after optimal label matching. Its primary window is epochs 150–199. Preliminary support requires the expected direction in every seed, mean ARI contrast at least .05, mean churn contrast at least .02, and reproduction of every original final test endpoint within .002.

Table 6: **Routing trajectories have the predicted ordering but fail the endpoint-fidelity gate.** Contrasts average epochs 150–199; positive values mean less stable assignments at $M = 16$.

Seed	$\text{ARI}_4 - \text{ARI}_{16}$	$\text{churn}_{16} - \text{churn}_4$	maximum endpoint deviation
0	.0411	.0740	.0101
1	.0555	.0823	.00503
2	.0837	.1055	.0105
Mean	.0601	.0873	—

All paired directions and both mean effect-size thresholds pass. However, five of six budget–seed endpoints miss the .002 fidelity tolerance (absolute deviations .00249–.01046; the remaining deviation is .00006). The rerun configurations differ from the originals only in diagnostic logging, output path, and an unused hard-routing temperature field, so there is no identified configuration confound; the deviations are compatible with training nondeterminism. Under the frozen rule, the diagnostic therefore *fails* and I retain routing nonstationarity only as a descriptive association, not a mechanistic explanation of the $M = 16$ performance boundary. I do not add seeds 3–4 or relax the endpoint tolerance.

Table 7: **Five-seed exact-forward assignment-gradient control.** Entries are final-epoch PascalVOC-SP test macro-F1 from the uniform evaluator. S_M denotes straight-through minus hard routing within each paired seed.

Setting	seed 0	seed 1	seed 2	seed 3	seed 4	mean	SD
Hard $M = 4$	.44334	.45544	.44760	.44782	.42950	.44474	.00957
ST $M = 4$	.44150	.44193	.44691	.44600	.43657	.44258	.00413
S_4	$-.00183$	$-.01351$	$-.00068$	$-.00182$	.00707	$-.00216$	.00735
Hard $M = 16$	.44317	.44566	.44767	.44833	.45139	.44725	.00307
ST $M = 16$	.44545	.44296	.44590	.43611	.43710	.44150	.00462
S_{16}	.00228	$-.00270$	$-.00177$	$-.01222$	$-.01430$	$-.00574$	.00715

D.1 Exact-forward straight-through routing control

This control was frozen on 2026-08-11 after the hard-routing result was known but before inspecting any straight-through outcome. It preserves farthest-first initialization, eight hard Lloyd updates, cluster means, the hard forward assignment, optimizer, schedule, parameter count, and the five paired seeds. Only the backward rule for the final receiver-to-prototype assignment changes. If H is the hard one-hot assignment and P is a softmax over negative squared projected-query distances, the backward surrogate is

$$\widetilde{H} = H + P - \mathrm{stopgrad}(P).$$

Distances are divided by detached per-graph projected-query variance and the fixed temperature is one. Cluster-center estimation remains hard, so this is a targeted test of missing gradients through the final assignment, not a fully soft clustering method or a learned-prototype baseline. Unit tests verify identical hard and straight-through forward outputs to numerical precision, unchanged parameters, finite gradients, padding behavior, and exact $M = 1$ and full-query endpoints.

The first one-epoch implementation smoke test normalized by the converged hard residual and produced non-finite gradients because that residual can approach zero. Before any valid task outcome or full run was inspected, the protocol was amended to the projected-query variance normalization above. The temperature was never tuned.

The preregistered support rule for the artifact explanation required all five S_{16} values to be positive, a two-sided paired-bootstrap interval excluding zero, and mean $S_{16} \geq .005$. None holds: its 95% interval is [$-.01138$,$-.00033$]. At $M = 4$, the interval is [$-.00861$,.00351]. The specificity contrast $S_{16} - S_4$ is $-.00359$ with interval [$-.01408$,.00575]. Thus adding this assignment-gradient path is significantly detrimental at $M = 16$, not merely unable to rescue it, and weakens the simplest hard-boundary account. Because Lloyd center assignments remain discrete, the experiment does not rule out a fully differentiable router, learned prototypes, or other optimization changes.

E Weight-frozen query-compression intervention

The capacity sweep loads the three complete epoch-199 standard-GPS checkpoints for each dataset. No model weight is optimized. For every graph and attention layer, it performs the following operations:

1. exclude padded nodes and project the real receiver queries;
2. choose up to M representatives by deterministic farthest-first initialization;
3. apply eight Lloyd assignment/update iterations in projected-query Euclidean distance;
4. average the pre-projection receiver states within each cluster, apply the unchanged query projection, and evaluate attention for the representative;
5. broadcast that attention output to the receivers assigned to the mode.

For $M = 1$, the pre-projection average is exactly the graph-mean query. If M is at least the number of real nodes, every receiver retains its original query, exactly recovering standard GPS. Unit tests verify

Table 8: **Frozen COCO-to-Pascal exploratory prediction.** Pascal task labels do not enter the predictor. V_M is retained query variation and C_M is query-compression robustness.

M	COCO F1	COCO V_M	Pascal V_M	predicted C_M	observed C_M
1	.16268	.00000	.00000	.00000	.00000
2	.25338	.26788	.28677	.43266	.59934
4	.32632	.46183	.48988	.75622	.83525
8	.36181	.59962	.63157	.90140	.93307
16	.37713	.70325	.73592	.95663	.97604
32	.38213	.78567	.81572	.97808	.99477
64	.38614	.85524	.87980	.99025	.99676
full	.38880	1.00000	1.00000	1.00000	1.00000

both endpoints, determinism, padding exclusion, parameter preservation, and agreement between the repeated and prototype executors.

Let SSE_M be pooled projected-query clustering error and SSE_1 the error around the graph mean. Retained query variation is

$$V_M = 1 - \frac{\mathrm{SSE}_M}{\mathrm{SSE}_1}, \tag{13}$$

pooled over real receivers, graphs, layers, and seeds. Hence $V_1 = 0$ and $V_{\mathrm{full}} = 1$. This statistic uses no task labels.

For the historical normalized plots, define query-compression robustness

$$C_M = \frac{F_M - F_1}{F_{\mathrm{full}} - F_1}. \tag{14}$$

Both endpoints here come from the same full-query-trained checkpoint under an inference-time intervention. In particular, F_1 is not the performance of the independently trained one-query arm and C_M is not trained-capacity recovery.

F Exploratory cross-dataset predictor audit

The protocol in `experiments/GPS_QUERY_CAPACITY_PROTOCOL.md` was written after observing the trained $M = 1$ and full-attention endpoints but before evaluating any intermediate query budget. Its order was:

1. measure all intermediate budgets on COCO validation;
2. fit a monotone piecewise-linear map $V_M \mapsto C_M$ on COCO validation;
3. freeze the map, preprocessing, budgets, and numerical gates;
4. evaluate intermediate Pascal validation budgets once;
5. freeze $M = 16$ and only then evaluate the two test curves.

The six-point MAE is 0.0533, below the originally frozen 0.150 MAE gate, but that gate was permissive. Signed bias is -0.0533, RMSE is 0.0772, maximum absolute residual is 0.167 at $M = 2$, and integrated absolute error over $\log_2 M \in [1, 6]$ is 0.0467. Every mean prediction is below the observed Pascal curve. Restricting the audit post hoc to $M \geq 8$ gives MAE 0.0186 and maximum error 0.0317. These diagnostics support only an exploratory, high-budget approximation claim; they do not establish a predictive law.

Post-hoc same-budget baseline. After the prospective result, I added the strongest simple budget-only audit: use COCO's observed robustness at each M directly as the Pascal prediction. It receives the full COCO curve but no Pascal query-variation information.

The query-variation map lowers every aggregate error relative to the post-hoc same-budget baseline, but both predictors systematically underestimate Pascal. With one arbitrary map, one calibration dataset, and one closely related target dataset, this comparison cannot establish general predictive superiority.

Table 9: **Post-hoc predictor audit on frozen outputs.** Bias is prediction minus observation; integrated error is averaged over $\log_2 M$.

Predictor	MAE	bias	RMSE	max	integrated
Retained query variation	.0533	$-.0533$	.0772	.167	.0467
COCO robustness at same M	.0704	$-.0704$	.0964	.198	.0639

Table 10: **Complete frozen test curves.** Means over three fixed checkpoints; C_M is normalized only within each checkpoint's weight-frozen inference-time compression range.

	COCO test		Pascal test	
M	F1	C_M	F1	C_M
1	.17450	.00000	.22101	.00000
2	.25575	.38499	.35204	.55631
4	.32662	.72068	.42219	.85404
8	.36160	.88655	.44014	.93019
16	.37745	.96165	.44787	.96303
32	.38134	.98024	.45133	.97771
64	.38446	.99524	.45435	.99057
full	.38542	1.00000	.45657	1.00000

G Frozen test confirmation

Test labels were evaluated only after the predictor and selected $M = 16$ budget were frozen. Table 10 reports the complete curves.

At $M = 16$, raw test macro-F1 is 0.377 on COCO and 0.448 on Pascal, with absolute gaps of 0.00797 and 0.00871 from uncompressed inference. The normalized values in the table are retained for auditability, not interpreted against the much smaller independently trained endpoint gap.

H Implementation and efficiency falsification

The screening implementation repeats each representative query at its assigned receiver positions, so it changes function without claiming computational savings. A second executor evaluates attention once per prototype and broadcasts the result. With eight Lloyd updates, the two executors agree within 4.3×10^{-6} validation F1.

On Pascal validation, seed 0, batch size 50, and one Lloyd update, full attention obtains F1 .449589 in 5.100 seconds with 503.8 MB peak allocated memory. The $M = 16$ prototype executor obtains .440156 in 5.924 seconds with 382.2 MB. Peak allocation falls by 24.1%, but runtime increases by 16.2% relative to the optimized dense kernel. The practical runtime gate fails, so I stop this branch and make no speed or deployment claim.

I Prospective graph-level scope screen

We prospectively screened Peptides-func, an official LRGB graph-level multilabel benchmark, before adding another capacity curve. This task uses a graph-mean prediction head and is therefore less directly aligned with our multi-receiver hypothesis than nodewise segmentation. The screen uses the official tuned six-layer GPS recipe, 479,234 parameters, 250 epochs, average precision (AP), and paired seeds 0–2. Its only architectural intervention is the same graph-mean receiver query used by the endpoint study. The frozen continuation gate inspected final-epoch validation AP only: full-minus-one-query had to be positive in all three seeds and have mean at least .010 AP.

The gate fails both its sign and magnitude requirements, and the prespecified $M = 4$ continuation is stopped. The apparent .00908 one-query advantage at the final epoch is not seedwise robust under validation selection: best-validation full-minus-one gaps are $+.00245$, $-.00238$, and $-.01585$. Test AP at those selected checkpoints favors one query in two seeds but full queries in the third, with

Table 11: **Peptides-func validation-only continuation gate.** The full-query advantage is absent in every paired seed, so the planned capacity continuation is stopped.

Seed	full queries	one query	full minus one
0	.65528	.65598	$-.00070$
1	.63581	.64716	$-.01135$
2	.64734	.66254	$-.01520$
Mean	.64614	.65523	$-.00908$

paired one-minus-full differences $+.0146$, $+.0123$, and $-.0342$. I therefore do not interpret the final-epoch mean as a demonstrated regularization benefit.

After observing this null, I froze a label-free post-hoc mechanism audit. It asked whether Peptides has an unusually fast retained-query-variation curve, requiring mean $V_4 > .65$ and $V_{16} > .85$ before inspecting the values. The result rejects that explanation.

Table 12: **Post-hoc Peptides query-variation audit.** Peptides does not saturate unusually early relative to the two superpixel tasks. AP is from frozen compression of the full-query Peptides checkpoints.

M	COCO V_M	Pascal V_M	Peptides V_M	Peptides frozen AP
1	.000	.000	.000	.574
4	.462	.490	.460	.622
8	.600	.632	.660	.637
16	.703	.736	.812	.642
32	.786	.816	.929	.644
full	1.000	1.000	1.000	.645

In particular, Peptides $V_4 = .460$ is indistinguishable at this resolution from COCO's .462, and $V_{16} = .812$ misses the fixed .85 gate. Graph-mean readout does not imply that intermediate node representations have only one effective receiver. This audit therefore does not make V_M a transfer predictor. Its AP column is a validation-only diagnostic and is not mixed with the trained endpoint metric below.

Matched test estimand. After the prospective gate and V_M explanation had failed, I froze a separate post-hoc analysis before running the uniform test evaluator. For each seed it loads the final epoch-249 full checkpoint twice—once unmodified and once with $M = 1$ imposed only at inference—and the final trained-$M = 1$ checkpoint once. All predictions use the official test split and the same concatenated multilabel AP implementation.

Table 13: **Peptides matched-estimand audit on the test split.** Δ_1 is trained $M = 1$ minus frozen $M = 1$. The large frozen deficit vanishes after training, but three post-hoc seeds do not establish equivalence between the two trained endpoints.

Setting	seed 0	seed 1	seed 2	mean
Frozen $M = 1$	.60093	.53399	.58209	.57234
Trained $M = 1$	.67675	.65906	.63641	.65741
Δ_1	+.07582	+.12507	+.05432	+.08507
Trained full	.65313	.64673	.66271	.65419
Full minus trained $M = 1$	$-.02362$	$-.01233$	+.02630	$-.00322$

All three Δ_1 values are positive and their descriptive seed-block bootstrap interval is [.0543,.1251]. The frozen comparison says that one query costs .0819 AP relative to the same full checkpoint; the trained comparison is $-.00322$ AP on average and changes sign across seeds. This post-hoc result motivates, but is not substituted for, the prospective intermediate-budget test below.

Prospective intermediate-budget estimand test. Before training at $M = 4$, I froze a distinct protocol that explicitly did not reopen the failed capacity continuation. It instead tests whether the frozen-versus-trained estimand gap persists under a less severe intervention. The recipe, seeds 0–2, final epoch-249 checkpoint, test evaluator, and support rule were fixed in advance. Support requires every paired correction to be positive and their mean to be at least .015 AP.

Table 14: **Prospective Peptides $M = 4$ estimand test.** Every paired correction is positive and the mean exceeds the prespecified .015-AP threshold. Full and $M = 4$ trained endpoints are not consistently ordered across seeds.

Setting	seed 0	seed 1	seed 2	mean
Frozen $M = 4$	.63828	.62147	.62225	.62733
Trained $M = 4$	.67058	.66553	.64958	.66190
Δ_4	+.03231	+.04406	+.02733	+.03457
Trained full	.65313	.64673	.66271	.65419
Full minus trained $M = 4$	$-.01745$	$-.01879$	+.01312	$-.00771$

The seed-block interval for Δ_4 is [.0273,.0441], so the prospective support rule passes. Moreover, $|\Delta_4| = .0346 < |\Delta_1| = .0851$, as the intervention-intensity account predicts. Magnitude ordering is therefore not our novelty claim. The non-trivial result is replication at an intermediate budget on an unrelated task, alongside the opposite-signed Pascal $M = 16$ correction: frozen sensitivity does not determine the sign or scale of the adaptive correction. Because trained full-minus-$M = 4$ changes sign, this test does *not* rescue the failed capacity-axis prediction, establish trained endpoint equivalence, or validate V_M across tasks. The original endpoint screen remains an honest scope boundary: the receiver-specific effect on two nodewise segmentation tasks does not automatically transfer to graph-mean prediction.

We retain the preregistered three-seed screen rather than silently extending it after failure. Two additional post-hoc seeds could refine the mean but would not turn a failed superiority screen into an equivalence test; establishing practical equivalence would require a new margin and a prospectively powered design.

J Reproducibility and artifact map

Endpoint and historical frozen-curve means report sample standard deviations across three seeds. The trained-capacity study reports five paired seeds and seed-block bootstrap intervals exactly as frozen in its protocol.

K Scope boundaries

The central conclusion is deliberately narrow. The theorem uses explicit finite-information messages; the spectral result is linear and squared-loss; neither converts neural width directly into bits or covariance rank. Controlled retrieval graphs identify a mechanism but are not natural benchmark evidence. The endpoint evidence covers one GPS recipe, two related superpixel datasets, and three seeds; the prospective intermediate-budget evidence covers only PascalVOC-SP with five seeds. The exact-forward straight-through control changes only the final assignment gradient and is not a fully soft router. The Peptides-func screen is an unrelated public task, but its endpoint gate fails and therefore supplies a boundary rather than a replicated capacity curve. Its post-hoc V_M audit rejects low query variation as the explanation. Its separate prospective $M = 4$ test replicates the frozen-versus-adaptive estimand gap, not the capacity effect; it uses three paired seeds and does not establish equivalence between trained endpoints. COCO-to-Pascal transfer is an exploratory within-family audit, not universal cross-domain prediction. The same-budget predictor comparison and the original $M = 16$ mechanism diagnostics are post-hoc. The one-query model is a causal ablation, not a new leaderboard method, and the $M = 16$ executor does not improve runtime. I establish neither a universal capacity law nor a new optimized routing method.