ATTRIBUTION VERGENCE: A MECHANISTIC INDICATOR ACROSS GROKKING AND ADVERSARIAL TRANSFER

Anonymous authors
Paper under double-blind review

ABSTRACT

Connecting internal computational mechanisms to macroscopic model behaviors, such as grokking and adversarial transfer, is one of the central goals of mechanistic interpretation. However, while existing methods analyze isolated components or global representations, whether the structure of attributed contributions supporting model decisions admits a general macroscopic observable remains fundamentally open. We introduce **Attribution Vergence**, a measure that quantifies this structural reorganization across feature aggregation by measuring the change in Rényi-2 effective dimensionality. Under a unified *Aggregation Node* formulation encompassing attention and convolution, we ground the measurement via attribution preservation, axiomatic attribution compatibility conditions, an information-theoretic interpretation, and analysis on invariance, bounds, stability and computation. Experiments on grokking and generative adversarial transfer show stronger behavioral responses than prior analysis tools and additional gains from the formulation. The profiles reveal perturbation-dependent response locations, source indicators shared across transfer targets, and attribution reorganization across layers after training accuracy saturates. Together, these results establish a principled measurement linking attribution organization to macroscopic behaviors, as a step toward a scientific understanding of AI models.

1 INTRODUCTION

Deep neural networks are empirically powerful but mechanistically opaque, leaving scaling behavior, safety, and trustworthiness largely characterized by empirical modeling rather than structural accounts of how decisions are formed. Traditional approaches to post-hoc explanation focus on accounting for outputs that a model has already produced (Mittelstadt et al., 2019; Ribeiro et al., 2016; Lundberg & Lee, 2017). However, a scientific understanding of models also requires studying how internal mechanisms relate to patterns of behavior across architectures and phenomena, especially for behaviors that are not directly observed in ordinary performance benchmark outcomes: in grokking, a model can memorize the training set long before validation performance abruptly improves (Power et al., 2021); in adversarial transfer, perturbations optimized on a source model can retain decision effects on target models (Papernot et al., 2017).

Two common approaches seek to explain such phenomena. One derives theoretical accounts from simplified models, such as linear explanations of adversarial vulnerability (Goodfellow et al., 2015) and solvable linear models of grokking (Levi et al., 2024). However, exact results for linear models or idealized fully connected networks do not by themselves account for the full range of behavior in modern architectures. Modern neural networks are complex systems whose structural interactions may give rise to collective patterns absent from those simplified models. Another approach characterizes internal structures associated with model behavior. Mechanistic interpretability methods provide detailed accounts of computational components and their interactions, as in Fourier analysis of the circuits learned for modular addition (Nanda et al., 2023). These analyses often depend on task-specific structures, making it difficult to derive common macroscopic observables across tasks and architectures. In parallel, representation statistics such as local intrinsic dimensionality (LID) characterize the across-sample representation geometry (Ma et al., 2018; Ansuini et al., 2019; Ruppik et al., 2025), and training attribution relates model predictions to parameter, data, and

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.

training-step influences (Eichin et al., 2026). These advances motivate a new question: whether the *structure of attributed contributions supporting model decisions* exhibits regularities associated with different macroscopic phenomena, *e.g.*, grokking and adversarial transfer. Can this structure provide the basis for a common observable, and how can we construct such a measurement?

In classical algorithm analysis and network science, measures such as graph conductance and modularity provide common structural descriptors across networks (Jerrum & Sinclair, 1988; Newman, 2006). However, the behavior of deep neural models is closely tied to the tasks they perform. Constructing a common structural observable therefore requires a principled abstraction that preserves the decision-relevant meaning of internal contributions across tasks and architectures. The resulting measure should remain well-defined and consistently interpretable across these settings, while admitting desirable mathematical properties and ensuring computational feasibility.

We introduce **Attribution Vergence**, a framework that quantifies the effective *vergence* of attributed contributions under feature aggregation. Our *Aggregation Node Formulation* aligns native computations across attention and convolution, with the *attribution preservation* property grounding the measurement to a well-formed attribution problem. We establish an axiomatic foundation through four *Attribution Compatibility Conditions*, covering integrated gradient-based and Owen-Shapley axiomatic attribution, as a formal bridge connecting attribution rules with structural measurement, and show the information-theoretic connection, invariance, stability, and computational tractability[1].

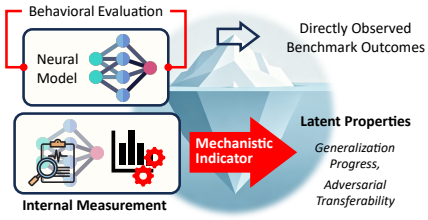

Figure 1: Internal attribution organization provides indicators of grokking progress and adversarial transfer.

To test the indicative ability illustrated in Figure 1, we evaluate the framework on four algorithmic grokking tasks and six generative adversarial transfer directions using held-out seeds and image subsets. The Attribution Vergence profile achieves larger mean responses than prior analysis tools operating on model representations, inputs and parameters. The response gains of the proposed formulation are also validated. The profiles reveal the source-model adversarial transfer indicators shared across targets, and, in grokking, memorization-associated attribution reorganization across layers after training accuracy saturates. These findings connect behavioral variation to where attributed contributions converge and how this organization evolves. Our main contributions are:

- **Measurement Framework.** We introduce Attribution Vergence as a measure of the organization of attributed contributions under aggregation. The *Aggregation Node Formulation* applies this measurement to modules across attention and convolution, yielding structural profiles under a common principle for studying relations between internal organization and model behavior.
- **Theoretical Foundations and Axiomatic Grounding.** Building on the *Aggregation Node Formulation*, we ground Attribution Vergence through attribution preservation and axiomatic *Attribution Compatibility Conditions*. We establish an information-theoretic entropy interpretation, signed extremal bounds, coordinate invariance and stability guarantees, and execution schedules with formal time and memory bounds relative to a training step.
- **Empirical indicators and structural findings.** As a novel macroscopic observable of internal aggregation, the Attribution Vergence profile yields stronger mean behavioral responses than prior analysis tools across grokking and adversarial transfer. Its structural profiles uncover perturbation-dependent attribution organization, source-layer transfer indicators shared across targets, and layerwise attribution reorganization during grokking.

2 Preliminaries

Feature attribution. Given a neural network and a scalar objective $F(x)$ of its final output, feature attribution assigns contributions to input features to account for an observed output or its change relative to a reference. *Shapley attribution* treats features as players in a cooperative game, with a value function $v(U)$ specifying the model score for each feature coalition U. For a fixed value function, Efficiency, Symmetry, Dummy, and Linearity uniquely determine the Shapley allocation (Shapley,

[1]The implementation of the corresponding algorithm is in the supplementary material.

1953). The Owen value extends this allocation to games with prescribed player groups (Owen, 1977), with hierarchical extensions applied to image feature attribution (Rashid et al., 2026). *Integrated Gradients* (IG), motivated by the continuous extension of Shapley Value, assigns contributions to input features by integrating their gradients along the straight path from a baseline to the observed input (Sundararajan et al., 2017). The objective, reference, and intervention or path specify the attribution. Appendix B gives a more detailed mathematical background.

Internal attribution and mechanistic interpretation. Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ denote the network's computation graph, whose nodes are operations and whose edges carry activation tensors. Internal attribution assigns contributions to selected activation coordinates with respect to the scalar output F. Neuron Conductance extends IG to internal activations (Dhamdhere et al., 2019), producing a signed internal attribution field $\mathbf{S}$ for each internal representation. Signed contributions indicate support for or opposition to that change under the specified attribution protocol. AttnLRP attributes transformer outputs to input and latent representations (Achtibat et al., 2024), while EAP-IG scores computational edges to identify circuits that preserve task behavior (Hanna et al., 2024). *Mechanistic interpretability* connects internal components to the computations that produce model behavior. Circuit Tracing (Ameisen et al., 2025) constructs attribution graphs over learned interpretable features in a local replacement model and tests their interactions through interventions. Building on attribution and circuit analyses and extending beyond interpretation, we seek a common measurement of how attributed contributions are organized across computations.

Behavioral phenomena and interpretations. *Grokking* exhibits delayed generalization after memorization (Power et al., 2021). To explore its mechanism, researchers track Fourier circuit formation in modular addition (Nanda et al., 2023), compare the efficiency of memorizing and generalizing circuits (Varma et al., 2023), and relate representation-space mean local intrinsic dimension and parameter-space regularization influences to learning progress (Ruppik et al., 2025; Eichin et al., 2026). *Adversarial transfer* carries perturbation effects from a source model to target models (Papernot et al., 2017). Prior work improves transfer by suppressing interactions among input perturbation units (Wang et al., 2021c) and amplifying intermediate representation displacements (Huang et al., 2019). Extended background is provided in Appendix B. These phenomena motivate us to construct measurements of attribution organization as a model learns or its inputs are perturbed, connecting internal contributions to generalization and transfer behavior.

3 The Attribution Vergence Measurement Framework

Feature aggregation is a recurring operation in neural computation. Attention combines contributions from source tokens, convolution combines local input contributions, and affine projections combine activation coordinates. We quantify the effective *vergence* of attributed contributions under these operations. Our framework combines a common computational formulation with an axiomatic foundation for attribution methods, providing a basis for structural measurement profiles. Appendix A summarizes the notation.

3.1 Formulation

A common measurement across modules requires a consistent attribution interpretation of the contributions being compared. Shapley Flow and Recursive Shapley develop attribution methods for assigning and propagating credit on graphs (Wang et al., 2021b; Singal et al., 2021). By contrast, our formulation focuses on specifying a common attribution problem for native aggregation operations across modules.

Aggregation Operator and Attribution Semantics. We formalize this correspondence at the level of the computation graph $\mathcal{G}$, and model a native aggregation process in $\mathcal{G}$ as an *aggregation node* a, which is a multi-input, single-output computation node. For each neural module, an architecture-specific coordinate mapping instantiates the *Aggregation Node Formulation* and its corresponding attribution-space *Aggregation Operator* $\mathbb{A}$ acting on a route-indexed pre-aggregation attribution field $\mathbf{S}$. This formulation establishes a structure of attribution problem that supports attribution preservation across aggregation: for Neuron Conductance, $\mathbb{A}[\mathbf{S}]$ recovers output Conductance, while grouped Owen attribution recovers output Shapley values, as established in Theorem 3.1.

Definition 3.1: Aggregation Node Formulation

Let $a \in \mathcal{V}$ be an aggregation node that combines route-indexed feature inputs into a single output tensor $\mathbf{O} \in \mathbb{R}^d$ according to their associated routing weights $w_{y,z}$. Let $\mathcal{Y} = \{1, \ldots, d\}$ index the unreduced output coordinates y, and let $\mathcal{Z} = \{1, \ldots, K\}$ index the reduced routing coordinates z. Each (y, z) defines an input edge $e_{y,z} := (e^w_{y,z}, e^X_{y,z})$, whose constituent incoming graph edges carry $w_{y,z}(\xi)$ and $\mathbf{X}_{y,z}(\xi)$, respectively, for network input ξ. Let $\mathbf{S}_{y,z}$ be an internal attribution of each edge $e_{y,z}$, forming an attribution field $\mathbf{S} \in \mathbb{R}^{d \times K}$. The node computation and corresponding attribution-space Aggregation Operator $\mathbb{A}[\mathbf{S}]$ are defined by:

$$\mathbf{O}_y(\xi) = \sum_{z=1}^{K} w_{y,z}(\xi)\mathbf{X}_{y,z}(\xi), \quad \mathbf{S}_{y,z} \longleftrightarrow e_{y,z}, \quad \mathbb{A}[\mathbf{S}] = \sum_{z=1}^{K} \mathbf{S}_{:,z} \in \mathbb{R}^d. \tag{1}$$

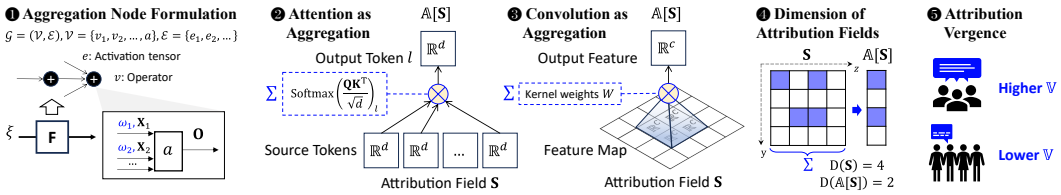

Figure 2: Aggregation, as a shared form of information processing across modules, reorganizes attributed contributions from $\mathbf{S}$ into $\mathbb{A}[\mathbf{S}]$. Our measurement is based on the aggregation node formulation a on the computation graph $\mathcal{G}$, measuring the change in effective dimensionality D.

Theorem 3.1 (Attribution Preservation under Aggregation (Informal))**.** *Let F be a scalar objective of the network's final output, and let $\mathbf{S}$ assign attribution to the input edges of selected aggregation operations. Under matched attribution contexts (attribution path γ or output-coordinate game $\bar{v}$), aggregation preserves the corresponding output attribution:*

$$\mathbb{A}[\mathbf{S}]_y = \begin{cases} \mathrm{Cond}_y(F; \gamma), & \mathbf{S} \textit{ is the input-edge Neuron Conductance field}, \\ \phi_y(\bar{v}), & \mathbf{S} \textit{ sums Owen credits grouped by output}. \end{cases} \tag{2}$$

Formal conditions and proofs are given in Corollary C.5 and Theorem C.2.

Measuring the Change of Effective Dimensionality. We characterize attribution organization through effective dimensionality, which accounts for how contributions are distributed across coordinates. To measure the effective dimensionality of attribution $\mathbf{S}$ (distribution shown in Figure 3), we adapt the inverse collision probability $1/\sum_i p(i)^2$ of normalized absolute attribution (Definition 3.2). Related constructions appear in effective party counts (Laakso & Taagepera, 1979) and sparsity measures (Hoyer, 2004). Entropy statistics and sparsity have also been used to summarize attribution distributions when evaluating explanation quality (Bhatt et al., 2020; Chalasani et al., 2020).

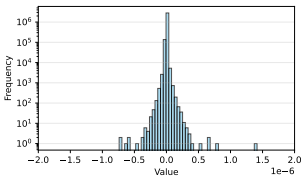

Figure 3: Distribution of $\mathbf{S}$.

Definition 3.2 (Effective Dimensionality)**.** For a nonzero vector $\boldsymbol{v} = (\boldsymbol{v}_i)_{i=1}^d \in \mathbb{R}^d$, define its absolute-mass distribution by $p_{|\boldsymbol{v}|}(i) = |\boldsymbol{v}_i|/\|\boldsymbol{v}\|_1$. Its effective dimensionality is the exponential of the order-2 Rényi entropy $H_2(p_{|\boldsymbol{v}|})$ (Rényi, 1961):

$$\mathrm{D}(\boldsymbol{v}) = 2^{H_2(p_{|\boldsymbol{v}|})} = \frac{1}{\sum_{i=1}^d p_{|\boldsymbol{v}|}(i)^2} = \frac{\|\boldsymbol{v}\|_1^2}{\|\boldsymbol{v}\|_2^2}. \tag{3}$$

If k coordinates have equal nonzero magnitudes, as an example in Figure 2, $\mathrm{D}(\boldsymbol{v}) = k$, more mass-concentrated attribution yields a smaller effective count. Its expression through first and second moments also supports the analysis and computation developed in the next subsection. To measure how a neural aggregation node reorganizes the attribution, we introduce **Attribution Vergence** as the logarithmic change in effective dimensionality from $\mathbf{S}$ to $\mathbb{A}[\mathbf{S}]$, quantifying the effective vergence of attributed contributions.

Definition 3.3: Attribution Vergence

For a non-degenerate attribution field $\mathbf{S}$ and aggregation operator $\mathbb{A}$, flattening over (y, z) gives $\mathrm{D}(\mathbf{S}) := (\sum_z \|\mathbf{S}_{:,z}\|_1)^2 / \sum_z \|\mathbf{S}_{:,z}\|_2^2$, and **Attribution Vergence** is defined as the signed logarithmic effective-dimensionality ratio:

$$\mathbb{V}_{\mathbb{A}}(\mathbf{S}) := \log_2 \mathrm{D}(\mathbf{S}) - \log_2 \mathrm{D}(\mathbb{A}[\mathbf{S}]). \tag{4}$$

Unlike a *divergence*, which is conventionally non-negative, $\mathbb{V}_{\mathbb{A}}$ can be positive, zero, or negative, respectively corresponding to a decrease, preservation, or increase in effective dimensionality under aggregation. Non-negative $\mathbf{S}$ satisfy $\mathbb{V}_{\mathbb{A}}(\mathbf{S}) \geq 0$, whereas signed fields admit all three cases (See Theorem 3.3 in the next subsection). We therefore use the neutral term *Vergence* to denote this signed dimensional change without implying a non-negative *divergence*.

Architecture Mappings and Attribution Compatibility. As shown in Figure 2, attention, convolution, and linear projections instantiate this formulation through their native coordinate mappings. Each attention output token and head, convolution output position, or linear projection at a token defines an aggregation node. These mappings identify the input and output edges on which attribution is evaluated (Appendix C.1.6). We further introduce **Attribution Compatibility Conditions** (Definition C.6 in Appendix) to formalize the requirements on the paired attribution rules. Motivated by the *Owen axioms*, we retain Linearity and Null response, restrict Symmetry to matched attribution problems, and replace global Efficiency with local Aggregation consistency. Together, these conditions formalize the preservation established above and connect transformations of the attribution problem to the measurement invariances analyzed in the next subsection.

3.2 Analysis of Attribution Vergence

Building on the aggregation formulation, we first establish what Attribution Vergence measures and how signed contributions affect its range. We then characterize the transformations that preserve its value, its sensitivity to attribution error, and the computational cost of evaluating profiles across multiple aggregation nodes, with the corresponding algorithmic implementation induced. The logical roadmap of the analysis is shown in Figure C.1 in Appendix C.

Coordinate-Positioning Entropy Interpretation and Signed Bounds Since $\mathrm{D}(\cdot)$ is defined by Rényi effective dimensionality, $\log_2 \mathrm{D}(\mathbf{S}) = H_2(p)$ naturally holds the information-theoretic interpretation: by definition, it measures the *uncertainty in selecting an activation coordinate based on the attribution density that defines p on the joint coordinate space $\mathcal{Y} \times \mathcal{Z}$*. Intuitively, larger values indicate greater routing-source heterogeneity. For a component-wise non-negative $\mathbf{S}$, the following theorem shows that $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ exactly quantifies the Coordinate-Positioning Entropy reduction induced by reducing the route coordinate Z. The proof is in Appendix C.5 by showing $\mathbb{A}$ is marginalization over Z in this case.

Theorem 3.2 (Coordinate-Positioning Entropy Interpretation)**.** *Suppose $\mathbf{S} \neq 0$ and every pre-aggregation attribution entry is component-wise non-negative: $\mathbf{S}_{y,z} \geq 0$ for all (y, z). Then*

$$p(Y = y, Z = z) = \frac{\mathbf{S}_{y,z}}{\sum_{z',y'} \mathbf{S}_{y',z'}}, \quad \mathbb{V}_{\mathbb{A}}(\mathbf{S}) = H_2(Y, Z) - H_2(Y). \tag{5}$$

$H_2(Y, Z) - H_2(Y)$ is a form of conditional Rényi entropy (Fehr & Berens, 2014). In the signed case, opposite-sign contributions can cancel under $\mathbb{A}$, so the absolute-mass distribution induced by $\mathbb{A}[\mathbf{S}]$ is generally not the Y-marginal of that induced by $\mathbf{S}$. In this case, $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ measures the difference in effective dimensionality before and after aggregation, accounting for both attribution distribution and cancellation. Whether $\mathbf{S}$ is component-wise non-negative also affects the range of $\mathbb{V}_{\mathbb{A}}$; nevertheless, the quantity remains bounded, as stated in the following theorem. The proof and endpoint cases are in Appendix C.6. In the signed case, the edge cases hold when the dominant or non-dominant items in $\mathbf{S}$ during the summation in $\mathbb{A}$ are canceled. Note that $\mathbb{V}_{\mathbb{A}}(\mathbf{S}) = 0$ if $K = 1$.

Theorem 3.3 (Lower and Upper Bounds)**.** *For a signed non-degenerate field $\mathbf{S}$ with $K > 1$ and $d \geq 1$, the bound of $\mathbb{V}_{\mathbb{A}}$ and its component-wise non-negative refinement are:*

$$\min\{0, 1 - \log_2 d\} \leq \mathbb{V}_{\mathbb{A}}(\mathbf{S}) \leq \log_2(Kd); \qquad \mathbf{S}_{y,z} \geq 0 \implies 0 \leq \mathbb{V}_{\mathbb{A}}(\mathbf{S}) \leq \log_2 K. \tag{6}$$

Measurement Invariances and Stability A structural measurement should not change when attribution units change or inactive coordinates are padded. The following result specifies these invariances under a specified coordinate mapping.

Theorem 3.4 (Representation Invariances under a Fixed Coordinate Mapping)**.** *Attribution Vergence* $\mathbb{V}_{\mathbb{A}}$ *satisfies* ***(i) attribution-scale invariance:*** $\mathbb{V}_{\mathbb{A}}(c\mathbf{S}) = \mathbb{V}_{\mathbb{A}}(\mathbf{S})$ *for* $c \neq 0$*;* ***(ii) padding immunity:*** *Axes of* $\mathbf{S}$ *with* $\mathbf{S}_{:,z} = 0$ *or* $\mathbf{S}_{y,:} = 0$ *can be added or removed with* $\mathbb{V}_{\mathbb{A}}$ *unchanged; and* ***(iii) duplication invariance:*** *for* $r \in \mathbb{N}_+$ *and* $a \neq 0$*, let* C_r^a *duplicate every unreduced coordinate* r *times with common scale* a*. If* $\mathbf{S}' = C_r^a \mathbf{S}$ *and* $\mathbb{A}'[\mathbf{S}'] = C_r^a \mathbb{A}[\mathbf{S}]$*, then* $\mathbb{V}_{\mathbb{A}'}(\mathbf{S}') = \mathbb{V}_{\mathbb{A}}(\mathbf{S})$.

The proof is given in Appendix C.2. Both $\log_2 \mathrm{D}(\mathbf{S})$ and $\log_2 \mathrm{D}(\mathbb{A}[\mathbf{S}])$ satisfy attribution-scale invariance and padding immunity. Under the duplication in (iii), however, each increases by $\log_2 r$. Applying this to the attribution framework, under the *Attribution Compatibility Conditions* (Definition C.6), $\mathbb{V}_{\mathbb{A}}$ is invariant under **(1)** rescaling of the value function, **(2)** extension of model activations with inactive routes or output coordinates and **(3)** matched replication of activation coordinates at the aggregation node. On the other hand, the estimation of the attribution field $\mathbf{S}$ may induce error. The following result further gives a conditional stability bound.

Theorem 3.5 (Perturbation Stability)**.** *For a perturbation field* ϵ*, flatten the pre-aggregation fields over* (y, z) *and let* η *denote the largest relative perturbation in the* L_1 *and* L_2 *norms. If* $\eta < 1$*, then*

$$|\mathbb{V}_{\mathbb{A}}(\mathbf{S} + \epsilon) - \mathbb{V}_{\mathbb{A}}(\mathbf{S})| \leq 4 \log_2 \frac{1+\eta}{1-\eta}, \quad \eta = \max_{p \in \{1,2\}} \left\{ \frac{\|\epsilon\|_p}{\|\mathbf{S}\|_p}, \frac{\|\mathbb{A}[\epsilon]\|_p}{\|\mathbb{A}[\mathbf{S}]\|_p} \right\}. \quad (\mathbf{S}, \mathbb{A}[\mathbf{S}] \neq 0) \tag{7}$$

Attribution Instantiation and Computation. We instantiate $\mathbf{S}$ by *decomposing* the Neuron Conductance of each aggregation output $\mathbf{O}_y$ across the input edges of the corresponding aggregation node a. Along the chosen attribution path γ, $\mathbf{S}_{y,z} = \int_\gamma \partial F / \partial \mathbf{O}_y \cdot \mathrm{d}(w_{y,z} \mathbf{X}_{y,z})$. Paired with the corresponding output Conductance, this construction satisfies the Attribution Compatibility Conditions (Theorem C.4). To provide a measurement profile for the entire network, we therefore analyze the time and memory required to compute these attribution fields and their measurements across N_{node} aggregation nodes by connecting with the computation in model training.

Theorem 3.6 (Attribution Vergence Complexity over Multiple Aggregation Nodes (Informal))**.** *Let* T_{step} *and* $\mathrm{Mem}_{\mathrm{step}}$ *denote the time and peak memory of one ordinary forward/backward step on the same model, batch, and precision. Consider* N_{node} *probed aggregation nodes and an IG path divided into* n *path-discretization intervals. Under the execution and storage conditions formalized in Appendix C.3, the discretized statistics can be evaluated in*

$$T_{\mathrm{total}} = O(nT_{\mathrm{step}}), \quad \mathrm{Mem}_{\mathrm{peak}} = O(\min\{n, N_{\mathrm{node}}\}\mathrm{Mem}_{\mathrm{step}}). \tag{8}$$

The analysis bridges the single training step costs and the measurement costs by reusing $\mathcal{G}$. For measurement, one forward/backward pass supplies the IG step of all probed aggregation nodes. However, additional storage is required since the $w_{y,z}, \mathbf{X}_{y,z}$ across different nodes may correspond to shared intermediate activations during training, but represent distinct routing operations during measurement. As a step forward, streaming nodewise IG and caching pathwise graph states exchange the N_{node} and n storage axes, yielding the stated time–memory bound. The formal theorem and proof are in Appendix C.3, and the corresponding multi-head-attention, convolution, and MLP measurement algorithms are given in Appendix D. For module-level reduction, we average local $\log_2 \mathrm{D}(\mathbf{S})$, $\log_2 \mathrm{D}(\mathbb{A}[\mathbf{S}])$, and $\mathbb{V}_{\mathbb{A}}$ values with weights $\|\mathbb{A}[\mathbf{S}]\|_1$, which gives greater weight to nodes with larger post-aggregation attribution mass, reducing the influence of low-mass nodes that can be more sensitive to attribution errors, following Theorem 3.5 (reduction details appear in Appendix D).

4 Experiments

Given attribution settings, the *Attribution Vergence* framework produces a measurement on each neural module, resulting in a structural profile of attribution organization. We evaluate the measurement in grokking and adversarial transfer through two questions: ❶ **Behavioral responsiveness (RQ1).** How strongly does the proposed measurement respond to the studied behavioral properties, and what gains does the framework's formulation provide? ❷ **Attribution organization (RQ2).** What task- and property-dependent patterns does the measurement of microscopic attribution organization reveal in grokking and adversarial transfer?

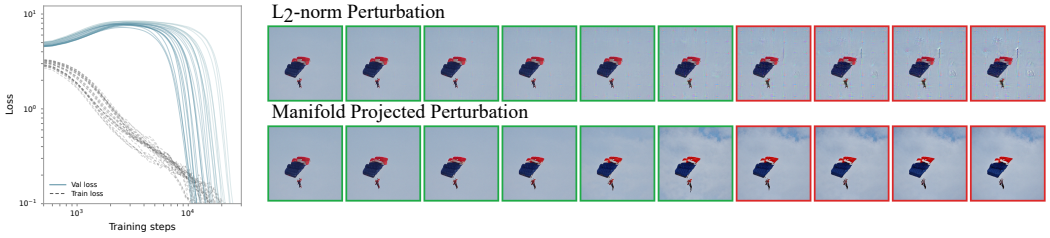

Figure 4: Illustration of phenomena studied in experiments. Left (Grokking): The validation loss drop is delayed compared to the training loss. Training configurations produce different validation-loss trajectories. Right (Adversarial): Green and red denote correct and incorrect classifications by vision encoders. Perturbations with different constraint structures exhibit different transferability.

4.1 Experimental Settings and Statistical Protocol

Statistical protocol. Each setting specifies the property, measurement sites, response correlation statistic $\mathrm{CorrStat}$, analysis groups, and data split. To assess whether a measure reflects a property, we select the site with the strongest response separately within each group and data fold:

$$\mathrm{site}^* = \arg\max_{\mathrm{site}} \left| \mathrm{CorrStat}\big(\mathrm{property}, \mathrm{measure}^{(\mathrm{site})}; \mathrm{selection\ data}\big) \right|. \tag{9}$$

In each fold, we fix the selected site and recompute its response on held-out data, separating selection from evaluation (Cawley & Talbot, 2010). We average response magnitudes over the designated partitions and analysis groups.

Grokking setting. The construction of the grokking model and task is based on (Ruppik et al., 2025). ❶ **Measures:** We measure eight Attention and MLP profile coordinates in a two-block Transformer over 100 trajectories: four tasks *Add P97, S5 Product, S5 Conjugation, and S5 Sandwich*, five configurations varying learning rate and training fraction per task, and five seeds per configuration. Profiles are estimated from a fixed subset of **training samples** (1024 samples for the main experiment). Prior-work analysis baselines are representation LID (Ruppik et al., 2025) and regularization importance from Eichin et al. (2026). ❷ **Analysis groups & data splits:** Tasks define the analysis groups. For each task, two seeds select sites and the remaining three evaluate them, and all ten seed partitions are evaluated and averaged. Each side pools trajectories across all five configurations. ❸ **Properties:** We examine training phase, training accuracy, and validation accuracy as properties. The five phases are underfitting, memorization, transition, generalization, and complete accuracy, defined by training and validation accuracies. ❹ **Response statistics:** For phase, $\mathrm{CorrStat}$ is the *one-way ANOVA effect size* η^2; for accuracy, it is *absolute Spearman correlation* $|\rho|$ or *squared Pearson correlation* R^2. Appendix E.1 specifies the settings.

Adversarial-transfer setting. ❶ **Construction:** We study a flow-matching instantiation of generative semantic attacks (Zhang et al., 2024) on vision encoders. A perturbation is optimized at position t_p on a learned trajectory and transported by the remaining flow, controlling the form of constraint as shown in Figure 4. Generator identity G, budget ϵ and momentum β vary this construction. ViT-B/32, ViT-L/14, and ConvNeXt-L each serve as source model for the other two targets. Each source has 120 configurations with G spanning Pixel Mean Flow (PMF) (Lu et al., 2026), Representation Autoencoder (RAE) (Zheng et al., 2026), and Scalable Interpolant Transformer (SiT) (Ma et al., 2024). Each configuration contains 100 generated clean–adversarial pairs, ten per ImageNette-10 class. Clean endpoints are generator-specific and remain fixed across t_p, ϵ, and β. ❷ **Measures:** We measure **source-model** attention or depthwise-convolution sites using our framework, and compare with the Shapley interaction reference $\widehat{I}_{\mathrm{grid}}$ (Wang et al., 2021c) and intermediate featured displacement (IFD) constructed from the loss in (Huang et al., 2019) ❸ **Groups and splits:** We evaluate generation responses within each source model and transfer responses within each source→target direction. We average 10,000 random class-balanced 50/50 partitions, exchanging selection and evaluation halves within each partition. Configuration means are computed separately on each side. ❹ **Properties & responses statistics:** $\mathrm{CorrStat}$ is ANOVA η^2 for generator identity G, and $|\rho|$ or R^2 for injection position t_p. Target responses use partial $|\rho|$ or partial R^2 with correct-class image–text similarity, controlling source similarity. Appendix F.1 provides the full settings.

Table 1: Property responses in grokking and adversarial transfer. Bold marks the largest response for each statistic. $\mathbb{V}_{\mathbb{A}}$ has the largest mean phase, validation, and target responses.

	Task	Grokking					Adversarial transfer				
	Property	Phase type	Train acc.		Val. acc.		Generation config.			Target response	
	Correlation Stat.	η^2	$\lvert\rho\rvert$	R^2	$\lvert\rho\rvert$	R^2	η^2_G	$\lvert\rho\rvert_{t_p}$	$R^2_{t_p}$	Partial $\lvert\rho\rvert$	Partial R^2
Measure	IFD (Huang et al., 2019)	—	—	—	—	—	0.387	0.401	0.163	0.661	0.497
	$\widehat{I}_{\text{grid}}$ (Wang et al., 2021c)	—	—	—	—	—	0.416	0.220	0.058	0.298	0.152
	LID (Ruppik et al., 2025)	0.614	0.715	0.427	0.693	0.511	—	—	—	—	—
	Reg. Importance (Eichin et al., 2026)	0.575	0.744	**0.591**	0.745	0.522	—	—	—	—	—
	$\log_2 \mathrm{D}(\mathbf{S})$ (Ours)	0.761	0.778	0.377	0.791	0.662	0.492	0.592	0.368	0.816	0.612
	$\log_2 \mathrm{D}(\mathbb{A}[\mathbf{S}])$ (Ours)	0.612	0.693	0.303	0.757	0.527	0.642	0.553	0.324	0.772	0.571
	$\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ (Ours)	**0.819**	**0.812**	0.404	**0.826**	**0.769**	**0.645**	**0.711**	**0.420**	**0.853**	**0.622**

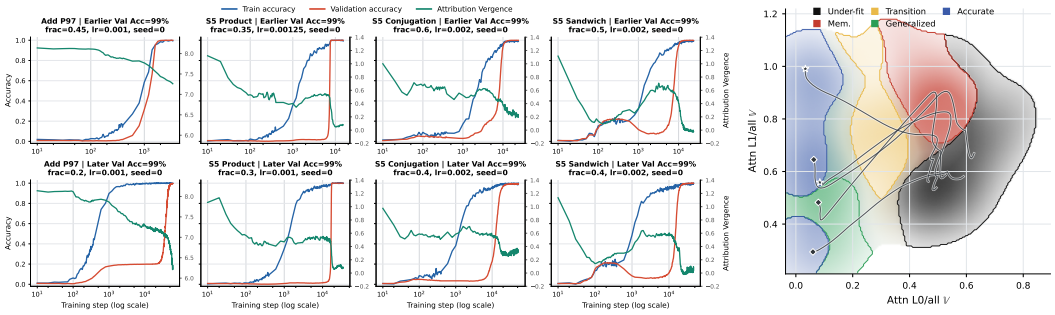

Figure 5: $\mathbb{V}_{\mathbb{A}}$ tracks grokking progress and reveals phase-dependent organization. Left: accuracy and measurement trajectories. Right: S5 Product exhibits distinct regions associated with memorization and generalization, making the change in internal organization visible.

4.2 Results

Behavioral responses and comparison (RQ1). ❶ Comparison of analysis tools from prior work. As shown in table 1, for grokking, $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ has larger mean phase and validation accuracy responses than LID and *Reg. Importance*, despite exhibiting a weaker R^2 correlation with training accuracy, which demonstrates that $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ serves as an effective indicator of grokking progress rather than training accuracy. For Val. Acc., $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$'s mean R^2 reaches 0.769, compared with 0.511 for LID and 0.522 for *Reg. Importance*. The mean phase and validation advantages of $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ also persist across different numbers of measured samples $N = 64, 256, 1024$ (Table E.2). For adversarial transfer, it exceeds both $\widehat{I}_{\text{grid}}$ and IFD in all five mean responses. In particular, its average transferability correlation (source-controlled target similarity) reaches 0.853 in partial $|\rho|$ and 0.622 in partial R^2, compared with 0.661 and 0.497, respectively, for IFD. **❷ Gains from the framework formulation.** Both $\log_2 \mathrm{D}(\mathbf{S})$ and $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ have stronger mean responses than $\log_2 \mathrm{D}(\mathbb{A}[\mathbf{S}])$ for grokking phase and validation accuracy, injection position, and target transfer. The logical route field made accessible by our formalization and computation thus provides organization information that the aggregated output summary alone does not express as strongly. By comparing the input-side and output-side measurements, $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ further strengthens mean responses: $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ has higher mean responses than $\log_2 \mathrm{D}(\mathbf{S})$ across all ten evaluated response properties. We further assess the robustness of the comparisons between measurements *under configuration resampling and random image or training-seed partitions*, supporting the reliability of the main findings (Appendix G, more variants of formulation are also compared). Task-, architecture-, and direction-specific analyses appear in Appendix E.2 for Grokking and F.2 for Adversarial Transfer.

Revealing internal reorganization (RQ2). Figure 5 shows that $\mathbb{V}_{\mathbb{A}}$ continues to change after training accuracy has saturated in several trajectories, with marked changes around the rise in validation accuracy. After controlling training accuracy and $\log t$ together, its mean validation responses remain 0.486 for partial $|\rho|$ and 0.618 for partial R^2 (Table E.3), demonstrating monotonic tracking of validation progress beyond training performance and temporal proxies. As a scalar indicator of generative adversarial transfer, $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ exhibits stronger mean source-controlled responses than injection position t_p or momentum β considered individually (Tables F.2 and G.2), and is supported by the configuration and image sample bootstrap in Appendix G.

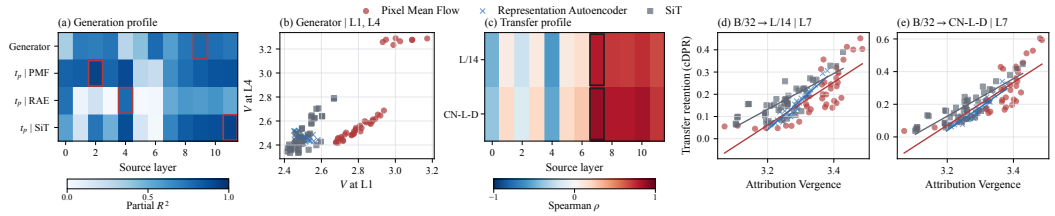

Figure 6: Generation and adversarial transfer in ViT-B. (a) Generation response profiles; outlined cells mark selected maximal response sites. (b) Generator organization in the L1–L4 $\mathbb{V}_{\mathbb{A}}$ plane; (c) Transfer response profiles; (d,e) Associations with transfer retention for ViT-L and ConvNeXt.

Generator-dependent attribution organization (RQ2). For ViT-B, within-generator t_p responses peak at L2 for PMF, L4 for RAE, and L11 for SiT, while joint L1–L4 measurements show distinct generator-associated distributions (Figure 6). Across all three source models, PMF produces pronounced early-layer t_p responses, consistent with its operation directly in pixel space rather than the latent spaces used by RAE and SiT (Figure F.1). Meanwhile, different targets share a transfer-response peak within each source, where greater $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ accompanies stronger transfer (Figure 6c–e). These results suggest that **attribution organization responds to different perturbation properties at different layers**, and the transfer-retention peaks are distinct from the generation-response peaks. Wang et al. (2021c) link several conventional transfer attacks to reduced Shapley interactions among input perturbations. In our experiments, the positive association between $\mathbb{V}_{\mathbb{A}}$ and transfer concerns attribution organization within source-model aggregation operations. $\mathbb{V}_{\mathbb{A}}$ characterizes how distributed microscopic contributions combine into evidence for the model's decision. Complex semantic features shared across models may require combinations of distributed microscopic contributions. Generative perturbations that engage these combinations could therefore produce misleading evidence shared across models, with higher $\mathbb{V}_{\mathbb{A}}$ reflecting more effective input attribution dimensions relative to those present in the aggregated output. Together with the shared source-layer peak, these observations suggest that transferable adversarial features in generative perturbations engage a common organization of attributed contributions in the source model.

Task-dependent reorganization of attribution (RQ2). Whereas Eichin et al. (2026) associate late generalization with greater regularization influence and embedding-decoder alignment, our results suggest focusing on the organization of attribution. Figure 5 shows the $\mathbb{V}_{\mathbb{A}}$ trajectories with measurement samples fixed within each run. More multi-site trajectory and phase diagrams are shown in Figure E.2. Although the evolution of $\mathbb{V}_{\mathbb{A}}$ fluctuates, the trajectories of multiple sites embody statistical regularities. From the first measured checkpoint to Train Acc=95% (Train95), 6/100 trajectories show net increases at all 8 sites and 0/100 show net decreases at all sites. During the subsequent progression toward generalization (Train95–Val99), 0/97 trajectories show net increases at all sites, whereas 34/97 show net decreases at all sites (detailed in Table E.4). These stage-level changes reveal a reorganization of attributed contributions across layers. The effective diversity of contributions brought together by a module can be viewed as a form of local complexity, which $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ characterizes relative to the aggregation output. This perspective complements Liu et al. (2023), who relate grokking to a reduction in nonlinear network complexity, by examining where complexity changes within the network. The different directions of $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ change suggest that generalization may involve **reorganizing local complexity** across layers to match task requirements, with possible increases at some sites accompanying decreases at others. Although implicit gradient regularization favors flatter local loss geometry (Barrett & Dherin, 2021), establishing a suitable organization across layers may require slower, coordinated changes in model complexity. The difficulty of this reorganization offers a plausible hypothesis for prolonged memorization.

5 Conclusion

We introduce Attribution Vergence as a common, computable measurement of attribution organization under feature aggregation. Experiments support response gains from the formulation and relative measurement, while revealing shared source indicators of generative adversarial transfer and reorganization across layers during algorithmic grokking. Overall, the framework discover an novel observable and provides a measurement tool toward a scientific understanding of AI models. Limitations and potential future work are discussed in Appendix H.

AI USE STATEMENT

Method parts. The following parts were constructed by human authors without using generative AI tools: formulation, theorem proposal, proof sketch initiation, experiment task construction, algorithm construction, organization of the main paper and the appendix. The following parts were discussed with LLMs and finalized by the authors: experimental statistical protocols, proofs of the upper and lower bounds (the extreme cases were constructed by human, and the bound extension was then refined with LLM). We adapted the feedbacks from PAT automated pre-submission feedback to refine the experiment and retrieving related works. We checked LLM feedback for potential plagiarism through a manual literature survey, and we verified the proof using Lean.

Execution parts. The following parts have been implemented by human authors with LLM checks & feedback: Detailed construction of formalization & axioms. Algorithm implementation and coding architecture design. Experiment arrangement and settings instantiation. Writing of the main paper. The following parts are partially implemented by LLMs with human author & LLM checks: Experiment script coding. Algorithm implementation refinement. Experiments execution. Proof writing. Polishing the Main Paper and writing the Appendices. We have reviewed all AI-assisted work and code.

We take responsibility for the final content of this work, including text or artifacts produced with the aid of generative AI.

REFERENCES

Reduan Achtibat, Sayed Mohammad Vakilzadeh Hatefi, Maximilian Dreyer, Aakriti Jain, Thomas Wiegand, Sebastian Lapuschkin, and Wojciech Samek. AttnLRP: Attention-aware layer-wise relevance propagation for transformers. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 135–168. PMLR, 2024.

Emmanuel Ameisen, Jack Lindsey, Adam Pearce, Wes Gurnee, Nicholas L. Turner, Brian Chen, Craig Citro, David Abrahams, Shan Carter, Basil Hosmer, Jonathan Marcus, Michael Sklar, Adly Templeton, Trenton Bricken, Callum McDougall, Hoagy Cunningham, Thomas Henighan, Adam Jermyn, Andy Jones, Andrew Persic, Zhenyi Qi, T. Ben Thompson, Sam Zimmerman, Kelley Rivoire, Thomas Conerly, Chris Olah, and Joshua Batson. Circuit tracing: Revealing computational graphs in language models. *Transformer Circuits Thread*, 2025. URL `https://transformer-circuits.pub/2025/attribution-graphs/methods.html`.

Laurent Amsaleg, Oussama Chelly, Teddy Furon, Stéphane Girard, Michael E. Houle, Ken-ichi Kawarabayashi, and Michael Nett. Estimating local intrinsic dimensionality. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '15, pp. 29–38, New York, NY, USA, 2015. Association for Computing Machinery. ISBN 9781450336642.

Alessio Ansuini, Alessandro Laio, Jakob H Macke, and Davide Zoccolan. Intrinsic dimension of data representations in deep neural networks. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

David Barrett and Benoit Dherin. Implicit gradient regularization. In *International Conference on Learning Representations*, 2021.

Walter Baur and Volker Strassen. The complexity of partial derivatives. *Theoretical computer science*, 22(3):317–330, 1983.

Atilim Gunes Baydin, Barak A. Pearlmutter, Alexey Andreyevich Radul, and Jeffrey Mark Siskind. Automatic differentiation in machine learning: A survey. *Journal of Machine Learning Research*, 18(153):1–43, 2018.

Umang Bhatt, Adrian Weller, and José M. F. Moura. Evaluating and aggregating feature-based model explanations. In Christian Bessiere (ed.), *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pp. 3016–3022. International Joint Conferences on Artificial Intelligence Organization, 7 2020. Main track.

J. Lawrence Carter and Mark N. Wegman. Universal classes of hash functions. *Journal of Computer and System Sciences*, 18(2):143–154, 1979.

Gavin C. Cawley and Nicola L. C. Talbot. On over-fitting in model selection and subsequent selection bias in performance evaluation. *Journal of Machine Learning Research*, 11:2079–2107, 2010.

Prasad Chalasani, Jiefeng Chen, Amrita Roy Chowdhury, Xi Wu, and Somesh Jha. Concise explanations of neural networks using adversarial training. In Hal Daumé III and Aarti Singh (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 1383–1391. PMLR, 13–18 Jul 2020.

Aditya Chattopadhyay, Piyushi Manupriya, Anirban Sarkar, and Vineeth N. Balasubramanian. Neural network attributions: A causal perspective. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 981–990. PMLR, 2019.

Hugh Chen, Scott M Lundberg, and Su-In Lee. Explaining a series of models by propagating shapley values. *Nature communications*, 13(1):4512, 2022.

Yanbo Chen and Weiwei Liu. A theory of transfer-based black-box attacks: Explanation and implications. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Jiali Cheng, Ziheng Chen, Chirag Agarwal, and Hadi Amiri. A mechanistic perspective and circuit-guided difficulty metric for unlearning. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 10950–10964. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.findings-acl.532.

Arthur Conmy, Augustine N. Mavor-Parker, Aengus Lynch, Stefan Heimersheim, and Adrià Garriga-Alonso. Towards automated circuit discovery for mechanistic interpretability. In *Advances in Neural Information Processing Systems*, volume 36, pp. 16318–16352. Curran Associates, Inc., 2023. doi: 10.52202/075280-0719.

Ambra Demontis, Marco Melis, Maura Pintor, Matthew Jagielski, Battista Biggio, Alina Oprea, Cristina Nita-Rotaru, and Fabio Roli. Why do adversarial attacks transfer? explaining transferability of evasion and poisoning attacks. In *28th USENIX Security Symposium*, pp. 321–338. USENIX Association, 2019.

Kedar Dhamdhere, Mukund Sundararajan, and Qiqi Yan. How important is a neuron. In *International Conference on Learning Representations*, 2019.

Florian Eichin, Yupei Du, Philipp Mondorf, Maria Matveev, Barbara Plank, and Michael A. Hedderich. ExPLAIND: Unifying model, data, and training attribution to study model behavior. In *Forty-third International Conference on Machine Learning*, 2026.

Elena Facco, Maria d'Errico, Alex Rodriguez, and Alessandro Laio. Estimating the intrinsic dimension of datasets by a minimal neighborhood information. *Scientific Reports*, 7:12140, 2017. doi: 10.1038/s41598-017-11873-y.

Serge Fehr and Stefan Berens. On the conditional rényi entropy. *IEEE Transactions on Information Theory*, 60(11):6801–6810, 2014. doi: 10.1109/TIT.2014.2357799.

Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations*, 2015.

Michael Hanna, Sandro Pezzelle, and Yonatan Belinkov. Have faith in faithfulness: Going beyond circuit overlap when finding model mechanisms. In *First Conference on Language Modeling*, 2024.

Yihuai Hong, Lei Yu, Haiqin Yang, Shauli Ravfogel, and Mor Geva. Intrinsic test of unlearning using parametric knowledge traces. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 19513–19535. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.emnlp-main.985.

Patrik O Hoyer. Non-negative matrix factorization with sparseness constraints. *Journal of machine learning research*, 5(Nov):1457–1469, 2004.

Jin Hu, Jiakai Wang, Linna Jing, Haolin Li, Haodong Liu, Haotong Qin, Aishan Liu, Ke Xu, and Xianglong Liu. Exploring semantic-constrained adversarial example with instruction uncertainty reduction. In *Advances in Neural Information Processing Systems*, volume 38, pp. 102640–102692. Curran Associates, Inc., 2025.

Qian Huang, Isay Katsman, Horace He, Zeqi Gu, Serge Belongie, and Ser-Nam Lim. Enhancing adversarial example transferability with an intermediate level attack. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4733–4742, 2019. doi: 10.1109/ICCV.2019.00483.

Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. Adversarial examples are not bugs, they are features. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

Mark Jerrum and Alistair Sinclair. Conductance and the rapid mixing property for markov chains: the approximation of permanent resolved. In *Proceedings of the Twentieth Annual ACM Symposium on Theory of Computing*, STOC '88, pp. 235–244, New York, NY, USA, 1988. Association for Computing Machinery. ISBN 0897912640.

Tanishq Kumar, Blake Bordelon, Samuel J. Gershman, and Cengiz Pehlevan. Grokking as the transition from lazy to rich training dynamics. In *The Twelfth International Conference on Learning Representations*, 2024.

Markku Laakso and Rein Taagepera. "effective" number of parties: A measure with application to west europe. *Comparative Political Studies*, 12(1):3–27, 1979.

Noam Levi, Alon Beck, and Yohai Bar-Sinai. Grokking in linear estimators – a solvable model that groks without understanding. In *The Twelfth International Conference on Learning Representations*, 2024.

Ziming Liu, Ouail Kitouni, Niklas S. Nolte, Eric Michaud, Max Tegmark, and Mike Williams. Towards understanding grokking: An effective theory of representation learning. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

Ziming Liu, Ziqian Zhong, and Max Tegmark. Grokking as simplification: A nonlinear complexity perspective. In *UniReps: the First Workshop on Unifying Representations in Neural Models*, 2023.

Yiyang Lu, Susie Lu, Qiao Sun, Hanhong Zhao, Zhicheng Jiang, Xianbang Wang, Tianhong Li, Zhengyang Geng, and Kaiming He. One-step latent-free image generation with pixel mean flows. In *Forty-third International Conference on Machine Learning*, 2026.

Scott M. Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Daniel D Lundstrom, Tianjian Huang, and Meisam Razaviyayn. A rigorous study of integrated gradients method and extensions to internal neuron attributions. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 14485–14508. PMLR, 17–23 Jul 2022.

Haotian Ma, Hao Zhang, Fan Zhou, Yinqing Zhang, and Quanshi Zhang. Quantification and analysis of layer-wise and pixel-wise information discarding. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 14664–14698. PMLR, 2022.

Nanye Ma, Mark Goldstein, Michael S Albergo, Nicholas M Boffi, Eric Vanden-Eijnden, and Saining Xie. Sit: Exploring flow and diffusion-based generative models with scalable interpolant transformers. In *European Conference on Computer Vision*, pp. 23–40. Springer, 2024.

Xingjun Ma, Bo Li, Yisen Wang, Sarah M. Erfani, Sudanthi Wijewickrema, Grant Schoenebeck, Dawn Song, Michael E. Houle, and James Bailey. Characterizing adversarial subspaces using local intrinsic dimensionality. In *International Conference on Learning Representations*, 2018.

Joseph Miller, Bilal Chughtai, and William Saunders. Transformer circuit faithfulness metrics are not robust. In *First Conference on Language Modeling*, 2024.

Brent Mittelstadt, Chris Russell, and Sandra Wachter. Explaining explanations in AI. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pp. 279–288, 2019.

Aaron Mueller, Atticus Geiger, Sarah Wiegreffe, Dana Arad, Iván Arcuschin, Adam Belfki, Yik Siu Chan, Jaden Fried Fiotto-Kaufman, Tal Haklay, Michael Hanna, Jing Huang, Rohan Gupta, Yaniv Nikankin, Hadas Orgad, Nikhil Prakash, Anja Reusch, Aruna Sankaranarayanan, Shun Shao, Alessandro Stolfo, Martin Tutek, Amir Zur, David Bau, and Yonatan Belinkov. MIB: A mechanistic interpretability benchmark. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 45069–45108. PMLR, 2025.

Neel Nanda, Lawrence Chan, Tom Lieberum, Jess Smith, and Jacob Steinhardt. Progress measures for grokking via mechanistic interpretability. In *The Eleventh International Conference on Learning Representations*, 2023.

M. E. J. Newman. Modularity and community structure in networks. *Proceedings of the National Academy of Sciences*, 103(23):8577–8582, 2006.

Guilliermo Owen. Values of games with a priori unions. In Rudolf Henn and Otto Moeschlin (eds.), *Mathematical Economics and Game Theory*, pp. 76–88, Berlin, Heidelberg, 1977. Springer Berlin Heidelberg. ISBN 978-3-642-45494-3.

Nicolas Papernot, Patrick McDaniel, Ian Goodfellow, Somesh Jha, Z. Berkay Celik, and Ananthram Swami. Practical black-box attacks against machine learning. In *Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security*, pp. 506–519, 2017.

Alethea Power, Yuri Burda, Harri Edwards, Igor Babuschkin, and Vedant Misra. Grokking: Generalization beyond overfitting on small algorithmic datasets. In *ICLR MATH-AI Workshop*, 2021.

Muhammad Rashid, Elvio G Amparore, Enrico Ferrari, and Damiano Verda. ShapBPT: Image Feature Attributions Using Data-Aware Binary Partition Trees. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 25099–25107, 2026.

Alfréd Rényi. On measures of entropy and information. *Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Contributions to the Theory of Statistics*, 1961.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should i trust you?": Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1135–1144, 2016.

Benjamin Matthias Ruppik, Julius von Rohrscheidt, Carel van Niekerk, Michael Heck, Renato Vukovic, Shutong Feng, Hsien-chin Lin, Nurul Lubis, Bastian Rieck, Marcus Zibrowius, and Milica Gašić. Less is more: Local intrinsic dimensions of contextual language models. In *Advances in Neural Information Processing Systems*, volume 38, 2025.

Wojciech Samek, Grégoire Montavon, Sebastian Lapuschkin, Christopher J Anders, and Klaus-Robert Müller. Explaining deep neural networks and beyond: A review of methods and applications. *Proceedings of the IEEE*, 109(3):247–278, 2021.

Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pp. 618–626, 2017.

Lloyd S Shapley. A value for n-person games. *Contributions to the Theory of Games*, 2(28):307–317, 1953.

Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning important features through propagating activation differences. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 3145–3153. PMLR, 2017.

Raghav Singal, George Michailidis, and Hoiyi Ng. Flow-based attribution in graphical models: A recursive shapley approach. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 9733–9743. PMLR, 18–24 Jul 2021.

Edward Stevinson, Lucas Prieto, Melih Barsbey, and Tolga Birdal. Adversarial vulnerability from interference between features in superposition. In *Forty-third International Conference on Machine Learning*, 2026.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *International conference on machine learning*, pp. 3319–3328. PMLR, 2017.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In *International Conference on Learning Representations*, 2014.

Florian Tramèr, Nicolas Papernot, Ian Goodfellow, Dan Boneh, and Patrick McDaniel. The space of transferable adversarial examples. *arXiv preprint arXiv:1704.03453*, 2017.

Vikrant Varma, Rohin Shah, Zachary Kenton, János Kramár, and Ramana Kumar. Explaining grokking through circuit efficiency. *arXiv preprint arXiv:2309.02390*, 2023.

Jiakai Wang, Aishan Liu, Zixin Yin, Shunchang Liu, Shiyu Tang, and Xianglong Liu. Dual attention suppression attack: Generate adversarial camouflage in physical world. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8565–8574, 2021a.

Jiaxuan Wang, Jenna Wiens, and Scott Lundberg. Shapley flow: A graph-based approach to interpreting model predictions. In Arindam Banerjee and Kenji Fukumizu (eds.), *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, pp. 721–729. PMLR, 13–15 Apr 2021b.

Xin Wang, Jie Ren, Shuyun Lin, Xiangming Zhu, Yisen Wang, and Quanshi Zhang. A unified approach to interpreting and boosting adversarial transferability. In *International Conference on Learning Representations*, 2021c.

Andi Zhang, Mingtian Zhang, and Damon Wischik. Constructing semantics-aware adversarial examples with probabilistic perspective. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

Cenwei Zhang, Lin Zhu, Manxi Lin, and Lei You. Axiomatic on-manifold shapley via optimal generative flows. *arXiv e-prints*, pp. arXiv–2603, 2026.

Boyang Zheng, Nanye Ma, Shengbang Tong, and Saining Xie. Diffusion transformers with representation autoencoders. In *The Fourteenth International Conference on Learning Representations*, 2026.

CONTENTS

1 Introduction . . . 1

2 Preliminaries . . . 2

3 The Attribution Vergence Measurement Framework . . . 3
- 3.1 Formulation . . . 3
- 3.2 Analysis of Attribution Vergence . . . 5

4 Experiments . . . 6
- 4.1 Experimental Settings and Statistical Protocol . . . 7
- 4.2 Results . . . 8

5 Conclusion . . . 9

A Notation . . . 16

B Background and Related Work . . . 16
- B.1 Key Phenomena Studied . . . 16
- B.2 Mathematical Background of Shapley Value and Rényi Entropy . . . 17
- B.3 Mechanistic Attribution of Deep Networks . . . 18

C Detailed Mathematical Proofs of Attribution Vergence . . . 19
- C.1 Pre-Aggregation Attribution Fields . . . 19
- C.2 Representation Invariances under a Fixed Coordinate Mapping . . . 26
- C.3 $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ Complexity over Multiple Aggregation Nodes . . . 28
- C.4 Perturbation Stability . . . 30
- C.5 Coordinate-Positioning Entropy Interpretation . . . 31
- C.6 Lower and Upper Bounds . . . 32

D Implementation of Attribution Vergence . . . 36

E Experimental Details: Grokking . . . 40
- E.1 Experimental Setup . . . 40
- E.2 Detailed Measurement-Profile Results . . . 42

F Experimental Details: Adversarial Transferability . . . 47
- F.1 Experimental Setup . . . 47
- F.2 Measurement Profiles and Sitewise Response Analyses . . . 50
- F.3 Source-dependent Attribution Organization . . . 53

G Paired Comparisons of Property Responses . . . 54
- G.1 Rényi Orders and Vergence Components . . . 56
- G.2 Integration Precision and Computational Cost . . . 58

H Limitations and Future Work . . . 60

A NOTATION

Symbol	Description	Ref.
Aggregation Node Formulation		
F	Scalar objective of the network's final output.	Sec. 2
$\mathcal{G} = (\mathcal{V}, \mathcal{E})$, a	Computation graph and an aggregation node.	Def. C.1
$\mathcal{Y}$, d	Unreduced output-coordinate set and its size $d = \lvert\mathcal{Y}\rvert$.	Def. 3.1
$\mathcal{Z}$, K	Routing-coordinate set and its size $K = \lvert\mathcal{Z}\rvert$.	Def. 3.1
$w_{y,z}$, $\mathbf{X}_{y,z}$, $\mathbf{O}_y$	Routing weight, feature input and output.	Def. 3.1
$e_{y,z}$	Logical input edge for output y and route z.	Def. C.1
$\mathbf{S} \in \mathbb{R}^{d\times K}$	Signed input-edge attribution field, with credit $\mathbf{S}_{y,z}$ assigned to $e_{y,z}$.	Def. 3.1
$\mathbb{A}$	Attribution-space aggregation operator, $\mathbb{A}[\mathbf{S}]_y = \sum_z \mathbf{S}_{y,z}$.	Def. 3.1
Attribution, Games, and Compatibility		
$\mathrm{IG}_i(x)$	Integrated Gradients attribution of input feature i.	App. B
$\mathrm{Cond}(F;\gamma,\mathbf{O}_y)$	Neuron Conductance of output coordinate $\mathbf{O}_y$ along attribution path γ.	Thm. 3.1
$\mathcal{N}_y$, $\mathcal{N}$, $\mathcal{P}$	Constituent-edge group for output y, full player set, and routing partition.	Def. C.3
v, $\bar{v}$	Input-edge and output-coordinate games for objective F.	Def. C.3
$\phi_i(v)$	Shapley allocation to player i in game v.	App. B
$\mathrm{Ow}_e(v,\mathcal{P})$	Owen allocation to constituent edge e under partition $\mathcal{P}$.	Thm. C.2
$\mathbf{S}^F_{\mathrm{post},y}$	Independently specified attribution to output coordinate y.	Def. C.6
Effective Dimensionality and Attribution Vergence		
$p_{\lvert\boldsymbol{v}\rvert}(i)$	Normalized absolute attribution mass $\lvert\boldsymbol{v}_i\rvert/\lVert\boldsymbol{v}\rVert_1$.	Def. 3.2
$H_2(p)$	Base-2 Rényi entropy of order two.	Def. 3.2
$\mathrm{D}(\boldsymbol{v})$	Effective dimensionality $\lVert\boldsymbol{v}\rVert_1^2/\lVert\boldsymbol{v}\rVert_2^2$, for $\boldsymbol{v} \neq 0$.	Def. 3.2
$\mathrm{D}(\mathbf{S})$, $\mathrm{D}(\mathbb{A}[\mathbf{S}])$	Effective dimensionalities before and after aggregation.	Def. 3.3
$\mathbb{V}_{\mathbb{A}}(\mathbf{S})$	Attribution Vergence $\log_2 \mathrm{D}(\mathbf{S}) - \log_2 \mathrm{D}(\mathbb{A}[\mathbf{S}])$.	Def. 3.3
Computational Complexity		
n, N_{node}	Numbers of integration intervals and probed aggregation nodes.	App. C.3
T_{step}, $\mathrm{Mem}_{\mathrm{step}}$	Work and peak storage of one forward–reverse execution.	Def. C.8
Experimental Responses		
ρ, $\lvert\rho\rvert$	Spearman rank correlation and its magnitude.	App. E.1
R^2, partial R^2	Squared Pearson correlation and squared Pearson partial correlation.	App. E.2
η^2	One-way ANOVA effect size for phase or generator identity.	Sec. 4.1
G, t_p	Generator identity and perturbation injection position.	Sec. 4.1
ϵ, β	Attack budget and optimizer momentum coefficient.	App. F.1
sim_S, sim_T	Mean source and target correct-class image–text similarities.	App. F.2
cDPR	Conditional Directional Preservation Ratio for transfer retention.	Eq. 155

B BACKGROUND AND RELATED WORK

This section positions Attribution Vergence relative to prior work on delayed generalization and adversarial transferability, and then reviews the attribution and information-theoretic foundations used by our framework.

B.1 KEY PHENOMENA STUDIED

Grokking Grokking describes a long-delayed transition in which a model can remain fitted to its training set for an extended interval before it acquires a solution that generalizes Power et al. (2021). This gap between abrupt behavior and gradual learning has prompted accounts based on structured representations and phase diagrams Liu et al. (2022), circuit efficiency Varma et al. (2023), nonlinear compression Liu et al. (2023), and lazy-to-rich feature learning Kumar et al. (2024). More directly, the reverse engineering of small Transformers by Nanda et al. (2023) reveals Fourier-based circuits and continuous progress measures spanning memorization, circuit formation, and cleanup on the *Add p97* task. Ruppik et al. (2025) use local intrinsic dimension (LID) Amsaleg et al. (2015) to characterize dataset-level hidden-representation geometry with a task-independent statistic. Attribution Vergence complements task-specific circuit recovery and dataset-level representation geometry

by measuring target-conditioned attribution reorganization at aggregation nodes with a computation-graph-grounded, aggregation-relative observable.

Adversarial Transfer An adversarial example $x_{\mathrm{adv}} = x + \delta$ changes a model's prediction while keeping δ small under a norm or perceptual constraint Szegedy et al. (2014); Goodfellow et al. (2015). Transferability occurs when a perturbation optimized on a source model also affects held-out target models, enabling practical black-box attacks Papernot et al. (2017). Research has therefore followed a chain from the vulnerable signal to the shared structure that makes it transferable. Non-robust features Ilyas et al. (2019) and superposition Stevinson et al. (2026) describe what perturbations may exploit in learned representations, whereas adversarial subspaces Tramèr et al. (2017), gradient alignment Demontis et al. (2019), and manifold anchoring Chen & Liu (2023) place transfer in shared decision geometry.

Related to internal mechanism and attribution, the Intermediate Level Attack of Huang et al. (2019) improves black-box transfer by amplifying a source-model intermediate-feature displacement and gives a source-only procedure for choosing the attacked layer. Attribution-based approaches use input Shapley interactions to analyze and enhance families of transfer attacks Wang et al. (2021c), or use Grad-CAM-derived attention to construct transferable attacks Wang et al. (2021a). Building on these findings about intermediate features, output deviations, and attribution structure, we measure how target-conditioned decision credit reorganizes at an internal aggregation node. Recent generative semantic attacks move beyond perturbations whose final form is restricted to a fixed L_p ball Zhang et al. (2024). Our experiments instantiate this direction with flow-matching trajectories, while related diffusion-based adversarial attacks suggest that transport through learned generative structure can offer a potential advantage for cross-model transfer Hu et al. (2025). Our source-model statistic is fixed before any target evaluation. The target is held out during perturbation generation and source-profile computation and enters only through post-hoc transfer-retention or target-similarity evaluation.

B.2 Mathematical Background of Shapley Value and Rényi Entropy

Shapley Value and Linearity Axioms The formal quest for model interpretability often seeks to distribute the outcome of a machine learning system among its input features. This credit-assignment problem maps directly to cooperative game theory, where a characteristic function $v : 2^N \to \mathbb{R}$ evaluates the payoff of a coalition of features $S \subseteq N$. The Shapley value Shapley (1953) is the unique allocation satisfying Efficiency, Symmetry, Dummy, and Linearity, where Linearity combines Additivity and scalar homogeneity:

$$\phi_i(v) = \sum_{S \subseteq N \setminus \{i\}} \frac{|S|!(|N| - |S| - 1)!}{|N|!} \left[v(S \cup \{i\}) - v(S) \right] \tag{10}$$

Efficiency guarantees that the attributions recover the total payoff difference, $\sum_{i \in N} \phi_i(v) = v(N) - v(\emptyset)$, whereas Linearity states that $\phi_i(av + bw) = a\phi_i(v) + b\phi_i(w)$. The former supplies attribution completeness and the latter permits attributed decision credit to be combined without changing its accounting semantics.

Graph-Based and Compositional Attribution Graph structure has previously been incorporated into attribution through several complementary formulations. Neural networks have been treated as structural causal models for interventional attribution Chattopadhyay et al. (2019). Shapley Flow assigns credit to edges of a directed acyclic graph and characterizes a graph-level extension of the Shapley axioms Wang et al. (2021b), while Recursive Shapley Value describes how source effects propagate through graphical models under flow-based axioms Singal et al. (2021). DeepLIFT propagates reference-relative contributions through neural layers Shrikumar et al. (2017), and G-DeepSHAP propagates local attributions through a series of models while enforcing layerwise Efficiency Chen et al. (2022). Building on these graph-based and compositional approaches, the Aggregation Node Formulation uses the computation graph to define a local multiply–add aggregation node, distinguishes shared activation tensors from distinct aggregation-node input edges, and assigns decision credit as input-edge attribution.

Rényi Entropy, Collision Entropy, and Effective Hash Dimension For a discrete probability distribution p, the Rényi entropy of order $\alpha \neq 1$ is Rényi (1961)

$$H_\alpha(p) = \frac{1}{1-\alpha} \log_2 \left(\sum_i p_i^\alpha \right). \tag{11}$$

At $\alpha = 2$, the inner sum has an operational collision meaning: for independent draws $X, X' \sim p$,

$$C_2(p) := \Pr[X = X'] = \sum_i p_i^2, \qquad H_2(p) = -\log_2 C_2(p). \tag{12}$$

Thus the exponential $N_{\text{eff}}^{(2)}(p) := 2^{H_2(p)} = 1/\sum_i p_i^2$ is the effective number of equally occupied states.

The same identity gives an equivalent hashing interpretation. A uniform hash into M buckets has pairwise collision probability $1/M$ Carter & Wegman (1979); hence

$$b_{\text{eff}}(p) := -\log_2 C_2(p) = H_2(p), \qquad M_{\text{eff}}(p) := 2^{b_{\text{eff}}(p)} = N_{\text{eff}}^{(2)}(p). \tag{13}$$

We call b_{eff} the *effective hash dimension*: it is the number of ideal uniform hash bits that would produce the same two-sample collision rate, while M_{eff} is the corresponding effective bucket count. This is an equivalent-bit semantics, not a claim that a fractional b_{eff} is a literal hash width or a geometric rank. Compared with Shannon entropy, the order-2 quantity places greater weight on dominant coordinates, matching our goal of measuring how attribution mass is organized across active coordinates.

B.3 Mechanistic Attribution of Deep Networks

Integrated Gradients and Path Completeness Exact discrete Shapley attribution is combinatorially expensive in high-dimensional neural networks. Integrated Gradients (IG) Sundararajan et al. (2017) instead integrates feature gradients along a straight path from a baseline x' to the input x:

$$\text{IG}_i(x) = (x_i - x_i') \int_0^1 \frac{\partial F(x' + \alpha(x - x'))}{\partial x_i} \, \mathrm{d}\alpha. \tag{14}$$

The path integral preserves completeness, $\sum_i \text{IG}_i(x) = F(x) - F(x')$, providing the accounting semantics needed to compare the pre-aggregation attribution field with the post-aggregation attribution field. Complementary work analyzes the axiomatic scope and input sensitivity of IG and gives an efficient method for identifying internal neurons that contribute to specified regions of an IG attribution map Lundstrom et al. (2022).

Internal-Coordinate and Edge Attribution Neuron Conductance Dhamdhere et al. (2019) lifts path attribution from input features to internal activation coordinates. Let $y = h(x)$ be an internal representation and $F(x) = g(h(x))$ the subsequent network branch. The conductance of coordinate j follows from the chain rule along the same input path:

$$\text{Cond}_j(x) = \int_0^1 \frac{\partial g(h(x' + \alpha(x - x')))}{\partial h_j} \frac{\partial h_j(x' + \alpha(x - x'))}{\partial \alpha} \, \mathrm{d}\alpha. \tag{15}$$

It therefore provides path-wise attribution accounting over internal coordinates. Layer-wise Relevance Propagation (LRP) redistributes a prediction backward through the network under layerwise conservation rules Samek et al. (2021), and AttnLRP adapts this family of rules to attention blocks in Transformers Achtibat et al. (2024). Grad-CAM Selvaraju et al. (2017) localizes class evidence using a single-point spatial average of gradients, $\alpha_k^c = Z^{-1} \sum_{i,j} \partial Y^c / \partial \mathbf{A}_{i,j}^k$, while Neuron Conductance provides path-integrated completeness accounting over internal activation coordinates.

At the level of internal computational connections, EAP-IG attributes a task loss to graph edges Hanna et al. (2024). For an edge $u \to v$, its score has the form $(z_u' - z_u)^\top \overline{\nabla}_{a_v} L$: the change from the clean sender output to the corrupted sender output is paired with the loss gradient at the receiver, averaged along an interpolation path from corrupted to clean inputs. Conductance integrates the product of an output sensitivity and the activation coordinate's path velocity, $(\partial g/\partial h_j)(\partial h_j/\partial \alpha)$, and thereby decomposes the path-wise output difference. EAP-IG gives a first-order approximation to the loss change caused by patching an edge, and Grad-CAM supplies feature-map localization. These methods provide complementary attribution objects: activation coordinates for Conductance, feature maps for Grad-CAM, and computational edges for EAP-IG.

Mechanism-Level Macroscopic Indicators Beyond localizing individual activation coordinates or connections, prior work constructs quantitative summaries of internal organization. Layer-wise and pixel-wise information discarding measures how much input information a representation loses and makes comparisons across layers Ma et al. (2022). ACDC extracts task-relevant circuit edges by automated causal intervention Conmy et al. (2023), while recent benchmarks standardize circuit and causal-variable localization with compactness-aware scores Mueller et al. (2025). Parametric knowledge traces assess whether unlearning removes internally encoded concepts Hong et al. (2025), and Circuit-guided Unlearning Difficulty converts EAP-IG circuit similarities into a continuous, sample-level difficulty score Cheng et al. (2026). These quantities target representation information, circuit recovery, concept probes, or task-specific reference circuits, and circuit-faithfulness scores can vary substantially with the ablation design Miller et al. (2024). Attribution Vergence complements them with a summary of target-conditioned attribution organization at an aggregation node, using a common measurement rule instantiated through architecture-specific coordinate mappings.

Computation-Graph Cost Accounting Reverse-mode automatic differentiation obtains adjoints for intermediate variables of a scalar computation graph in a shared reverse sweep Baydin et al. (2018); the classical Baur–Strassen result gives a constant-overhead precedent for gradients of arithmetic straight-line programs Baur & Strassen (1983). Applying this shared-sweep principle at each attribution-path state, our conditional complexity statement additionally charges each route-indexed attribution materialization volume to its native aggregation-node work.

C Detailed Mathematical Proofs of Attribution Vergence

Figure C.1 summarizes the proof dependencies. Solid arrows denote direct dependencies, and dashed arrows denote conceptual grounding.

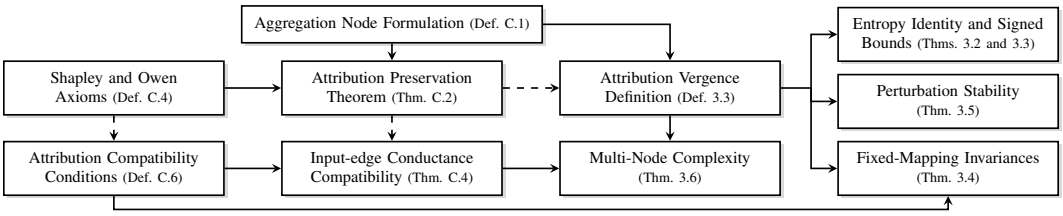

Figure C.1: Attribution closure grounds the measurement; invariances, stability, and complexity characterize its mathematical and computational behavior.

This appendix proves the Aggregation Node Formulation used in the main text. Each field entry $\mathbf{S}_{y,z}$ is assigned to the input edge $e_{y,z}$ jointly specified by the routing-weight input $w_{y,z}$ and the feature-tensor input $\mathbf{X}_{y,z}$. Each column $\mathbf{S}_{:,z}$ collects the coordinatewise assignments associated with route z.

C.1 Pre-Aggregation Attribution Fields

Throughout this appendix, $F = \rho \circ f$ is a scalar objective function of the neural network's final output. For an internal intervention I, $F(I)$ is evaluated by executing the intervened network under the given external context. The network parameters, external inputs, and stochastic choices are fixed across coalition evaluations; affected downstream operations are recomputed.

Definition C.1 (Computation Graph and Aggregation Node)**.** Fix a scalar value function F, a baseline intervention, and a directed acyclic computation graph $\mathcal{G}[F] = (\mathcal{V}, \mathcal{E})$ with individually identified edges. Each input edge identifies a distinct use of an upstream tensor. An intervention changes the value supplied through that edge, while other uses of the tensor remain unchanged. Distinct input edges remain distinct when their values coincide or share storage or tensor views. Let $a \in \mathcal{V}$ be a tensor-valued multiply–add aggregation node, let $\mathcal{Y} = \{1, \dots, d\}$ be its unreduced output coordinates, and let $\mathcal{Z} = \{1, \dots, K\}$ index its designated computational routes. For every $(y, z) \in \mathcal{Y} \times \mathcal{Z}$, the node receives a routing-weight input $w_{y,z}$ and a feature-tensor input $\mathbf{X}_{y,z}$, carried by the incoming graph edges $e^w_{y,z}$ and $e^X_{y,z}$, respectively. For every route-indexed coordinate (y, z), the Aggregation Node Formulation defines an aggregation-node input edge $e_{y,z}$, jointly specified by

a routing-weight input and a feature-tensor input. Set $e_{y,z} := (e^w_{y,z}, e^X_{y,z})$. Multiplication is an internal primitive of the node, while $e^w_{y,z}$ and $e^X_{y,z}$ remain its constituent incoming graph edges. Their association with the unreduced and reduced coordinates defines the node's coordinate mapping and its edge-level granularity. For every network input ξ, the node computes

$$\mathbf{O}_y(\xi) = \sum_{z=1}^{K} w_{y,z}(\xi)\mathbf{X}_{y,z}(\xi), \qquad y \in \mathcal{Y}. \tag{16}$$

Each output coordinate $\mathbf{O}_y$ corresponds to one logical output edge e^O_y; all downstream uses receive the value assigned to that edge. The reduction acts on z while preserving y. The routing weights may depend jointly on the complete network input ξ, so the definition includes dynamic and coupled routing. Reuse of an upstream tensor is represented by distinct incoming graph edges at the aggregation node and therefore by distinct input edges, as illustrated in Figure C.2.

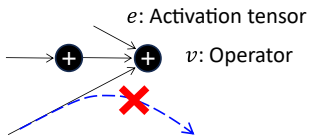

Figure C.2: We assign distinct attribution fields to the input edges of the aggregation nodes. The single-exit structure of graph edges supports the properties of the attribution node.

Definition C.2 (Pre-Aggregation Input-Edge Attribution Assignment)**.** Fix an aggregation node from Definition C.1. A route-indexed pre-aggregation attribution field $\mathbf{S} \in \mathbb{R}^{d\times K}$ is admissibly assigned when every field entry is assigned to the corresponding input edge:

$$\mathbf{S}_{y,z} \longleftrightarrow e_{y,z} \equiv \big(e^w_{y,z}, e^X_{y,z}\big) \equiv \big(w_{y,z}, \mathbf{X}_{y,z}\big), \qquad (y,z) \in \mathcal{Y}\times\mathcal{Z}. \tag{17}$$

Thus $\mathbf{S}_{y,z}$ is the signed input-edge attribution of $e_{y,z}$ and jointly accounts for its routing-weight and feature-tensor inputs at coordinate pair (y,z). Distinct incoming graph edges specify distinct input edges and receive distinct field entries even when they are views of the same upstream tensor.

The associated attribution-space Aggregation Operator reduces the routing coordinate:

$$\mathbb{A}[\mathbf{S}]_y = \sum_{z=1}^{K} \mathbf{S}_{y,z}, \qquad \mathbb{A}[\mathbf{S}] = \sum_{z=1}^{K} \mathbf{S}_{:,z} \in \mathbb{R}^d. \tag{18}$$

For $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$, we additionally require $\mathbf{S} \neq 0$ and $\mathbb{A}[\mathbf{S}] \neq 0$ whenever either quantity occurs in a denominator or logarithm.

C.1.1 Input and Output Games

Definition C.3 (Input and Output Attribution Games)**.** Let $\mathcal{Y}$ index all unreduced output coordinates of the selected aggregation nodes, with distinct indices across nodes. Each constituent edge $e^w_{y,z}$ or $e^X_{y,z}$ is a player controlling the value supplied through that edge. The pair $e_{y,z} = (e^w_{y,z}, e^X_{y,z})$ remains the coordinate of the measured attribution field. All controlled input edges are distinct, and other uses of shared tensors remain unchanged. Define

$$\mathcal{N}_y = \{e^w_{y,z}, e^X_{y,z} : z \in \mathcal{Z}\}, \qquad \mathcal{N} = \bigcup_{y\in\mathcal{Y}} \mathcal{N}_y, \qquad \mathcal{P} = \{\mathcal{N}_y : y \in \mathcal{Y}\}. \tag{19}$$

Here $\mathcal{Z} = \{1,\dots,K\}$ is a common route index set. Fix baseline and target states for every input player. For $U \subseteq \mathcal{N}$, let I_U set players in U to target and all others to baseline under a fixed external context. For $s \in \{\text{baseline}, \text{target}\}$, write $\mathbf{O}^s_y = \sum_z w^s_{y,z}\mathbf{X}^s_{y,z}$. For $D \subseteq \mathcal{Y}$, let J_D set the value $\mathbf{O}_y$ on each selected output edge e^O_y to $\mathbf{O}^{\text{target}}_y$ when $y \in D$ and to $\mathbf{O}^{\text{baseline}}_y$ otherwise. Both interventions evaluate the same final network objective. Define

$$v(U) = F(I_U) - F(I_\varnothing), \qquad \bar{v}(D) = F(J_D) - F(J_\varnothing), \tag{20}$$

where $v : 2^{\mathcal{N}} \to \mathbb{R}$ and $\bar{v} : 2^{\mathcal{Y}} \to \mathbb{R}$.

C.1.2 SHAPLEY AND OWEN ATTRIBUTION AXIOMS

Definition C.4 (Shapley and Owen Attribution Axioms)**.** Let $v : 2^{\mathcal{N}} \to \mathbb{R}$ satisfy $v(\varnothing) = 0$. For a partition $\mathcal{P} = \{\mathcal{N}_y : y \in \mathcal{Y}\}$, write $\phi_i(v, \mathcal{P})$ for a player's allocation. For ordinary Shapley attribution, we omit $\mathcal{P}$ and write $\phi_i(v)$. Both attribution schemes satisfy the following requirements.

1. **Efficiency.** $\sum_{i \in \mathcal{N}} \phi_i(v, \mathcal{P}) = v(\mathcal{N})$.
2. **Linearity.** For games on the same player set and fixed partition, $\phi_i(c_1 v + c_2 w, \mathcal{P}) = c_1 \phi_i(v, \mathcal{P}) + c_2 \phi_i(w, \mathcal{P})$.
3. **Null player.** If $v(U \cup \{i\}) = v(U)$ for every $U \subseteq \mathcal{N} \setminus \{i\}$, then $\phi_i(v, \mathcal{P}) = 0$.

Players i, j are symmetric in v when $v(U \cup \{i\}) = v(U \cup \{j\})$ for every $U \subseteq \mathcal{N} \setminus \{i, j\}$. The Shapley rule requires *Symmetry*: all such players receive equal allocations. The Owen rule imposes two symmetry requirements:

1. **Symmetry within unions.** Symmetric players belonging to the same $\mathcal{N}_y$ receive equal allocations.
2. **Coalitional symmetry.** Define the quotient game $v_{\mathcal{P}}(D) = v(\bigcup_{y \in D} \mathcal{N}_y)$ on $\mathcal{Y}$. If y, y' are symmetric players in $v_{\mathcal{P}}$, then

$$\sum_{i \in \mathcal{N}_y} \phi_i(v, \mathcal{P}) = \sum_{i \in \mathcal{N}_{y'}} \phi_i(v, \mathcal{P}). \tag{21}$$

The common requirements with Symmetry characterize the Shapley value $\phi_i(v)$ (Shapley, 1953); with the two symmetry requirements for unions, they characterize the Owen value $\mathrm{Ow}_i(v, \mathcal{P})$ (Owen, 1977). For an unnormalized payoff, the rules apply after subtracting its empty-coalition value.

The Shapley value averages marginal contributions over all player permutations. The Owen value first orders the groups and then the players within each group, uniformly, so each group appears consecutively. Its quotient-game property equates a group's total allocation to its Shapley value in the quotient game (Owen, 1977).

C.1.3 OWEN-TO-SHAPLEY CLOSURE

Graph-based attribution assigns credit to edges and propagates effects through graphical models (Wang et al., 2021b; Singal et al., 2021); attribution propagation also applies to composed model pipelines (Chen et al., 2022). Here the routing partition connects the input allocation to the Shapley allocation of the output game.

Lemma C.1 (Correspondence of Input-Group and Output Games)**.** *For the games in Definition C.3,*

$$v\left(\bigcup_{y \in D} \{e^w_{y,z}, e^X_{y,z} : z \in \mathcal{Z}\}\right) = \bar{v}(D), \qquad D \subseteq \mathcal{Y}. \tag{22}$$

Proof. Fix a selected aggregation node $a \in \mathcal{V}$ and constituent input edges $e^w_{y,z}, e^X_{y,z} \in \mathcal{E}$ entering a. By Definition C.1, an intervention changes only the values supplied through these edges, other uses of the upstream tensors remain unchanged. Since $e^w_{y,z}, e^X_{y,z}$ share a same tail a, every directed path carrying this change to F must pass through a and one of its logical output edges. Since $\mathcal{Y}$ includes all output coordinates of every selected aggregation node, $\{e^O_y : y \in \mathcal{Y}\}$ intersects every path by which the input interventions can affect F.

Under $I_{\bigcup_{y \in D} \mathcal{N}_y}$, all constituent-edge players in each selected group switch together, so both the weight and feature values take their target states. The native aggregation gives

$$\mathbf{O}_y(I_{\bigcup_{y \in D} \mathcal{N}_y}) = \begin{cases} \sum_z w^{\text{target}}_{y,z} \mathbf{X}^{\text{target}}_{y,z} = \mathbf{O}^{\text{target}}_y, & y \in D, \\ \sum_z w^{\text{baseline}}_{y,z} \mathbf{X}^{\text{baseline}}_{y,z} = \mathbf{O}^{\text{baseline}}_y, & y \notin D. \end{cases} \tag{23}$$

Hence $I_{\bigcup_{y\in D}\mathcal{N}_y}$ and J_D assign identical values to every e_y^O. These edges intercept all effects of the input interventions, Under the same fixed game configuration, evaluating the remaining operations in $\mathcal{G}[F]$ therefore gives $F(I_{\bigcup_{y\in D}\mathcal{N}_y}) = F(J_D)$. Taking $D = \varnothing$ also gives $F(I_\varnothing) = F(J_\varnothing)$; subtracting these equalities proves $v(\bigcup_{y\in D}\mathcal{N}_y) = \bar{v}(D)$. □

Theorem C.2 (Owen-to-Shapley Closure under Aggregation)**.** *Under Lemma C.1, define the attribution of each paired input edge by*

$$\mathbf{S}_{y,z} = \mathrm{Ow}_{e^w_{y,z}}(v,\mathcal{P}) + \mathrm{Ow}_{e^X_{y,z}}(v,\mathcal{P}). \tag{24}$$

Then

$$\mathbb{A}[\mathbf{S}]_y = \sum_{z\in\mathcal{Z}} \mathbf{S}_{y,z} = \phi_y(\bar{v}), \qquad y \in \mathcal{Y}. \tag{25}$$

Proof. By the quotient-game property of the Owen value (Owen, 1977), $\sum_{e\in\mathcal{N}_y} \mathrm{Ow}_e(v,\mathcal{P}) = \phi_y(v_\mathcal{P})$ Since $v_\mathcal{P} = \bar{v}$ by Lemma C.1, we obtain

$$\mathbb{A}[\mathbf{S}]_y = \sum_{z\in\mathcal{Z}} \mathbf{S}_{y,z} = \sum_{e\in\mathcal{N}_y} \mathrm{Ow}_e(v,\mathcal{P}) = \phi_y(\bar{v}). \tag{26}$$

□

C.1.4 Attribution Compatibility Conditions

We formulate two local requirements alongside Linearity and Null response. Symmetry is restricted to attribution problems with matched interventions, references, routing groups, and paths. Aggregation consistency requires incoming credits to recover independently specified output attribution across the aggregation operation. This local conservation requirement concerns a different comparison from Efficiency, which fixes the total allocation to the payoff difference of a complete game. First we define *Intervention Equivalence* for the restricted Symmetry definition.

Definition C.5 (Intervention Equivalence of Attribution Problems)**.** Consider attribution problems on $\mathcal{G}[F]$ and $\mathcal{G}'[F']$, each with specified input edges, a baseline, target states, and an intervention context. A coordinate is null if changing the values carried by its input edges leaves the objective unchanged throughout the specified intervention domain, or if its baseline and target input-edge values coincide and remain constant along any specified attribution path.

After omitting any added or removed null coordinates, let τ be a bijection between the specified aggregation-node input edges. The relation

$$\tau(e_{y,z}) = e'_{y',z'}, \qquad e_{y,z} = (e^w_{y,z}, e^X_{y,z}), \quad e'_{y',z'} = (e'^w_{y',z'}, e'^X_{y',z'}) \tag{27}$$

defines the correspondence between attribution coordinates (y,z) and (y',z') through their assigned graph edges. It matches the two constituent input edges in the displayed order. Each complete specified player set and its intervention context are matched together. The induced bijection $\widehat{\tau}$ on constituent edges satisfies $\widehat{\tau}(e^w_{y,z}) = e'^w_{y',z'}$ and $\widehat{\tau}(e^X_{y,z}) = e'^X_{y',z'}$.

The two attribution problems are *intervention-equivalent under* τ when their allowed interventions correspond in both directions, their baseline and target assignments match, and every corresponding pair I, I' satisfies

$$\begin{gathered}\big(I'(e'^w_{y',z'}), I'(e'^X_{y',z'})\big) = \big(I(e^w_{y,z}), I(e^X_{y,z})\big),\\ F'(I') = F(I).\end{gathered} \tag{28}$$

For games with independent constituent-edge players, the allowed interventions include every coalition of these players, including mixed baseline–target states within each pair. Here the objective F is evaluated in the corresponding specified context, including inputs ξ and labels. When attribution paths γ, γ' are specified, the corresponding input-edge values also agree at every path parameter α. For differentiable path attribution, the intervention domain includes independent variations of the values on matched input edges in a neighborhood of each path state, with the remaining context matched at that state.

An *intervention symmetry* is such a correspondence from an attribution problem to itself. This relation concerns the specified graph edges and their intervention responses and permits different

intermediate graph operations. When routing groups are compared, their index sets include only nonempty groups of matched coordinates.

Definition C.6 (Attribution Compatibility Conditions)**.** For fixed reference choices, routing groups, and a path when applicable, consider paired rules $F \mapsto (\mathbf{S}^F, \mathbf{S}^F_{\text{post}})$ for the same scalar network-output objective F. The input field assigns credit to specified edges; the output field is independently specified by the corresponding attribution rule on output coordinates. The pair is compatible with aggregation when:

1. **Linearity.** For downstream objectives F_1, F_2 and scalars c_1, c_2,
$$\mathbf{S}^{c_1F_1+c_2F_2} = c_1\mathbf{S}^{F_1} + c_2\mathbf{S}^{F_2}. \tag{29}$$

2. **Dummy/Null response.** A coordinate that is null under Definition C.5 receives zero attribution: $\mathbf{S}^F_{y,z} = 0$.

3. **Reduced symmetry and consistency.** For two attribution problems that are intervention-equivalent under τ in Definition C.5, let their corresponding routing groups be
$$\mathcal{N}_y = \{e^w_{y,z}, e^X_{y,z} : z \in \mathcal{Z}\}, \qquad \mathcal{N}'_{y'} = \{e'^w_{y',z'}, e'^X_{y',z'} : z' \in \mathcal{Z}'\}. \tag{30}$$
If
$$\exists \text{ a bijection } \sigma : \mathcal{Y} \to \mathcal{Y}' \quad \text{s.t.} \quad \{\widehat{\tau}(e) : e \in \mathcal{N}_y\} = \mathcal{N}'_{\sigma(y)} \quad \forall y \in \mathcal{Y}, \tag{31}$$
then, for every pair $\tau(e_{y,z}) = e'_{y',z'}$,
$$\mathbf{S}^{F'}_{y',z'} = \mathbf{S}^F_{y,z}. \tag{32}$$

4. **Aggregation consistency.** Under the specified aggregation mapping and matched reference semantics,
$$\mathbb{A}[\mathbf{S}^F]_y = \mathbf{S}^F_{\text{post},y}. \tag{33}$$

Classical Shapley symmetry compares coalition marginal contributions. Within a single attribution problem, reduced symmetry additionally matches the intervention responses, reference states, routing groups, and any attribution paths. These stronger matching conditions restrict when equal attribution is required, weakening the symmetry constraint on the attribution rule. Consistency extends the equality to two matched problems. The graph and attribution protocols are implicit in the paired fields. Global completeness additionally requires the specified coordinates to account for the full objective change.

Theorem C.3 (Compatibility of Shapley and Owen Attribution Fields)**.** *Under the conditions of Lemma C.1, define the input and output attribution fields by*

$$\begin{aligned} \mathbf{S}_{y,z} &= \mathrm{Ow}_{e^w_{y,z}}(v, \mathcal{P}) + \mathrm{Ow}_{e^X_{y,z}}(v, \mathcal{P}), \\ \mathbf{S}_{\text{post},y} &= \phi_y(\bar{v}). \end{aligned} \tag{34}$$

The pair satisfies all four Attribution Compatibility Conditions in Definition C.6. Replacing the two input Owen allocations with ordinary Shapley allocations also yields an input field satisfying the first three conditions.

Proof. The induced coalition payoff is linear in the downstream objective, so Linearity of either the Owen or ordinary Shapley rule, followed by addition of the two constituent allocations, gives Linearity of the field. If a paired coordinate is null throughout the independent intervention domain, varying either constituent value leaves the objective unchanged. If both values are fixed at their baseline states, their inclusion likewise changes no coalition payoff. Each constituent is therefore a null player, and both allocations vanish.

For consistency, first match problems with no added or removed null players. Intervention equivalence gives identical payoffs for corresponding coalitions of constituent edges. The induced bijection $\widehat{\tau}$ maps player permutations bijectively and preserves every corresponding marginal contribution. Averaging over all permutations proves Shapley consistency. For Owen attribution, σ also matches the groups. Thus $\widehat{\tau}$ maps the permutations with consecutive groups bijectively, preserving their probabilities and marginal contributions. Averaging gives equal corresponding constituent allocations for both rules. Adding the two allocations in each matched pair proves consistency of $\mathbf{S}$.

Adding a null paired coordinate introduces two null players. Adding null players preserves the uniform relative order of the original players in the Shapley average. In the Owen average, adding null players within a group preserves the uniform relative order of its original members, and adding a group of null players preserves the uniform relative order of the original groups. Null players contribute no marginal change in either case. Removing null players reverses these arguments. Hence both rules satisfy Eq. 32 for the specified correspondences, including null extensions.

For the stated Owen input field, Theorem C.2 gives $\mathbb{A}[\mathbf{S}]_y = \phi_y(\bar{v}) = \mathbf{S}_{\mathrm{post},y}$. This establishes Aggregation consistency and hence all four conditions for the paired fields. □

C.1.5 Conductance: Compatibility and Local Closure

Theorem C.4 (Compatibility of the Input-Edge Neuron Conductance Field)**.** *Fix an aggregation node a in the directed acyclic graph of Definition C.1, a scalar network-output objective F, and a differentiable attribution path $\boldsymbol{x}(\alpha) = \gamma(\alpha)$ with integrable attribution derivatives. Let $q_{y,z} = w_{y,z}\mathbf{X}_{y,z}$ denote the local multiplication output entering $\mathbf{O}_y = \sum_z q_{y,z}$. Define the paired attribution fields by*

$$\begin{aligned} \mathbf{S}^F_{y,z} &= \int_0^1 \frac{\partial F}{\partial q_{y,z}} \frac{\mathrm{d}q_{y,z}}{\mathrm{d}\alpha}\,\mathrm{d}\alpha = \int_0^1 \frac{\partial F}{\partial \mathbf{O}_y} \frac{\mathrm{d}(w_{y,z}\mathbf{X}_{y,z})}{\mathrm{d}\alpha}\,\mathrm{d}\alpha, \\ \mathbf{S}^F_{\mathrm{post},y} &= \mathrm{Cond}(F;\gamma,\mathbf{O}_y) = \int_0^1 \frac{\partial F}{\partial \mathbf{O}_y} \frac{\mathrm{d}\mathbf{O}_y}{\mathrm{d}\alpha}\,\mathrm{d}\alpha. \end{aligned} \tag{35}$$

All quantities are evaluated along γ. The derivative with respect to $q_{y,z}$ concerns its specific internal use in the aggregation sum. The input field decomposes Neuron Conductance (Dhamdhere et al., 2019) across the specified input edges. The pair satisfies all four conditions in Definition C.6.

Proof. Within node a, the multiplication output $q_{y,z}$ enters only the sum defining $\mathbf{O}_y$. Its effects on F pass through the logical output edge e^O_y, and acyclicity excludes feedback into a. Holding other local route coordinates fixed, the chain rule gives

$$\frac{\partial F}{\partial q_{y,z}} = \frac{\partial F}{\partial \mathbf{O}_y}\frac{\partial \mathbf{O}_y}{\partial q_{y,z}} = \frac{\partial F}{\partial \mathbf{O}_y}. \tag{36}$$

This establishes the two expressions for $\mathbf{S}^F_{y,z}$ in Eq. 35.

For Linearity, differentiation is linear in the downstream objective:

$$\frac{\partial(c_1F_1 + c_2F_2)}{\partial \mathbf{O}_y} = c_1\frac{\partial F_1}{\partial \mathbf{O}_y} + c_2\frac{\partial F_2}{\partial \mathbf{O}_y}. \tag{37}$$

Substituting this identity into the path integral gives $\mathbf{S}^{c_1F_1+c_2F_2} = c_1\mathbf{S}^{F_1} + c_2\mathbf{S}^{F_2}$.

For Dummy/Null response, view the values on the two constituent input edges as independent intervention coordinates. The local chain rule gives

$$\frac{\partial F}{\partial \mathbf{O}_y}\frac{\mathrm{d}}{\mathrm{d}\alpha}[w_{y,z}\mathbf{X}_{y,z}] = \frac{\partial F}{\partial w_{y,z}}\frac{\mathrm{d}w_{y,z}}{\mathrm{d}\alpha} + \frac{\partial F}{\partial \mathbf{X}_{y,z}}\frac{\mathrm{d}\mathbf{X}_{y,z}}{\mathrm{d}\alpha}. \tag{38}$$

The derivatives on the right are with respect to the independently addressable incoming graph edges. If their states cannot affect the objective throughout the intervention domain, both partial derivatives vanish along the path. If their states are fixed, both path derivatives vanish. In either case the Conductance integral is zero.

For Reduced symmetry and consistency, suppose $\tau(e_{y,z}) = e'_{y',z'}$. Intervention equivalence gives identical objective functions of the corresponding input-edge values in a neighborhood of each matched path state. Their partial derivatives agree, and the matched input-edge paths have identical derivatives. Applying Eq. 38 on each graph therefore gives

$$\frac{\partial F'}{\partial \mathbf{O}'_{y'}}\frac{\mathrm{d}}{\mathrm{d}\alpha}[w'_{y',z'}\mathbf{X}'_{y',z'}] = \frac{\partial F}{\partial \mathbf{O}_y}\frac{\mathrm{d}}{\mathrm{d}\alpha}[w_{y,z}\mathbf{X}_{y,z}]. \tag{39}$$

Integrating over α yields $\mathbf{S}^{F'}_{y',z'} = \mathbf{S}^F_{y,z}$, as required. An intervention symmetry within one problem gives equality of the exchanged entries as a special case.

For Aggregation consistency, finite summation along the common path gives

$$\begin{aligned}\mathbb{A}[\mathbf{S}^F]_y &= \int_0^1 \frac{\partial F}{\partial \mathbf{O}_y} \frac{\mathrm{d}}{\mathrm{d}\alpha} \left[\sum_z w_{y,z} \mathbf{X}_{y,z} \right] \mathrm{d}\alpha \\ &= \int_0^1 \frac{\partial F}{\partial \mathbf{O}_y} \frac{\mathrm{d}\mathbf{O}_y}{\mathrm{d}\alpha} \mathrm{d}\alpha = \mathbf{S}^F_{\mathrm{post},y}.\end{aligned} \tag{40}$$

For dynamic routing, the product derivative includes both $w_{y,z}\,\mathrm{d}\mathbf{X}_{y,z}$ and $\mathbf{X}_{y,z}\,\mathrm{d}w_{y,z}$. Thus the paired fields satisfy all four conditions. □

Corollary C.5 (Local Neuron Conductance Closure)**.** *Under Theorem C.4,*

$$\mathbb{A}[\mathbf{S}^F]_y = \mathrm{Cond}(F; \gamma, \mathbf{O}_y). \tag{41}$$

This is the Aggregation consistency established in the theorem.

The preceding results specify compatible attribution constructions. We now describe how the common aggregation coordinates represent native modules.

C.1.6 Architecture-Specific Coordinate Mappings

Definition C.7 (Architecture Coordinate-Mapping Rules)**.** The architecture-specific coordinate mappings use the following node and coordinate assignments.

1. **Attention.** For a fixed head h, every output token q defines one aggregation node. Within that node, $y = r$ indexes the output/value coordinate, the routing coordinate $z = k$ indexes the source-token position, and $\mathbf{S}^{(q,h)}_{r,k}$ is assigned to the input edge jointly specified by $\mathbf{A}^{(h)}_{q,k}$ and $\mathbf{V}^{(h)}_{k,r}$.

2. **Convolution.** Every output position u defines one aggregation node. Within that node, $y = o$ indexes the output channel, the routing coordinate $z = (c, \delta)$ indexes the input channel c and kernel offset δ, and $\mathbf{S}^{(u)}_{o,(c,\delta)}$ is assigned to the input edge jointly specified by $\mathbf{W}_{o,c,\delta}$ and $\mathbf{X}_{c,u+\delta}$.

3. **Position-wise MLP.** Every token position t of an affine projection defines one node. The unreduced coordinate is the output activation coordinate $y = o$, the route coordinate is the input activation coordinate $z = i$, and $\mathbf{S}^{(t)}_{o,i}$ is assigned to the input edge jointly specified by $\mathbf{W}^{\mathrm{mlp}}_{o,i}$ and $\mathbf{H}_{t,i}$. The bias is fixed context.

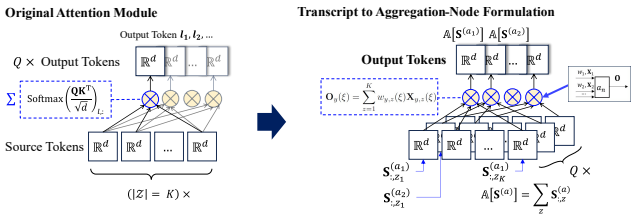

Figure C.3: Illustration of the mapping from the view of the classical Attention module and aggregation node on the computation graph. Each source token is decomposed into Q attribution fields, and the attention module is transcribed into Q aggregation nodes.

Any additive bias is treated as a constant context independent of inputs and interventions, and is excluded from the route set. The routed aggregation therefore gives the corresponding output minus this fixed bias. Regarding routing dynamics, parameter weights in convolution and position-wise MLP projections remain fixed during evaluation ($\frac{\mathrm{d}w}{\mathrm{d}\alpha} = 0$), so only activation inputs vary along the path. In contrast, multi-head self-attention employs input-dependent dynamic routing, where attention routing weights $w_{y,z}(\alpha)$ vary simultaneously with value activations $\mathbf{X}_{y,z}(\alpha)$ along the path, with the attribution integrand natively accounting for both product-rule derivative components.

Theorem C.6 (Architecture Modules as Aggregation-Node Compositions)**.** *Under Definition C.7, fixed-head attention, convolution, and position-wise affine MLP modules are compositions of aggregation nodes satisfying Definitions C.1 and C.2. At every constituent node, the induced attribution-space operator is* $\mathbb{A}[\mathbf{S}]_y = \sum_z \mathbf{S}_{y,z}$.

Proof. We apply the coordinate assignments in Definition C.7 to the native operations below. In each case, the pair $(w_{y,z}, \mathbf{X}_{y,z})$ specifies the input edge carrying attribution $\mathbf{S}_{y,z}$, and $\mathbb{A}[\mathbf{S}]_y = \sum_z \mathbf{S}_{y,z}$ follows from Definition C.2. Distinct uses of a shared physical tensor remain distinct incoming graph edges under this assignment.

Attention. Fix a head h with Q output tokens, L source tokens, and value width d_h. For input ξ, each output token q computes

$$\mathrm{Attn}_q^{(h)}\big(\mathbf{A}_{q,:}^{(h)}(\xi), \mathbf{V}^{(h)}(\xi)\big)_r = \mathbf{O}_{q,r}^{(h)}(\xi) = \sum_{k=1}^{L} \mathbf{A}_{q,k}^{(h)}(\xi)\mathbf{V}_{k,r}^{(h)}(\xi), \qquad r = 1, \dots, d_h. \tag{42}$$

Taking $y = r$ and $z = k$, set

$$\mathbf{X}_{r,k}^{(q,h)}(\xi) = \mathbf{V}_{k,r}^{(h)}(\xi), \qquad w_{r,k}^{(q,h)}(\xi) = \mathbf{A}_{q,k}^{(h)}(\xi). \tag{43}$$

This reduces the source-token coordinate and leaves the value coordinate unreduced. The input dependence of attention weights, including their coupled softmax computation, remains part of the dynamic routing.

Convolution. For input channels $c = 1, \dots, C_{\mathrm{in}}$, output channels $o = 1, \dots, C_{\mathrm{out}}$, and kernel offsets $\delta \in \Delta$, each output position u computes

$$\mathrm{Conv}_u(\mathbf{W}, \mathbf{X})_o = \mathbf{Y}_{o,u} = \boldsymbol{b}_o^{\mathrm{conv}} + \sum_{c=1}^{C_{\mathrm{in}}} \sum_{\delta \in \Delta} \mathbf{W}_{o,c,\delta}\mathbf{X}_{c,u+\delta}. \tag{44}$$

Taking $y = o$ and $z = (c, \delta)$, set

$$\mathbf{X}_{o,(c,\delta)}^{(u)} = \mathbf{X}_{c,u+\delta}, \qquad w_{o,(c,\delta)}^{(u)} = \mathbf{W}_{o,c,\delta}. \tag{45}$$

Subtracting the fixed bias gives the required aggregation with static routing weights. The node index u distinguishes uses of overlapping input patches.

MLP. For a position-wise affine projection with input width d_{in} and output width d_{out}, each token t computes

$$\mathrm{MLP}_t\big(\mathbf{W}^{\mathrm{mlp}}, \mathbf{H}_t(\xi)\big)_o = \mathbf{M}_{t,o}(\xi) = \boldsymbol{b}_o^{\mathrm{mlp}} + \sum_{i=1}^{d_{\mathrm{in}}} \mathbf{W}_{o,i}^{\mathrm{mlp}}\mathbf{H}_{t,i}(\xi), \qquad o = 1, \dots, d_{\mathrm{out}}. \tag{46}$$

Taking $y = o$ and $z = i$, set

$$\mathbf{X}_{o,i}^{(t)}(\xi) = \mathbf{H}_{t,i}(\xi), \qquad w_{o,i}^{(t)} = \mathbf{W}_{o,i}^{\mathrm{mlp}}. \tag{47}$$

Subtracting the fixed bias again gives the required aggregation. Collecting nodes over (h, q), u, or t reconstructs the corresponding module output. The fixed-bias and independent-batch conventions are those of Definition C.7. □

C.2 Representation Invariances under a Fixed Coordinate Mapping

Aggregation consistency grounds the interpretation of the paired fields. The invariance results below depend on the first three compatibility conditions and the stated transformations of those fields. Linearity connects objective scaling to attribution scaling. For an extension by null coordinates that preserves intervention equivalence, Reduced symmetry and consistency preserve the original entries, while Dummy/Null response gives zero on added entries. The norm identities below then establish the corresponding measurement invariances.

Lemma C.7 (Aggregation Compatibility under Matched Routing Groups)**.** *Suppose $F \mapsto \mathbf{S}^F$ satisfies the first three conditions in Definition C.6. Aggregation preserves Linearity, zero contributions from null entries, and consistency between corresponding unreduced-coordinate groups. In particular, for a correspondence as in Definition C.6,*

$$\mathbb{A}'[\mathbf{S}^{F'}]_{\sigma(y)} = \mathbb{A}[\mathbf{S}^F]_y. \tag{48}$$

Proof. Linearity follows by distributing the finite aggregation sum over a linear combination of fields. Null entries contribute zero. Under τ, each summand in an unreduced-coordinate group matches one in its corresponding group, with any extra null entries equal to zero. Reindexing therefore gives

$$\mathbb{A}'[\mathbf{S}^{F'}]_{\sigma(y)} = \sum_{z' \in \mathcal{Z}'} \mathbf{S}^{F'}_{\sigma(y),z'} = \sum_{z \in \mathcal{Z}} \mathbf{S}^F_{y,z} = \mathbb{A}[\mathbf{S}^F]_y. \tag{49}$$

□

Lemma C.8 (Scale Invariance)**.** $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ *is invariant when all $\mathbf{S}_{:,z}$ are multiplied by a common nonzero scalar. For an attribution rule satisfying Linearity in Definition C.6, this includes rescaling the downstream objective by that scalar.*

Proof. When the scaling is induced by $F \mapsto cF$, Linearity gives $\mathbf{S}^{cF} = c\mathbf{S}^F$. It therefore suffices to verify the following algebraic field-scaling identity. For $c \neq 0$, linearity of the aggregation sum gives

$$\mathbb{A}[c\mathbf{S}] = c\mathbb{A}[\mathbf{S}]. \tag{50}$$

Both effective-dimensionality values are homogeneous of degree zero:

$$\mathrm{D}(c\mathbf{S}) = \mathrm{D}(\mathbf{S}), \qquad \mathrm{D}(c\mathbb{A}[\mathbf{S}]) = \mathrm{D}(\mathbb{A}[\mathbf{S}]). \tag{51}$$

Their logarithmic ratio is therefore unchanged. □

Lemma C.9 (Padding Immunity)**.** *Extending a fixed field by zero routes or zero unreduced coordinates, or removing such coordinates, leaves $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ unchanged. For a rule satisfying the first three conditions in Definition C.6, this includes graph extensions by null coordinates that preserve intervention equivalence and the original aggregation assignments.*

Proof. For a graph extension satisfying the stated conditions, intervention consistency preserves the original entries and Dummy/Null response assigns zero to the added entries. Thus $\mathbf{S}^+$ is $\mathbf{S}$ with zero padding, and $\mathbb{A}^+[\mathbf{S}^+]$ is $\mathbb{A}[\mathbf{S}]$ with at most zero padding. For $p \in \{1, 2\}$,

$$\|\mathbf{S}^+\|_p = \|\mathbf{S}\|_p, \qquad \|\mathbb{A}^+[\mathbf{S}^+]\|_p = \|\mathbb{A}[\mathbf{S}]\|_p. \tag{52}$$

Both effective dimensionalities and their logarithmic ratio are unchanged. Removing zero coordinates reverses this operation. □

Lemma C.10 (Duplication Invariance)**.** *For $r \in \mathbb{N}_+$ and a common nonzero scale a, let C_r^a duplicate every unreduced coordinate r times. Suppose the transformed pre-aggregation attribution field and aggregation operator satisfy*

$$\mathbf{S}' = C_r^a \mathbf{S}, \qquad \mathbb{A}'[\mathbf{S}'] = C_r^a \mathbb{A}[\mathbf{S}]. \tag{53}$$

Then

$$\mathbb{V}_{\mathbb{A}'}(\mathbf{S}') = \mathbb{V}_{\mathbb{A}}(\mathbf{S}). \tag{54}$$

Proof. For any nonzero vector x, direct summation gives

$$\|C_r^a x\|_1 = r|a|\|x\|_1, \qquad \|C_r^a x\|_2^2 = ra^2\|x\|_2^2, \qquad \mathrm{D}(C_r^a x) = r\,\mathrm{D}(x). \tag{55}$$

The same identities apply to the flattened field $\mathbf{S}$. The two matched transformation conditions therefore give

$$\mathbb{V}_{\mathbb{A}'}(\mathbf{S}') = \log_2 \frac{r\,\mathrm{D}(\mathbf{S})}{r\,\mathrm{D}(\mathbb{A}[\mathbf{S}])} = \mathbb{V}_{\mathbb{A}}(\mathbf{S}). \tag{56}$$

□

One graph construction realizing this field transformation makes r independent copies of the original attribution problem, with matched baselines, paths, and aggregation assignments. Write $F^{(j)}$ for the objective acting on copy j, and set $F' = a\sum_{j=1}^{r} F^{(j)}$. For each $F^{(j)}$, the other copies have null attribution coordinates. Intervention consistency preserves the original field in copy j, and Linearity gives $a\mathbf{S}$ in each copy for the combined objective. When the copies remain distinct unreduced coordinates, this yields $\mathbf{S}' = C_r^a\mathbf{S}$ and $\mathbb{A}'[\mathbf{S}'] = C_r^a\mathbb{A}[\mathbf{S}]$. Choosing $a = 1/r$ also preserves the objective value along the copied paths.

Theorem C.11 (Representation Invariances under a Fixed Coordinate Mapping)**.** *For a rule satisfying the first three conditions in Definition C.6, $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ is invariant under objective scaling and under null-coordinate graph extensions that preserve intervention equivalence and the original aggregation assignments. It is also algebraically invariant under zero-field padding and the matched duplication in Lemma C.10.*

Proof. The three conclusions are Lemmas C.8, C.9, and C.10. □

C.3 $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ Complexity over Multiple Aggregation Nodes

Consider distinct aggregation nodes $a_1, \ldots, a_{N_{\text{node}}} \in \mathcal{V}$, where a_i has feature dimension d_i and routing size K_i. Evaluating their attribution fields along a common path requires computations across both nodes and integration steps (Figure C.2). The central challenge is to share the required computations while controlling the storage of intermediate tensors and attribution fields.

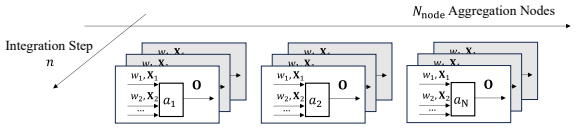

Figure C.4: Computing attribution fields across aggregation nodes and integration steps. The same underlying weight w or feature tensor $\mathbf{X}$ may be reused by multiple aggregation nodes a, but each node requires a separate attribution field. This creates different storage requirements for the training process and the node-specific attribution fields.

Our complexity analysis resuse the computation-graph model on $\mathcal{G}$ that counts scalar operations and peak tensor storage during a forward–reverse execution. We use the standard reverse-mode automatic differentiation on a computation graph Baydin et al. (2018). Assumption C.1 specifies how graph operations correspond to the actual computation and relates the cost of attribution updates to the forward–reverse execution cost.

Definition C.8 (Forward–Reverse Computation-Graph Execution)**.** For a network input ξ, one forward–reverse execution of $\mathcal{G}[F] = (\mathcal{V}, \mathcal{E})$ computes the scalar objective and the forward edge tensors, and returns their reverse-mode adjoints:

$$(F(\xi), \{\mathbf{X}_e(\xi)\}_{e\in\mathcal{E}}, \{\nabla_{\mathbf{X}_e} F(\xi)\}_{e\in\mathcal{E}}) \,. \tag{57}$$

Let $c(v) \geq 0$ be the scalar work of node v. The work and peak scalar storage of this execution are

$$T_{\text{step}} = \Theta\left(\sum_{v\in\mathcal{V}} c(v)\right), \qquad \text{Mem}_{\text{step}} = \max_{\tau} \sum_{\substack{\mathbf{X} \text{ live} \\ \text{at } \tau}} \text{size}(\mathbf{X}). \tag{58}$$

The tensor families in Eq. 57 form the logical execution trace; the peak-storage term in Eq. 58 counts its simultaneously live physical tensors. When the graph is represented explicitly, its node and edge records require structural storage

$$\text{Mem}_{\text{graph}} = \Theta(|\mathcal{V}| + |\mathcal{E}|) \,. \tag{59}$$

This structural description is separate from the live-tensor storage Mem_{step}; the latter is determined by the execution frontier in Eq. 58.

Assumption C.1 (Execution-Faithful Computation Graph)**.** The nodes and edge tensors of $\mathcal{G}[F] = (\mathcal{V}, \mathcal{E})$ coincide, up to constant-cost partitions and tensor views, with the operations and tensor uses in the actual forward and reverse-mode execution. The graph is reduced relative to the forward map, the required reverse-mode adjoints, and the specified coordinate mapping: deleting or identifying a remaining node or edge changes at least one of these objects. Every remaining edge is accessed by its target operation in the forward or reverse execution. Consequently, the execution admits the edge-access charging condition

$$|\mathcal{E}| = O\left(\sum_{v\in\mathcal{V}} c(v)\right) = O(T_{\text{step}}). \tag{60}$$

For the probed aggregation nodes, physical execution work is assigned through disjoint partitions or a uniformly bounded-overlap cover: there is a constant $\kappa = O(1)$, independent of n, N_{node}, and the model dimensions, such that

$$\sum_{i=1}^{N_{\text{node}}} c(a_i) \le \kappa \sum_{v\in\mathcal{V}} c(v). \tag{61}$$

Uniformly over the model dimensions,

$$d_i K_i = O(c(a_i)), \qquad i = 1, \dots, N_{\text{node}}. \tag{62}$$

The execution in Definition C.8 supplies the output-side gradients, and forming all route-indexed attribution updates at node a_i costs $O(d_i K_i)$. The scalar volume of any attribution-required subset of one forward–reverse execution trace is $O(T_{\text{step}})$.

Theorem C.12 ($\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ Complexity over Multiple Aggregation Nodes)**.** *Under Assumption C.1, suppose the local physical tensors needed at one path state and every node-local pre-aggregation attribution field occupy $O(\text{Mem}_{\text{step}})$ storage. For $N_{\text{node}} \ge 1$ probed aggregation nodes and the common attribution path discretized into $n \ge 1$ intervals, the discretized statistics can be computed with*

$$\begin{aligned} T_{\text{total}} &= O(nT_{\text{step}}), \\ \text{Mem}_{\text{peak}} &= O\big(\big(1 + \min\{N_{\text{node}}, n\}\big)\text{Mem}_{\text{step}}\big) = O(\min\{N_{\text{node}}, n\}\text{Mem}_{\text{step}})\,. \end{aligned} \tag{63}$$

Proof. Because the a_i are distinct nodes in the execution-faithful graph, Assumption C.1 gives

$$\sum_i d_i K_i = O\left(\sum_i c(a_i)\right) = O\left(\kappa \sum_{v\in\mathcal{V}} c(v)\right) = O(T_{\text{step}}). \tag{64}$$

The second equality uses Eq. 61; disjoint physical partitions are its special case $\kappa = 1$. Equation 60 also gives the coarse work lower bound $T_{\text{step}} = \Omega(|\mathcal{E}|)$: every active graph edge is charged to an operation that consumes it. Graph-description compression is accounted for by the reduced execution graph in the assumption, whereas peak tensor storage continues to follow the live frontier in Eq. 58. We evaluate the field in Eq. 35 using a right-endpoint discretization along the common attribution path. Let $\xi_0, \dots, \xi_n$ be its $n+1$ path states. By Eq. 36, the output gradient supplies the sensitivity of each local route product. The numerical increments and their accumulated approximation are therefore

$$\Delta\mathbf{S}^{(i,j)}_{:,z} = \nabla_{\mathbf{O}^{(i)}} F(\xi_j) \odot \left[w^{(i)}_{:,z}(\xi_j)\mathbf{X}^{(i)}_{:,z}(\xi_j) - w^{(i)}_{:,z}(\xi_{j-1})\mathbf{X}^{(i)}_{:,z}(\xi_{j-1})\right], \tag{65}$$

$$\mathbf{S}^{(i,n)}_{:,z} = \sum_{j=1}^{n} \Delta\mathbf{S}^{(i,j)}_{:,z}, \qquad z = 1, \dots, K_i, \tag{66}$$

which is a right-endpoint discretization of the product path $w^{(i)}_{:,z}\mathbf{X}^{(i)}_{:,z}$. At path state ξ_j, Definition C.8 returns $F(\xi_j)$, the node tensors in Eq. 66, and $\nabla_{\mathbf{O}^{(i)}} F(\xi_j)$ for every probed node in one execution of work T_{step}. Accumulating Eq. 66 costs $O(\sum_i d_i K_i)$ per path state, and reducing the completed pre-aggregation attribution fields costs another $O(\sum_i d_i K_i)$. The total work is therefore

$$O\left(n\left(T_{\text{step}} + \sum_i d_i K_i\right)\right) = O(nT_{\text{step}}), \tag{67}$$

where the equality follows from Eq. 64.

For the storage analysis, let $\mathcal{C}_j$ be the set of distinct physical tensors required to recover all route-indexed increments at path state j, and set

$$\mathrm{Mem}_{\{\Delta\mathbf{S}\}} := \max_{0\le j\le n} \sum_{\mathbf{X}\in\mathcal{C}_j} \mathrm{size}(\mathbf{X}). \tag{68}$$

The execution-faithful volume condition and the storage premise of the theorem give

$$\mathrm{Mem}_{\{\Delta\mathbf{S}\}} = O(T_{\mathrm{step}}), \qquad \mathrm{Mem}_{\{\Delta\mathbf{S}\}} = O(\mathrm{Mem}_{\mathrm{step}}), \qquad \max_i d_i K_i = O(\mathrm{Mem}_{\mathrm{step}}). \tag{69}$$

Moreover, $\sum_i d_i K_i = O(N_{\mathrm{node}}\mathrm{Mem}_{\mathrm{step}})$. Tensor sharing is counted once in $\mathrm{Mem}_{\{\Delta\mathbf{S}\}}$, whereas route-indexed field accumulators remain indexed by logical route. The schedules below implement path-state materialization across path states and aggregation nodes. To control the storage requirement, we use the *Incremental* schedule and the *Cache* schedule.

The *Incremental* schedule keeps one graph state and all nodewise route-indexed field accumulators, updating

$$\mathbf{S}^{(i,0)} = 0, \qquad \mathbf{S}^{(i,j)} = \mathbf{S}^{(i,j-1)} + \Delta\mathbf{S}^{(i,j)}, \qquad {\scriptstyle i=1,\dots,N_{\mathrm{node}},\ j=1,\dots,n}. \tag{70}$$

Consequently,

$$T_{\mathrm{inc}} = O\left(n\left(T_{\mathrm{step}} + \sum_i d_i K_i\right) + \sum_i d_i K_i\right) = O(nT_{\mathrm{step}}), \tag{71}$$

$$\mathrm{Mem}_{\mathrm{inc}} = O\left(\mathrm{Mem}_{\mathrm{step}} + \mathrm{Mem}_{\{\Delta\mathbf{S}\}} + \sum_i d_i K_i\right). \tag{72}$$

The *Cache* schedule stores the distinct physical tensors $\mathcal{C}_0, \dots, \mathcal{C}_n$ and reconstructs one nodewise field at a time. For node a_i, let $\mathcal{C}_{i,j} \subseteq \mathcal{C}_j$ collect its local coefficient tensor $w^{(i)}(\xi_j)$, source tensor $\mathbf{X}^{(i)}(\xi_j)$, and, for $j \ge 1$, output-side adjoint $\nabla_{\mathbf{O}^{(i)}} F(\xi_j)$. The local tensors at two adjacent path states recover the increment in Eq. 65:

$$\mathcal{C}_{i,j-1} \cup \mathcal{C}_{i,j} \longmapsto \Delta\mathbf{S}^{(i,j)}. \tag{73}$$

Summing these recovered increments as in Eq. 66 reconstructs one complete nodewise field. A shared physical tensor is stored once at each path state. The graph cache and one node-local pre-aggregation attribution field therefore give

$$T_{\mathrm{cache}} = O\left(nT_{\mathrm{step}} + n\mathrm{Mem}_{\{\Delta\mathbf{S}\}} + n\sum_i d_i K_i\right) = O(nT_{\mathrm{step}}), \tag{74}$$

$$\mathrm{Mem}_{\mathrm{cache}} = O\Big(\mathrm{Mem}_{\mathrm{step}} + n\mathrm{Mem}_{\{\Delta\mathbf{S}\}} + \max_i d_i K_i\Big). \tag{75}$$

Choosing the smaller schedule and applying Eq. 69 gives

$$\mathrm{Mem}_{\mathrm{peak}} = O\left(\mathrm{Mem}_{\mathrm{step}} + \min\left\{\mathrm{Mem}_{\{\Delta\mathbf{S}\}} + \sum_i d_i K_i,\ n\mathrm{Mem}_{\{\Delta\mathbf{S}\}} + \max_i d_i K_i\right\}\right) \tag{76}$$

$$= O\big((1 + \min\{N_{\mathrm{node}}, n\})\mathrm{Mem}_{\mathrm{step}}\big) \tag{77}$$

$$= O(\min\{N_{\mathrm{node}}, n\}\mathrm{Mem}_{\mathrm{step}}). \tag{78}$$

Together with the work bounds, this proves Eq. 63. Concrete route updates and reconstruction procedures are given in Appendix D. □

C.4 Perturbation Stability

For a perturbation field ϵ, define

$$\eta = \max_{p\in\{1,2\}} \left\{\frac{\|\epsilon\|_p}{\|\mathbf{S}\|_p}, \frac{\|\mathbb{A}[\epsilon]\|_p}{\|\mathbb{A}[\mathbf{S}]\|_p}\right\}, \tag{79}$$

where pre-aggregation attribution fields are flattened over (y, z).

Theorem C.13 (Perturbation Stability)**.** *Suppose $\eta < 1$ in Eq. 79. Then*

$$|\mathbb{V}_{\mathbb{A}}(\mathbf{S}+\epsilon) - \mathbb{V}_{\mathbb{A}}(\mathbf{S})| \le 4\log_2\frac{1+\eta}{1-\eta} = \frac{8}{\ln 2}\eta + O(\eta^3) = \Theta(\eta) \quad (\eta \to 0). \tag{80}$$

Proof. For each $p \in \{1, 2\}$, the triangle and reverse-triangle inequalities give

$$\|\mathbf{S}\|_p - \|\epsilon\|_p \le \|\mathbf{S}+\epsilon\|_p \le \|\mathbf{S}\|_p + \|\epsilon\|_p. \tag{81}$$

Linearity of aggregation gives $\mathbb{A}[\mathbf{S}+\epsilon] = \mathbb{A}[\mathbf{S}] + \mathbb{A}[\epsilon]$, and hence

$$\|\mathbb{A}[\mathbf{S}]\|_p - \|\mathbb{A}[\epsilon]\|_p \le \|\mathbb{A}[\mathbf{S}+\epsilon]\|_p \le \|\mathbb{A}[\mathbf{S}]\|_p + \|\mathbb{A}[\epsilon]\|_p. \tag{82}$$

By Eq. 79, both ratios therefore satisfy

$$1-\eta \le \frac{\|\mathbf{S}+\epsilon\|_p}{\|\mathbf{S}\|_p} \le 1+\eta, \qquad 1-\eta \le \frac{\|\mathbb{A}[\mathbf{S}+\epsilon]\|_p}{\|\mathbb{A}[\mathbf{S}]\|_p} \le 1+\eta. \tag{83}$$

Expanding the change in $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ gives

$$\begin{aligned}\mathbb{V}_{\mathbb{A}}(\mathbf{S}+\epsilon) - \mathbb{V}_{\mathbb{A}}(\mathbf{S}) = 2\log_2\frac{\|\mathbf{S}+\epsilon\|_1}{\|\mathbf{S}\|_1} - 2\log_2\frac{\|\mathbf{S}+\epsilon\|_2}{\|\mathbf{S}\|_2} \\ - 2\log_2\frac{\|\mathbb{A}[\mathbf{S}+\epsilon]\|_1}{\|\mathbb{A}[\mathbf{S}]\|_1} + 2\log_2\frac{\|\mathbb{A}[\mathbf{S}+\epsilon]\|_2}{\|\mathbb{A}[\mathbf{S}]\|_2}.\end{aligned} \tag{84}$$

The preceding interval bounds each positive term by $2\log_2(1+\eta)$ and each negative term by $-2\log_2(1-\eta)$. Applying the same argument to the negative of the displayed difference yields

$$|\mathbb{V}_{\mathbb{A}}(\mathbf{S}+\epsilon) - \mathbb{V}_{\mathbb{A}}(\mathbf{S})| \le 4\log_2\frac{1+\eta}{1-\eta}, \tag{85}$$

and the Taylor expansion $\ln((1+\eta)/(1-\eta)) = 2\eta + \frac{2}{3}\eta^3 + O(\eta^5)$ gives the stated asymptotic form. □

C.5 Coordinate-Positioning Entropy Interpretation

Theorem C.14 (Coordinate-Positioning Entropy Interpretation)**.** *If $\mathbf{S} \neq 0$ and $\mathbf{S}_{y,z} \ge 0$ for all (y, z), then*

$$\mathbb{V}_{\mathbb{A}}(\mathbf{S}) = H_2(Y, Z) - H_2(Y), \tag{86}$$

where

$$p(Y = y, Z = z) = \frac{\mathbf{S}_{y,z}}{\sum_{y',z'} \mathbf{S}_{y',z'}}. \tag{87}$$

Proof. Because every pre-aggregation attribution entry is non-negative and the field is nonzero,

$$\sum_{y,z} \mathbf{S}_{y,z} > 0. \tag{88}$$

For each route,

$$\sum_{y} \mathbf{S}_{y,z} = \sum_{y} |\mathbf{S}_{y,z}| = \|\mathbf{S}_{:,z}\|_1. \tag{89}$$

Summing over routes and then over unreduced coordinates gives the mass-conservation identities

$$\sum_{y,z} \mathbf{S}_{y,z} = \sum_{z} \|\mathbf{S}_{:,z}\|_1 = \sum_{y} \mathbb{A}[\mathbf{S}]_y = \|\mathbb{A}[\mathbf{S}]\|_1. \tag{90}$$

It follows immediately that the proposed joint mass is normalized:

$$\sum_{y,z} p(Y = y, Z = z) = \frac{\sum_{y,z} \mathbf{S}_{y,z}}{\sum_{y',z'} \mathbf{S}_{y',z'}} = 1. \tag{91}$$

Marginalizing over routes gives

$$p(Y=y)=\sum_z p(Y=y,Z=z)=\frac{\sum_z \mathbf{S}_{y,z}}{\sum_{y',z'}\mathbf{S}_{y',z'}}=\frac{\mathbb{A}[\mathbf{S}]_y}{\|\mathbb{A}[\mathbf{S}]\|_1}. \tag{92}$$

On every coordinate with $\mathbb{A}[\mathbf{S}]_y>0$, the associated conditional route distribution is

$$p(Z=z\mid Y=y)=\frac{p(Y=y,Z=z)}{p(Y=y)}=\frac{\mathbf{S}_{y,z}}{\mathbb{A}[\mathbf{S}]_y}, \tag{93}$$

and it sums to one over z. Thus route aggregation is exactly marginalization of the same coordinate-positioning mass.

The joint collision probability is

$$\sum_{y,z}p(Y=y,Z=z)^2=\frac{\sum_{y,z}\mathbf{S}_{y,z}^2}{\left(\sum_{y',z'}\mathbf{S}_{y',z'}\right)^2}=\frac{\sum_z\|\mathbf{S}_{:,z}\|_2^2}{\left(\sum_z\|\mathbf{S}_{:,z}\|_1\right)^2}, \tag{94}$$

so

$$H_2(Y,Z)=\log_2\frac{\left(\sum_z\|\mathbf{S}_{:,z}\|_1\right)^2}{\sum_z\|\mathbf{S}_{:,z}\|_2^2}=\log_2\mathrm{D}(\mathbf{S}). \tag{95}$$

Similarly,

$$\sum_y p(Y=y)^2=\frac{\sum_y\mathbb{A}[\mathbf{S}]_y^2}{\|\mathbb{A}[\mathbf{S}]\|_1^2}=\frac{\|\mathbb{A}[\mathbf{S}]\|_2^2}{\|\mathbb{A}[\mathbf{S}]\|_1^2}, \tag{96}$$

and hence

$$H_2(Y)=\log_2\frac{\|\mathbb{A}[\mathbf{S}]\|_1^2}{\|\mathbb{A}[\mathbf{S}]\|_2^2}=\log_2\mathrm{D}(\mathbb{A}[\mathbf{S}]). \tag{97}$$

Subtracting the two entropies gives

$$H_2(Y,Z)-H_2(Y)=\log_2\frac{\|\mathbb{A}[\mathbf{S}]\|_2^2}{\sum_z\|\mathbf{S}_{:,z}\|_2^2}=\log_2\frac{\mathrm{D}(\mathbf{S})}{\mathrm{D}(\mathbb{A}[\mathbf{S}])}=\mathbb{V}_{\mathbb{A}}(\mathbf{S}), \tag{98}$$

where the second equality uses the common L_1 mass identity above. This proves the chain-difference identity. □

C.6 Lower and Upper Bounds

Theorem C.15 (Lower and Upper Bounds on $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$)**.** *For a non-degenerate field with K routes and $d\geq 1$ unreduced output coordinates,*

$$\begin{cases}0, & K=1,\\ \min\{0,1-\log_2 d\}, & K\geq 2\end{cases}\leq\mathbb{V}_{\mathbb{A}}(\mathbf{S})\leq\log_2(Kd). \tag{99}$$

If $\mathbf{S}_{y,z}\geq 0$ component-wise, then

$$0\leq\mathbb{V}_{\mathbb{A}}(\mathbf{S})\leq\log_2\frac{\left(\sum_z\|\mathbf{S}_{:,z}\|_2\right)^2}{\sum_z\|\mathbf{S}_{:,z}\|_2^2}\leq\log_2 K. \tag{100}$$

Proof. Step 1: expand the four norm states. Definitions 3.2 and 3.3 directly give

$$\mathbb{V}_{\mathbb{A}}(\mathbf{S})=2\log_2\frac{\sum_z\|\mathbf{S}_{:,z}\|_1}{\|\mathbb{A}[\mathbf{S}]\|_1}+\log_2\frac{\|\mathbb{A}[\mathbf{S}]\|_2^2}{\sum_z\|\mathbf{S}_{:,z}\|_2^2}. \tag{101}$$

Step 2: prove the signed upper bound. For any nonzero $\boldsymbol{v}\in\mathbb{R}^n$,

$$\|\boldsymbol{v}\|_1^2=\left(\sum_{i=1}^n|\boldsymbol{v}_i|\right)^2\geq\sum_{i=1}^n|\boldsymbol{v}_i|^2=\|\boldsymbol{v}\|_2^2, \tag{102}$$

while Cauchy–Schwarz gives

$$\|\boldsymbol{v}\|_1 = \sum_{i=1}^{n} 1 \cdot |\boldsymbol{v}_i| \le \sqrt{n} \left(\sum_{i=1}^{n} |\boldsymbol{v}_i|^2 \right)^{1/2} = \sqrt{n}\, \|\boldsymbol{v}\|_2. \tag{103}$$

Therefore

$$1 \le \mathrm{D}(\boldsymbol{v}) = \frac{\|\boldsymbol{v}\|_1^2}{\|\boldsymbol{v}\|_2^2} \le n. \tag{104}$$

The flattened pre-aggregation attribution field has Kd coordinates, whereas $\mathbb{A}[\mathbf{S}]$ has d coordinates. Hence

$$\mathrm{D}(\mathbf{S}) \le Kd, \qquad \mathrm{D}(\mathbb{A}[\mathbf{S}]) \ge 1. \tag{105}$$

Consequently, the general outer estimate is

$$\mathbb{V}_{\mathbb{A}}(\mathbf{S}) = \log_2 \frac{\mathrm{D}(\mathbf{S})}{\mathrm{D}(\mathbb{A}[\mathbf{S}])} \le \log_2(Kd) = \log_2 K + \log_2 d. \tag{106}$$

Step 3: prove the signed lower bound. If $K = 1$, then $\mathbb{A}[\mathbf{S}] = \mathbf{S}$ and $\mathbb{V}_{\mathbb{A}}(\mathbf{S}) = 0$. Hence it remains to consider $K \ge 2$. If $d = 1$, the nonzero aggregated field is scalar, so $\mathrm{D}(\mathbb{A}[\mathbf{S}]) = 1$. Since every nonzero field satisfies $\mathrm{D}(\mathbf{S}) \ge 1$, it follows directly that

$$\mathbb{V}_{\mathbb{A}}(\mathbf{S}) \ge 0 = \min\{0, 1 - \log_2 d\}. \tag{107}$$

Hence it remains to consider $d \ge 2$.

To approach the smallest ratio $\mathrm{D}(\mathbf{S})/\mathrm{D}(\mathbb{A}[\mathbf{S}])$, $\mathrm{D}(\mathbb{A}[\mathbf{S}])$ should approach its largest possible value d, while $\mathrm{D}(\mathbf{S})$ remains as small as signed cancellation permits. The limiting configuration therefore concentrates the dominant entries of $\mathbf{S}$ into one positive and one negative entry that cancel, while a vanishing residual is distributed over the d unreduced coordinates. This gives the candidate ratio $2/d$; the construction in Step 5 realizes this limit.

The proof bounds every pre-aggregation attribution field by this extremal pattern. The motivation is to make the squared sum $\sum_{y,z} \mathbf{S}_{y,z}^2$ as large as possible: for fixed $\sum_{y,z} |\mathbf{S}_{y,z}|$ and $\mathbb{A}[\mathbf{S}]$, this minimizes $\mathrm{D}(\mathbf{S})$. Coordinate-wise, for fixed $\sum_z |\mathbf{S}_{y,z}|$ and $\sum_z \mathbf{S}_{y,z}$, the quantity $\sum_z \mathbf{S}_{y,z}^2$ is largest when the non-negative entries are concentrated into one entry and the negative entries into another. By scale invariance, assume

$$\sum_{y,z} |\mathbf{S}_{y,z}| = 1. \tag{108}$$

For each y, adding and subtracting $\sum_z |\mathbf{S}_{y,z}|$ and $\sum_z \mathbf{S}_{y,z} = \mathbb{A}[\mathbf{S}]_y$ gives

$$\sum_{z:\, \mathbf{S}_{y,z} \ge 0} \mathbf{S}_{y,z} = \frac{1}{2} \left(\sum_z |\mathbf{S}_{y,z}| + \mathbb{A}[\mathbf{S}]_y \right), \qquad - \sum_{z:\, \mathbf{S}_{y,z} < 0} \mathbf{S}_{y,z} = \frac{1}{2} \left(\sum_z |\mathbf{S}_{y,z}| - \mathbb{A}[\mathbf{S}]_y \right). \tag{109}$$

All summands in the two sums on the left are non-negative. Therefore

$$\begin{aligned} \sum_z \mathbf{S}_{y,z}^2 &= \sum_{z:\, \mathbf{S}_{y,z} \ge 0} \mathbf{S}_{y,z}^2 + \sum_{z:\, \mathbf{S}_{y,z} < 0} (-\mathbf{S}_{y,z})^2 \\ &\le \left(\sum_{z:\, \mathbf{S}_{y,z} \ge 0} \mathbf{S}_{y,z} \right)^2 + \left(- \sum_{z:\, \mathbf{S}_{y,z} < 0} \mathbf{S}_{y,z} \right)^2 = \frac{1}{2} \left[\left(\sum_z |\mathbf{S}_{y,z}| \right)^2 + \mathbb{A}[\mathbf{S}]_y^2 \right]. \end{aligned} \tag{110}$$

Summing over y gives

$$\sum_{y,z} \mathbf{S}_{y,z}^2 \le \frac{1}{2} \left[\sum_y \left(\sum_z |\mathbf{S}_{y,z}| \right)^2 + \|\mathbb{A}[\mathbf{S}]\|_2^2 \right]. \tag{111}$$

The triangle inequality and the normalization imply

$$0 < \|\mathbb{A}[\mathbf{S}]\|_1 \le 1, \qquad |\mathbb{A}[\mathbf{S}]_y| \le \sum_z |\mathbf{S}_{y,z}|. \tag{112}$$

We bound the first term in Eq. 111 by separating the strict and equality cases of this normalized L_1 relation. *Case (i):* $\|\mathbb{A}[\mathbf{S}]\|_1 = 1$. The non-negative quantities $\sum_z |\mathbf{S}_{y,z}| - |\mathbb{A}[\mathbf{S}]_y|$ sum to zero and hence vanish for every y. Therefore the bound established in Case (ii) below is attained with equality.

Case (ii): $\|\mathbb{A}[\mathbf{S}]\|_1 < 1$. In this case,

$$\sum_z |\mathbf{S}_{y,z}| = \|\mathbb{A}[\mathbf{S}]\|_1 \frac{|\mathbb{A}[\mathbf{S}]_y|}{\|\mathbb{A}[\mathbf{S}]\|_1} + (1 - \|\mathbb{A}[\mathbf{S}]\|_1) \frac{\sum_z |\mathbf{S}_{y,z}| - |\mathbb{A}[\mathbf{S}]_y|}{1 - \|\mathbb{A}[\mathbf{S}]\|_1}.$$

Collecting these identities over y gives a convex combination of two component-wise non-negative vectors with unit coordinate sum. Each therefore has squared Euclidean norm at most one. Define the local function

$$f(\boldsymbol{v}) = \|\boldsymbol{v}\|_2^2, \qquad \boldsymbol{v} \in \mathbb{R}^d. \tag{113}$$

Its Hessian is $\nabla^2 f(\boldsymbol{v}) = 2\boldsymbol{I}_d \succeq 0$, so f is convex. Applying the convexity inequality to the preceding decomposition, with coefficient $\|\mathbb{A}[\mathbf{S}]\|_1$, gives

$$\begin{aligned}
\sum_y \left(\sum_z |\mathbf{S}_{y,z}| \right)^2 &= f\left(\sum_z |\mathbf{S}_{:,z}| \right) \\
&\leq \|\mathbb{A}[\mathbf{S}]\|_1 f\left(\frac{|\mathbb{A}[\mathbf{S}]|}{\|\mathbb{A}[\mathbf{S}]\|_1} \right) + (1 - \|\mathbb{A}[\mathbf{S}]\|_1) f\left(\frac{\sum_z |\mathbf{S}_{:,z}| - |\mathbb{A}[\mathbf{S}]|}{1 - \|\mathbb{A}[\mathbf{S}]\|_1} \right) \\
&\leq \frac{\|\mathbb{A}[\mathbf{S}]\|_2^2}{\|\mathbb{A}[\mathbf{S}]\|_1} + 1 - \|\mathbb{A}[\mathbf{S}]\|_1.
\end{aligned} \tag{114}$$

Combining Eqs. 111 and 114 gives the preceding upper bound

$$\sum_{y,z} \mathbf{S}_{y,z}^2 \leq \frac{1}{2} \left[\frac{\|\mathbb{A}[\mathbf{S}]\|_2^2}{\|\mathbb{A}[\mathbf{S}]\|_1} + \|\mathbb{A}[\mathbf{S}]\|_2^2 + 1 - \|\mathbb{A}[\mathbf{S}]\|_1 \right]. \tag{115}$$

Next, we show that the upper bound in Eq. 115 is at most $d\|\mathbb{A}[\mathbf{S}]\|_2^2/(2\|\mathbb{A}[\mathbf{S}]\|_1^2)$. By Cauchy–Schwarz,

$$\|\mathbb{A}[\mathbf{S}]\|_2^2 \geq \frac{\|\mathbb{A}[\mathbf{S}]\|_1^2}{d}. \tag{116}$$

Since $d \geq 2$ and $0 < \|\mathbb{A}[\mathbf{S}]\|_1 \leq 1$,

$$\begin{aligned}
&\frac{d\|\mathbb{A}[\mathbf{S}]\|_2^2}{\|\mathbb{A}[\mathbf{S}]\|_1^2} - \left[\frac{\|\mathbb{A}[\mathbf{S}]\|_2^2}{\|\mathbb{A}[\mathbf{S}]\|_1} + \|\mathbb{A}[\mathbf{S}]\|_2^2 + 1 - \|\mathbb{A}[\mathbf{S}]\|_1 \right] \\
&\quad = \frac{d - \|\mathbb{A}[\mathbf{S}]\|_1 - \|\mathbb{A}[\mathbf{S}]\|_1^2}{\|\mathbb{A}[\mathbf{S}]\|_1^2} \|\mathbb{A}[\mathbf{S}]\|_2^2 - (1 - \|\mathbb{A}[\mathbf{S}]\|_1) \\
&\quad \geq \frac{\|\mathbb{A}[\mathbf{S}]\|_1 (d - 1 - \|\mathbb{A}[\mathbf{S}]\|_1)}{d} \geq 0.
\end{aligned} \tag{117}$$

Together with Eq. 115, this gives

$$\sum_{y,z} \mathbf{S}_{y,z}^2 \leq \frac{d\|\mathbb{A}[\mathbf{S}]\|_2^2}{2\|\mathbb{A}[\mathbf{S}]\|_1^2}. \tag{118}$$

Restoring the common scale gives

$$\frac{\|\mathbb{A}[\mathbf{S}]\|_2^2}{\sum_z \|\mathbf{S}_{:,z}\|_2^2} \geq \frac{2}{d} \left(\frac{\|\mathbb{A}[\mathbf{S}]\|_1}{\sum_z \|\mathbf{S}_{:,z}\|_1} \right)^2. \tag{119}$$

Combining the direct expansion in Eq. 101 with the coupled squared-L_2 estimate in Eq. 119 gives

$$\begin{aligned}
\mathbb{V}_{\mathbb{A}}(\mathbf{S}) &\geq 2 \log_2 \frac{\sum_z \|\mathbf{S}_{:,z}\|_1}{\|\mathbb{A}[\mathbf{S}]\|_1} + \log_2 \left[\frac{2}{d} \left(\frac{\|\mathbb{A}[\mathbf{S}]\|_1}{\sum_z \|\mathbf{S}_{:,z}\|_1} \right)^2 \right] \\
&= \log_2 \frac{2}{d} = 1 - \log_2 d.
\end{aligned} \tag{120}$$

Step 4: refine the bound for non-negative pre-aggregation attribution entries. If $\mathbf{S}_{y,z} \geq 0$, direct summation gives

$$\sum_z \|\mathbf{S}_{:,z}\|_1 = \|\mathbb{A}[\mathbf{S}]\|_1, \tag{121}$$

so the first logarithm in Eq. 101 vanishes and

$$\mathbb{V}_{\mathbb{A}}(\mathbf{S}) = \log_2 \frac{\|\sum_z \mathbf{S}_{:,z}\|_2^2}{\sum_z \|\mathbf{S}_{:,z}\|_2^2}. \tag{122}$$

Every pairwise inner product is non-negative because both vectors are component-wise non-negative. Hence

$$\left\|\sum_z \mathbf{S}_{:,z}\right\|_2^2 = \sum_z \|\mathbf{S}_{:,z}\|_2^2 + 2\sum_{z<z'} \langle \mathbf{S}_{:,z}, \mathbf{S}_{:,z'}\rangle \geq \sum_z \|\mathbf{S}_{:,z}\|_2^2. \tag{123}$$

The ratio inside the logarithm is at least one, proving $\mathbb{V}_{\mathbb{A}}(\mathbf{S}) \geq 0$.

For the upper bound, the triangle inequality gives

$$\left\|\sum_z \mathbf{S}_{:,z}\right\|_2 \leq \sum_z \|\mathbf{S}_{:,z}\|_2. \tag{124}$$

Therefore

$$\mathbb{V}_{\mathbb{A}}(\mathbf{S}) \leq \log_2 \frac{\left(\sum_z \|\mathbf{S}_{:,z}\|_2\right)^2}{\sum_z \|\mathbf{S}_{:,z}\|_2^2} \leq \log_2 K. \tag{125}$$

The last inequality follows directly from Cauchy–Schwarz:

$$\left(\sum_z \|\mathbf{S}_{:,z}\|_2\right)^2 \leq K \sum_z \|\mathbf{S}_{:,z}\|_2^2. \tag{126}$$

This proves the complete non-negative interval.

Step 5: illustrative signed constructions. For $K \geq 2$, let

$$\mathbf{S}_{:,1} = [1+\epsilon, \epsilon, \ldots, \epsilon], \qquad \mathbf{S}_{:,2} = [-1, 0, \ldots, 0], \qquad \mathbf{S}_{:,z} = \mathbf{0} \quad (z \geq 3), \qquad \epsilon > 0. \tag{127}$$

The pre-aggregation and post-aggregation effective-dimensionality values are

$$\mathrm{D}(\mathbf{S}) = \frac{(2+d\epsilon)^2}{2 + 2\epsilon + d\epsilon^2} \longrightarrow 2, \qquad \mathbb{A}[\mathbf{S}] = [\epsilon, \ldots, \epsilon], \qquad \mathrm{D}(\mathbb{A}[\mathbf{S}]) = d. \tag{128}$$

Consequently,

$$\mathbb{V}_{\mathbb{A}}(\mathbf{S}) \longrightarrow \log_2 \frac{2}{d} = 1 - \log_2 d. \tag{129}$$

For $d \geq 2$ this construction approaches the signed lower infimum $1 - \log_2 d$; the limit is negative when $d > 2$. For $K = 1$, $\mathbb{A}[\mathbf{S}] = \mathbf{S}$ and hence $\mathbb{V}_{\mathbb{A}}(\mathbf{S}) = 0$.

For $K = 2$, set

$$\mathbf{S}_{:,1} = [1, 1, \ldots, 1], \qquad \mathbf{S}_{:,2} = [1, -1, \ldots, -1]. \tag{130}$$

Then

$$\mathrm{D}(\mathbf{S}) = 2d, \qquad \mathbb{A}[\mathbf{S}] = [2, 0, \ldots, 0], \qquad \mathrm{D}(\mathbb{A}[\mathbf{S}]) = 1, \tag{131}$$

and hence

$$\mathbb{V}_{\mathbb{A}}(\mathbf{S}) = 1 + \log_2 d. \tag{132}$$

This realizes the general signed upper bound at $K = 2$. □

D IMPLEMENTATION OF ATTRIBUTION VERGENCE

This appendix specifies the module increments, field statistics, and two storage schedules analyzed in Appendix C.3. The Cache schedule stores path states and reconstructs one node's field at a time; the Incremental schedule accumulates selected fields along the path. Across these schedules, the implementation uses the canonical tensor layout

$$\mathbf{S} \in \mathbb{R}^{B\times R_1\times\cdots\times R_m\times Z_1\times\cdots\times Z_k\times Y_1\times\cdots\times Y_p}, \qquad \mathbb{A}[\mathbf{S}] \in \mathbb{R}^{B\times R_1\times\cdots\times R_m\times Y_1\times\cdots\times Y_p}. \tag{133}$$

Here B is the sample axis, R contains axes indexing aggregation nodes, Z contains route axes reduced by $\mathbb{A}$, and Y contains unreduced output coordinates. Each tuple of indices along the R axes identifies an aggregation node a_i. Effective dimensionalities are computed independently for each sample b and node a_i: the pre-aggregation state is flattened only over $Z \times Y$, and the post-aggregation state only over Y. Neither the sample axis nor any axis indexing aggregation nodes participates in an effective-dimensional coordinate.

Concrete aggregation nodes. Let $0 = \alpha_0 < \cdots < \alpha_n = 1$ and $x_j = \gamma(\alpha_j)$, giving n path-discretization intervals and $n+1$ path states. The formulas below use the right-endpoint Riemann–Stieltjes discretization of the product path $w\mathbf{X}$. At each right-endpoint state, $F_j = F(x_j)$ is the scalar target. For a probed aggregation node a_i, the implementation needs the post-side gradient $\boldsymbol{g}^{(i,j)} := \nabla_{\mathbf{O}^{(i)}} F(\gamma(\alpha_j))$, the current and previous module inputs, and the native routing coefficients at the same path points. We write $\nu^{(i,j)}_{z,:} \in \mathbb{R}^{d_i}$ for the multiplicative coefficient section that maps source section $\mathbf{X}^{(u(i),j)}_{z,:}$ to its route-indexed contribution $\nu^{(i,j)}_{z,:} \odot \mathbf{X}^{(u(i),j)}_{z,:}$ before the native sum over z. Here $u(i)$ indexes the upstream input tensor feeding node a_i. Each scalar attribution entry is bound coordinatewise to the input edge jointly specified by $\nu^{(i,j)}_{z,y}$ and $\mathbf{X}^{(u(i),j)}_{z,y}$. Its forward contribution is $\nu^{(i,j)}_{z,y}\mathbf{X}^{(u(i),j)}_{z,y}$. The exact right-endpoint product difference gives

$$\begin{aligned}
\Delta\mathbf{S}^{(i,j)}_{z,:} &= \boldsymbol{g}^{(i,j)} \odot \left[\nu^{(i,j-1)}_{z,:} \odot \left(\mathbf{X}^{(u(i),j)}_{z,:} - \mathbf{X}^{(u(i),j-1)}_{z,:}\right) + \left(\nu^{(i,j)}_{z,:} - \nu^{(i,j-1)}_{z,:}\right) \odot \mathbf{X}^{(u(i),j)}_{z,:}\right], \\
\mathbf{S}^{(i,n)} &= \sum_{j=1}^{n} \Delta\mathbf{S}^{(i,j)}.
\end{aligned} \tag{134}$$

The Aggregation Operator and pre-aggregation statistics are evaluated on the final pre-aggregation attribution field $\mathbf{S}^{(i,n)}$, with $\mathbb{A}^{(i)}[\mathbf{S}^{(i,n)}] = \sum_z \mathbf{S}^{(i,n)}_{z,:}$; the coefficient $\nu^{(i,j)}_{z,:}$ is already absorbed into $\mathbf{S}^{(i,n)}_{z,:}$ and is not applied again. The first term in brackets attributes source-value variation under the previous routing state; the second attributes routing variation itself. For fixed-routing operators the second term is identically zero. The formulas below specialize Eq. 134. For partitions derived from Gauss–Legendre abscissae, those abscissae specify the path states; the finite differences in Eq. 134 receive no additional quadrature weights.

Multi-head attention aggregation nodes. For an input sequence of shape $B \times L \times D$, let H be the number of heads and $d_h = D/H$. At path state j, the attention-weight tensor is $\mathbf{A}^{(j)} \in \mathbb{R}^{B\times H\times L\times L}$, with entries $\mathbf{A}^{(j)}_{b,h,q,k}$; the value tensor has entries $\mathbf{V}^{(j)}_{b,k,h,r}$, and the attention output is

$$\mathbf{O}^{(j)}_{b,h,q,r} = \sum_{k=1}^{L} \mathbf{A}^{(j)}_{b,h,q,k}\mathbf{V}^{(j)}_{b,k,h,r}. \tag{135}$$

For aggregation node (h,q) and source token k, the algorithmic routing coefficient is $\nu^{(h,q,j)}_{b,k,r} = \mathbf{A}^{(j)}_{b,h,q,k}$ for every unreduced value coordinate r. The local increment is therefore

$$\Delta\mathbf{S}^{(h,q,j)}_{b,k,r} = \frac{\partial F_j}{\partial \mathbf{O}^{(j)}_{b,h,q,r}} \left[\mathbf{A}^{(j-1)}_{b,h,q,k}\left(\mathbf{V}^{(j)}_{b,k,h,r} - \mathbf{V}^{(j-1)}_{b,k,h,r}\right) + \left(\mathbf{A}^{(j)}_{b,h,q,k} - \mathbf{A}^{(j-1)}_{b,h,q,k}\right)\mathbf{V}^{(j)}_{b,k,h,r}\right]. \tag{136}$$

The post-aggregation increment is $\mathbb{A}^{(h,q)}_{\text{attn}}[\Delta\mathbf{S}^{(h,q,j)}]_{b,r} = \sum_{k=1}^{L} \Delta\mathbf{S}^{(h,q,j)}_{b,k,r}$. Summing the bracketed transition over k gives exactly $\mathbf{O}^{(j)}_{b,h,q,r} - \mathbf{O}^{(j-1)}_{b,h,q,r}$, so both changing values and changing attention

routes enter the Conductance field. Vectorizing all local nodes gives $\mathbf{S}_{b,h,q,k,r} \in \mathbb{R}^{B\times H\times L\times L\times d_h}$ and $\mathbb{A}_{\text{attn}}[\mathbf{S}]_{b,h,q,r} \in \mathbb{R}^{B\times H\times L\times d_h}$. Thus (h, q) index aggregation nodes, k is the route coordinate, and r is the unreduced output coordinate. The implementation first computes one local statistic for every (b, h, q), over (k, r) before aggregation and over r afterward.

General, grouped, and depthwise convolution aggregation nodes. For convolution operations expressed in standard channel-first coordinates, $\mathbf{X}^{(j)} \in \mathbb{R}^{B\times C_{\text{in}}\times H_{\text{in}}\times W_{\text{in}}}$ and $\mathbf{Y}^{(j)} \in \mathbb{R}^{B\times C_{\text{out}}\times H_{\text{out}}\times W_{\text{out}}}$. Let $u = (u_1, u_2)$ be an output position, $\delta \in \mathcal{K}$ a kernel offset, and $\pi(u, \delta)$ the corresponding input position under the specified stride, dilation, and padding. For grouped convolution, let $\mathcal{C}(o)$ be the input channels connected to output channel o. The native operator is

$$\mathbf{Y}^{(j)}_{b,o,u} = \boldsymbol{b}_o + \sum_{c\in\mathcal{C}(o)}\sum_{\delta\in\mathcal{K}} \mathbf{W}_{o,c,\delta}\mathbf{X}^{(j)}_{b,c,\pi(u,\delta)}. \tag{137}$$

The aggregation node is the output position u, the unreduced output coordinate is $y = o$, and the route coordinate is $z = (c, \delta)$. Because the convolution weights are fixed along the path, Eq. 134 becomes

$$\Delta\mathbf{S}^{(u,j)}_{b,(c,\delta),o} = \mathbf{W}_{o,c,\delta}\frac{\partial F_j}{\partial \mathbf{Y}^{(j)}_{b,o,u}}\left(\mathbf{X}^{(j)}_{b,c,\pi(u,\delta)} - \mathbf{X}^{(j-1)}_{b,c,\pi(u,\delta)}\right), \qquad c \in \mathcal{C}(o), \tag{138}$$

and

$$\mathbb{A}^{(u)}_{\text{conv}}[\Delta\mathbf{S}^{(u,j)}]_{b,o} = \sum_{c\in\mathcal{C}(o)}\sum_{\delta\in\mathcal{K}} \Delta\mathbf{S}^{(u,j)}_{b,(c,\delta),o}. \tag{139}$$

The routing tensor implementation forms the intermediate tensor $[B, C_{\text{out}}, H_{\text{out}}, W_{\text{out}}, Z]$, where $Z = (C_{\text{in}}/G)|\mathcal{K}|$ for G groups, and canonicalizes it as $[B, H_{\text{out}}, W_{\text{out}}, Z, C_{\text{out}}]$. Hence u indexes the aggregation node, the packed input-channel–offset index is the route coordinate, and output channel is unreduced. The pre state is evaluated over $Z \times C_{\text{out}}$, whereas the post state is evaluated over C_{out}; the convolutional post field is therefore a channel vector.

Depthwise convolution is the special case $G = C_{\text{in}} = C_{\text{out}}$ and $\mathcal{C}(o) = \{o\}$. Writing the unreduced channel as c, its canonical local field is $\mathbf{S}^{(u)}_{b,\delta,c}$, its route coordinate is only δ, and

$$\mathbb{A}^{(u)}_{\text{conv}}[\mathbf{S}^{(u)}]_{b,c} = \sum_{\delta\in\mathcal{K}} \mathbf{S}^{(u)}_{b,\delta,c}. \tag{140}$$

Thus depthwise convolution leaves the full output-channel vector unreduced, and its post-aggregation effective dimensionality is not identically one. In all cases, the native kernel weight is already absorbed into the route-indexed attribution field and is not applied again.

Position-wise affine MLP projection. Let the position-wise MLP hidden tensor after GELU and dropout be $\mathbf{H}^{(j)} \in \mathbb{R}^{B\times L\times d_{\text{mlp}}}$, and let $\mathbf{W}^{(2)} \in \mathbb{R}^{d_{\text{model}}\times d_{\text{mlp}}}$ and $\boldsymbol{b}^{(2)} \in \mathbb{R}^{d_{\text{model}}}$ be the fixed second-projection parameters. Its residual-stream update is

$$\mathbf{M}^{(j)}_{b,t,o} = \boldsymbol{b}^{(2)}_o + \sum_{i=1}^{d_{\text{mlp}}} \mathbf{W}^{(2)}_{o,i}\mathbf{H}^{(j)}_{b,t,i}. \tag{141}$$

Each token position t is one aggregation node, with unreduced coordinate $y = o$ and route coordinate $z = i$. The bias is fixed context and is not included as a route. For every (b, t, i, o), the field coordinate is assigned to the input edge jointly specified by $\mathbf{W}^{(2)}_{o,i}$ and $\mathbf{H}_{b,t,i}$. Because the routing coefficient is fixed, the local increment is

$$\Delta\mathbf{S}^{(t,j)}_{b,i,o} = \mathbf{W}^{(2)}_{o,i}\frac{\partial F_j}{\partial \mathbf{M}^{(j)}_{b,t,o}}\left(\mathbf{H}^{(j)}_{b,t,i} - \mathbf{H}^{(j-1)}_{b,t,i}\right), \qquad \mathbb{A}^{(t)}_{\text{mlp}}[\Delta\mathbf{S}^{(t,j)}]_{b,o} = \sum_i \Delta\mathbf{S}^{(t,j)}_{b,i,o}. \tag{142}$$

The batch index b labels independent samples and is excluded from the unreduced and route coordinates used by D and $\mathbb{V}$; t remains a local node index until the designated all-token or target-token reduction. The vectorized position-wise layouts are therefore $\mathbf{S} \in \mathbb{R}^{B\times L\times d_{\text{mlp}}\times d_{\text{model}}}$ and $\mathbb{A}_{\text{mlp}}[\mathbf{S}] \in \mathbb{R}^{B\times L\times d_{\text{model}}}$. More generally, for fixed-routing linear layers, every non-batch prefix axis shared by its input and output is treated as an axis indexing aggregation nodes, the input activation coordinate as the route coordinate, and the output activation coordinate as the unreduced output coordinate.

Computing statistics from local fields. After accumulating the full path, compute $\|\mathbf{S}_b^{(i,n)}\|_1$ and $\|\mathbf{S}_b^{(i,n)}\|_2^2$ for each sample b and node a_i. The norms flatten the local route and unreduced coordinates according to Eq. 133; convolution sums follow the connected channel topology. $\mathbb{A}^{(i)}[\mathbf{S}^{(i,n)}]$ sums only route coordinates. These quantities give the local $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$:

$$\mathbb{V}_{\mathbb{A}^{(i)}}(\mathbf{S}_b^{(i,n)}) = 2\log_2 \frac{\|\mathbf{S}_b^{(i,n)}\|_1}{\|\mathbb{A}^{(i)}[\mathbf{S}^{(i,n)}]_{b,:}\|_1} + \log_2 \frac{\|\mathbb{A}^{(i)}[\mathbf{S}^{(i,n)}]_{b,:}\|_2^2}{\|\mathbf{S}_b^{(i,n)}\|_2^2}. \tag{143}$$

Many-node schedule. The Cache schedule stores the tensors needed by the module increments and reconstructs one node's field at a time. Attention requires weights and values at states $j = 0, \ldots, n$, plus output gradients at $j = 1, \ldots, n$. Convolution requires input differences and output gradients; MLP projections require hidden states at $j = 0, \ldots, n$ and output gradients at $j = 1, \ldots, n$. Fixed weights remain available throughout. Each physical tensor shared by multiple nodes is stored once per path state. All modules reconstruct their fields by the sum in Eq. 134, with increments from Eqs. 136, 138, and 142, respectively.

Algorithm 1 Cached Profile Computation across Aggregation Nodes

Require: Partition states $\boldsymbol{x}_j = \gamma(\alpha_j)$ for $j = 0, \ldots, n$; probed attention, convolution, and MLP nodes.
Ensure: Local attribution statistics and their module-level reductions.
  Store baseline attention weights and values, convolution inputs, and MLP hidden states.
  for $j = 1, \ldots, n$ **do**
    Run one forward/backward pass for F_j.
    Cache the tensors needed by the probed modules' increment formulas.
  end for
  for each probed aggregation node a_i **do**
    Reconstruct $\mathbf{S}^{(i,n)}$ by summing its module increments over j.
    Sum route coordinates to obtain $\mathbb{A}^{(i)}[\mathbf{S}^{(i,n)}]$.
    Compute local norms and dimensional statistics by Eq. 143.
    Record local statistics and $w_{b,i}$ as defined in Eq. 144.
  end for
  Compute $\bar{x}_b$ and $\bar{x}$ by Eq. 144.

Single-token MHAttention schedule. For a fixed query $q_\star$ and selected heads, the Incremental schedule stores their accumulated fields and the routing and value tensors needed at adjacent path states. Each increment follows Eq. 136 with $q = q_\star$.

Complexity accounting. Under the execution and storage assumptions in Appendix C.3, both schedules require $O(nT_{\text{step}})$ work. Cache storage scales as $O(n\text{Mem}_{\text{step}})$, while accumulating the selected fields scales as $O(N_{\text{node}}\text{Mem}_{\text{step}})$. The smaller storage estimate guides the schedule choice. Norm statistics are evaluated on the final accumulated fields without an additional norm estimator; the path integral is approximated by the chosen discretization scheme.

Local-to-module reduction. The definitions and computations above apply to individual aggregation nodes. A module contains aggregation nodes $a_1, \ldots, a_M$: head–query pairs in attention, spatial positions in convolution, and token positions in an MLP. We summarize their measurements as follows.

Definition D.1 (Module-level measurement)**.** For sample b and node a_i in a module, let $x_{b,i}$ be the value of $\log_2 \mathrm{D}(\mathbf{S})$, $\log_2 \mathrm{D}(\mathbb{A}[\mathbf{S}])$, or $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$, computed under Eq. 133. Define the module measurement $\bar{x}_b$ by weighting node values with their post-aggregation attribution mass $w_{b,i}$, and average over B samples to obtain $\bar{x}$:

$$w_{b,i} = \left\| \mathbb{A}^{(i)}[\mathbf{S}^{(i,n)}]_{b,:} \right\|_1, \qquad \bar{x}_b = \frac{\sum_{i:\, w_{b,i}>0} w_{b,i} x_{b,i}}{\sum_{i=1}^{M} w_{b,i}}, \qquad \bar{x} = \frac{1}{B}\sum_{b=1}^{B} \bar{x}_b, \tag{144}$$

with $\bar{x}_b = 0$ when $\sum_{i=1}^{M} w_{b,i} = 0$. Values $x_{b,i}$ are evaluated only where $w_{b,i} > 0$.

Algorithm 2 Single-Token MHAttention $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ Statistics

Require: Query token $q_\star$, selected heads, partition states $\boldsymbol{x}_j = \gamma(\alpha_j)$ for $j = 0, \dots, n$.
Ensure: $\mathbb{V}_{\mathrm{attn}}^{(h,q_\star)}$ for each selected head.
  for each selected head h **do**
    Initialize $\mathbf{S}^{(h,q_\star)} \leftarrow 0$ and $\mathbb{A}_{\mathrm{attn}}^{(h,q_\star)}[\mathbf{S}] \leftarrow 0$. ▷ store one token-level field
  end for
  Store the baseline attention row for $q_\star$ and values for all source tokens in the selected heads.
  for $j = 1, \dots, n$ **do**
    Run one forward/backward pass for F_j. ▷ $O(T_{\mathrm{step}})$ time
    for each selected head h **do**
      Form $\Delta\mathbf{S}^{(h,q_\star,j)}$ by Eq. 136 with $q = q_\star$. ▷ uses current tensors
      $\mathbf{S}^{(h,q_\star)} \leftarrow \mathbf{S}^{(h,q_\star)} + \Delta\mathbf{S}^{(h,q_\star,j)}$.
      For each sample b, add $\sum_{k=1}^{L} \Delta\mathbf{S}_{b,k,:}^{(h,q_\star,j)}$ to $\mathbb{A}_{\mathrm{attn}}^{(h,q_\star)}[\mathbf{S}]_{b,:}$.
    end for
    Update $(\mathbf{A}^{(j-1)}, \mathbf{V}^{(j-1)}) \leftarrow (\mathbf{A}^{(j)}, \mathbf{V}^{(j)})$.
  end for
  for each selected head h **do**
    Compute the local norms and $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ from the final field by Eq. 143.
  end for
  If a module-level value is reported, apply Eq. 144 over the selected heads and then average samples.

This weighting also reduces the influence of local values with small $w_{b,i}$, including those affected by strong cancellation. When $\sum_{i=1}^{M} w_{b,i}$ is bounded away from zero, input-field perturbations small relative to this sum also produce small changes in $w_{b,i} / \sum_{j=1}^{M} w_{b,j}$. Together with local perturbation control, this motivates the reduction as a stable module summary; its numerical sensitivity is examined in Appendix G.2.

Empirical validation. We benchmarked the single-sample, single-layer $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ computation on a synthetic 224×224 input with a zero baseline and batch size one. Timings were measured after warmup on an RTX 4090 GPU using PyTorch in eager mode. Its endpoint-inclusive nonuniform path partitions were derived from Gauss–Legendre abscissae; the $n = 64$ run provides the finest-discretization reference in Table D.1. Multi-layer costs across integration precisions are reported in Table G.4.

Table D.1: Runtime, normalized peak memory, and path-discretization error for single-sample, single-layer MHAttention and depthwise convolution $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$. The training-step baselines are $(T_{\mathrm{step}}, \mathrm{Mem}_{\mathrm{step}}) = (0.017512\,\mathrm{s}, 938.18\,\mathrm{MiB})$ for ViT-B/32 and $(0.025703\,\mathrm{s}, 3218.94\,\mathrm{MiB})$ for ConvNeXt-Large-D.

Probe	n	Time (s)	T/T_{step}	$\mathrm{Mem}/\mathrm{Mem}_{\mathrm{step}}$	Median relative error
ViT-B/32 MHA	4	0.1032	5.89	0.752	0.111627
	8	0.2040	11.65	0.800	0.089786
	16	0.3869	22.10	0.799	0.040141
	32	0.7306	41.72	0.816	0.007668
	64	1.4382	82.12	0.816	0.000000
ConvNeXt-Large-D DWConv	4	0.1552	6.04	0.641	0.141101
	8	0.2899	11.28	0.645	0.075439
	16	0.5704	22.19	0.644	0.027974
	32	1.0179	39.60	0.646	0.008592
	64	1.9216	74.76	0.645	0.000000

The final column is the median coordinate-wise relative error defined by Eq. 145:

$$E_n = \operatorname*{median}_{k \in \Omega} \frac{|\mathbb{V}_{n,k} - \mathbb{V}_{64,k}|}{|\mathbb{V}_{64,k}|}, \tag{145}$$

where k indexes entries of the single-layer $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ tensor. For ViT-B/32, Ω is the non-zero support of the $n = 64$ reference; for ConvNeXt-Large-D, it is the full flattened tensor.

E Experimental Details: Grokking

E.1 Experimental Setup

Tasks and datasets. We study four finite algorithmic tasks. Add P97 maps an ordered pair $(a, b) \in \mathbb{Z}_{97}^2$ to $(a + b) \bmod 97$. For the remaining tasks, let $a, b \in S_5$ and define permutation composition by $(a \circ b)(i) = a(b(i))$. S5 Product, S5 Conjugation, and S5 Sandwich return $a \circ b$, $(a \circ b) \circ a^{-1}$, and $(a \circ b) \circ a$, respectively. Each example is presented as the four-token sequence $[a, \circ, b, =]$, and the model predicts the result at the final position. Add P97 contains all $97^2 = 9{,}409$ ordered pairs, while each S_5 task contains all $(5!)^2 = 14{,}400$ ordered pairs. For each seed, the complete set is randomly ordered once using the dataset seed; the specified training fraction is a prefix and the remaining pairs form a fixed validation split throughout the trajectory. The low/center/high training fractions are $0.20/0.35/0.45$ for Add P97, $0.30/0.35/0.40$ for S5 Product, $0.40/0.50/0.60$ for S5 Conjugation, and $0.40/0.45/0.50$ for S5 Sandwich.

Model and optimization. All tasks use the same two-block pre-norm Transformer with hidden size 128, four attention heads of dimension 32, learned token and positional embeddings, and dropout 0.1. Each block has a $128 \to 512 \to 128$ GELU MLP; a final layer normalization and linear classifier produce the tokenwise logits. Each task contributes five configurations: a task-specific center, low and high training fractions at the center learning rate, and low and high learning rates at the center training fraction. The corresponding low/center/high learning rates are $5{\times}10^{-4}/10^{-3}/2{\times}10^{-3}$ for Add P97, $8\times10^{-4}/10^{-3}/1.25\times10^{-3}$ for S5 Product, and $1.6\times10^{-3}/2\times10^{-3}/2.5\times10^{-3}$ for both S5 Conjugation and S5 Sandwich. Weight decay is fixed within task at 0.05, 0.5, 0.3, and 0.3, respectively. Seeds 0–4 for every configuration give 100 training trajectories. We use AdamW with $\beta = (0.9, 0.98)$, $\epsilon = 10^{-6}$, batch size 512, a constant learning rate after 10 warm-up steps, and gradient-norm clipping at 1.0. Each trajectory terminates at the smaller of 60,000 steps and twice the first refined $\mathrm{Val\ Acc} = 99\%$ crossing.

Checkpoint construction. Validation accuracy is evaluated deterministically on the complete fixed validation split. We locate the first crossings at $\mathrm{Train\ Acc}, \mathrm{Val\ Acc} = 10\%, 20\%, \ldots, 90\%, 99\%$; checkpoint replay and bisection refine each crossing to an integer step while restoring the complete training state. Structural measurements start at step 200 and are sampled every 200 optimizer steps. The main response analysis uses the union of these dense measurements and all reached threshold points; if a threshold event coincides with a regular checkpoint step, the exact threshold evaluation is used. At a threshold-only step, the crossing-axis accuracy is taken from the replay record and the other behavioral quantities are linearly interpolated on the dense trajectory. All training thresholds and all but three validation thresholds are observed; the remaining 99% validation events are treated as right-censored. For trajectory visualizations, additional measurements every ten steps through step 400 replace the early portion of the regular grid; this visualization schedule is separate from the main response sample.

Attribution measurements and aggregation nodes. The formalization in Definition 3.1 decomposes the attributed computation at a given node over its incoming computation-graph edges, producing the route-indexed pre-aggregation attribution field $\mathbf{S}$ before those edges are combined by $\mathbb{A}$. The scalar payoff for an example with correct class $y^\star$ is the final-token logit margin

$$F(x; y^\star) = f_{y^\star}(x) - \log \sum_{c \neq y^\star} \exp f_c(x). \tag{146}$$

The model input is represented as a one-hot tensor. The endpoint $x^{(1)}$ is the observed sequence, whereas the baseline $x^{(0)}$ sets operand positions 0 and 2 to zero and copies the observed operator and target-token marker vectors at positions 1 and 3. Thus only the two operand inputs vary along the path; positional embeddings, model parameters, the operator token, and the final $=$ token remain fixed. This model-input intervention is distinct from the module routes: every probed module retains all four contextualized source positions.

The outer path is a single straight segment from $x^{(0)}$ to $x^{(1)}$. We use 32 endpoint-transition intervals with endpoint-inclusive Gauss–Legendre refinement. The attention and MLP increments follow Eqs. 136 and 142.

We probe attention aggregation and the MLP output projection in both blocks. At an attention site, (h, q) is the native local node, source token k is the route coordinate, and per-head activation coordinate d is unreduced; the local field therefore has axes (k, d) at fixed (b, h, q). At an MLP site, the captured input is the complete post-GELU/dropout tensor $H \in \mathbb{R}^{B\times L\times 512}$ and the output is the residual-stream update in $\mathbb{R}^{B\times L\times 128}$. Token q is the local node, hidden unit i is the route coordinate, output coordinate o is unreduced, and the projection bias is fixed context; the local field has axes (i, o) at fixed (b, q), following Eq. 133.

Local statistics and their reduction follow Eqs. 143 and 144. For attention, the all-token scope includes all head–query pairs and the target-token scope includes heads at the final query position. For the MLP, these scopes include all token nodes or only the final token. The two module types, two layers, and two scopes form eight checkpoint-level profile coordinates. The logarithmic dimensions are reduced directly, preserving their difference as the averaged $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$.

All attribution measurements use the training split. Fixed, nested probes $64 \subset 256 \subset 1024$ preserve sample IDs and order at every dense and threshold checkpoint. The primary results use $N = 1024$; $N = 64$ and 256 are sampling sensitivity checks. $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ is recomputed as $\log_2 \mathrm{D}(\mathbf{S}) - \log_2 \mathrm{D}(\mathbb{A}[\mathbf{S}])$ throughout the dense–threshold union.

LID external reference. As an external representation-geometry reference related to the LID analysis of grokking (Amsaleg et al., 2015; Facco et al., 2017; Ruppik et al., 2025), we compute mean-local TwoNN LID on contextual hidden states immediately after residual blocks 0 and 1 and before the final Transformer normalization. The all-token scope uses positions $\{0, 1, 2, 3\}$ for each example, and the operand-position scope uses $\{0, 2\}$ after contextual mixing. Crossing two blocks with two scopes gives a four-site profile. The primary estimator uses pointwise TwoNN fitted with $k = 64$ nearest neighbors without smoothing, evaluated on the same fixed $N = 1024$ training-example prefix and matched checkpoint trajectory as the attribution profiles.

Global TwoNN on the same four residual-state sites is included as an estimator and sample-size comparative baseline at $N = 64, 256, 1024$. Here N always counts the fixed training examples, rather than the number of token-state vectors. Duplicate hidden-state vectors are removed prior to estimator fitting.

ExPLAIND importance references. We also evaluate ExPLAIND regularization importance (Eichin et al., 2026), adapted to the scalar payoff defined in Eq. 146. ❶ **Importance definition.** At checkpoint t, let $\Delta\theta_t = \theta_{t+1} - \theta_t$ denote the actual next optimizer update and $\Delta\theta_t^{\mathrm{reg}}$ its AdamW weight-decay component. For each parameter updated by AdamW, $\Delta\theta_{t,i}^{\mathrm{reg}} = -\eta_{t,i}\lambda_i\theta_{t,i}$, using the learning rate and weight decay of its optimizer parameter group at that update; parameters skipped by the optimizer contribute zero. For parameter group Θ_l, we compute

$$I_{t,l}^{\mathrm{reg}} = \frac{1}{N}\sum_{x\in P_N}\left|\sum_{i\in\Theta_l}\Delta\theta_{t,i}^{\mathrm{reg}}\int_0^1 \frac{\partial F_{\theta_t+u\Delta\theta_t}(x)}{\partial\theta_i}\,du\right|.$$

For the supplementary kernel-importance comparison, we replace $\Delta\theta_t^{\mathrm{reg}}$ in the same expression with $\Delta\theta_t^K = \Delta\theta_t - \Delta\theta_t^{\mathrm{reg}}$. This component includes the effects of the optimizer's momentum and preconditioning. ❷ **Computation.** For both components, the integral uses 32 Gauss–Legendre nodes along the simultaneous parameter update. We sum parameter contributions within each group before taking the absolute value for each probe and averaging probes. Measurements use the same nested training probes and all 18,590 checkpoints as the attribution profiles, with deterministic evaluation along the parameter path. ❸ **Evaluation.** We evaluate 8 candidate parameter groups: the embedding parameters, the output-head parameters, and three separate groups within each of the two Transformer blocks—attention, MLP, and normalization parameters. Responses use the original checkpoint behavior at t and the same selection and resampling protocol as our method.

Phases and pooled property responses. ❶ **Phase definitions.** Phase labels are assigned in the following mutually exclusive order: accurate when both accuracies equal 100%; generalized when

both are at least 95% but the checkpoint is not accurate; memorized when training accuracy is at least 95% and validation accuracy is below 50%; underfitting when training accuracy is below 95%; and transition otherwise. ❷ **Response statistics.** For each task, two selection seeds contribute ten trajectories and the remaining three seeds contribute fifteen evaluation trajectories. Each evaluation uses one of the ten choices of two selection seeds. We pool checkpoints within each side, assigning equal total weight to each trajectory and equal weight to its checkpoints. Different tasks are analyzed separately. Phase responsiveness is the weighted between-phase sum of squares divided by the weighted total sum of squares of the measure. Accuracy responsiveness is reported as

$$|\rho| = \left|\mathrm{corr}_w\big(\mathrm{rank}_w(\text{measure}), \mathrm{rank}_w(A)\big)\right|, \qquad R^2 = \mathrm{corr}_w^2(\text{measure}, A), \tag{147}$$

where A is training or validation accuracy and rank_w is the weighted midrank. A tied value receives the probability mass below its tie group plus half the mass of the group. Uniform weights recover ordinary midranks up to an affine transformation. The Pearson R^2 is the simple squared correlation, equivalently the fit of a single-measure linear model with an intercept. We report pooled responses because validation accuracy is often nearly monotonic within a training trajectory. In that setting, a within-trajectory Spearman response may only reflects whether the measure follows a monotonic ordering over training time. Pooling trajectories with different learning dynamics also evaluates the ordering of observations across runs, helping distinguish association with validation accuracy from monotonic progress along an individual trajectory. We further examine temporal associations through the controls in Appendix E.2. ❸ **Site selection and averaging.** Each measure and response definition selects its own site on the selection pool. We recompute the response at that frozen site in the evaluation pool, average over all ten two-seed selection combinations, and then give each task equal weight. Enumerating this finite partition space removes Monte Carlo variation from seed-partition averaging. ❹ **Resampling.** For uncertainty, 5,000 bootstrap draws resample the five configuration blocks within each task, sharing configuration multiplicities across seeds, measures, sample sizes, and response definitions. Each draw samples one of the ten seed partitions, shared across tasks and measures, to retain the variability of a single selection–evaluation split. The random stream is initialized with seed 0. Each draw recomputes pooled moments and weighted ranks, reselects sites, and evaluates them on held-out seeds. Task-specific intervals condition on the task. Overall intervals keep the four observed tasks fixed and average their responses with equal weights in every draw. The resulting distribution includes within-task configuration variation, random seed-partition assignment, and changes in the selected site. It provides a more demanding robustness assessment than fixing the selection–evaluation split, while conditioning on the observed tasks, trained runs, and measurement samples. Tables report 95% bootstrap percentile intervals for this resampling procedure, interpreted as described in Appendix G, with bold indicating the largest point estimate rather than a significance decision. All draws are estimable.

E.2 Detailed Measurement-Profile Results

Table E.1 resolves the pooled phase, training-accuracy, and validation-accuracy responses by task. We report absolute Spearman and squared Pearson accuracy responses separately, with matching resampling intervals.

Tracking delayed generalization. Within every task, $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ has stronger phase and validation responses than both LID and $\log_2 \mathrm{D}(\mathbb{A}[\mathbf{S}])$. Although this advantage does not extend to every training-accuracy response, validation progress is central to grokking because generalization improves after training accuracy has saturated. Although Reg. Importance has a slightly higher validation R^2 response on S5 Conjugation, $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ leads on the other three tasks and achieves the strongest mean validation responses. Even where $\log_2 \mathrm{D}(\mathbf{S})$ leads in some phase and validation rank responses by less than 0.012, $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ has stronger mean phase and validation responses, supporting its role in tracking delayed generalization. The input-side dimension measures the effective diversity of microscopic attribution contributions before aggregation. Locally, $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ normalizes this diversity by the output-side effective dimension through $\log_2[\mathrm{D}(\mathbf{S})/\mathrm{D}(\mathbb{A}[\mathbf{S}])]$, measuring the degree of local attribution convergence. Its stronger mean phase and validation responses support this aggregation-relative measurement of organization during generalization. Figure G.1 reports the paired intervals for these mean differences.

Sample-size sensitivity. Table E.2 applies the same pooled protocol to $N = 64, 256, 1024$. All measures use matched checkpoint support; the sampled examples are nested prefixes of the same

Table E.1: Task-resolved pooled grokking responses. Entries give point estimates and 95% resampling intervals; bold marks the largest displayed response within each task.

Group	Response	LID (mean-local)	Reg. Importance	Kernel Importance	$\log_2 \mathrm{D}(\mathbf{S})$	$\log_2 \mathrm{D}(\mathbb{A}[\mathbf{S}])$	$\mathbb{V}_{\mathbb{A}}(\mathbf{S})$
Add P97	Phase η^2	0.682 [0.588, 0.784]	0.374 [0.230, 0.819]	0.272 [0.193, 0.558]	0.567 [0.458, 0.789]	0.555 [0.354, 0.701]	**0.724** [0.590, 0.860]
	Train $\lvert\rho\rvert$	0.830 [0.746, 0.896]	0.575 [0.400, 0.915]	0.656 [0.464, 0.858]	0.673 [0.460, 0.903]	0.590 [0.312, 0.681]	**0.837** [0.746, 0.908]
	Train R^2	0.413 [0.310, 0.531]	0.332 [0.198, 0.795]	0.181 [0.115, 0.463]	**0.666** [0.478, 0.780]	0.323 [0.174, 0.487]	0.471 [0.368, 0.671]
	Val. $\lvert\rho\rvert$	0.720 [0.574, 0.893]	0.479 [0.316, 0.872]	0.575 [0.382, 0.857]	0.636 [0.410, 0.890]	0.657 [0.477, 0.793]	**0.801** [0.721, 0.921]
	Val. R^2	0.420 [0.286, 0.765]	0.259 [0.088, 0.706]	0.242 [0.137, 0.560]	0.441 [0.266, 0.766]	0.395 [0.194, 0.589]	**0.686** [0.524, 0.870]
S5 Product	Phase η^2	0.698 [0.468, 0.771]	0.632 [0.510, 0.741]	0.653 [0.550, 0.831]	**0.926** [0.907, 0.942]	0.720 [0.625, 0.783]	0.924 [0.869, 0.952]
	Train $\lvert\rho\rvert$	0.577 [0.414, 0.704]	**0.858** [0.779, 0.931]	0.846 [0.790, 0.908]	0.824 [0.779, 0.860]	0.813 [0.773, 0.847]	0.802 [0.753, 0.847]
	Train R^2	**0.663** [0.576, 0.741]	0.655 [0.517, 0.742]	0.562 [0.497, 0.645]	0.130 [0.028, 0.207]	0.449 [0.346, 0.537]	0.384 [0.161, 0.568]
	Val. $\lvert\rho\rvert$	0.560 [0.415, 0.714]	0.833 [0.765, 0.917]	**0.847** [0.794, 0.908]	0.834 [0.797, 0.879]	0.794 [0.757, 0.846]	0.828 [0.787, 0.872]
	Val. R^2	0.568 [0.446, 0.694]	0.538 [0.418, 0.642]	0.635 [0.543, 0.795]	0.888 [0.859, 0.915]	0.638 [0.555, 0.712]	**0.894** [0.832, 0.933]
S5 Conjugation	Phase η^2	0.420 [0.133, 0.561]	0.618 [0.451, 0.731]	0.354 [0.183, 0.497]	**0.750** [0.607, 0.824]	0.565 [0.413, 0.719]	0.742 [0.665, 0.833]
	Train $\lvert\rho\rvert$	0.620 [0.508, 0.751]	**0.774** [0.633, 0.917]	0.542 [0.309, 0.688]	0.771 [0.652, 0.847]	0.636 [0.490, 0.763]	0.772 [0.677, 0.848]
	Train R^2	0.363 [0.302, 0.470]	**0.673** [0.554, 0.771]	0.636 [0.531, 0.738]	0.328 [0.227, 0.409]	0.275 [0.150, 0.425]	0.331 [0.263, 0.471]
	Val. $\lvert\rho\rvert$	0.642 [0.271, 0.780]	**0.836** [0.702, 0.943]	0.619 [0.383, 0.734]	0.804 [0.686, 0.880]	0.757 [0.568, 0.843]	0.793 [0.690, 0.850]
	Val. R^2	0.439 [0.184, 0.604]	**0.702** [0.497, 0.867]	0.422 [0.244, 0.540]	0.623 [0.533, 0.759]	0.576 [0.409, 0.732]	0.667 [0.538, 0.788]
S5 Sandwich	Phase η^2	0.655 [0.582, 0.752]	0.676 [0.607, 0.781]	0.647 [0.602, 0.721]	0.802 [0.671, 0.861]	0.607 [0.483, 0.728]	**0.888** [0.865, 0.915]
	Train $\lvert\rho\rvert$	0.832 [0.792, 0.877]	0.769 [0.724, 0.912]	0.825 [0.786, 0.873]	**0.846** [0.828, 0.869]	0.732 [0.643, 0.770]	0.838 [0.794, 0.867]
	Train R^2	0.270 [0.210, 0.321]	0.703 [0.597, 0.851]	**0.713** [0.645, 0.797]	0.383 [0.323, 0.433]	0.164 [0.089, 0.223]	0.429 [0.269, 0.486]
	Val. $\lvert\rho\rvert$	0.849 [0.789, 0.918]	0.831 [0.793, 0.915]	0.833 [0.804, 0.876]	**0.890** [0.848, 0.918]	0.820 [0.725, 0.862]	0.881 [0.839, 0.915]
	Val. R^2	0.619 [0.558, 0.697]	0.589 [0.531, 0.713]	0.583 [0.540, 0.665]	0.695 [0.610, 0.787]	0.497 [0.238, 0.602]	**0.829** [0.792, 0.871]
Mean	Phase η^2	0.614 [0.535, 0.669]	0.575 [0.529, 0.713]	0.481 [0.448, 0.591]	0.761 [0.718, 0.822]	0.612 [0.532, 0.672]	**0.819** [0.777, 0.864]
	Train $\lvert\rho\rvert$	0.715 [0.650, 0.768]	0.744 [0.696, 0.856]	0.717 [0.653, 0.791]	0.778 [0.720, 0.841]	0.693 [0.610, 0.738]	**0.812** [0.766, 0.845]
	Train R^2	0.427 [0.382, 0.477]	**0.591** [0.543, 0.731]	0.523 [0.486, 0.610]	0.377 [0.318, 0.426]	0.303 [0.241, 0.370]	0.404 [0.348, 0.492]
	Val. $\lvert\rho\rvert$	0.693 [0.611, 0.768]	0.745 [0.706, 0.857]	0.719 [0.654, 0.804]	0.791 [0.729, 0.860]	0.757 [0.688, 0.803]	**0.826** [0.795, 0.868]
	Val. R^2	0.511 [0.435, 0.627]	0.522 [0.470, 0.660]	0.470 [0.432, 0.580]	0.662 [0.617, 0.760]	0.527 [0.437, 0.595]	**0.769** [0.718, 0.835]

fixed order. Phase and both accuracy responses are recomputed with independent site selection at each sample size. The $N = 1024$ rows reproduce the pooled main-experiment results, and the same bootstrap draws are shared across sample sizes. Both Mean-local and Global TwoNN are included. At every sample size, $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ has stronger mean phase and validation responses than both LID estimates and both importance measures. Each of these three $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ responses changes by

less than 0.025 across the evaluated sample sizes, supporting stable measurement of generalization-related organization over this range.

Table E.2: Sample-size sensitivity under matched pooled evaluation. Entries give mean responses and 95% resampling intervals.

Group	Response	LID (mean-local)	LID (global)	Reg. Importance	Kernel Importance	$\log_2 \mathrm{D}(\mathbf{S})$	$\log_2 \mathrm{D}(\mathbb{A}[\mathbf{S}])$	$\mathbb{V}_{\mathbb{A}}(\mathbf{S})$
$N = 64$	Phase η^2	0.446 [0.377, 0.545]	0.354 [0.324, 0.496]	0.569 [0.526, 0.707]	0.458 [0.426, 0.561]	0.751 [0.706, 0.817]	0.577 [0.505, 0.642]	**0.806** [0.762, 0.855]
	Train $\lvert\rho\rvert$	0.632 [0.559, 0.699]	0.442 [0.400, 0.625]	0.738 [0.692, 0.849]	0.705 [0.647, 0.780]	0.776 [0.719, 0.840]	0.673 [0.594, 0.720]	**0.812** [0.772, 0.843]
	Train R^2	0.246 [0.189, 0.297]	0.254 [0.206, 0.321]	**0.586** [0.544, 0.710]	0.468 [0.433, 0.542]	0.374 [0.318, 0.419]	0.288 [0.224, 0.351]	0.404 [0.341, 0.488]
	Val. $\lvert\rho\rvert$	0.698 [0.611, 0.754]	0.581 [0.467, 0.666]	0.741 [0.699, 0.851]	0.706 [0.643, 0.790]	0.784 [0.712, 0.854]	0.737 [0.668, 0.787]	**0.812** [0.784, 0.861]
	Val. R^2	0.417 [0.340, 0.500]	0.303 [0.236, 0.405]	0.521 [0.467, 0.658]	0.448 [0.412, 0.547]	0.645 [0.604, 0.752]	0.504 [0.412, 0.571]	**0.744** [0.694, 0.825]
$N = 256$	Phase η^2	0.578 [0.540, 0.662]	0.429 [0.372, 0.534]	0.577 [0.527, 0.713]	0.476 [0.444, 0.585]	0.757 [0.721, 0.821]	0.606 [0.522, 0.668]	**0.817** [0.777, 0.862]
	Train $\lvert\rho\rvert$	0.769 [0.692, 0.820]	0.506 [0.422, 0.618]	0.744 [0.695, 0.854]	0.716 [0.654, 0.790]	0.779 [0.720, 0.842]	0.690 [0.611, 0.735]	**0.814** [0.763, 0.845]
	Train R^2	0.446 [0.367, 0.494]	0.336 [0.261, 0.413]	**0.589** [0.544, 0.726]	0.512 [0.474, 0.596]	0.380 [0.323, 0.425]	0.297 [0.233, 0.365]	0.406 [0.348, 0.492]
	Val. $\lvert\rho\rvert$	0.777 [0.721, 0.847]	0.494 [0.436, 0.629]	0.744 [0.704, 0.857]	0.716 [0.653, 0.801]	0.793 [0.731, 0.862]	0.756 [0.689, 0.801]	**0.825** [0.793, 0.869]
	Val. R^2	0.556 [0.482, 0.661]	0.292 [0.232, 0.432]	0.521 [0.469, 0.660]	0.467 [0.427, 0.573]	0.664 [0.616, 0.760]	0.523 [0.432, 0.590]	**0.764** [0.712, 0.832]
$N = 1024$	Phase η^2	0.614 [0.535, 0.669]	0.353 [0.302, 0.469]	0.575 [0.529, 0.713]	0.481 [0.448, 0.591]	0.761 [0.718, 0.822]	0.612 [0.532, 0.672]	**0.819** [0.777, 0.864]
	Train $\lvert\rho\rvert$	0.715 [0.650, 0.768]	0.533 [0.429, 0.593]	0.744 [0.696, 0.856]	0.717 [0.653, 0.791]	0.778 [0.720, 0.841]	0.693 [0.610, 0.738]	**0.812** [0.766, 0.845]
	Train R^2	0.427 [0.382, 0.477]	0.474 [0.389, 0.547]	**0.591** [0.543, 0.731]	0.523 [0.486, 0.610]	0.377 [0.318, 0.426]	0.303 [0.241, 0.370]	0.404 [0.348, 0.492]
	Val. $\lvert\rho\rvert$	0.693 [0.611, 0.768]	0.448 [0.357, 0.561]	0.745 [0.706, 0.857]	0.719 [0.654, 0.804]	0.791 [0.729, 0.860]	0.757 [0.688, 0.803]	**0.826** [0.795, 0.868]
	Val. R^2	0.511 [0.435, 0.627]	0.299 [0.213, 0.444]	0.522 [0.470, 0.660]	0.470 [0.432, 0.580]	0.662 [0.617, 0.760]	0.527 [0.437, 0.595]	**0.769** [0.718, 0.835]

Controls for training accuracy and time. We also report absolute Spearman partial correlations and squared Pearson partial correlations with validation accuracy, controlling training accuracy, time, or both. For Pearson, we regress the original measure and validation accuracy separately on an intercept and the original control values, using $\log t$ for time, then square the weighted correlation of the residuals. For Spearman, the measure, validation accuracy, and controls are first converted to weighted midranks; the same residual calculation yields a signed coefficient whose absolute value is reported. The rank analysis evaluates the ordering of validation progress without requiring proportional changes on the original measurement scales. It is invariant to strictly increasing transformations of the measure, accuracy, and controls, while adjusting for linear associations among their ranks. In particular, $\mathrm{rank}_w(\log t) = \mathrm{rank}_w(t)$, so the two time controls operate on different representations.

Site selection and evaluation use the same task-specific trajectory pools and weighting as the main analysis, with an independent site search for each measure and control definition. Ranks and residual fits are recomputed in each pool. These statistics describe residual association. Squared partial Pearson measures the fraction of residual validation variation explained by the added measure; absolute partial Spearman measures the strength of the residual rank association.

Although LID has the largest mean squared Pearson partial response (Table E.3), $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ has the largest mean absolute Spearman partial response under all three control sets: 0.482, 0.584, and 0.486 under training, time, and joint controls, respectively, compared with 0.349, 0.402, and 0.342 for LID. This difference may reflect nonlinear components in the association between $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ and

Table E.3: Validation responses after controlling training accuracy, logarithmic time, or both. Entries give means and 95% resampling intervals.

Group	Response	LID (mean-local)	Reg. Importance	Kernel Importance	$\log_2 \mathrm{D}(\mathbf{S})$	$\log_2 \mathrm{D}(\mathbb{A}[\mathbf{S}])$	$\mathbb{V}_{\mathbb{A}}(\mathbf{S})$
Partial $\lvert\rho\rvert$	Training Acc	0.349 [0.214, 0.453]	0.359 [0.332, 0.555]	0.369 [0.290, 0.486]	0.458 [0.340, 0.584]	0.400 [0.302, 0.480]	**0.482** [0.386, 0.563]
	$\log t$	0.402 [0.258, 0.501]	0.478 [0.367, 0.659]	0.460 [0.352, 0.586]	0.493 [0.317, 0.629]	0.439 [0.292, 0.526]	**0.584** [0.396, 0.646]
	Both	0.342 [0.202, 0.439]	0.319 [0.268, 0.543]	0.354 [0.270, 0.494]	0.393 [0.262, 0.549]	0.360 [0.241, 0.441]	**0.486** [0.328, 0.568]
Partial R^2	Training Acc	**0.762** [0.682, 0.834]	0.460 [0.421, 0.612]	0.445 [0.411, 0.565]	0.644 [0.601, 0.739]	0.510 [0.443, 0.582]	0.739 [0.684, 0.809]
	$\log t$	**0.630** [0.492, 0.742]	0.351 [0.321, 0.511]	0.303 [0.261, 0.444]	0.504 [0.438, 0.609]	0.289 [0.200, 0.389]	0.606 [0.517, 0.675]
	Both	**0.699** [0.595, 0.791]	0.310 [0.257, 0.489]	0.283 [0.240, 0.422]	0.497 [0.416, 0.581]	0.316 [0.219, 0.396]	0.618 [0.504, 0.666]

validation accuracy after adjustment for the specified controls. Its corresponding squared Pearson partial responses are 0.739, 0.606, and 0.618. Under all three control sets, its mean responses exceed those of Reg. Importance and Kernel Importance for both statistics. The consistently stronger residual rank association supports $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ as an indicator of the ordering of validation progress beyond the rank associations captured by training accuracy and time. This complements the continued structural changes after training accuracy saturates in Figure E.1, connecting the observed internal evolution to validation progress across trajectories.

Trajectory views. Figure E.1 shows that $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ continues to change after training accuracy has saturated in several trajectories, with marked changes accompanying validation progress. Plotting behavior and sitewise measurements on the same logarithmic training axis makes this internal evolution visible.

Stage-level net changes. Table E.4 compares the first measured checkpoint, the first measured checkpoint with training accuracy at least 95%, and the first with validation accuracy at least 99%. Writing $\Delta\mathbb{V}_s = \mathbb{V}_s(t_b) - \mathbb{V}_s(t_a)$ for the net change at site s between the two stage endpoints, we classify

$$\begin{aligned} \text{All increase}: &\quad \min_{s=1,\dots,8} \Delta\mathbb{V}_s > 10^{-3}, \\ \text{All decrease}: &\quad \max_{s=1,\dots,8} \Delta\mathbb{V}_s < -10^{-3}. \end{aligned} \tag{148}$$

The remaining trajectories are classified as *Other*. Changes are reported in bits. The three runs without an observed subsequent Val99 endpoint contribute only to the first interval. These comparisons describe net changes between endpoints and allow fluctuations within each interval. Before Train95, 6 of 100 trajectories increase at every site and none decreases at every site. From Train95 to Val99, none of 97 trajectories increases at every site and 34 decrease at every site. This stage asymmetry shows that generalization is accompanied by a reorganization of the $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ profile across layers, including uniform decreases and changes with different directions across sites.

Task heterogeneity in stage net changes. Uniform net decreases from Train95 to Val99 occur in 17/25 S5 Product, 10/22 S5 Conjugation, and 7/25 S5 Sandwich runs, whereas all 25 Add P97 runs fall into *Other*, which includes mixed directions or changes within the tolerance (Table E.4). This task difference may be caused by the closer timing of training and validation convergence in Add P97, as illustrated in Figure E.1. The resulting overlap may mix attribution reorganization associated with fitting the training data and with generalization, obscuring a common direction across sites. In the S_5 tasks, the greater temporal separation allows reorganization associated with generalization to be more distinct from that associated with training during the Train95–Val99 interval.

Phase-specific diagram. We fit two-site phase-probability diagrams to the regular $N = 1024$ trajectories, combining early measurements through step 400 with later dense measurements from

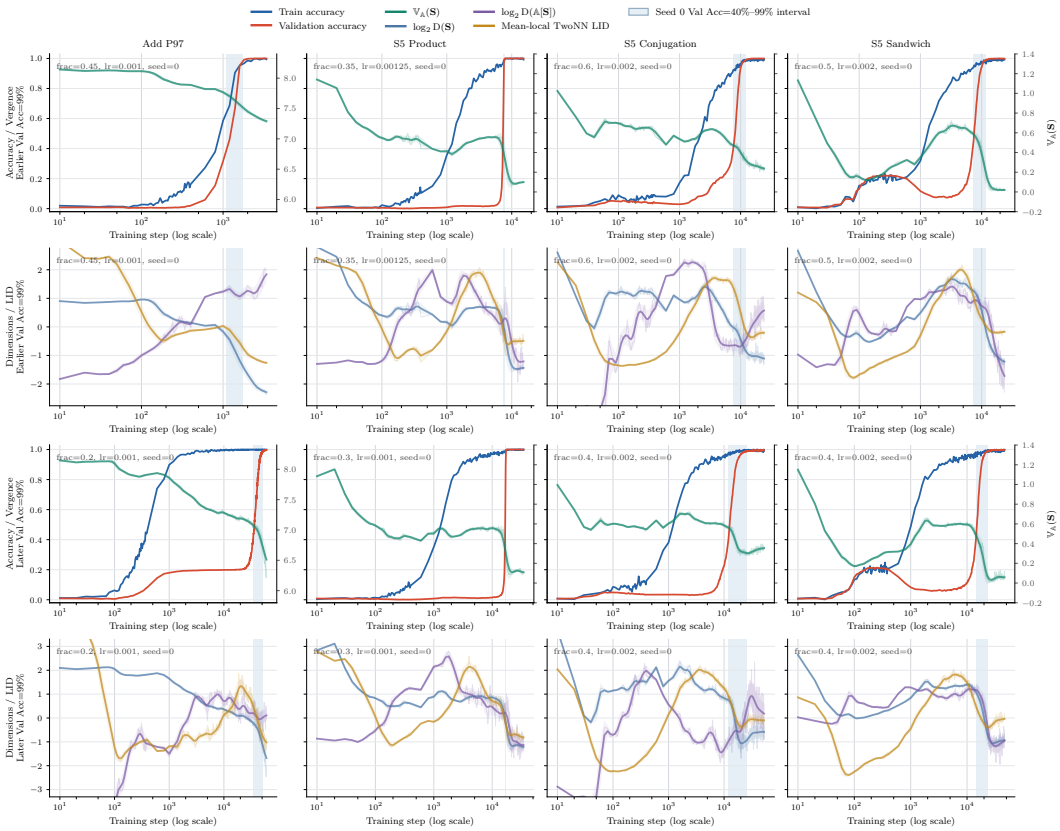

Figure E.1: Behavioral and structural trajectories for the four tasks. $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ reveals internal changes after training accuracy has saturated. Each task shows configurations with earlier and later median validation-99% crossings. Upper rows overlay training/validation accuracy and $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$; lower rows show the two dimensional terms and mean-local TwoNN LID. All measurement curves use $N = 1024$ training examples. Curves use seed 0, and blue bands mark the Val40–Val99 interval.

Table E.4: Stage-level net changes across all eight $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ sites. Uniform net increases occur before Train95 but not between Train95 and Val99; most trajectories have mixed directions or include unchanged sites.

Group	Task	Runs	All increase	All decrease	Other
First to Train95	Add P97	25	0	0	25
	S5 Product	25	0	0	25
	S5 Conjugation	25	3	0	22
	S5 Sandwich	25	3	0	22
	All	100	6	0	94
Train95 to Val99	Add P97	25	0	0	25
	S5 Product	25	0	17	8
	S5 Conjugation	22	0	10	12
	S5 Sandwich	25	0	7	18
	All	97	0	34	63

the five configurations and five seeds of each task. A phase is displayed as dominant when its fitted probability reaches 50%; non-dominant regions mix the fitted phase probabilities.

Figure E.2 exposes task-dependent regions associated with memorization and generalization. The trajectories bend or turn through these regions as attribution organization changes. Together with the stage-level net changes, these patterns raise the question of how memorization-associated organization may obstruct the subsequent reorganization required for generalization.

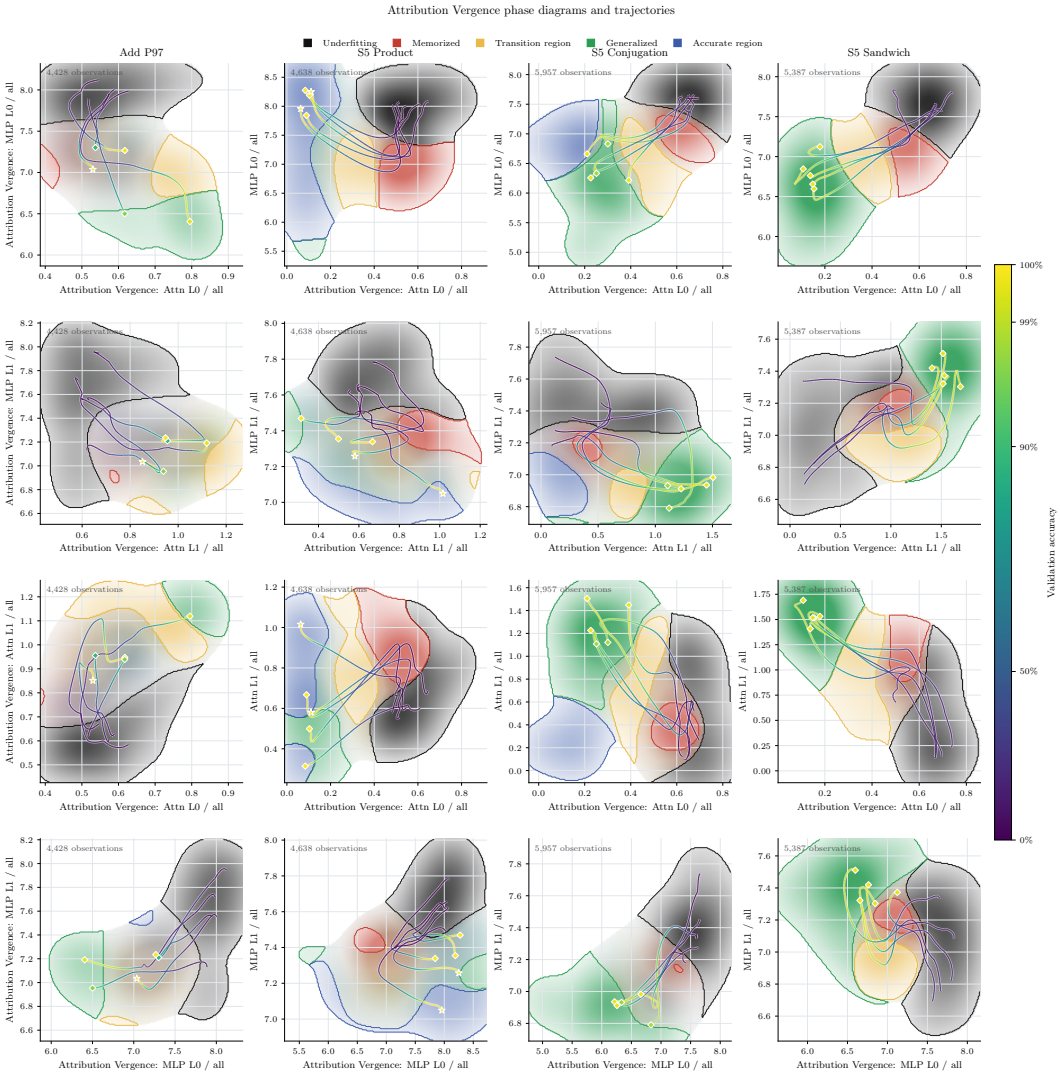

Figure E.2: Empirical two-site $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ phase-probability diagrams and training trajectories across grokking stages. Rows 1–2 pair Attention and MLP sites within layers 0 and 1; rows 3–4 pair layers 0 and 1 within Attention and MLP. Each panel fits all 25 runs. Pure colors and boundaries denote fitted phase probability at least 50%; non-dominant regions use probability-weighted mixtures, while low-support regions remain blank. Validation-colored curves select the center, low/high training-fraction, and low/high learning-rate configurations with seeds 0–4, respectively. Stars mark endpoints with 100% training and validation accuracy; diamonds mark the remaining endpoints.

F Experimental Details: Adversarial Transferability

This appendix provides the complete experimental configuration, computational methodology, and detailed statistical analyses supporting the adversarial transferability results presented in Section 4.1.

F.1 Experimental Setup

Model architectures. We evaluate adversarial transfer across three source models and six directed source–target pairs:

- **Source models (white-box):** OpenAI CLIP ViT-B/32, OpenAI CLIP ViT-L/14, and OpenCLIP ConvNeXt-Large-DataComp (`convnext_large_d`, `laion2b_s26b_b102k_augreg`).

- **Target models (transfer evaluation):** for each source, the other two architectures are evaluated post hoc, producing six transfer directions.

Each model uses its released preprocessing pipeline and native evaluation resolution. The attack generator and every coordinate of the layerwise measurement profile use only the source model. Target models enter after the adversarial examples and source measurements have been fixed, through post-hoc transfer-retention and target-similarity response analysis.

Perturbation configuration sets. The main sweep is the Cartesian product of three generators, $\epsilon \in \{0.01, 0.025, 0.05, 0.1\}$, $\beta \in \{0, 0.5\}$, and $t_p \in \{0, 0.2, 0.4, 0.6, 0.8\}$, giving 120 manifold-constrained attack configurations per source. Here β is the momentum coefficient in the adversarial optimizer. These configurations form a flow-matching-based instantiation of recent generative semantic attacks: the perturbation is introduced on a learned trajectory and transported by the remaining flow, rather than being restricted in its final form to a fixed L_p ball. The three generator families are Pixel Mean Flow (PMF), Representation Autoencoder (RAE), and Scalable Interpolant Transformer (SiT). The coordinate t_p selects the perturbation-injection position on the generative trajectory and therefore controls how much subsequent generative transport acts on it; the analysis does not impose a monotonic relation between t_p and transfer.

Attack objective and updates. The attack minimizes correct-class image–text cosine similarity:

$$\mathcal{L}_{\text{attack}}(x; y^\star) = \frac{\langle f_S(x), f_S^{\text{text}}(y^\star)\rangle}{\|f_S(x)\|_2 \, \|f_S^{\text{text}}(y^\star)\|_2}, \tag{149}$$

where f_S and f_S^{text} are the source image and text encoders, and $y^\star$ is the correct class. Batch losses are averaged. The update direction g_k is the negative gradient of this loss with respect to the optimized generator state, differentiated through the approximate remaining flow and image decoder. Momentum satisfies

$$m_1 = g_1, \qquad m_k = \beta m_{k-1} + (1-\beta) g_k \quad (k \geq 2). \tag{150}$$

Sample generation and attribution paths. Each generator produces both the clean image and its adversarial counterpart. Thus changing G changes the clean-image distribution as well as the perturbation-generation process. Within a generator, the clean endpoint for each sample index remains fixed across t_p, ϵ, and β. Algorithm 3 constructs the sample pair and intermediate images defining the attribution path. This construction connects attribution to a path induced by a generative model (Zhang et al., 2026). Here, state interpolation followed by generative transport defines the path. The measurement depends jointly on this path, the source model, and the attribution objective.

Algorithm 3 Sample Generation and Attribution-Path Construction

Require: Generator G, source model, class $y^\star$, generation seed, injection position t_p, momentum β, budget ϵ.
Ensure: Clean–adversarial pair $(x^{(0)}, x^{(10)})$ and intermediate images $x^{(1)}, \ldots, x^{(9)}$.
  Sample initial noise at $t = 1$ using the generation seed.
  Denoise from 1 to t_p conditioned on $y^\star$; denote the state by z_{clean}.
  Set $z_{\text{adv}} \leftarrow z_{\text{clean}}$.
  for $k = 1, \ldots, 10$ **do**
    Compute g_k through the approximate remaining flow and decoder using Eq. 149.
    Update m_k using β by Eq. 150.
    $z_{\text{adv}} \leftarrow \text{clip}_{\epsilon, z_{\text{clean}}}\left(z_{\text{adv}} + 0.15\epsilon \frac{m_k}{\sqrt{\sum_i m_{k,i}^2 / \dim(m_k)}}\right)$.
  end for
  for $j = 0, \ldots, 10$ **do**
    Set $z \leftarrow (1 - j/10) z_{\text{clean}} + (j/10) z_{\text{adv}}$.
    Denoise z from t_p to 0 with the same conditioning and decode it as $x^{(j)}$.
  end for
  Store the sample pair and ordered intermediate images for piecewise-linear attribution.

The perturbation budget is imposed on the generator state using the root-mean-square norm, $\|\delta\|_{\mathrm{RMS}} = \|\delta\|_2/\sqrt{d}$, where d is the number of state coordinates. The operator $\mathrm{clip}_{\epsilon,z_{\mathrm{clean}}}$ projects onto $\|z - z_{\mathrm{clean}}\|_{\mathrm{RMS}} \le \epsilon$, equivalently an L_2 ball of radius $\epsilon\sqrt{d}$ at the injection state.

Analysis unit. Each configuration contains 100 generated clean–adversarial pairs, with ten pairs per ImageNette-10 class. Class labels and sample indices are aligned across configurations and generators, while image content is generator specific. The sweep therefore contains 12,000 image pairs per source. Raw class names are passed to each model's tokenizer and text encoder without an additional prompt template.

Attribution settings. ❶ **Attribution target.** The attribution target is the source-model image-embedding displacement from the clean image to the current path sample:

$$\mathcal{L}_{\mathrm{attr}}(x_{\mathrm{cur}}; x_{\mathrm{clean}}) = \|f_S(x_{\mathrm{clean}}) - f_S(x_{\mathrm{cur}})\|_2 . \tag{151}$$

The complete measurement profiles are computed from the source model alone; target measurements enter only in the subsequent association evaluation. ❷ **Path integration.** For the stored path $x^{(0)}, \ldots, x^{(10)}$, we further interpolate between each adjacent pair of images and refine that segment independently with the endpoint-inclusive Integrated-Gradients rule. The baseline setting specifies $n = 10$ integration intervals within each outer segment. The Gauss–Legendre implementation retains both segment endpoints, inserts the interior quadrature nodes, and deduplicates shared outer endpoints. With n intervals per segment, the complete path has $10n+1$ distinct evaluation positions; the baseline therefore has 101 positions. The integration precision analysis varies n while keeping the stored trajectory fixed (Appendix G.2). ❸ **Measurement sites.** For ConvNeXt-Large-DataComp sources, we probe all 36 depthwise convolution layers. For ViT-B/32 and ViT-L/14, we probe all 12 and 24 multi-head attention layers, respectively. At an attention layer, the canonical per-sample pre-aggregation and post-aggregation attribution fields have shapes $[H, Q, K, D]$ and $[H, Q, D]$: head and query are local sites, key is the route coordinate, and the head-dimension activation coordinate is unreduced. At a ConvNeXt depthwise 7×7 convolution, the fields have shapes $[H, W, 49, C]$ and $[H, W, C]$: spatial output is local, the 49 kernel offsets are routes, and the channel activation coordinate is unreduced. The tensor-level increments follow Appendix D. For either architecture, local values are reduced to one layer value per image using the weighted average in Eq. 144. The same local weights apply to $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ and both logarithmic dimensions.

Perturbation-interaction external reference. We use a sampled grid bipartition-interaction reference based on the released construction of Wang et al. (2021c), adapted to the source-model utility used here. The players are the cells of a 16×16 image grid $\mathcal{P}$. For each clean/adversarial pair, we draw $B = 32$ coalitions by sampling 32 cell identifiers with replacement; repeated identifiers collapse to one binary mask. We evaluate the source-side semantic utility of the full perturbation, the perturbation with G_b removed, the perturbation restricted to G_b, and the clean image. This gives

$$\widehat{I}_{\mathrm{grid}}(\delta) = \frac{1}{B}\sum_{b=1}^{B}\left[u_S(\delta) - u_S(\delta_{\mathcal{P}\setminus G_b}) - u_S(\delta_{G_b}) + u_S(0)\right], \tag{152}$$

where $u_S(\eta) = S_{\cos}(f_S(x), w_c^S) - S_{\cos}(f_S(x+\eta), w_c^S)$. We average the raw signed $\widehat{I}_{\mathrm{grid}}$ over the same images as the other measures in each selection or evaluation split. This CLIP-space estimator enters both response statistics below. We evaluate the raw signed statistic: $\widehat{I}_{\mathrm{grid}}$ measures input-level combinatorial interaction, whereas $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ measures effective-dimensional reorganization from a pre-aggregation attribution field to post-aggregation attribution evidence at an aggregation site.

Intermediate feature displacement reference. Motivated by the feature-disruption perspective of ILA (Huang et al., 2019), we compare with intermediate feature displacement (IFD). For the flattened source feature $h_l(x)$ with d_l coordinates, we compute

$$\mathrm{IFD}_l = \frac{\|h_l(x_{\mathrm{adv}}) - h_l(x)\|_2}{\sqrt{d_l}}. \tag{153}$$

The features are the attention aggregation outputs before the output projection in ViT and the depthwise-convolution outputs in ConvNeXt. Each score is computed per image before configuration averaging. IFD selects among the same 12, 24, or 36 source sites with the same image

partitions and configuration resampling as $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$. This endpoint displacement measure provides a comparison with feature change magnitude under the source-similarity control.

F.2 Measurement Profiles and Sitewise Response Analyses

Measurement profile and analysis unit. For a statistic stat, the source-computed object is the complete layerwise measurement profile

$$\mathrm{MeasurementProfile}(\mathrm{stat}; \text{configuration}) = \left(\mathrm{stat}^{(\text{site})}(\text{configuration})\right)_{\text{site}\in\mathcal{S}}. \tag{154}$$

$\mathbb{V}_{\mathbb{A}}(\mathbf{S})$, $\log_2 \mathrm{D}(\mathbf{S})$, and $\log_2 \mathrm{D}(\mathbb{A}[\mathbf{S}])$ use identical source observations and candidate layers specified in the attribution settings above. The external $\widehat{I}_{\mathrm{grid}}$ baseline has one source-computed value per configuration and one candidate site. In addition, for parameter comparison, we replace the internal measure with the adversarial generation parameter t_p or β, which are designed to control the strength of manifold regularization or momentum regularization and are associated with transferability.

Generation-property responses. For generator identity G, the response is the one-way ANOVA effect size η^2: the between-generator sum of squares divided by the total sum of squares of the original measure. For numeric injection position t_p, we report absolute Spearman correlation $|\rho|$ and squared Pearson correlation R^2. These statistics are computed independently at each site.

Target responses. For each configuration, sim_S and sim_T are mean correct-class image–text similarities in the source and target models. We report absolute Spearman partial correlation and squared Pearson partial correlation between each measure and sim_T, controlling sim_S. Pearson residualizes the original values on an intercept and source similarity; Spearman first ranks all three variables separately within the selection and evaluation pools, then applies the same residualization within each pool. Target responses inform site selection only after the attack and source measurements are fixed. These responses quantify associations with target similarity after controlling for source similarity.

Descriptive profile and scatter protocol. Descriptive full-data profiles average all 100 images. For the descriptive profiles, let $\Delta_{\mathcal{M}}$ denote the clean-to-adversarial decrease in correct-class similarity. Transfer retention is the Conditional Directional Preservation Ratio

$$\mathrm{cDPR} = \frac{\mathbb{E}[\Delta_{\mathrm{Target}} \mid \Delta_{\mathrm{Source}} > 0]}{\mathbb{E}[\Delta_{\mathrm{Source}} \mid \Delta_{\mathrm{Source}} > 0]}. \tag{155}$$

Generation profiles use rank partial R^2, controlling $\log_2 \epsilon$, t_p, and β for G, and controlling $\log_2 \epsilon$ and β for t_p within each generator. Transfer profiles show signed Spearman correlations with cDPR across all 120 configurations. The highlighted peaks are selected on this full configuration set. Shaded scatter bands are pointwise intervals for the fitted mean regression line; these peaks and bands are distinct from the held-out site selections and bootstrap response intervals in the main analysis. Per-panel correlation tests are descriptive and use no adjustment across the displayed layers.

Output-side reference and logical route measurements. The Conductance field at an aggregation output can be computed directly on its physical output coordinates. Local closure identifies this field with $\mathbb{A}[\mathbf{S}]$, so its effective dimensionality is available without recovering the full logical route field. The proposed formalization and computation additionally make $\log_2 \mathrm{D}(\mathbf{S})$ and Attribution $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ accessible. Comparing these two metrics with the output-side reference evaluates the empirical value of resolving contribution organization before aggregation.

Image cross-fit protocol. ❶ **Partitioning.** We partition the 100 aligned sample indices into two class-balanced halves of 50, with five indices per class in each half, using a random stream initialized with seed 0. Each generator contributes its own generated image pairs at these indices. All records sharing a sample index stay on the same side across configurations, generators, sources, targets, and measures. ❷ **Site selection and evaluation.** One half selects sites and the other evaluates them; exchanging the halves gives two folds. Each side recomputes configuration means over its assigned images, ranks, nuisance residuals, and response profiles. Every measure and correlation statistic selects its own site on the selection side and evaluates that frozen site on the other side.

❸ **Response averaging.** We average the two fold responses within each partition, then average 10,000 independently drawn partitions for the point estimate. The three sources receive equal weight for generation and the six directions receive equal weight for transfer. Generator identity varies within each source's configuration grid.

Configuration and partition variability. We use 10,000 configuration bootstrap replicates. Each replicate draws 40 (t_p, ϵ, β) cells with replacement, retaining all three generators within each cell and sharing the draw across sources, targets, measures, and folds. Within each replicate, we generate one new class-balanced 50/50 partition of the fixed generated pairs, repeat site selection and evaluation in both directions, and average the two fold responses. We then compute the fixed-group mean and subtract measures within the shared draw to obtain paired differences. Intervals are the 2.5th and 97.5th percentiles of the corresponding replicate values. All main-response aggregates have 10,000 valid replicates. We report these as 95% bootstrap percentile intervals for the resampling procedure. They describe the spread of configuration-resampled, single-partition responses, conditional on the observed generated pairs and model set. They do not include fresh sampling from the image population or model retraining. Their interpretation and the distinction from confidence intervals for partition-averaged estimates are given in Appendix G.

Tables F.1 and F.2 report the response means and intervals. Selection scores tied within relative and absolute tolerances of 10^{-12} are resolved by the lowest site index.

Table F.1: Source-resolved generation responses. Mean weights the three sources equally. Entries give responses and 95% intervals.

Group	Response	$\widehat{I}_{\mathrm{grid}}$	IFD	$\log_2 \mathrm{D}(\mathbf{S})$	$\log_2 \mathrm{D}(\mathbb{A}[\mathbf{S}])$	$\mathbb{V}_{\mathbb{A}}(\mathbf{S})$
ViT-B/32	$G : \eta^2$	0.582 [0.495, 0.682]	0.379 [0.309, 0.480]	0.293 [0.198, 0.450]	**0.737** [0.684, 0.802]	0.671 [0.512, 0.767]
	$t_p : \lvert\rho\rvert$	0.276 [0.159, 0.380]	0.357 [0.204, 0.571]	0.497 [0.416, 0.686]	0.479 [0.213, 0.681]	**0.780** [0.669, 0.826]
	$t_p : R^2$	0.069 [0.018, 0.146]	0.119 [0.034, 0.317]	0.260 [0.186, 0.455]	0.244 [0.062, 0.456]	**0.346** [0.264, 0.403]
ViT-L/14	$G : \eta^2$	0.209 [0.136, 0.335]	0.421 [0.354, 0.520]	0.579 [0.392, 0.675]	**0.852** [0.819, 0.889]	0.449 [0.277, 0.642]
	$t_p : \lvert\rho\rvert$	0.353 [0.122, 0.549]	0.377 [0.230, 0.593]	0.694 [0.515, 0.808]	0.580 [0.404, 0.749]	**0.698** [0.536, 0.797]
	$t_p : R^2$	0.102 [0.005, 0.280]	0.152 [0.050, 0.351]	**0.481** [0.313, 0.613]	0.362 [0.181, 0.524]	**0.481** [0.320, 0.601]
ConvNeXt-L	$G : \eta^2$	0.457 [0.378, 0.553]	0.360 [0.297, 0.490]	0.603 [0.528, 0.697]	0.336 [0.270, 0.464]	**0.814** [0.697, 0.892]
	$t_p : \lvert\rho\rvert$	0.032 [0.010, 0.211]	0.469 [0.360, 0.621]	0.583 [0.416, 0.745]	0.599 [0.401, 0.753]	**0.656** [0.458, 0.753]
	$t_p : R^2$	0.003 [0.000, 0.046]	0.217 [0.121, 0.381]	0.362 [0.222, 0.558]	0.365 [0.218, 0.562]	**0.433** [0.244, 0.540]
Mean	$G : \eta^2$	0.416 [0.344, 0.510]	0.387 [0.326, 0.490]	0.492 [0.402, 0.593]	0.642 [0.604, 0.703]	**0.645** [0.560, 0.735]
	$t_p : \lvert\rho\rvert$	0.220 [0.131, 0.368]	0.401 [0.273, 0.593]	0.592 [0.487, 0.741]	0.553 [0.349, 0.726]	**0.711** [0.593, 0.774]
	$t_p : R^2$	0.058 [0.013, 0.155]	0.163 [0.071, 0.347]	0.368 [0.260, 0.537]	0.324 [0.159, 0.513]	**0.420** [0.312, 0.503]

The Mean blocks give the equally weighted source or direction averages; their intervals are computed from these averages within each bootstrap replicate. The tables also resolve the variation behind the averages. Spatial organization is examined through the complete profiles in Figure F.1. All intervals follow the paired-comparison protocol in Appendix G. A positive interval for an absolute correlation does not by itself reject zero population correlation.

Injection position t_p and momentum β are also evaluated as scalar measures in Table F.2, with the same source-similarity control and image partitions. These comparisons test whether an internal attribution measure tracks transfer variation more strongly than either individual generation parameter

Table F.2: Direction-resolved target responses controlling source similarity. All responses are partial correlations; t_p and β are scalar measurement baselines. Mean weights the six directions equally. Entries give point estimates and 95% resampling intervals.

Group	Response	t_p	β	$\widehat{I}_{\mathrm{grid}}$	IFD	$\log_2 \mathrm{D}(\mathbf{S})$	$\log_2 \mathrm{D}(\mathbb{A}[\mathbf{S}])$	$\mathbb{V}_{\mathbb{A}}(\mathbf{S})$
ViT-B/32→ ViT-L/14	$\lvert\rho\rvert$	0.743 [0.647,0.788]	0.071 [0.011,0.302]	0.190 [0.046,0.331]	0.673 [0.565,0.772]	0.720 [0.631,0.797]	0.571 [0.446,0.666]	**0.758** [0.628,0.845]
	R^2	0.509 [0.399,0.594]	0.006 [0.000,0.098]	0.050 [0.005,0.126]	0.451 [0.319,0.552]	0.512 [0.409,0.596]	0.363 [0.240,0.472]	**0.543** [0.366,0.648]
ViT-B/32→ ConvNeXt-L	$\lvert\rho\rvert$	0.761 [0.664,0.804]	0.042 [0.009,0.272]	0.079 [0.018,0.223]	0.637 [0.509,0.746]	0.759 [0.668,0.815]	0.538 [0.404,0.633]	**0.831** [0.685,0.896]
	R^2	0.527 [0.433,0.602]	0.002 [0.000,0.078]	0.007 [0.000,0.051]	0.430 [0.313,0.531]	0.507 [0.398,0.590]	0.333 [0.211,0.432]	**0.549** [0.395,0.657]
ViT-L/14→ ViT-B/32	$\lvert\rho\rvert$	0.754 [0.686,0.785]	0.046 [0.010,0.267]	0.390 [0.204,0.528]	0.621 [0.487,0.750]	0.735 [0.641,0.799]	**0.832** [0.739,0.881]	0.788 [0.634,0.845]
	R^2	0.497 [0.408,0.565]	0.003 [0.000,0.076]	0.158 [0.048,0.276]	0.520 [0.307,0.620]	0.537 [0.430,0.644]	**0.591** [0.400,0.692]	0.544 [0.364,0.671]
ViT-L/14→ ConvNeXt-L	$\lvert\rho\rvert$	0.768 [0.686,0.796]	0.038 [0.008,0.258]	0.299 [0.131,0.430]	0.702 [0.541,0.802]	0.774 [0.637,0.820]	0.804 [0.737,0.852]	**0.835** [0.721,0.871]
	R^2	0.538 [0.469,0.588]	0.003 [0.000,0.072]	0.105 [0.020,0.203]	0.560 [0.384,0.642]	0.572 [0.488,0.661]	0.590 [0.437,0.676]	**0.632** [0.507,0.718]
ConvNeXt-L→ ViT-B/32	$\lvert\rho\rvert$	0.754 [0.681,0.795]	0.057 [0.009,0.284]	0.380 [0.167,0.510]	0.655 [0.484,0.750]	0.963 [0.921,0.970]	0.950 [0.913,0.959]	**0.964** [0.931,0.973]
	R^2	0.446 [0.311,0.562]	0.005 [0.000,0.084]	0.248 [0.114,0.350]	0.496 [0.304,0.608]	**0.768** [0.661,0.822]	0.766 [0.682,0.820]	0.741 [0.638,0.805]
ConvNeXt-L→ ViT-L/14	$\lvert\rho\rvert$	0.722 [0.640,0.774]	0.076 [0.009,0.301]	0.448 [0.229,0.575]	0.680 [0.522,0.774]	**0.944** [0.893,0.958]	0.935 [0.889,0.950]	0.940 [0.895,0.958]
	R^2	0.403 [0.263,0.530]	0.006 [0.000,0.086]	0.341 [0.191,0.446]	0.525 [0.373,0.628]	0.775 [0.665,0.834]	**0.785** [0.687,0.842]	0.719 [0.618,0.795]
Mean	$\lvert\rho\rvert$	0.750 [0.681,0.781]	0.055 [0.019,0.278]	0.298 [0.181,0.402]	0.661 [0.557,0.744]	0.816 [0.762,0.847]	0.772 [0.711,0.807]	**0.853** [0.784,0.884]
	R^2	0.486 [0.397,0.555]	0.004 [0.000,0.081]	0.152 [0.079,0.222]	0.497 [0.385,0.574]	0.612 [0.547,0.669]	0.571 [0.495,0.624]	**0.622** [0.539,0.687]

under a common source-similarity control. The positive paired intervals support $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ as a stronger scalar indicator in this comparison (Table G.2).

Property dependence and aggregation granularity. Within every source and transfer-direction group, $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ exceeds $\widehat{I}_{\mathrm{grid}}$ in all reported generation and target responses (Tables F.1 and F.2). Although $\log_2 \mathrm{D}(\mathbb{A}[\mathbf{S}])$ responds more strongly to G in ViT-L/14, changing G also changes the clean samples, so these distinctions may already be expressed in the output attribution distribution. The central variable t_p instead probes generative transport within a fixed generator and clean sample set. For this perturbation-specific variation, $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ exceeds the output-side dimension in both ViT-L/14 responses and gives the largest or tied-largest displayed response in all six source-specific comparisons. The transfer results further suggest a role for aggregation granularity. Despite the component dimensions' larger partial R^2 in ConvNeXt, $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ exhibits similarly strong rank associations, consistent with a nonlinear relation that preserves the ordering of transfer strength. Even for ViT-L/14 to ViT-B/32, where the output-side dimension has the strongest response, $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ improves on $\log_2 \mathrm{D}(\mathbf{S})$ in both transfer statistics. The output-side advantage may reflect fine-grained organization specific to the larger source, with aggregated evidence more directly related to features shared with the smaller target. Across all six directions, $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ raises the minimum observed response over either component: the minima are 0.758 versus 0.720 and 0.538 for partial $|\rho|$, and 0.543 versus 0.507 and 0.333 for partial R^2, respectively. Since $\mathbf{S}$ resolves contributions before aggregation at a fine computational granularity, these observations motivate studying ViT-L/14 on more abstract attribution graphs that organize microscopic contributions into higher-level features.

F.3 Source-dependent Attribution Organization

Figure F.1 shows the generation and transfer profiles for all three sources, including the ViT-B profiles illustrated in Figure 6.

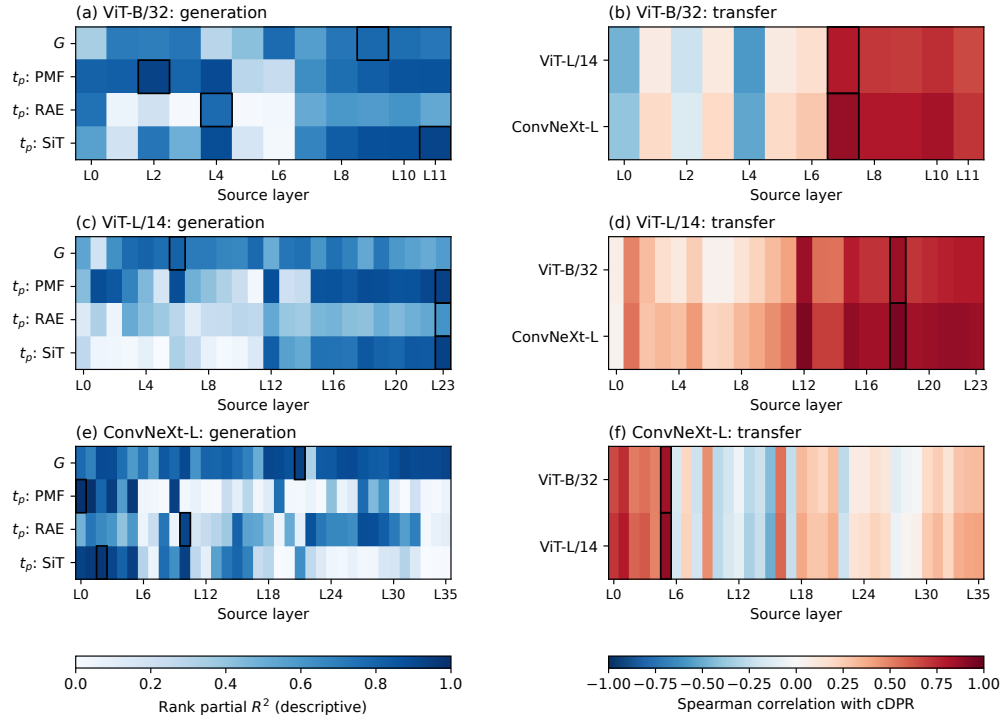

Figure F.1: Generation and transfer response profiles for ViT-B, ViT-L, and ConvNeXt. Outlined cells mark maximal response magnitudes. The two targets share a transfer-retention peak within each source, while generation responses show source-dependent organization.

Across all three source models, PMF produces pronounced t_p responses in early network modules; at the first measured site, its response exceeds those of both RAE and SiT. This pattern is consistent with the different spaces in which the generators evolve their denoising states: PMF operates directly in image space, whereas RAE and SiT operate in latent spaces. Changing t_p in PMF may therefore alter perturbation features that engage early source-model computations more directly. These profiles suggest that the layers at which $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ responds depend on the match between the generated perturbation features and the model's aggregation operations.

Transfer responses reveal a complementary pattern. Within each source, the two targets share a cDPR response peak: L7 in ViT-B, L18 in ViT-L, and L5 in ConvNeXt. This shared location across targets of different sizes or architectures suggests that adversarial features transferable to multiple models may share a source-model level of attribution convergence. Together, the generation and transfer profiles relate feature-specific attribution organization to the source-model locations associated with transferability.

Configuration-level scatter views. Figure F.2 shows the corresponding configuration-level associations at source layers L7, L18, and L5. Figures F.3 and F.4 display all measured layers for the other two source architectures. These views show how configuration-level variation in $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ relates to transfer retention and generator identity.

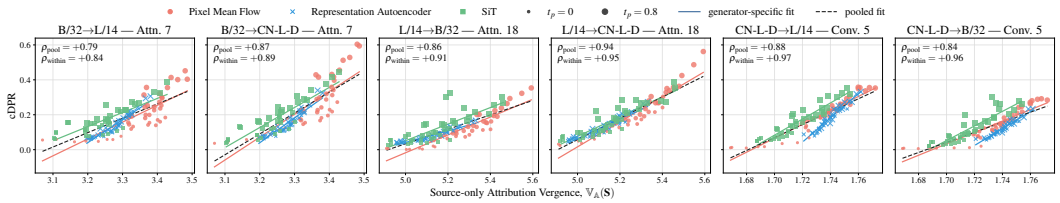

Figure F.2: Configuration-level $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ and transfer retention across six directions. The scatter views complement the shared within-source response peaks and reveal variation within generator families.

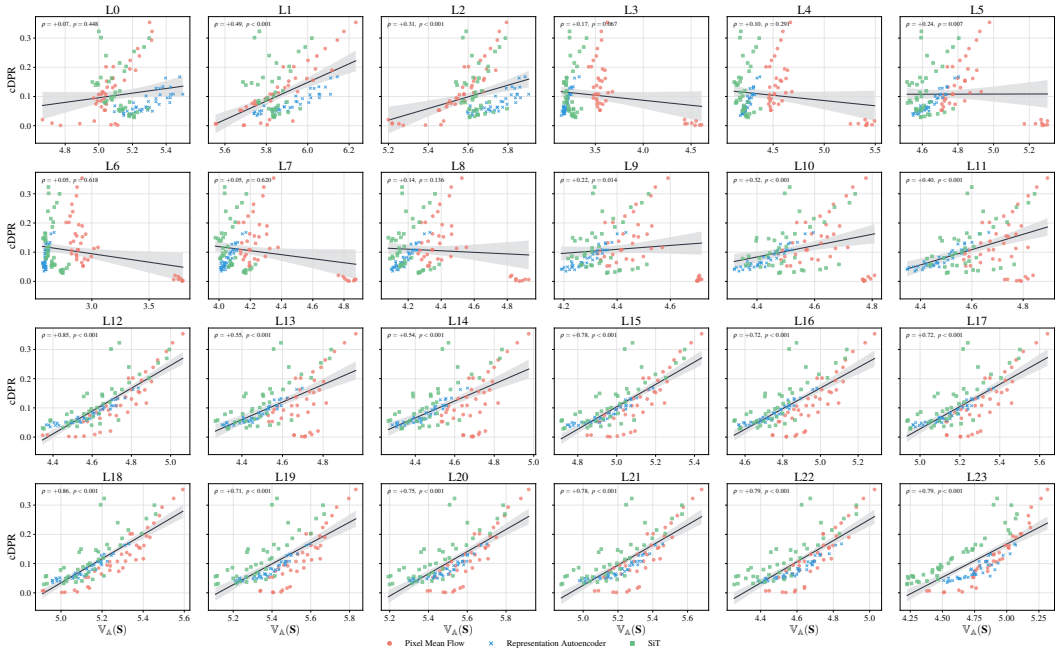

Figure F.3: Layerwise $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ and cDPR for ViT-L as source and ViT-B as target. Transfer-retention associations are concentrated in deeper attention layers. Correlations and regression lines describe the displayed configurations.

G Paired Comparisons of Property Responses

Using the protocols in Appendices E and F, Figure G.1 compares order-2 $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ with the reference measures and alternative Rényi orders. For each property, we subtract the reference response from the $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ response within each shared bootstrap replicate, then form the interval from the resulting mean differences. Each comparison follows the observation sets, group weights, and repeated site selection specified in the corresponding experimental protocol, with image partitions defined in Appendix F.2. The intervals are individual 95% bootstrap percentile intervals without adjustment for multiple comparisons.

Interpretation of the resampling intervals. The analysis challenges each comparison across configuration resampling, random partition assignment, and the resulting site-selection variation. They are not standard bootstrap confidence intervals for the repeated-partition average. However, retaining these sources of variation provides a more demanding robustness assessment than conditioning on a fixed partition. For a fixed dataset, a single random partition and repeated partition averaging have the same expectation over partitions, but different variability; configuration resampling can also shift the mean relative to the original point estimate. The reported intervals summarize the central 95% of the resampled single-partition responses or paired differences. A paired interval entirely above zero supports a robust advantage under this configuration-and-partition resampling protocol.

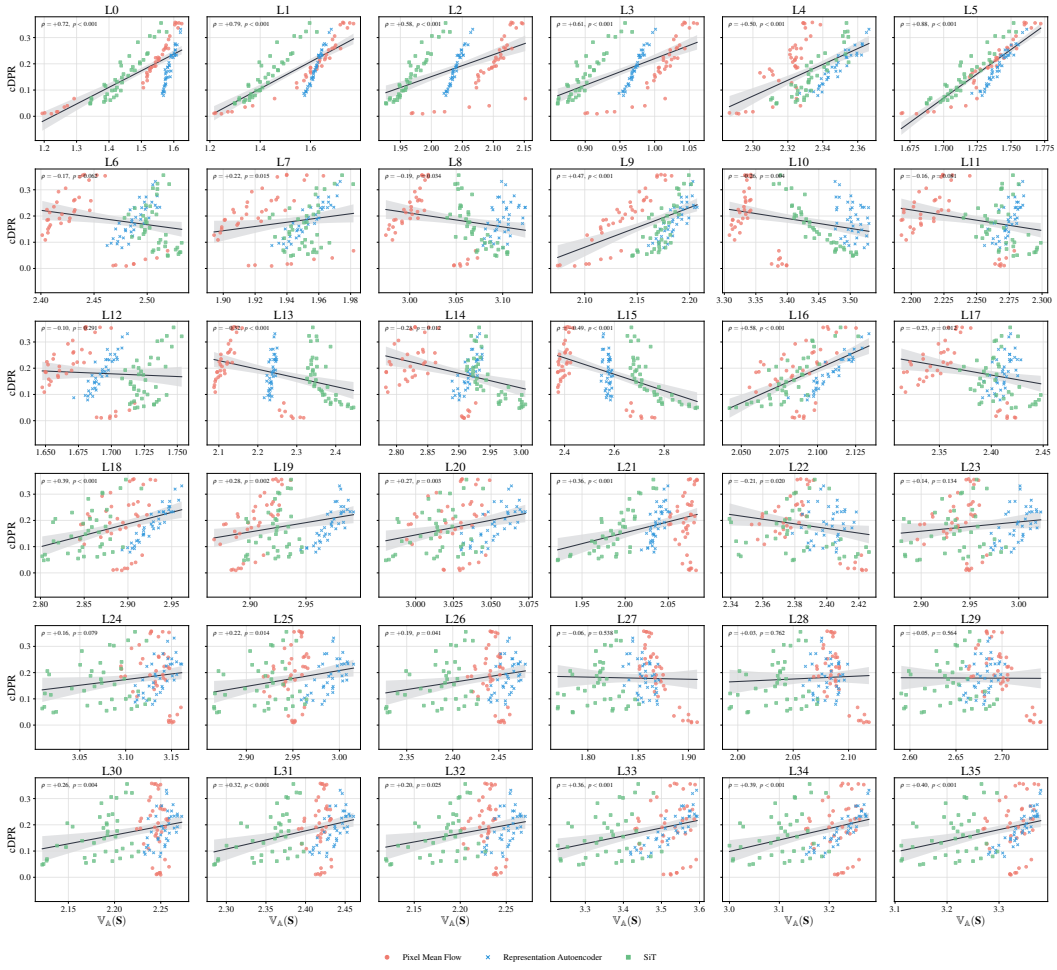

Figure F.4: Layerwise $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ and cDPR for ConvNeXt as source and ViT-L as target. The early depthwise layers show the clearest positive transfer-retention associations.

Alternative Rényi orders are defined in Appendix G.1.

Gains over prior analysis tools. For grokking, Figure G.1 shows entirely positive paired intervals against mean-local TwoNN LID for phase, training rank, and both validation responses; the training R^2 interval includes zero. Although training R^2 favors both importance measures, $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ has higher mean responses for the other four properties, with entirely positive intervals against Kernel Importance. Against Reg. Importance, the phase and validation R^2 intervals are entirely positive, while the two rank-response intervals include zero. These comparisons support the sensitivity of $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ to generalization progress.

For adversarial transfer, all five mean-response differences against each of $\widehat{I}_{\text{grid}}$ and IFD have entirely positive paired intervals (Table G.1). Together, LID and $\widehat{I}_{\text{grid}}$ account for nine positive intervals across ten properties. Although some direction-specific comparisons with IFD include zero, the mean advantages support attribution organization as a stronger indicator of the studied properties than feature-displacement magnitude alone.

Gains from the formulation and relative measurement. The component comparisons distinguish the benefit of recovering microscopic contributions from that of measuring their convergence under aggregation. For $\log_2 \mathrm{D}(\mathbf{S}) - \log_2 \mathrm{D}(\mathbb{A}[\mathbf{S}])$, four paired response intervals are positive: grokking phase and validation R^2, adversarial injection-position R^2, and target rank response. Although the generator-identity comparison favors the output-side summary, these gains support the

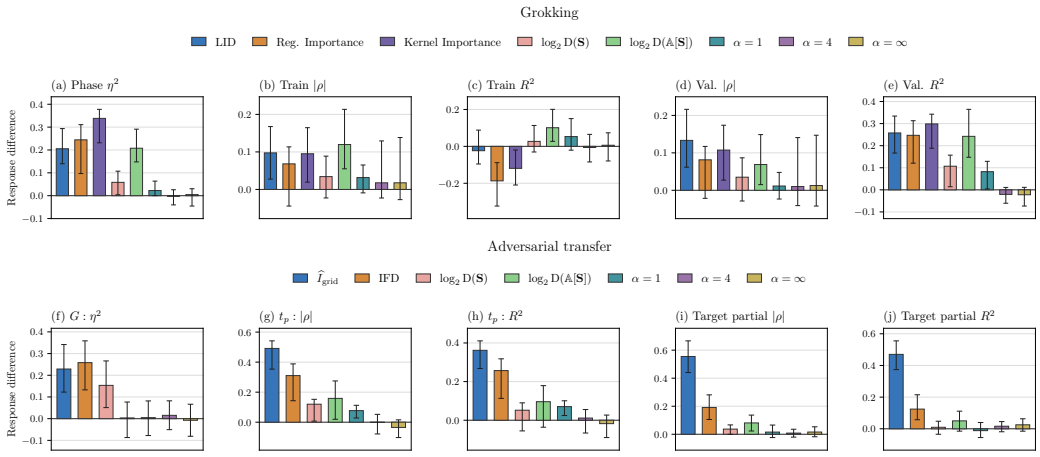

Figure G.1: Paired differences in mean property responses, computed as order-2 $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ minus each reference. Error bars show 95% bootstrap percentile intervals; positive differences favor $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$. The top row shows grokking and the bottom row adversarial transfer. Each row has its own reference legend. Grokking includes ExPLAIND regularization and kernel importance. Adversarial comparisons also include intermediate feature displacement (IFD). $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ improves several responses over the reference measures and Shannon entropy; paired intervals against orders 4 and ∞ include zero for every property.

value of the pre-aggregation field recovered by the proposed formulation and computation. $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ further improves on $\log_2 \mathrm{D}(\mathbf{S})$: all ten mean differences are positive, with intervals entirely above zero for grokking phase and validation R^2, generator identity, and injection-position rank response. Relative to the output-side summary, $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ has seven positive intervals and no negative intervals. Thus, the paired comparisons support both access to microscopic attribution organization and the additional benefit of measuring it relative to the aggregation output.

Table G.1 classifies intervals using unrounded endpoints: positive and negative intervals lie entirely above and below zero, respectively. Intervals including zero are further separated by the sign of the unrounded difference between the reported mean-response point estimates. These two columns distinguish the direction of the average difference without implying an entirely positive or negative interval. Component and Rényi-order comparisons each cover ten responses. LID, Reg. Importance, and Kernel Importance each cover five grokking responses; the interaction and IFD references each cover five adversarial responses. Each parameter comparison covers the two target-response statistics.

Table G.2 compares $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ with injection position and momentum as scalar measures under the same source control. The four paired intervals are positive, indicating stronger mean transfer responses than either individual parameter.

G.1 Rényi Orders and Vergence Components

Table G.3 compares orders $\alpha \in \{1, 2, 4, \infty\}$ on the same observations and candidate sites. Each local entropy is computed from normalized absolute attribution mass, and the local pre-aggregation minus post-aggregation entropy is reduced with the same weights from Eq. 144 for every order. Both experiments use their full main-experiment observation sets. Each order selects its own site for each property and statistic under the main held-out protocol.

The measurements constructed with $\alpha = 4$ and $\alpha = \infty$ yield property responses close to those of $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ at $\alpha = 2$. Although the higher orders have larger point responses for some properties, their paired intervals against order 2 include zero for all ten properties (Table G.1). With comparable empirical responses, we adopt $\alpha = 2$ for its established foundation as an effective count and its connection to sparsity, together with a direct collision-probability interpretation and moment-based computation (Rényi, 1961; Laakso & Taagepera, 1979; Hoyer, 2004). Relative to Shannon entropy,

Table G.1: Counts of paired 95% resampling intervals by sign. $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ has positive intervals for all five adversarial responses against each of the interaction and feature-displacement references. Within Includes zero, Mean denotes the unrounded difference between the reported mean-response point estimates, rather than the bootstrap mean. The nonzero Negative entries, from top to bottom, correspond to adversarial generator identity G (η^2) (row 3); grokking training accuracy (R^2) (row 5); grokking training accuracy (R^2) (row 6); adversarial injection position t_p (R^2) (row 14); adversarial generator identity G (η^2) (row 15); adversarial generator identity G (η^2) (row 16). Rows count data rows only.

Paired difference	Positive	Includes zero		Negative	Total
		Mean > 0	Mean < 0		
$\mathbb{V}_{\mathbb{A}}(\mathbf{S}) - \log_2 \mathrm{D}(\mathbf{S})$	4	6	0	0	10
$\mathbb{V}_{\mathbb{A}}(\mathbf{S}) - \log_2 \mathrm{D}(\mathbb{A}[\mathbf{S}])$	7	3	0	0	10
$\log_2 \mathrm{D}(\mathbf{S}) - \log_2 \mathrm{D}(\mathbb{A}[\mathbf{S}])$	4	5	0	1	10
$\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ minus LID	4	0	1	0	5
$\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ minus Reg. Importance	2	2	0	1	5
$\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ minus Kernel Importance	4	0	0	1	5
$\mathbb{V}_{\mathbb{A}}(\mathbf{S}) - \widehat{I}_{\mathrm{grid}}$	5	0	0	0	5
$\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ minus IFD	5	0	0	0	5
$\alpha = 2$ minus $\alpha = 1$ (Shannon)	4	5	1	0	10
$\alpha = 2$ minus $\alpha = 4$	0	6	4	0	10
$\alpha = 2$ minus $\alpha = \infty$	0	6	4	0	10
$\mathbb{V}_{\mathbb{A}}(\mathbf{S}) - t_p$	2	0	0	0	2
$\mathbb{V}_{\mathbb{A}}(\mathbf{S}) - \beta$	2	0	0	0	2
$\mathbb{V}_{\mathbb{A}}(\mathbf{S}) - \mathbb{V}_{\mathbb{A}}^{L_1}$	3	4	2	1	10
$\mathbb{V}_{\mathbb{A}}(\mathbf{S}) - \mathbb{V}_{\mathbb{A}}^{L_2}$	5	2	2	1	10
$\mathbb{V}_{\mathbb{A}}^{L_1} - \mathbb{V}_{\mathbb{A}}^{L_2}$	5	2	2	1	10

Table G.2: Paired mean response differences between $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ and individual generation parameters. All four intervals are positive.

Reference	Partial $\lvert\rho\rvert$ difference	Partial R^2 difference
t_p	+0.102 [+0.052, +0.170]	+0.135 [+0.054, +0.225]
β	+0.798 [+0.565, +0.848]	+0.617 [+0.507, +0.677]

Table G.3: Grokking and adversarial responses across Rényi orders and norm-based components. Order $\alpha = 2$ gives $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$; the last two rows give $\mathbb{V}_{\mathbb{A}}^{L_1}$ and $\mathbb{V}_{\mathbb{A}}^{L_2}$ from Eq. 156. Entries give mean responses and 95% resampling intervals. Bold marks the largest displayed mean within each column.

Measure	Grokking					Adversarial transfer				
	Phase type	Train acc.		Val. acc.		Generation config.			Target response	
	η^2	$\lvert\rho\rvert$	R^2	$\lvert\rho\rvert$	R^2	η_G^2	$\lvert\rho\rvert_{t_p}$	$R_{t_p}^2$	Partial $\lvert\rho\rvert$	Partial R^2
$\alpha = 1$	0.797 [0.738,0.844]	0.781 [0.743,0.817]	0.351 [0.273,0.448]	0.815 [0.772,0.856]	0.687 [0.634,0.785]	0.640 [0.570,0.719]	0.634 [0.521,0.713]	0.349 [0.248,0.441]	0.837 [0.751,0.879]	**0.633** [0.538,0.705]
$\alpha = 2$ ($\mathbb{V}_{\mathbb{A}}$)	0.819 [0.777,0.864]	**0.812** [0.766,0.845]	0.404 [0.348,0.492]	**0.826** [0.795,0.868]	0.769 [0.718,0.835]	0.645 [0.560,0.735]	0.711 [0.593,0.774]	0.420 [0.312,0.503]	**0.853** [0.784,0.884]	0.622 [0.539,0.687]
$\alpha = 4$	**0.822** [0.780,0.868]	0.795 [0.679,0.843]	0.410 [0.371,0.524]	0.816 [0.686,0.874]	0.790 [0.742,0.855]	0.630 [0.556,0.732]	0.713 [0.622,0.773]	0.408 [0.324,0.509]	0.844 [0.773,0.878]	0.605 [0.528,0.667]
$\alpha = \infty$	0.815 [0.783,0.868]	0.795 [0.672,0.844]	0.398 [0.363,0.511]	0.813 [0.678,0.875]	**0.791** [0.749,0.860]	0.652 [0.586,0.749]	**0.746** [0.651,0.801]	0.437 [0.352,0.535]	0.837 [0.758,0.871]	0.596 [0.517,0.658]
$\mathbb{V}_{\mathbb{A}}^{L_1}$	0.715 [0.674,0.789]	0.791 [0.752,0.825]	0.398 [0.335,0.491]	0.788 [0.733,0.845]	0.710 [0.641,0.797]	0.717 [0.658,0.785]	0.730 [0.610,0.834]	**0.555** [0.425,0.675]	0.798 [0.724,0.839]	0.495 [0.390,0.579]
$\mathbb{V}_{\mathbb{A}}^{L_2}$	0.519 [0.447,0.577]	0.564 [0.473,0.646]	**0.479** [0.448,0.534]	0.551 [0.477,0.650]	0.335 [0.254,0.430]	**0.779** [0.745,0.830]	0.661 [0.577,0.722]	0.432 [0.359,0.499]	0.760 [0.687,0.805]	0.573 [0.495,0.676]

order 2 has positive paired difference intervals for grokking phase, validation R^2, and both adversarial injection-position responses. Shannon-based responses may be more sensitive to numerous

non-negligible background contributions in the microscopic field. If these contributions carry limited information about the property, they can weaken the response of the resulting summary. Order 2 gives greater weight to dominant contributions, providing a possible explanation for its stronger responses.

Norm-based components. The norm identities in the proof of Theorem C.14 also motivate a decomposition of Vergence. For a local field with $\mathbb{A}[\mathbf{S}] \neq 0$,

$$\mathbb{V}_{\mathbb{A}}^{L_p}(\mathbf{S}) = -2\log_2 \frac{\|\mathbb{A}[\mathbf{S}]\|_p}{\|\mathbf{S}\|_p}, \qquad p \in \{1, 2\},$$
$$\mathbb{V}_{\mathbb{A}}(\mathbf{S}) = \mathbb{V}_{\mathbb{A}}^{L_1}(\mathbf{S}) - \mathbb{V}_{\mathbb{A}}^{L_2}(\mathbf{S}). \tag{156}$$

The L_1 component measures the relative absolute-mass loss caused by sign cancellation. For non-negative fields it vanishes, and $-\mathbb{V}_{\mathbb{A}}^{L_2}(\mathbf{S}) = H_2(Y, Z) - H_2(Y)$ gives the conditional Rényi entropy difference. For signed fields, the L_2 component also depends on cancellation through the aggregated squared norm. We apply the same local weighting and sample averaging to both components and select their sites independently.

Both components respond to the studied properties (Table G.3). Although $\mathbb{V}_{\mathbb{A}}^{L_2}$ gives a stronger generator-identity response and $\mathbb{V}_{\mathbb{A}}^{L_1}$ a stronger injection-position R^2, the complete $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ has larger mean responses than the respective components for 8/10 and 7/10 properties. Its paired differences have 5 positive and 1 negative interval against $\mathbb{V}_{\mathbb{A}}^{L_2}$, and 3 positive and 1 negative interval against $\mathbb{V}_{\mathbb{A}}^{L_1}$ (Table G.1). In particular, both target-transfer intervals against $\mathbb{V}_{\mathbb{A}}^{L_1}$ are positive. These results support the combined measurement for tracking behavioral properties and motivate further study of how cancellation and attribution geometry jointly shape these responses.

G.2 Integration Precision and Computational Cost

We vary integration precision while fixing the model state, examples, attribution target, path endpoints, and stored outer path segments. The adversarial precision analysis covers all 120 configurations for ViT-B at 20 fixed sample indices, evaluating the corresponding generated pairs for each generator. These indices are split into two class-balanced halves of ten for selection and evaluation in both directions. Each pair has the stored ten-segment attribution path described in Appendix F.1. We vary the further interpolation within every segment using $n \in \{2, 5, 10, 20, 40\}$ integration intervals; $n = 10$ is the baseline. These settings give $10n + 1$ evaluation positions along the complete path. This evaluation therefore measures numerical refinement of a fixed stored trajectory. The grokking precision analysis covers 497 observed stage checkpoints from 100 runs, with 1, 2, 4, 8, 16, 32, 64, and 128 integration intervals on its single straight segment; 32 is the baseline. The step columns in the following tables count intervals per outer segment in both settings. Unreached stages are not replaced by successful runs. The finest tested precision provides an approximate numerical reference.

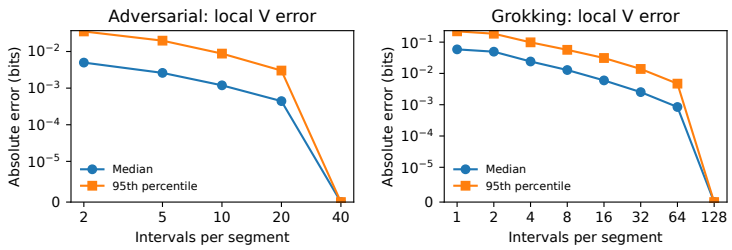

Figure G.2: Local $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ errors across integration precisions on fixed adversarial and grokking evaluation subsets. Increasing precision reduces typical errors relative to the finest tested setting.

At baseline precision, median absolute $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ errors are 0.00119 for adversarial measurements and 0.00253 for grokking; the corresponding 95th percentiles are 0.00877 and 0.01389. Response evaluation preserves the configuration or trajectory structure of each evaluation subset and repeats site selection at each precision (Tables G.5 and G.6). These subset responses assess numerical sensitivity

Table G.4: Integration accuracy and measured cost on the fixed audit subsets. Errors use the finest tested precision as a reference. Time is the sum of recorded job times; memory is peak GPU allocation.

Group	Intervals/segment	Median error	95th percentile	Max error	Job hours	GiB
Adversarial	2	0.00498	0.03531	0.59270	1.18	1.71
	5	0.00261	0.01978	0.43276	1.01	3.17
	10	0.00119	0.00877	0.24598	1.13	3.83
	20	0.00044	0.00302	0.09359	1.46	4.52
	40	0.00000	0.00000	0.00000	1.90	5.59
Grokking	1	0.05869	0.22138	0.74035	0.84	2.20
	2	0.04949	0.18588	0.73010	0.92	1.94
	4	0.02406	0.09845	0.67212	1.16	1.54
	8	0.01284	0.05677	0.32026	1.25	1.67
	16	0.00598	0.03109	0.16538	1.71	1.94
	32	0.00253	0.01389	0.06977	2.63	2.07
	64	0.00085	0.00471	0.02908	4.50	2.20
	128	0.00000	0.00000	0.00000	8.17	2.20

Table G.5: Adversarial responses across integration precisions with balanced 10-image/10-image cross-fitting (a 50/50 split of the 20 fixed audit images). Bold identifies the baseline; site changes are relative to its selections.

Intervals/segment	$G : \eta^2$	$t_p : \lvert\rho\rvert$	$t_p : R^2$	Target: $\lvert\rho\rvert$	Target: R^2	Site changes
2	0.387	0.706	0.343	0.790	0.375	0.0%
5	0.392	0.701	0.341	0.789	0.372	0.0%
10	0.396	0.701	0.341	0.783	0.371	0.0%
20	0.399	0.702	0.341	0.767	0.370	5.0%
40	0.399	0.703	0.341	0.765	0.370	5.0%

Table G.6: Grokking property responses across integration precisions on the fixed audit subset. Responses vary non-monotonically at low step counts and stabilize at finer settings. Sites are reselected at each precision; the final column averages changes from the 32-interval baseline over properties, tasks, and folds. Bold marks the baseline.

Intervals/segment	Phase η^2	Train $\lvert\rho\rvert$	Train R^2	Val. $\lvert\rho\rvert$	Val. R^2	Site changes
1	0.769	0.518	0.454	0.801	0.652	46.0%
2	0.773	0.523	0.450	0.827	0.740	40.0%
4	0.734	0.531	0.522	0.802	0.704	23.0%
8	0.726	0.523	0.504	0.822	0.740	10.5%
16	0.718	0.529	0.513	0.811	0.733	7.5%
32	0.717	0.527	0.522	0.806	0.724	0.0%
64	0.716	0.527	0.527	0.806	0.721	1.5%
128	0.719	0.531	0.528	0.805	0.719	3.0%

on the fixed evaluation subsets. Our experiments primarily examine $\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ computed from sufficiently accurate approximations to the specified attribution integrals. We therefore adopt relatively fine discretizations and assess their numerical convergence. Increasing the grokking resolution from 32 to 128 intervals changes both validation responses by less than 0.005, supporting the adopted setting.

Although very coarse discretizations produce larger measurement errors, some exhibit strong behavioral responses: at two intervals, the grokking validation rank response is 0.827, compared with 0.806 at 32 intervals, while the median local error rises to 0.04949 . This suggests that fewer integration steps may suffice for some measurement applications, motivating further study of the accuracy–cost trade-off and the computational structures underlying these responses. Table G.4 reports tail errors and actual multi-layer measurement costs, including the sum of recorded job times and peak memory.

H LIMITATIONS AND FUTURE WORK

$\mathbb{V}_{\mathbb{A}}(\mathbf{S})$ is defined relative to the model, attribution objective, reference, path, and computational coordinates. Comparisons across models therefore require explicit alignment of measurement sites and coordinate mappings. The observed behavioral associations motivate further investigation, with causal interpretations requiring targeted interventions.

Automating the construction of global measurement profiles is an important next step. The current implementation specifies aggregation nodes and coordinate mappings for selected module types. Extending this procedure across a model's computational graph would enable systematic coverage of its internal computations. This requires identifying compatible aggregation nodes, organizing their measurements across modules, and controlling the computational cost of profile construction.

Our two experimental settings address controlled aspects of two foundational problems: learning algorithmic reasoning and maintaining vision–language alignment under perturbations. Further validation should examine whether the observed associations extend to realistic tasks involving deception, hallucination, and robustness. Such studies would assess whether the profiles provide useful indicators of these behaviors across models, datasets, and deployment conditions.