ARA: ASYNCHRONOUS RESTORATION ATTRIBUTION FOR ATTENTION HEADS IN DIFFUSION LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Diffusion large language models (DLLMs) reconstruct masked tokens through successive denoising stages, yet the importance of their attention heads throughout this process remains unclear. We study how head importance changes with the available context, using masking rate as a controlled proxy for denoising stage. Head rankings retain shared structure but generally become less similar as masking rates move farther apart, revealing structured variation that a single global ranking can obscure. To quantify head contributions under these different reconstruction conditions, we introduce Asynchronous Restoration Attribution (ARA). The method averages contributions over paths that gradually restore suppressed head outputs according to independently sampled schedules. Unlike synchronized restoration, it evaluates each head while others are restored to different degrees. This allows it to capture interaction-dependent contributions that a synchronized path can miss. Experiments on two DLLM families across eight tasks support the effectiveness of ARA in identifying functionally important heads through dynamic pruning and guiding adaptive sparse attention.

1 INTRODUCTION

Diffusion large language models (DLLMs) have emerged as an alternative to autoregressive language models (Nie et al., 2025; Ye et al., 2025; Gong et al., 2025). Rather than generating text strictly from left to right, they repeatedly reconstruct masked tokens using context from both directions (Sahoo et al., 2024; Shi et al., 2024). Attention heads route this contextual information between token positions and thus play a central role in reconstruction. Understanding which heads contribute to successful reconstruction is therefore a fundamental interpretability problem for DLLMs. It connects the model's ability to recover missing tokens with the internal components that support this behavior.

Recent work has developed several approaches to studying head importance in autoregressive language models (Wang et al., 2023; Todd et al., 2024; Gould et al., 2024). CoKV evaluates head contributions through coalitions, while causal head gating (CHG) learns gates over head outputs (Sun et al., 2026; Nam et al., 2025). More general attribution methods include gradient-based attention attribution and relevance propagation with ATTNLRP (Hao et al., 2021; Achtibat et al., 2024). These approaches provide useful tools for analyzing component contributions. However, DLLMs introduce a different reconstruction setting. Rather than extending a sequence one token at a time, they repeatedly recover masked tokens within the same sequence. As denoising proceeds, newly recovered tokens provide additional context for predicting the remaining masked positions. Different heads may use this new context in different ways, potentially changing their relative contributions to reconstruction. This raises a natural question: *do the same heads remain important throughout denoising, or does their relative importance change across denoising stages?*

To answer this question, we need to compare head contributions across reconstruction conditions. Averaging these contributions into a single score per head can obscure the variation of interest. Along actual denoising trajectories, the amount of visible context changes together with the model-generated content. For a controlled comparison, we mask the same reference completions at different rates and retain separate attribution scores at each rate. These conditions provide proxies for denoising stages while keeping the underlying examples fixed. At each rate, assigning credit also requires

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.

accounting for interactions among heads. Integrated gradients provides a natural starting point for this allocation. When adapted to continuous head interventions, it accumulates each head's contribution as suppressed outputs are restored. Its completeness property ensures that these contributions sum to the overall change in reconstruction quality (Sundararajan et al., 2017; Lundstrom et al., 2022). However, its standard straight-line path restores all heads at the same pace and evaluates them only at equal restoration levels. This can miss contributions that appear at unequal restoration levels, even when the path integral is computed exactly.

To examine contributions beyond these synchronized configurations, we propose *Asynchronous Restoration Attribution* (ARA). ARA keeps the model fixed and gradually restores head outputs from zero on independently sampled schedules. Each head is evaluated while others are suppressed or restored to different degrees. Averaging contributions over these paths broadens the configurations considered while preserving completeness. The scores sum to the change in reconstruction quality from restoring all heads. Applying ARA at each masking rate yields a head-importance profile that retains variation across reconstruction conditions.

We investigate head importance through complementary analyses and interventions. Controlled comparisons across masking rates reveal changes in relative head importance that a single global ranking can obscure. We then evaluate ARA through head pruning and adaptive sparse attention on two DLLM families across eight knowledge, reasoning, and code-generation tasks. In the main baseline comparison, removing ARA's top-ranked components causes the largest average performance drop for both models. It also supports effective sparse inference across a range of allocation budgets. An ablation study further demonstrates the benefit of asynchronous restoration over the synchronized diagonal path.

Our main contributions are summarized as follows:

- As far as we know, we are the first to study how DLLM head importance varies across masking rates and distinguish rate-specific contributions from their global summary.
- We propose ARA to assign contributions through asynchronous head restoration. We establish full-dimensional support of its path-induced distributions over head configurations and show that it preserves completeness.
- We evaluate ARA on LLaDA-1.5 and Dream-v0-Instruct-7B through head pruning and adaptive sparse attention across knowledge, reasoning and code tasks, demonstrating its effectiveness in both settings.

2 Related Work

2.1 Diffusion Language Models and Their Interpretation

Discrete diffusion models develop score-based and masked-token objectives for language modeling (Lou et al., 2024; Sahoo et al., 2024; Shi et al., 2024). Block diffusion combines autoregressive and diffusion generation (Arriola et al., 2025). Recent work on DLLMs has advanced large-scale training (Nie et al., 2025), adaptation from autoregressive models (Gong et al., 2025; Ye et al., 2025), and parallel decoding (Israel et al., 2025). DLM-Scope extracts and intervenes on sparse-autoencoder features (Wang et al., 2026), while studies of attention sinks find that sink locations change during generation (Rulli et al., 2026). These works analyze feature representations or attention patterns. We instead measure the functional contributions of attention heads under different masking conditions to investigate how their relative importance varies.

2.2 Attention-Head and Component Attribution

Early head-pruning studies identified specialized functions alongside substantial redundancy (Voita et al., 2019; Michel et al., 2019). Circuit analyses characterize task-specific head functions (Wang et al., 2023; Todd et al., 2024; Gould et al., 2024), while automated circuit discovery searches for interacting components (Conmy et al., 2023). Head-attribution approaches include coalition-based scoring and cooperative-game KV-cache allocation (Held & Yang, 2022; Sun et al., 2026), information-bottleneck scoring (Qian et al., 2025), and learned head gates (Nam et al., 2025). Attention attribution and ATTNLRP use gradients or relevance propagation (Hao et al., 2021; Achtibat et al., 2024). More

recent work connects relevance scores to pruning and targeted correction (Hatefi et al., 2025). AtP* localizes component contributions through approximate activation patching (Kramár et al., 2024), while inactive-head analysis uses head-output activity (Sandoval-Segura et al., 2026). These methods differ in their attribution targets and treatment of component interactions. These studies primarily investigate which components support a given task, rather than how head importance changes across DLLM denoising stages. In contrast, we estimate masking-rate-specific head contributions through asynchronous restoration, retaining stage-dependent variation while evaluating interactions at unequal restoration levels.

2.3 Path-Based Attribution

Integrated gradients attributes an output difference along a straight baseline-to-input path (Sundararajan et al., 2017), with extensions to internal neurons (Lundstrom et al., 2022) and circuit discovery (Hanna et al., 2024). Alternative paths adapt to model gradients (Kapishnikov et al., 2021) or data geometry (Zaher et al., 2024). Randomized path sampling also aggregates input attributions to reduce path-dependent variability (Jeon et al., 2023). These alternative-path methods primarily address input attribution. In contrast, ARA uses independently sampled head-restoration schedules to attribute masked-reconstruction quality within the model. It evaluates interacting heads at unequal restoration levels while preserving completeness, and retains separate scores across masking rates.

3 Preliminaries: Denoising Stages and Attention Heads

We introduce the reference-based masking setup used for head attribution and define the attention-head outputs that serve as attribution units.

3.1 Controlled Masking and Denoising Stages

Let $\mathcal{D}_{\text{attr}}$ denote the validation dataset used for head attribution. Each example $\mathbf{x} = (x_1, \ldots, x_n) \in \mathcal{D}_{\text{attr}}$ is a clean token sequence consisting of a prompt and a reference completion. We select several masking rates and construct attribution inputs by randomly masking the reference completion at each rate. At each selected rate $\rho \in (0, 1]$, we keep the prompt visible and independently mask each completion token with probability ρ. For each completion position i, the corruption distribution is

$$q_\rho(\tilde{x}_i \mid x_i) = \begin{cases} 1-\rho, & \tilde{x}_i = x_i, \\ \rho, & \tilde{x}_i = \texttt{[MASK]}, \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

Let $\mathcal{M}$ denote the masked positions and $\tilde{\mathbf{x}}_{\mathcal{M}}$ the resulting corrupted sequence, including the visible prompt. We study how attention heads contribute to recovering the original tokens at positions in $\mathcal{M}$. At each rate, we reuse each reference example with multiple random masks to obtain diverse attribution inputs. Averaging head attributions over masks and examples estimates each head's contribution while reducing dependence on particular masked positions.

During denoising, recovering masked tokens generally makes more context available for subsequent predictions. The selected masking rates serve as proxies for different denoising stages, with higher rates representing earlier stages and lower rates representing later stages. Using the same reference examples and sampling budget across rates allows us to examine how relative head importance changes with the amount of visible context.

3.2 Attention Heads as Attribution Units

Consider a Transformer-based DLLM with H attention heads in total, indexed by $h = 1, \ldots, H$ across all layers. For head h, the hidden states entering its layer determine the query, key and value representations $\mathbf{Q}_h$, $\mathbf{K}_h$ and $\mathbf{V}_h$. Its output before the layer's output projection is

$$\mathbf{O}_h = \text{softmax}\left(\frac{\mathbf{Q}_h \mathbf{K}_h^\top}{\sqrt{d_k}}\right) \mathbf{V}_h, \tag{2}$$

where d_k is the query and key dimension and softmax is applied row-wise. Each output row is a weighted combination of value vectors from the token positions attended to by that head.

Attention weights specify where a head reads information, whereas its output also depends on the value vectors being combined. We therefore use the entire head output as the attribution unit rather than individual token-to-token attention weights.

Within each layer, the head outputs are concatenated and mixed by a shared output projection before entering the residual stream. We intervene before this projection, where each head's output can still be modified separately.

4 Stage-Conditioned Head Attribution

We study head contributions under controlled reconstruction conditions. Masking rate determines how much context is visible, whereas head gates specify interventions on the model at a fixed rate. Applying the same attribution rule across rates yields an importance profile for each head.

4.1 Head-Gated Reconstruction Objective

At a fixed masking rate ρ, we evaluate head contributions using the masked reference inputs defined in Section 3. Let $\Pi_\rho(\mathbf{x})$ denote the sampling distribution over nonempty completion masks of $\mathbf{x}$ at rate ρ. The masked-token reconstruction objective is therefore well defined.

We freeze the DLLM parameters and introduce a continuous gate g_h on each head output before its layer's output projection:

$$\widetilde{\mathbf{O}}_h = g_h \mathbf{O}_h, \qquad h = 1, \dots, H, \qquad \mathbf{g} \in [0,1]^H. \tag{3}$$

The scalar g_h scales the entire output of head h across all token positions. Here $\mathbf{O}_h$ denotes the output before applying its own gate. All head outputs are recomputed for each gate configuration, so upstream interventions propagate to later layers. Setting $g_h = 0$ suppresses that head's output, whereas $g_h = 1$ leaves it unscaled. The vectors $\mathbf{0}$ and $\mathbf{1}$ therefore suppress all gated outputs and recover the original model, respectively. Residual connections and other network components remain in place in both cases. We denote the resulting model distribution by $p_\theta(\cdot \mid \tilde{\mathbf{x}}_\mathcal{M}; \mathbf{g})$.

We measure reconstruction quality at rate ρ by the expected mean log probability of the masked tokens:

$$J_\rho(\mathbf{g}) \coloneqq \mathbb{E}_{\substack{\mathbf{x}\sim\mathcal{D}_{\text{attr}} \\ \mathcal{M}\sim\Pi_\rho(\mathbf{x})}} \left[\frac{1}{|\mathcal{M}|} \sum_{i\in\mathcal{M}} \log p_\theta(x_i \mid \tilde{\mathbf{x}}_\mathcal{M}; \mathbf{g}) \right]. \tag{4}$$

Larger values indicate better reconstruction. The inner average is over masked tokens in one sequence; the expectation averages over examples and mask realizations at the fixed rate ρ. Thus ρ determines the reconstruction condition, while $\mathbf{g}$ determines which head outputs are available within that condition.

4.2 Rate-Specific Head Importance

At masking rate ρ, we seek a signed score $A_h(\rho)$ that assigns head h a share of the joint restoration effect $J_\rho(\mathbf{1}) - J_\rho(\mathbf{0})$. The same head operates under different reconstruction conditions as the amount of visible context changes. We therefore retain separate contributions at rates $\rho_1, \dots, \rho_T$:

$$\big(A_h(\rho_1), \dots, A_h(\rho_T)\big), \qquad h = 1, \dots, H. \tag{5}$$

These scores describe how each head's contribution depends on the reconstruction condition, as illustrated schematically in Figure 1(a). We use the same attribution-data distribution and attribution rule at every rate so that the comparison concerns changes in available context rather than differences in the underlying examples or attribution procedure.

5 Asynchronous Restoration Attribution

Heads can complement or compensate for one another, so the effect of restoring one head depends on the restoration levels of the others. Path-based attribution allocates the joint restoration effect by accumulating each head's marginal contribution along a restoration path. We first examine the synchronized path obtained by adapting integrated gradients to head gates, then introduce asynchronous paths that evaluate heads at different restoration levels (Figure 1(b)).

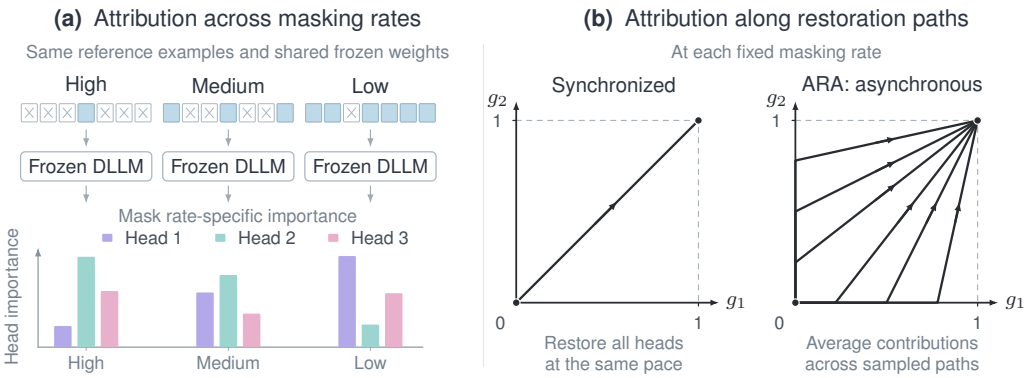

Figure 1: Stage-conditioned head attribution with ARA. (a) The same reference examples are masked at different rates, used as proxies for denoising stages. Filled and crossed tokens denote visible and masked positions. The importance profiles are schematic, not experimental results. (b) At a fixed rate, synchronized restoration uses the same schedule for all heads, whereas ARA averages contributions over paths with independently sampled activation times.

5.1 Limitation of Synchronized Restoration

To distribute the restoration effect among heads, consider an absolutely continuous, coordinate-wise nondecreasing path $\boldsymbol{\gamma} : [0,1] \to [0,1]^H$ with $\boldsymbol{\gamma}(0) = \mathbf{0}$ and $\boldsymbol{\gamma}(1) = \mathbf{1}$. The parameter τ describes progress along this counterfactual restoration path, not the denoising stage. The masking rate ρ remains unchanged while the gates vary. For a continuously differentiable J_ρ, the path assigns head h the contribution

$$\mathcal{A}_h(J_\rho; \boldsymbol{\gamma}) = \int_0^1 \partial_h J_\rho(\boldsymbol{\gamma}(\tau))\, \dot{\gamma}_h(\tau)\, d\tau. \tag{6}$$

Here $\partial_h J_\rho = \partial J_\rho / \partial g_h$ is the sensitivity to gate h, and $\dot{\gamma}_h(\tau) = d\gamma_h(\tau)/d\tau$ is its restoration rate. Their product accumulates the head's marginal effect weighted by how much its output is restored. Positive and negative scores represent net contributions that increase and decrease reconstruction quality along the path.

The chain rule gives completeness:

$$\sum_h \mathcal{A}_h(J_\rho; \boldsymbol{\gamma}) = J_\rho(\mathbf{1}) - J_\rho(\mathbf{0}). \tag{7}$$

Completeness fixes the total credit but does not determine how it is divided among heads. That division depends on the configurations encountered along the path.

Adapting standard integrated gradients (Sundararajan et al., 2017) to head gates gives the synchronized diagonal path (DP), $\boldsymbol{\gamma}^{\mathrm{DP}}(\tau) = \tau\mathbf{1}$. It restores all gates at the same rate and assigns

$$\mathcal{A}_h(J_\rho; \boldsymbol{\gamma}^{\mathrm{DP}}) = \int_0^1 \partial_h J_\rho(\tau\mathbf{1})\, d\tau. \tag{8}$$

Every head is evaluated while other heads are scaled by the same factor. This does not probe unequal restoration levels, where complementary or substitutable contributions may emerge. More integration steps only refine the same diagonal rather than expanding the configurations evaluated.

Proposition 5.1 (Limited support of synchronized paths)**.** *Let J_ρ be continuously differentiable. The synchronized attribution depends only on $\partial_h J_\rho$ along the diagonal. In particular, $\partial_h J_\rho(\tau\mathbf{1}) = 0$ for all $\tau \in [0,1]$ implies $\mathcal{A}_h(J_\rho; \boldsymbol{\gamma}^{\mathrm{DP}}) = 0$, even when $\partial_h J_\rho$ is nonzero away from the diagonal.*

The proof is in Appendix A.6.

Example. Consider the two-gate objective $\Psi(g_1, g_2) = g_2 + \int_0^{g_1} (s - g_2)^2\, ds$, whose derivative with respect to the first gate is $\partial_1 \Psi(g_1, g_2) = (g_1 - g_2)^2$. Along synchronized restoration, $g_1 = g_2 = \tau$,

so $\partial_1 \Psi(\tau, \tau) = 0$ throughout the path. DP therefore assigns zero credit to the first gate. Yet whenever $g_1 \neq g_2$, increasing g_1 while holding g_2 fixed increases the objective. The endpoint change is $\Psi(\mathbf{1}) - \Psi(\mathbf{0}) = 4/3$, which DP assigns entirely to the second gate. Completeness thus holds, but the first gate's positive marginal effect at unequal restoration levels is not reflected in its score.

5.2 Attribution over Asynchronous Paths

ARA varies the relative restoration of heads by sampling an independent activation time u_h for each gate. Collecting these times in $\mathbf{u}$, the sampled path $\boldsymbol{\gamma}_{\mathbf{u}}$ has coordinates

$$\gamma_{\mathbf{u},h}(\tau) = \mathrm{clip}_{[0,1]}\left(\frac{\tau - u_h}{1 - u_h}\right). \tag{9}$$

Each activation time u_h is sampled independently and uniformly from $[0, 1)$, and $\mathrm{clip}_{[0,1]}$ clips its argument to $[0, 1]$. Gate h stays at zero until u_h, then increases linearly to one at $\tau = 1$. Each path has the same endpoints as DP, but heads reach different restoration levels along the way (Figure 1). We compute each path's contribution with Equation (6) and average over activation times.

Definition 5.2 (**Asynchronous Restoration Attribution**). For the reconstruction objective J_ρ, ARA assigns head h the expected contribution over the asynchronous paths in Equation (9):

$$A_h(\rho) \coloneqq \mathbb{E}_{\mathbf{u}}\left[\mathcal{A}_h(J_\rho; \boldsymbol{\gamma}_{\mathbf{u}})\right] = \int_{[0,1)^H} \int_0^1 \partial_h J_\rho(\boldsymbol{\gamma}_{\mathbf{u}}(\tau))\, \dot{\gamma}_{\mathbf{u},h}(\tau)\, d\tau\, d\mathbf{u}. \tag{10}$$

Applying this rule separately at each masking rate gives the importance profiles introduced in Section 4.2. Path averaging takes place at a fixed rate and therefore retains dependence on ρ. Completeness holds at each rate because every sampled path has the same endpoints. Further axiomatic properties are stated and proved in Appendix A.5.

The following result characterizes the configurations considered by asynchronous attribution and makes the contrast with DP precise.

Proposition 5.3 (Full-dimensional support of asynchronous paths). *For every head h, asynchronous restoration induces a probability measure ν_h on $[0, 1]^H$ such that $A_h(\rho) = \mathbb{E}_{\mathbf{g}\sim\nu_h}[\partial_h J_\rho(\mathbf{g})]$ for every continuously differentiable J_ρ. Moreover, $\nu_h(O) > 0$ for every nonempty open set $O \subset (0, 1)^H$.*

The proof is in Appendix A.6. The measure ν_h weights a region by how much head h is restored while the joint configuration lies there. It is not uniform over gate vectors. Unlike DP, this weighting gives positive mass to every open interior region. For the two-gate example, ARA assigns $1/9$ to the first gate instead of zero (Appendix A.7). The support statement concerns the path distribution, not a guarantee that finitely many paths visit every region. Contributions of opposite signs can cancel.

In practice, we average finitely many paths and numerically integrate each one. The estimator weights gate gradients by the actual gate increments to account for different restoration schedules. Sampling settings, computational cost, and finite-sample error analysis are given in Appendix A.4.

6 Experiments

6.1 Experimental Setup

Models and tasks. We evaluate LLaDA-1.5 (Nie et al., 2025) and Dream-v0-Instruct-7B (Dream) (Ye et al., 2025) on eight tasks covering knowledge and multiple-choice reasoning (MMLU, CMMLU, C-Eval, and GPQA), mathematical reasoning (GSM8K and Minerva Math), and code generation (HumanEval and MBPP). Appendix A.1 gives splits, metrics, and generation settings.

Baseline methods. For head pruning and sparse inference, we compare ARA with ATTATTR (Hao et al., 2021), CoKV (Sun et al., 2026), and causal head gating (CHG) (Nam et al., 2025). These methods represent gradient-based attention attribution, cooperative-game attribution, and learned head gating, respectively. We adapt CoKV's utility and CHG's gating objective to masked reconstruction while keeping the DLLM parameters frozen. In Sections 6.3 and 6.4, all methods, including baselines, compute scores separately at each masking rate. Their precomputed rate-specific scores guide

dynamic pruning and sparse allocation as denoising proceeds; attribution is not recomputed online. Within each model and evaluation setting, all methods use the same intervention units and downstream protocol. Implementation details and attribution budgets are given in Appendix A.1.

Research questions. We examine **(R1)** how head importance varies across masking rates (Section 6.2); **(R2)** whether ARA identifies functionally important heads (Section 6.3); **(R3)** whether it guides adaptive sparse attention (Section 6.4); and **(R4)** the effects of synchronized versus asynchronous paths and global versus rate-specific rankings on pruning (Section 6.5).

6.2 Head Importance across Masking Rates

We estimate rate-specific ARA rankings for both models on HumanEval and Minerva Math. Within each model and task, the examples and estimator budget are fixed across masking rates 0.15, 0.30, 0.50, 0.70, and 0.90. Figure 2 compares ranking pairs using Spearman correlation. This measures agreement in relative head importance without conflating it with changes in score scale.

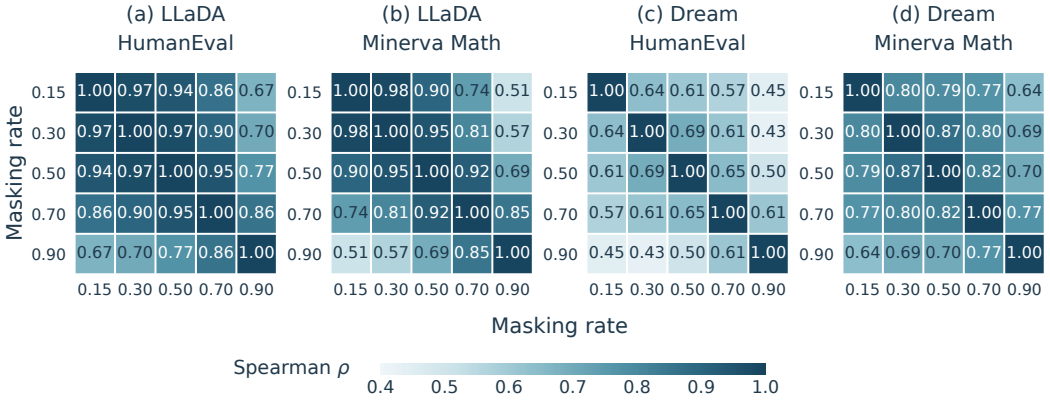

Figure 2: Head-ranking agreement across masking rates. Each cell is the Spearman correlation between the two indicated rates, computed over all query heads: 1,024 for LLaDA and 784 for Dream. Additional rate-wise analyses are provided in Appendix A.2. All panels share the same color scale; darker cells indicate greater agreement.

Nearby masking rates tend to produce more similar head rankings, whereas their agreement generally decreases as the rates move farther apart. Between rates 0.15 and 0.90, correlations are 0.67 and 0.51 for LLaDA, and 0.45 and 0.64 for Dream on HumanEval and Minerva Math, respectively. This distance-dependent pattern suggests a structured change in relative head importance as the amount of visible context varies.

For LLaDA, we also compare the sets of 103 highest-ranked heads across all layers (approximately 10%). Their mean overlap between adjacent rates is 84.95% on HumanEval and 86.41% on Minerva Math. Between 0.15 and 0.90, the overlap falls to 60.19% and 56.31%, respectively. These changes concern which heads enter the important-head set, not merely the ordering of lower-ranked heads. A complementary layer-wise analysis in Appendix A.2 further shows that LLaDA's highest-ranked heads shift toward shallower layers as the masking rate decreases. Together, these findings reveal both shared structure across nearby reconstruction conditions and changes over more widely separated conditions. Using controlled corruptions of reference sequences as proxies for denoising stages, they motivate stage-conditioned attribution rather than relying solely on a single global ranking.

6.3 Identifying Functionally Important Heads

We test whether attribution rankings identify functionally important heads by removing the top-ranked components and measuring performance degradation. All attribution methods use dynamic pruning: the current masking rate selects the corresponding precomputed ranking, allowing the pruned head set to change across denoising stages. Within each model, all methods use the same selection rule,

Table 1: Task performance under dynamic prune-most interventions. Every attribution method selects top-ranked components using its scores for the current masking rate. Within each model, lower is better because it indicates that the ranking identifies components with larger functional effects. Attribution-method entries report mean ± standard deviation over two runs (population standard deviation, $n = 2$); Dense is a reference. Macro averages cover all eight tasks.

	MMLU	CMMLU	C-Eval	GPQA	GSM8K	Minerva	HE	MBPP	Avg.
Dream *(Dense)*	43.42	34.59	34.03	29.50	82.50	39.90	39.52	54.50	44.75
+ ATTATTR	$33.49_{\pm 0.86}$	$30.51_{\pm 1.48}$	$29.46_{\pm 1.90}$	$25.50_{\pm 2.00}$	$47.25_{\pm 3.25}$	$29.56_{\pm 7.39}$	$37.90_{\pm 0.81}$	$55.00_{\pm 0.00}$	$36.08_{\pm 0.32}$
+ CoKV	$41.10_{\pm 1.18}$	$31.94_{\pm 0.30}$	$30.76_{\pm 1.19}$	$29.50_{\pm 1.50}$	$67.00_{\pm 2.50}$	$38.42_{\pm 0.99}$	$39.11_{\pm 0.40}$	$54.50_{\pm 0.50}$	$41.54_{\pm 0.22}$
+ CHG	$33.86_{\pm 3.86}$	$33.75_{\pm 0.47}$	$32.80_{\pm 2.79}$	$26.25_{\pm 1.25}$	$74.00_{\pm 4.00}$	$25.37_{\pm 2.71}$	$37.50_{\pm 1.21}$	$55.50_{\pm 0.00}$	$39.88_{\pm 0.36}$
+ ARA	$38.20_{\pm 0.00}$	$24.98_{\pm 0.39}$	$28.27_{\pm 1.00}$	$26.25_{\pm 2.25}$	$12.00_{\pm 2.50}$	$21.43_{\pm 0.25}$	$38.31_{\pm 0.40}$	$54.75_{\pm 0.75}$	$30.52_{\pm 0.78}$
LLaDA *(Dense)*	65.09	67.24	66.05	26.50	80.00	33.00	37.10	37.00	51.50
+ ATTATTR	$45.42_{\pm 0.42}$	$34.31_{\pm 3.38}$	$35.77_{\pm 2.86}$	$27.00_{\pm 0.00}$	$43.75_{\pm 4.75}$	$22.91_{\pm 0.25}$	$36.29_{\pm 0.00}$	$37.00_{\pm 0.00}$	$35.31_{\pm 0.24}$
+ CoKV	$54.10_{\pm 1.56}$	$59.03_{\pm 0.45}$	$59.14_{\pm 0.45}$	$29.25_{\pm 1.25}$	$59.00_{\pm 0.00}$	$28.33_{\pm 0.25}$	$37.50_{\pm 0.40}$	$37.25_{\pm 0.25}$	$45.45_{\pm 0.10}$
+ CHG	$48.77_{\pm 1.93}$	$50.39_{\pm 0.76}$	$48.37_{\pm 2.38}$	$30.75_{\pm 0.75}$	$46.00_{\pm 3.00}$	$25.62_{\pm 0.99}$	$36.29_{\pm 0.00}$	$37.00_{\pm 0.00}$	$40.40_{\pm 0.04}$
+ ARA	$44.54_{\pm 1.91}$	$35.09_{\pm 0.17}$	$32.91_{\pm 1.93}$	$25.25_{\pm 0.25}$	$21.50_{\pm 3.00}$	$21.43_{\pm 0.74}$	$35.89_{\pm 0.40}$	$36.50_{\pm 0.50}$	$31.64_{\pm 0.70}$

deletion budget, and evaluation protocol. Table 1 reports post-pruning task scores, where lower values indicate that removing the selected components has a larger functional effect.

Across the two runs, removing ARA's top-ranked components yields the lowest macro-average task score for both model families. The scores are 30.52 on Dream and 31.64 on LLaDA, compared with 36.08 and 35.31, respectively, for the closest baseline, ATTATTR. Since stage conditioning and the deletion budget are shared across methods, this comparison tests the attribution rules within the same dynamic intervention protocol. The larger degradation supports the functional relevance of the heads identified by ARA. At each rate, its asynchronous path averaging evaluates head contributions at unequal restoration levels, providing a plausible explanation for the stronger intervention effects.

6.4 Guiding Adaptive Sparse Attention

We evaluate whether head attribution preserves task performance under sparse attention budgets. Here, adaptive sparse attention retains a subset of token blocks and assigns larger block budgets to more important heads. All attribution methods use precomputed, rate-specific importance scores to adapt this allocation as denoising proceeds. Each method selects its scores according to the current masking rate, so the baselines and ARA all use dynamic sparse attention. All methods share the same sparse-allocation rule and generation protocol. We vary the nominal retained-block ratio from 10% to 50%. Figure 3 reports the macro-average task score over eight tasks alongside the dense reference. At a given nominal ratio, higher scores indicate better preservation of task performance.

Across both models, ARA gives the highest macro-average score at every retained-block ratio from 20% through 50%. At the aggressive 10% setting, all methods are tightly grouped, with ARA trailing the best baseline by 0.67 points on Dream and 0.52 on LLaDA. At 50%, ARA is within 1.97 and 0.76 points of the respective dense scores. Under the shared dynamic allocation protocol, these results support the practical utility of ARA's asynchronous-path scores for directing limited attention budgets toward influential heads.

6.5 Attribution Ablations

We compare three variants to separate restoration-path design from stage-dependent head selection (Table 2). DP (Dynamic) uses synchronized-path attributions computed separately at each masking rate, whereas ARA (Dynamic) uses asynchronous-path attributions. Both select the rate-specific ranking according to the current masking rate during denoising and update the pruned head set accordingly. ARA (Global) instead aggregates signed ARA scores across multiple masking rates into one global score per head. The resulting fixed ranking is reused whenever pruning is applied, yielding a static rather than stage-dependent selection.

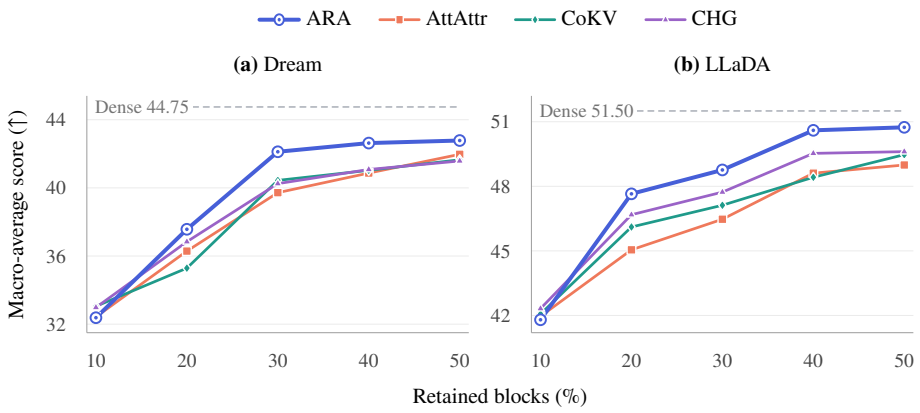

Figure 3: Attribution-guided sparse inference across allocation budgets. Each point is the unweighted macro average over the eight tasks, using the same per-task evaluation sizes as the prune-most study. Each curve uses the same precomputed attribution scores across budget settings. Every method, including the baselines, selects its rate-specific scores according to the current masking rate during denoising, yielding dynamic sparse attention. Gray dashed lines mark dense performance. Higher is better; vertical scales differ by model.

Table 2: Prune-most ablations of restoration paths and stage-dependent head selection (lower is better). DP (Dynamic) and ARA (Dynamic) use rate-specific rankings for dynamic pruning; ARA (Global) uses a multi-rate-aggregated ranking for static pruning. Entries report means and population standard deviations over two runs. Avg. summarizes the unweighted eight-task mean.

Model	Variant	MMLU	CMMLU	C-Eval	GPQA	GSM8K	Minerva	HE	MBPP	Avg.
Dream	DP (Dynamic)	$30.55_{\pm 0.33}$	$29.69_{\pm 1.29}$	$33.10_{\pm 1.08}$	$27.50_{\pm 2.50}$	$43.75_{\pm 3.25}$	$22.17_{\pm 0.50}$	$38.71_{\pm 0.00}$	$55.00_{\pm 0.00}$	$35.06_{\pm 0.05}$
	ARA (Global)	$37.30_{\pm 0.11}$	$28.45_{\pm 0.28}$	$29.27_{\pm 0.07}$	$23.75_{\pm 0.75}$	$20.00_{\pm 2.50}$	$22.17_{\pm 0.99}$	$38.71_{\pm 0.00}$	$55.00_{\pm 0.50}$	$31.83_{\pm 0.34}$
	ARA (Dynamic)	$38.20_{\pm 0.00}$	$24.98_{\pm 0.39}$	$28.27_{\pm 1.00}$	$26.25_{\pm 2.25}$	$12.00_{\pm 2.50}$	$21.43_{\pm 0.25}$	$38.31_{\pm 0.40}$	$54.75_{\pm 0.75}$	$30.52_{\pm 0.78}$
LLaDA	DP (Dynamic)	$45.53_{\pm 1.71}$	$45.60_{\pm 8.40}$	$33.36_{\pm 4.01}$	$27.00_{\pm 1.00}$	$38.25_{\pm 0.75}$	$22.17_{\pm 0.50}$	$35.89_{\pm 0.40}$	$36.50_{\pm 0.00}$	$35.54_{\pm 1.00}$
	ARA (Global)	$43.68_{\pm 0.22}$	$32.41_{\pm 1.25}$	$37.67_{\pm 4.61}$	$30.50_{\pm 2.00}$	$25.75_{\pm 4.25}$	$21.92_{\pm 1.23}$	$36.69_{\pm 0.40}$	$36.75_{\pm 0.25}$	$33.17_{\pm 1.05}$
	ARA (Dynamic)	$44.54_{\pm 1.91}$	$35.09_{\pm 0.17}$	$32.91_{\pm 1.93}$	$25.25_{\pm 0.25}$	$21.50_{\pm 3.00}$	$21.43_{\pm 0.74}$	$35.89_{\pm 0.40}$	$36.50_{\pm 0.50}$	$31.64_{\pm 0.70}$

ARA (Dynamic) lowers the macro-average post-pruning score relative to DP (Dynamic) by 4.54 points on Dream and 3.90 on LLaDA. Asynchronous paths evaluate heads at unequal restoration levels, potentially capturing contributions missed by the diagonal. Compared with ARA (Global), ARA (Dynamic) lowers scores by 1.31 and 1.53 points, respectively, supporting stage-dependent head selection.

7 Conclusion

In this paper, we investigated stage-dependent head importance in DLLMs using masking rates as controlled proxies for denoising stages. Head rankings retain shared structure but tend to diverge as masking rates move farther apart, revealing structured variation across reconstruction conditions. Motivated by this observation, we introduced Asynchronous Restoration Attribution (ARA) to quantify head contributions at each masking rate through masked reconstruction. Its asynchronous restoration paths evaluate heads at unequal restoration levels and capture interaction-dependent contributions beyond the synchronized diagonal path. Experiments on LLaDA and Dream across eight tasks support the effectiveness of ARA in identifying functionally important heads through dynamic pruning and guiding adaptive sparse attention. The path ablation further supports the benefit of asynchronous restoration. Future work will study cross-task and cross-model generalization of head importance and attention–feed-forward interactions.

AI USE STATEMENT

The authors developed the original research idea and initial experimental design and wrote the initial core implementation of DLLM head attribution. Generative AI tools, including Codex, assisted with debugging, adapting open-source implementations for sparse attention and head pruning, and implementing and debugging the baselines. AI tools also supported discussions of experimental comparisons, interpretation and presentation of results, literature search, and manuscript organization, translation, and language editing. They also assisted in checking mathematical proofs. The authors take responsibility for the final content of the paper and its associated implementations.

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, 2024.

Marianne Arriola, Aaron Gokaslan, Justin T. Chiu, Zhihan Yang, Zhixuan Qi, Jiaqi Han, Subham Sekhar Sahoo, and Volodymyr Kuleshov. Block diffusion: Interpolating between autoregressive and diffusion language models. In *The Thirteenth International Conference on Learning Representations*, pp. 50726–50753, 2025.

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, 2023.

Shansan Gong, Shivam Agarwal, Yizhe Zhang, Jiacheng Ye, Lin Zheng, Mukai Li, Chenxin An, Peilin Zhao, Wei Bi, Jiawei Han, et al. Scaling diffusion language models via adaptation from autoregressive models. In *The Thirteenth International Conference on Learning Representations*, 2025.

Rhys Gould, Euan Ong, George Ogden, and Arthur Conmy. Successor heads: Recurring, interpretable attention heads in the wild. In *The Twelfth International Conference on Learning Representations*, 2024.

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.

Yaru Hao, Li Dong, Furu Wei, and Ke Xu. Self-attention attribution: Interpreting information interactions inside transformer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 12963–12971, 2021.

Sayed Mohammad Vakilzadeh Hatefi, Maximilian Dreyer, Reduan Achtibat, Patrick Kahardipraja, Thomas Wiegand, Wojciech Samek, and Sebastian Lapuschkin. Attribution-guided pruning for compression, circuit discovery, and targeted correction in LLMs. *arXiv preprint arXiv:2506.13727*, 2025.

William Held and Diyi Yang. Shapley head pruning: Identifying and removing interference in multilingual transformers. *arXiv preprint arXiv:2210.05709*, 2022.

Daniel Mingyi Israel, Guy Van den Broeck, and Aditya Grover. Accelerating diffusion LLMs via adaptive parallel decoding. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

Giyoung Jeon, Haedong Jeong, and Jaesik Choi. Beyond single path integrated gradients for reliable input attribution via randomized path sampling. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2052–2061, 2023.

Andrei Kapishnikov, Subhashini Venugopalan, Besim Avci, Ben Wedin, Michael Terry, and Tolga Bolukbasi. Guided integrated gradients: An adaptive path method for removing noise. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5050–5058, 2021.

János Kramár, Tom Lieberum, Rohin Shah, and Neel Nanda. AtP*: An efficient and scalable method for localizing LLM behaviour to components. *arXiv preprint arXiv:2403.00745*, 2024.

Aaron Lou, Chenlin Meng, and Stefano Ermon. Discrete diffusion modeling by estimating the ratios of the data distribution. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 32819–32848, 2024.

Daniel D. Lundstrom, Tianjian Huang, and Meisam Razaviyayn. A rigorous study of integrated gradients method and extensions to internal neuron attributions. In *Proceedings of the 39th International Conference on Machine Learning*, pp. 14485–14508, 2022.

Paul Michel, Omer Levy, and Graham Neubig. Are sixteen heads really better than one? *Advances in neural information processing systems*, 32, 2019.

Andrew Nam, Henry Conklin, Yukang Yang, Thomas Griffiths, Jonathan D. Cohen, and Sarah-Jane Leslie. Causal head gating: A framework for interpreting roles of attention heads in transformers. In *Advances in Neural Information Processing Systems*, volume 38, 2025.

Shen Nie, Fengqi Zhu, Zebin You, Xiaolu Zhang, Jingyang Ou, Jun Hu, Jun Zhou, Yankai Lin, Ji-Rong Wen, and Chongxuan Li. Large language diffusion models. In *Advances in Neural Information Processing Systems*, volume 38, pp. 50608–50646, 2025.

Yukun Qian, Xuyi Zhuang, and Mingjiang Wang. Head information bottleneck (HIB): leveraging information bottleneck for efficient transformer head attribution and pruning. *EURASIP Journal on Audio, Speech, and Music Processing*, 2025(1):25, 2025.

Maximo Eduardo Rulli, Simone Petruzzi, Edoardo Michielon, Fabrizio Silvestri, Simone Scardapane, and Alessio Devoto. Attention sinks in diffusion language models. In *ICLR 2026 Workshop on Trustworthy AI*, 2026.

Subham Sekhar Sahoo, Marianne Arriola, Yair Schiff, Aaron Gokaslan, Edgar Marroquin, Justin T. Chiu, Alexander Rush, and Volodymyr Kuleshov. Simple and effective masked diffusion language models. In *Advances in Neural Information Processing Systems*, volume 37, pp. 130136–130184, 2024.

Pedro Sandoval-Segura, Xijun Wang, Ashwinee Panda, Micah Goldblum, Ronen Basri, Tom Goldstein, and David Jacobs. Identifying and evaluating inactive heads in pretrained LLMs. In *The Fourteenth International Conference on Learning Representations*, 2026.

Jiaxin Shi, Kehang Han, Zhe Wang, Arnaud Doucet, and Michalis K. Titsias. Simplified and generalized masked diffusion for discrete data. In *Advances in Neural Information Processing Systems*, volume 37, pp. 103131–103167, 2024.

Qiheng Sun, Hongwei Zhang, Junxu Liu, Haocheng Xia, Jinfei Liu, Kui Ren, and Haibo Hu. Efficient cooperation-aware key and value management for LLM inference. *Proceedings of the VLDB Endowment*, 19(9):2154–2167, 2026. doi: 10.14778/3819518.3819541.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning*, pp. 3319–3328, 2017.

Eric Todd, Millicent L. Li, Arnab Sen Sharma, Aaron Mueller, Byron C. Wallace, and David Bau. Function vectors in large language models. In *The Twelfth International Conference on Learning Representations*, 2024.

Elena Voita, David Talbot, Fedor Moiseev, Rico Sennrich, and Ivan Titov. Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned. In *Proceedings of the 57th annual meeting of the association for computational linguistics*, pp. 5797–5808, 2019.

Kevin Ro Wang, Alexandre Variengien, Arthur Conmy, Buck Shlegeris, and Jacob Steinhardt. Interpretability in the wild: A circuit for indirect object identification in GPT-2 small. In *The Eleventh International Conference on Learning Representations*, 2023.

Xu Wang, Bingqing Jiang, Yu Wan, Baosong Yang, Lingpeng Kong, and Difan Zou. DLM-Scope: Mechanistic interpretability of diffusion language models via sparse autoencoders. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026.

Jiacheng Ye, Zhihui Xie, Lin Zheng, Jiahui Gao, Zirui Wu, Xin Jiang, Zhenguo Li, and Lingpeng Kong. Dream 7B: Diffusion large language models. *arXiv preprint arXiv:2508.15487*, 2025.

Eslam Zaher, Maciej Trzaskowski, Quan Nguyen, and Fred Roosta. Manifold integrated gradients: Riemannian geometry for feature attribution. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 58090–58104, 2024.

A Appendix

A.1 Implementation Details

Reproducibility. For all attribution methods, we fix random seeds for dataset subsampling, mask generation, estimator-specific Monte Carlo sampling, and optimization when applicable. For each experiment, we record the attribution dataset, the normalized data weights $\{\omega_m\}_{m=1}^{M}$, the attribution/evaluation partition, and the intervention unit. These details help distinguish variation in attribution estimates from differences in the underlying data or intervention protocol. Fixed seeds specify the stochastic protocol but do not, by themselves, guarantee bitwise determinism in GPU-based gate optimization. Small numerical differences can accumulate over successive optimization updates.

Attribution and evaluation data. Table 3 summarizes the sample sizes for ARA attribution and downstream evaluation in the main intervention experiments. Attribution counts refer to distinct reference examples rather than their masked variants. For every dataset and attribution method, the attribution and evaluation sets are strictly disjoint at the example level. When drawn from the same benchmark split, they comprise non-overlapping subsets.

Table 3: Sample sizes for the main intervention experiments. Evaluation counts are shared by both models.

	ARA attribution		Evaluation
Dataset	Dream	LLaDA	Both models
MMLU	200	200	2,280
CMMLU	134	134	2,680
C-Eval	156	156	1,346
GPQA	199[†]	200	200
GSM8K	100	100	200
Minerva Math	100	100	203
HumanEval[‡]	40	40	124
MBPP	100	100	200

[†]Dream GPQA loads 200 examples; 199 have a usable completion.
[‡]For HumanEval, 40 test examples are used for attribution and the remaining 124 for evaluation; the two subsets are disjoint.

ARA **configuration.** Head contributions are estimated separately at each masking rate using the same path configuration, enabling controlled comparisons across reconstruction conditions. The full configuration of ARA comprises the masking rates $\{\rho_t\}_{t=1}^{T}$, mask samples per rate S, integration steps per path K, and sampled paths P. The reported estimates use $K = 8$ integration steps per path and $P = 4$ asynchronous paths. Here, K controls the numerical resolution of each path integral, whereas P determines the number of restoration schedules averaged. Appendix A.2 examines changes in head rankings, layer distributions, and top-head membership across masking rates with fixed examples and per-rate estimator budgets.

Budget sensitivity on LLaDA. We assess the default $(K, P) = (8, 4)$ on LLaDA-1.5 using 40 Minerva Math training examples, five masking rates, and $S = 2$ masks per example at each rate. We double either K or P while holding the examples and masks fixed. The $K = 16$ estimate uses the same four paths, and the $P = 8$ estimate retains the original four among its eight paths. We also compare two independently sampled four-path sets on the same examples and masks, and two disjoint 20-example subsets at $(K, P) = (8, 4)$. The latter comparison includes variation in examples, masks, and paths. Table 4 reports rate-specific rank agreement and overlap of the selected heads under the pruning rule of six heads per layer.

Increasing K or P gives mean Spearman correlations of 0.985 and 0.983, with selected-head overlaps of 92.2% and 88.3%, respectively. The independent four-path estimates retain a mean correlation of 0.936, while the disjoint data subsets give 0.849. These results indicate that $(K, P) = (8, 4)$ already yields reasonably stable aggregate rate-specific rankings in this LLaDA setting. We therefore retain this configuration rather than increasing the integration steps or number of sampled paths.

Table 4: Budget sensitivity of ARA on LLaDA-1.5 and Minerva Math. Each entry gives Spearman correlation across all 1,024 query heads / selected-head overlap (%). Overlap is the intersection size divided by 192, the number of heads selected by the per-layer pruning rule. Mean averages the five rates.

Mask rate	$K = 8$ vs. 16 ($P = 4$)	$P = 4$ vs. 8 ($K = 8$)	Independent four-path sets	Disjoint 20-example subsets
0.15	0.991 / 92.2	0.983 / 87.5	0.942 / 80.2	0.864 / 75.0
0.30	0.986 / 93.2	0.987 / 88.5	0.954 / 84.9	0.875 / 72.9
0.50	0.988 / 95.3	0.987 / 91.7	0.928 / 82.3	0.837 / 70.3
0.70	0.986 / 90.1	0.985 / 90.1	0.943 / 83.3	0.867 / 72.4
0.90	0.976 / 90.1	0.975 / 83.9	0.914 / 76.0	0.802 / 62.5
Mean	0.985 / 92.2	0.983 / 88.3	0.936 / 81.4	0.849 / 70.6

Attribution-score convention. Attribution scores are oriented so that larger values indicate greater importance. This convention does not specify the preferred direction of downstream task scores: lower post-removal performance indicates a stronger prune-most intervention, whereas higher performance is preferred for sparse inference. The ARA definition attributes J_ρ, the negative expected reconstruction loss at rate ρ; when an implementation accumulates gradients in the loss direction, we negate the resulting vector once. Scores remain separate across masking rates for every method in the dynamic interventions of Sections 6.3 and 6.4. At each rate, signed attribution vectors are averaged over examples using normalized data weights ω_m. For the static ARA (Global) ablation and global-ranking analyses, we then optionally aggregate these rate-specific scores using the prescribed rate weights. No element-wise transformation is applied before or after aggregation.

Attribution baselines. For ATTATTR (Hao et al., 2021) and CoKV, we use the same attribution examples and target component set as ARA. For CoKV, we adapt the coalition utility to negative masked-reconstruction loss, use coalition fractions 0.25, 0.5, and 0.75, and calibrate the task-specific Monte Carlo sample count against the measured attribution time of ARA.

We adapt CHG by replacing its autoregressive next-token objective with mean masked-reconstruction loss while keeping the DLLM frozen. A sigmoid gate is learned for every attention-head output. We first optimize an unregularized shared gate for 150 updates, fork that solution, and optimize its positive- and negative-logit-regularized branches for 150 updates each. We use the negative branch G^- as the facilitation ranking. The regularization magnitude is 0.1, the logit regularizer is clipped at 4, and the effective batch size is 64; Adam learning rates decay from 0.1 to 0.01 for the shared phase and from 0.5 to 0.1 for the regularized phases.

Model-specific intervention units. Dream uses grouped-query attention, so query-head scores are averaged within each seven-head KV group. We globally remove the top 5% of KV groups over layers 0–27, with at most one group removed per layer. We adopt this budget because Dream is sensitive to head removal. In our experiments, removing more than 10% of KV groups caused task

performance to collapse across all compared attribution methods. For LLaDA, we remove the top 20% of heads independently in every layer over layers 0–31. We compare methods within each model and do not interpret absolute degradation as directly comparable across these two intervention units.

Adaptive sparse inference protocol. For adaptive sparse inference, every attribution method uses the same rule for converting head scores into sparse allocation weights, with logical block size 32. The budget sweep varies the nominal retained-block ratio over $\{0.1, 0.2, 0.3, 0.4, 0.5\}$. Every method, including each baseline, selects precomputed rate-specific scores according to the current masking rate during denoising and updates its sparse allocation accordingly. Each method reuses the same collection of rate-specific scores across budget settings; this does not mean using a single fixed ranking throughout denoising. Dream averages the selected rate-specific query-head scores within KV groups before allocation. Model-specific harness settings include generation length, diffusion steps, prompt formatting, and answer extraction.

Head masking and pruning protocol. In Section 6.3, all attribution methods use dynamic pruning based on scores estimated separately at each masking rate. The current masking rate determines which precomputed ranking is used to select the removed components. The selected set can change across denoising stages, while the number or fraction removed follows the same model-specific budget and selection rule for every method. The explicitly labeled ARA (Global) variant in Section 6.5 is a separate static-pruning control.

Evaluation protocol and metrics. The scored sample counts are reported in Table 3. MMLU and CMMLU use 40 examples per subject, while Minerva Math uses 29 examples from each of seven subtasks. C-Eval uses the full validation split, while HumanEval uses the 124 test examples held out from attribution. Multiple-choice tasks are evaluated by accuracy. For mathematical reasoning, we report flexible-extraction exact match on GSM8K and accuracy based on `math_verify` on Minerva Math. Code generation is evaluated by pass@1. Model-specific generation settings and answer extraction follow the corresponding evaluation harness.

Attribution budgets. In the path-design ablation, ARA uses four asynchronous paths with eight integration steps each, whereas DP uses one diagonal path with 32 steps. Both configurations therefore use the same number of gate-gradient evaluations under identical attribution-data and masking settings. The comparison isolates the allocation of this budget across restoration paths rather than an increase in its size.

For CoKV, we calibrate the task-specific Monte Carlo sample count against the measured attribution time of ARA under the same hardware setting. For ATTATTR, we adapt the original attention-attribution procedure to masked reconstruction and verify that its attribution estimates have converged under the reported integration budget. For CHG, we retain the three-phase gate-optimization structure and verify convergence of the learned gate scores under the reported optimization budget. All configurations are fixed before downstream evaluation.

A.2 Additional Analyses of Rate-Specific Head Importance

Figure 2 shows that head rankings change across masking rates. Here we examine where these changes occur and whether they involve the highest-ranked heads. We focus on layer-wise redistribution, top-head membership, and within-layer rankings.

Layer-wise redistribution in LLaDA. Figure 4 locates the changes in LLaDA's rankings on HumanEval (code) and Minerva Math (math). Both tasks show a shift toward shallower layers as the masking rate decreases. For masking rates $0.90, 0.70, 0.50, 0.30, 0.15$, the mean layer indices of the highest-ranked 103 heads are $23.66, 23.07, 22.74, 22.44, 21.08$ for code and $25.83, 24.41, 23.89, 23.15, 23.05$ for math. The endpoint shift toward shallower layers also holds when selecting the highest-ranked 52 or 205 heads (approximately 5% or 20%). The corresponding high-to-low endpoint means are $23.13 \rightarrow 19.73$ and $23.94 \rightarrow 21.70$ for code, and $26.69 \rightarrow 21.87$ and $25.51 \rightarrow 23.11$ for math.

Top-head overlap. Whole-model rank correlation does not reveal whether the highest-ranked heads themselves change. We select the 103 highest-ranked heads (approximately 10%) at each rate,

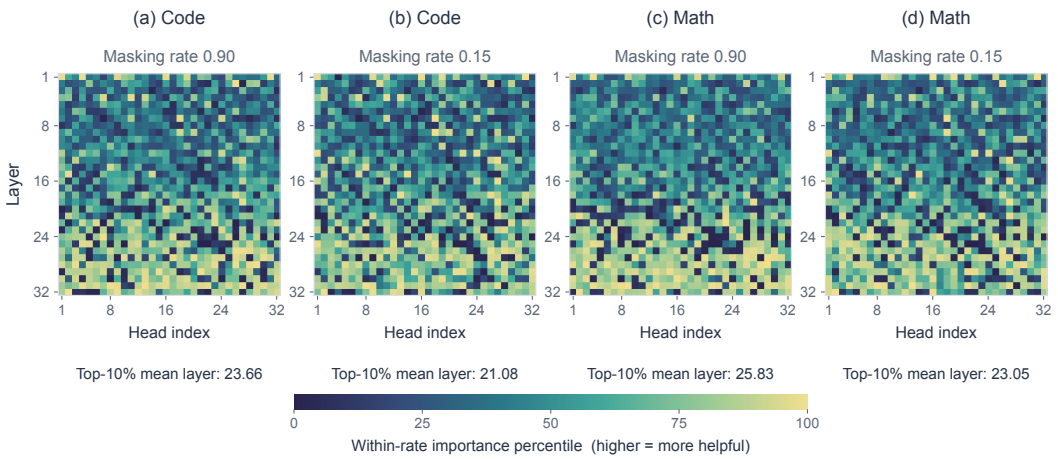

Figure 4: Layer-wise redistribution of head importance in LLaDA. The code panels use HumanEval test-set reference solutions, and the math panels use MATH training examples. Colors show within-rate importance rank percentiles across all 1,024 heads; brighter cells are ranked higher. Head and layer coordinates are fixed, with shallower layers at the top. Each panel reports the mean layer index of the highest-ranked 103 heads (approximately 10%).

breaking ties by ascending layer–head index. Overlap is the number of shared heads divided by 103. This whole-model diagnostic differs from the per-layer pruning rule.

Figure 5(a) compares each rate with $\rho = 0.90$. At $\rho = 0.15$, only 62 of 103 code heads and 58 of 103 math heads remain in the selected set, giving overlaps of 60.19% and 56.31%. In contrast, overlap averaged over the four adjacent rate pairs is 84.95% for code and 86.41% for math. The contrast persists when selecting the top 5% or 20%: mean adjacent-rate overlap remains higher than endpoint overlap in both tasks. Thus the cross-rate differences extend to the identity of the highest-ranked heads, rather than only to the ordering of less important heads.

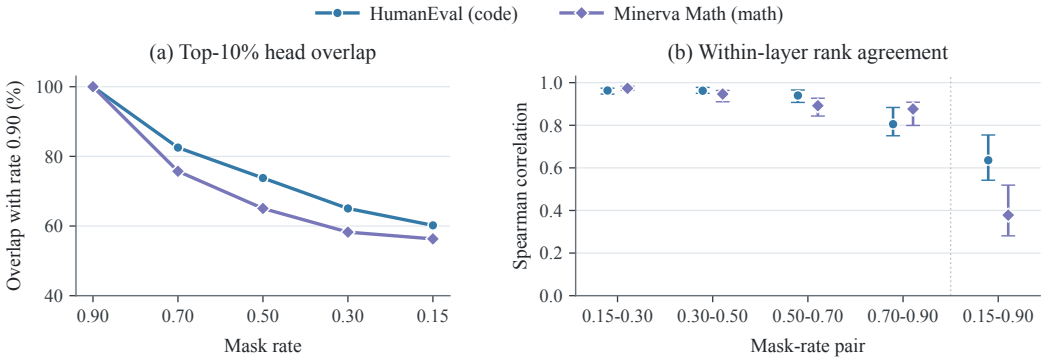

Figure 5: Changes in important-head membership and within-layer rankings in LLaDA. Code and math use the same reference data as Figure 4. (a) Fraction of the 103 highest-ranked heads shared with the selection at $\rho = 0.90$. The masking rate decreases from left to right. (b) Spearman correlation computed separately within each layer for the indicated rate pairs. Markers show medians over 32 layers and whiskers show interquartile ranges, not confidence intervals over independent runs. The separated final pair compares the endpoint rates.

Within-layer rankings. To distinguish changes within layers from shifts between layers, we compute Spearman correlation among the 32 heads of each LLaDA layer. Figure 5(b) summarizes the 32 layer-wise correlations for each of the four adjacent rate pairs and the endpoint pair.

For adjacent rates, the median correlation across layers ranges from 0.81 to 0.96 for code and from 0.88 to 0.97 for math. Between 0.15 and 0.90, these medians fall to 0.64 and 0.38, respectively. Hence the changes are not explained solely by layers moving together in the whole-model ranking: the relative ordering of heads also changes within individual layers.

A.3 Task-Specific Head Importance

We examine task dependence using the signed ARA scores of eight benchmarks. For each task, we first average signed scores equally over the five masking rates, then compute pairwise Spearman correlations across all query heads within each model (Figure 6). All estimates use $(K, P) = (8, 4)$.

Rankings differ substantially across tasks, supporting attribution on data representative of the target task rather than assuming a universal ranking. Correlations between the four multiple-choice tasks and the remaining tasks average 0.14 on Dream and 0.33 on LLaDA. The same pattern holds when within-layer correlations are averaged. Related task settings retain shared structure, with average correlations of 0.49 and 0.82 within the multiple-choice group. HumanEval and MBPP have correlations of 0.72 on Dream and 0.85 on LLaDA. Together with the rate-wise analysis, these results motivate conditioning head attribution on both the target task and the reconstruction condition.

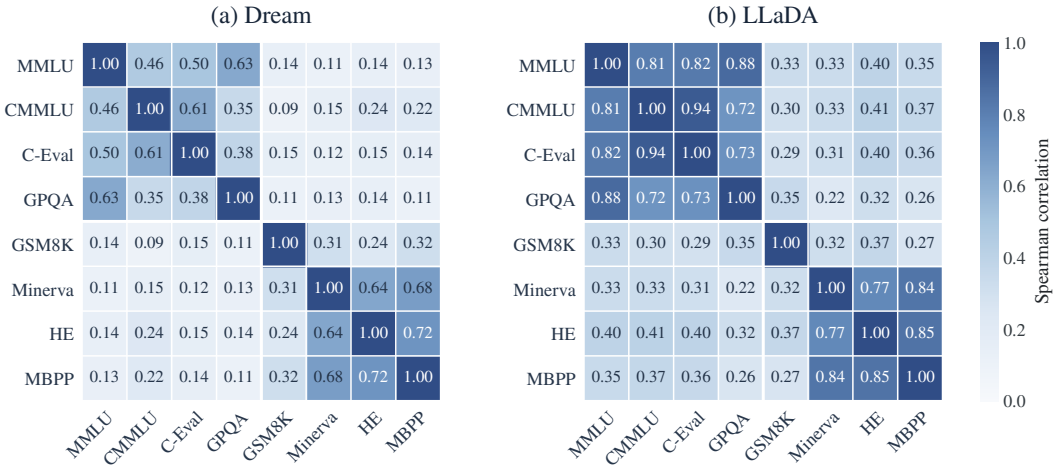

Figure 6: Cross-task Spearman correlations of ARA head rankings across eight tasks. Signed scores are averaged equally over five masking rates and compared across all 784 query heads in Dream and 1,024 in LLaDA. Higher values indicate greater agreement between task-specific rankings.

A.4 Practical Estimation and Error Analysis

For the estimation and proofs in this appendix, let $J : [0,1]^H \to \mathbb{R}$ denote a generic continuously differentiable gate objective and write $[H] = \{1, \dots, H\}$. We use $d \in [H]$ as a generic head coordinate, corresponding to the index h in the main text. We use the functional shorthand

$$\mathsf{ARA}_d(J) := \mathbb{E}_{\mathbf{u}}\left[\mathcal{A}_d(J; \boldsymbol{\gamma}_{\mathbf{u}})\right], \qquad \mathbf{u} \sim \mathcal{U}([0,1)^H), \tag{11}$$

and write $\mathsf{ARA}(J)$ for the vector of these coordinates. This is the same attribution rule as Equation (10), expressed as a functional of its objective: $\mathsf{ARA}_d(J_\rho) = A_d(\rho)$. Evaluating this rule separately at each masking rate yields the importance profiles in Section 4.2.

Algorithm 1 computes a separate attribution vector at each selected rate. For example m and rate ρ_t, let $J_{m,t}$ be the empirical version of Equation (4): we compute the mean log probability over masked tokens for each of S nonempty masks, then average these S values. The masks remain fixed across paths and integration steps for that example and rate.

For one sampled path, let $\tau_k = k/K$, $k = 0, \dots, K$, and define

$$\mathbf{g}^{(k)}(\mathbf{u}) = \boldsymbol{\gamma}_{\mathbf{u}}(\tau_k), \qquad \Delta\mathbf{g}^{(k)}(\mathbf{u}) = \mathbf{g}^{(k)}(\mathbf{u}) - \mathbf{g}^{(k-1)}(\mathbf{u}).$$

Algorithm 1: Rate-specific ARA for DLLMs

Input: Weighted attribution data $\{(\mathbf{x}^{(m)}, \omega_m)\}_{m=1}^{M}$; masking rates $\{\rho_t\}_{t=1}^{T}$; mask samples per example and rate S; integration steps K; path samples P.
Output: Rate-specific head scores $\{S_{\ell,h}(\rho_t)\}_{t=1}^{T}$.

for $t = 1$ **to** T **do**
  $\mathbf{S}_t \leftarrow \mathbf{0}$;
  for $m = 1$ **to** M **do**
    Sample S nonempty completion masks of $\mathbf{x}^{(m)}$ at rate ρ_t;
    Form $J_{m,t}$ from these masks and keep them fixed for all paths;
    $\mathbf{I} \leftarrow \mathbf{0}$;
    for $p = 1$ **to** P **do**
      Sample activation times $\mathbf{u}^{(p)} \sim \mathcal{U}([0,1)^H)$; $\mathbf{g}_{\text{prev}} \leftarrow \mathbf{0}$;
      for $k = 1$ **to** K **do**
        $\mathbf{g}_{\text{now}} \leftarrow \boldsymbol{\gamma}_{\mathbf{u}^{(p)}}(k/K)$ via (9);
        $\Delta\mathbf{g} \leftarrow \mathbf{g}_{\text{now}} - \mathbf{g}_{\text{prev}}$; $\mathbf{g}_{\text{prev}} \leftarrow \mathbf{g}_{\text{now}}$;
        $\mathbf{r} \leftarrow \nabla_{\mathbf{g}} J_{m,t}(\mathbf{g})\big|_{\mathbf{g}=\mathbf{g}_{\text{now}}}$;
        $\mathbf{I} \leftarrow \mathbf{I} + (\mathbf{r} \odot \Delta\mathbf{g})/P$;
    $\mathbf{S}_t \leftarrow \mathbf{S}_t + \omega_m \mathbf{I}$;
return *The layer–head coordinates* $S_{\ell,h}(\rho_t)$ *of each* $\mathbf{S}_t$.

The pathwise attribution vector is approximated by the Riemann sum

$$\widehat{\mathcal{A}}_K(J; \boldsymbol{\gamma}_{\mathbf{u}}) = \sum_{k=1}^{K} \nabla J(\mathbf{g}^{(k)}(\mathbf{u})) \odot \Delta\mathbf{g}^{(k)}(\mathbf{u}). \tag{12}$$

The d-th entry approximates $\mathcal{A}_d(J; \boldsymbol{\gamma}_{\mathbf{u}})$. The coordinate-wise increment $\Delta g_d^{(k)}(\mathbf{u})$ is part of the estimator, which is essential once different heads follow different activation schedules.

Although the continuous asynchronous-path derivative equals $1/(1-u_d)$ after gate d activates (and zero beforehand), the implementation need not evaluate this factor directly. The discrete estimator uses successive gate increments: for every coordinate d, the increments $\Delta g_d^{(k)}(\mathbf{u})$ are nonnegative and sum to one across the path. Late activation times can concentrate a coordinate's contribution near the end of the grid, but they do not create unbounded gate increments. This is why Algorithm 1 multiplies gate gradients by $\Delta\mathbf{g}$ rather than by an explicit analytic velocity.

We then draw P independent activation-time vectors $\mathbf{u}^{(1)}, \ldots, \mathbf{u}^{(P)} \sim \mathcal{U}([0,1)^H)$ and average the sampled path estimates:

$$\widehat{\mathsf{ARA}}_{P,K}(J) = \frac{1}{P} \sum_{p=1}^{P} \widehat{\mathcal{A}}_K(J; \boldsymbol{\gamma}_{\mathbf{u}^{(p)}}). \tag{13}$$

For comparison, the diagonal path is recovered by setting $\Delta g_d^{(k)} = 1/K$ for every coordinate and step.

Given normalized data weights $\omega_m \geq 0$ with $\sum_m \omega_m = 1$, the score for head h in layer ℓ at rate ρ_t is

$$S_{\ell,h}(\rho_t) = \sum_{m=1}^{M} \omega_m \left[\widehat{\mathsf{ARA}}_{P,K}(J_{m,t})\right]_{(\ell,h)}, \qquad t = 1, \ldots, T. \tag{14}$$

Signed attribution vectors are averaged directly, without taking magnitudes or applying rectification. By linearity, this estimates ARA for the rate-specific empirical objective $\sum_m \omega_m J_{m,t}$. Under $d \leftrightarrow (\ell, h)$, $S_{\ell,h}(\rho_t)$ estimates $A_d(\rho_t)$ up to mask-sampling, path-sampling, and numerical-integration error. Each rate retains its own descending head ranking for dynamic interventions and rate-wise analysis.

Only for the static ARA (Global) ablation and cross-task analysis, we aggregate the rate-specific scores using nonnegative weights w_t that sum to one:

$$\bar{A}_d = \sum_{t=1}^{T} w_t A_d(\rho_t). \tag{15}$$

This optional postprocessing is separate from Algorithm 1; dynamic interventions retain the individual rate-specific scores.

Algorithm 1 uses PK gate-gradient evaluations per example at each rate, or TPK across all rates. Each evaluation averages S mask variants, which can be batched but still contribute to the computational workload.

Proposition A.1 (Finite-sample approximation error)**.** *Assume that every partial derivative $\partial_d J$ is C-Lipschitz on $[0,1]^H$ with respect to $\|\cdot\|_2$, and that $|\mathcal{A}_d(J;\boldsymbol{\gamma}_{\mathbf{u}})| \le B$ for all $d \in [H]$ and $\mathbf{u} \in [0,1)^H$. Define the realized gate-space mesh*

$$\Delta_K(\mathbf{u}) = \max_{1\le k\le K} \left\|\Delta \mathbf{g}^{(k)}(\mathbf{u})\right\|_2 .$$

Then, for any $\delta \in (0,1)$, with probability at least $1-\delta$ over the P sampled paths,

$$\left\|\widehat{\mathsf{ARA}}_{P,K}(J) - \mathsf{ARA}(J)\right\|_\infty \le \frac{C}{P}\sum_{p=1}^{P} \Delta_K(\mathbf{u}^{(p)}) + B\sqrt{\frac{2\log(2H/\delta)}{P}}. \tag{16}$$

Proof. For an activation-time vector $\mathbf{u}$, let $\mathbf{a}(\mathbf{u}) = \mathcal{A}(J;\boldsymbol{\gamma}_{\mathbf{u}})$ denote the exact pathwise attribution and let $\widehat{\mathbf{a}}_K(\mathbf{u}) = \widehat{\mathcal{A}}_K(J;\boldsymbol{\gamma}_{\mathbf{u}})$ denote its Riemann approximation. We separate the error caused by discretizing each sampled path from the error caused by averaging finitely many paths.

Let $\bar{\mathbf{a}}_P = P^{-1}\sum_{p=1}^{P} \mathbf{a}(\mathbf{u}^{(p)})$ be the sample mean of the exact pathwise attributions. Since $\mathsf{ARA}(J) = \mathbb{E}_{\mathbf{u}}[\mathbf{a}(\mathbf{u})]$, the triangle inequality yields

$$\begin{aligned}\left\|\widehat{\mathsf{ARA}}_{P,K}(J) - \mathsf{ARA}(J)\right\|_\infty \le \frac{1}{P}\sum_{p=1}^{P} \left\|\widehat{\mathbf{a}}_K(\mathbf{u}^{(p)}) - \mathbf{a}(\mathbf{u}^{(p)})\right\|_\infty \\ + \left\|\bar{\mathbf{a}}_P - \mathbb{E}_{\mathbf{u}}[\mathbf{a}(\mathbf{u})]\right\|_\infty .\end{aligned} \tag{17}$$

We bound the two terms separately.

Discretization along a fixed path. Fix an activation-time vector $\mathbf{u}$ and a coordinate d. With $\tau_k = k/K$, the difference between the right-endpoint Riemann sum and the exact path integral can be written as

$$\begin{aligned}&\widehat{a}_{K,d}(\mathbf{u}) - a_d(\mathbf{u}) \\ &\quad = \sum_{k=1}^{K} \int_{\tau_{k-1}}^{\tau_k} \left[\partial_d J(\mathbf{g}^{(k)}(\mathbf{u})) - \partial_d J(\boldsymbol{\gamma}_{\mathbf{u}}(\tau))\right] \dot{\gamma}_{\mathbf{u},d}(\tau)\, d\tau.\end{aligned} \tag{18}$$

The path is coordinate-wise nondecreasing, so for $\tau \in [\tau_{k-1}, \tau_k]$, every coordinate of $\boldsymbol{\gamma}_{\mathbf{u}}(\tau)$ lies between the corresponding coordinates of $\mathbf{g}^{(k-1)}(\mathbf{u})$ and $\mathbf{g}^{(k)}(\mathbf{u})$. Hence

$$\left\|\mathbf{g}^{(k)}(\mathbf{u}) - \boldsymbol{\gamma}_{\mathbf{u}}(\tau)\right\|_2 \le \left\|\Delta \mathbf{g}^{(k)}(\mathbf{u})\right\|_2 .$$

Moreover, $\dot{\gamma}_{\mathbf{u},d}(\tau) \ge 0$ almost everywhere and $\int_{\tau_{k-1}}^{\tau_k} \dot{\gamma}_{\mathbf{u},d}(\tau)\, d\tau = \Delta g_d^{(k)}(\mathbf{u})$. Applying the C-Lipschitz assumption to Equation (18) therefore gives

$$\begin{aligned}|\widehat{a}_{K,d}(\mathbf{u}) - a_d(\mathbf{u})| &\le C\sum_{k=1}^{K} \left\|\Delta \mathbf{g}^{(k)}(\mathbf{u})\right\|_2 \Delta g_d^{(k)}(\mathbf{u}) \\ &\le C\Delta_K(\mathbf{u}) \sum_{k=1}^{K} \Delta g_d^{(k)}(\mathbf{u}) = C\Delta_K(\mathbf{u}).\end{aligned}$$

The final equality follows because each gate is restored from zero to one. Since the bound holds for every d, the first term in Equation (17) is at most $CP^{-1}\sum_p \Delta_K(\mathbf{u}^{(p)})$.

Sampling across paths. For a fixed coordinate d, the random variables $a_d(\mathbf{u}^{(1)}), \dots, a_d(\mathbf{u}^{(P)})$ are independent, have mean $\mathsf{ARA}_d(J)$, and lie in $[-B, B]$. Hoeffding's inequality gives, for every $t > 0$,

$$\Pr(|\overline{a}_{P,d} - \mathsf{ARA}_d(J)| \geq t) \leq 2\exp\left(-\frac{Pt^2}{2B^2}\right).$$

A union bound over all H coordinates then gives

$$\Pr(\|\overline{\mathbf{a}}_P - \mathsf{ARA}(J)\|_\infty \geq t) \leq 2H\exp\left(-\frac{Pt^2}{2B^2}\right).$$

Setting $t = B\sqrt{2\log(2H/\delta)/P}$ makes the right-hand side equal to δ. Substituting the resulting sampling bound and the deterministic discretization bound into Equation (17) proves Equation (16). □

The first term in Equation (16) bounds discretization error through the realized gate-space mesh and the Lipschitz constant C. The mesh is directly computable, but C need not be known. The second term is the Monte-Carlo error and decays as $O(P^{-1/2})$ up to the logarithmic dependence on the number of gates. The boundedness assumption holds with $B = G$ whenever $|\partial_d J(\mathbf{g})| \leq G$ on the gate cube, because each coordinate is restored by a total amount of one. We state the grid term through the realized mesh rather than claim a uniform $O(K^{-1})$ rate: activation times arbitrarily close to one can concentrate most of a coordinate's restoration in the final grid interval. In practice, K should therefore be increased until the realized meshes and resulting rankings stabilize, while P controls variation across sampled paths.

A.5 Axiomatic Guarantees

Proposition A.2 (Path-attribution axioms)**.** *Let $J, \widetilde{J}, J_1, J_2 : [0,1]^H \to \mathbb{R}$ be continuously differentiable. Then* ARA *satisfies:*

1. ***Completeness:*** $\sum_{d\in[H]} \mathsf{ARA}_d(J) = J(\mathbf{1}) - J(\mathbf{0})$.
2. ***Implementation invariance:*** *if $J(\mathbf{g}) = \widetilde{J}(\mathbf{g})$ for every $\mathbf{g} \in [0,1]^H$, then $\mathsf{ARA}(J) = \mathsf{ARA}(\widetilde{J})$.*
3. ***Dummy:*** *if J is independent of coordinate d, then $\mathsf{ARA}_d(J) = 0$.*
4. ***Linearity:*** $\mathsf{ARA}_d(aJ_1 + bJ_2) = a\mathsf{ARA}_d(J_1) + b\mathsf{ARA}_d(J_2)$ *for all $a, b \in \mathbb{R}$ and $d \in [H]$.*

Proof. Completeness. For any activation-time vector $\mathbf{u}$, the path has shared endpoints $\boldsymbol{\gamma}_\mathbf{u}(0) = \mathbf{0}$ and $\boldsymbol{\gamma}_\mathbf{u}(1) = \mathbf{1}$. The chain rule therefore gives

$$\begin{aligned}\sum_{d\in[H]} \mathcal{A}_d(J; \boldsymbol{\gamma}_\mathbf{u}) &= \int_0^1 \frac{d}{d\tau} J(\boldsymbol{\gamma}_\mathbf{u}(\tau))\, d\tau \\ &= J(\mathbf{1}) - J(\mathbf{0}).\end{aligned}$$

Taking expectation over $\mathbf{u}$ proves the claim.

Implementation invariance. If $J = \widetilde{J}$ on $[0,1]^H$, their gate gradients agree everywhere. Their pathwise integrands, and hence their expected attributions, are identical.

Dummy. Independence from coordinate d implies $\partial_d J(\mathbf{g}) = 0$ throughout $[0,1]^H$, so every path assigns zero attribution to that coordinate.

Linearity. Substitute $\partial_d(aJ_1 + bJ_2) = a\partial_d J_1 + b\partial_d J_2$ into the path integral in (6) and use linearity of the integral and expectation in (11). □

A.6 Proofs of the Support Propositions

Proof of Proposition 5.1. Equation (8) and the proposition's assumption give

$$\mathcal{A}_d(J_\rho; \boldsymbol{\gamma}^{\mathsf{DP}}) = \int_0^1 \partial_d J_\rho(\tau \mathbf{1})\, d\tau = 0. \qquad \square$$

Proof of Proposition 5.3. The proof has two steps. We first represent ARA as integration against a path-induced probability measure, and then show that this measure assigns positive mass to every open region in the interior of the gate cube.

Measure and representation. Intuitively, $\nu_d(E)$ records how much coordinate d is restored while the joint gate state lies in E, averaged over asynchronous paths. Formally, for each coordinate d and Borel set $E \subseteq [0,1]^H$, define

$$\nu_d(E) = \mathbb{E}_{\mathbf{u}} \left[\int_0^1 \mathbf{1}\{\boldsymbol{\gamma}_{\mathbf{u}}(\tau) \in E\} \dot{\gamma}_{\mathbf{u},d}(\tau)\, d\tau \right].$$

Nonnegativity and monotone convergence give countable additivity. Moreover,

$$\nu_d([0,1]^H) = \mathbb{E}_{\mathbf{u}} \left[\gamma_{\mathbf{u},d}(1) - \gamma_{\mathbf{u},d}(0) \right] = 1,$$

so ν_d is a probability measure. By its definition, any bounded measurable f satisfies

$$\int_{[0,1]^H} f(\mathbf{g})\, d\nu_d(\mathbf{g}) = \mathbb{E}_{\mathbf{u}} \left[\int_0^1 f(\boldsymbol{\gamma}_{\mathbf{u}}(\tau)) \dot{\gamma}_{\mathbf{u},d}(\tau)\, d\tau \right].$$

Because $\partial_d J_\rho$ is bounded on the compact gate cube, taking $f = \partial_d J_\rho$ yields

$$A_d(\rho) = \int_{[0,1]^H} \partial_d J_\rho(\mathbf{g})\, d\nu_d(\mathbf{g}) = \mathbb{E}_{\mathbf{g} \sim \nu_d}[\partial_d J_\rho(\mathbf{g})].$$

The measure construction is independent of the objective, so the same argument gives $\mathsf{ARA}_d(J) = \int_{[0,1]^H} \partial_d J(\mathbf{g})\, d\nu_d(\mathbf{g})$ for any continuously differentiable J under the shorthand in Equation (11).

Full support. Fix $\mathbf{g}^\star \in (0,1)^H$ and an open neighborhood $\mathcal{N}$ of $\mathbf{g}^\star$. We construct an asynchronous path through $\mathbf{g}^\star$ by choosing a common path time $\tau^\star$ and solving $\gamma_{\mathbf{u}^\star,i}(\tau^\star) = g_i^\star$ for every coordinate. Choose

$$\max_{i \in [H]} g_i^\star < \tau^\star < 1, \qquad u_i^\star = \frac{\tau^\star - g_i^\star}{1 - g_i^\star} \quad (i \in [H]).$$

The inequality $\tau^\star > \max_i g_i^\star$ ensures that $0 < u_i^\star < \tau^\star$, so every coordinate is on the linear, unclipped part of its restoration schedule at time $\tau^\star$. Substitution verifies that

$$\gamma_{\mathbf{u}^\star,i}(\tau^\star) = g_i^\star, \qquad \dot{\gamma}_{\mathbf{u}^\star,d}(\tau^\star) = \frac{1}{1 - u_d^\star} > 0.$$

Because the clipping is inactive at $(\tau^\star, \mathbf{u}^\star)$, continuity provides an open interval I_τ around $\tau^\star$ and an open box $B_{\mathbf{u}}$ around $\mathbf{u}^\star$ such that $\boldsymbol{\gamma}_{\mathbf{u}}(\tau) \in \mathcal{N}$ and $\tau > u_d$ throughout $I_\tau \times B_{\mathbf{u}}$. Thus every activation-time vector in $B_{\mathbf{u}}$ produces a path that remains inside $\mathcal{N}$ for a nonzero time interval while coordinate d is moving. Since the activation-time distribution has positive density on $(0,1)^H$,

$$\begin{aligned} \nu_d(\mathcal{N}) &\geq \mathbb{E}_{\mathbf{u}} \left[\mathbf{1}\{\mathbf{u} \in B_{\mathbf{u}}\} \int_{I_\tau} \dot{\gamma}_{\mathbf{u},d}(\tau)\, d\tau \right] \\ &\geq \mathbb{P}(\mathbf{u} \in B_{\mathbf{u}})\, |I_\tau| > 0. \end{aligned}$$

Thus every open neighborhood of every interior point has positive ν_d-mass. $\square$

Corollary A.3 (Off-diagonal detectability)**.** *Let $J : [0,1]^H \to \mathbb{R}$ be continuously differentiable. Suppose that, for some $d \in [H]$, nonempty open set $O \subset (0,1)^H$, and $c > 0$,*

$$\partial_d J(\tau \mathbf{1}) = 0 \quad \textit{for all } \tau \in [0,1], \qquad \partial_d J(\mathbf{g}) \geq c \quad \textit{for all } \mathbf{g} \in O.$$

Then $\mathcal{A}_d(J; \boldsymbol{\gamma}^{\mathsf{DP}}) = 0$, whereas the ARA *contribution from O is strictly positive:*

$$\int_O \partial_d J(\mathbf{g})\, d\nu_d(\mathbf{g}) \geq c\, \nu_d(O) > 0.$$

Proof. The proof of Proposition 5.1 applies to the generic objective J and gives the diagonal result, while Proposition 5.3 gives $\nu_d(O) > 0$ and hence the positive off-diagonal contribution. This is a local, population-level contribution: finitely many sampled paths need not enter O, and opposite-signed contributions outside O can still cancel it in the total attribution. □

A.7 Two-Gate Example

Consider the function from Section 5.1:

$$\Psi(g_1, g_2) = g_2 + \int_0^{g_1} (s - g_2)^2\, ds.$$

It has endpoint change $\Psi(\mathbf{1}) - \Psi(\mathbf{0}) = 4/3$ and partial derivative $\partial_1 \Psi(g_1, g_2) = (g_1 - g_2)^2$.

Synchronized restoration. Along the diagonal, $\partial_1 \Psi(\tau, \tau) = 0$, and therefore

$$\mathcal{A}_1(\Psi; \boldsymbol{\gamma}^{\mathsf{DP}}) = \int_0^1 \partial_1 \Psi(\tau, \tau)\, d\tau = 0.$$

Asynchronous restoration. For ARA, let $x = \gamma_{\mathbf{u},1}(\tau)$. For fixed u_1,

$$\tau = u_1 + (1 - u_1)x, \qquad \dot{\gamma}_{\mathbf{u},1}(\tau)\, d\tau = dx.$$

For each fixed x, the equivalent transformation $t = x + (1-x)u_1$ maps $u_1 \sim \mathcal{U}[0,1)$ to a uniform path time $t \in [x, 1)$, whose density is $1/(1-x)$. The second coordinate $\gamma_{\mathbf{u},2}(t)$ depends only on u_2, so its moments at fixed t require expectation only over u_2. Since $\partial_1 \Psi = (x - \gamma_{\mathbf{u},2}(t))^2$, expanding the square shows that we only need the first two moments of the second gate. Because $\gamma_{\mathbf{u},2}(t) = 0$ when $u_2 \geq t$, these moments are

$$\mu_r(t) := \mathbb{E}_{u_2}[\gamma_{\mathbf{u},2}(t)^r] = \int_0^t \left(\frac{t-u}{1-u}\right)^r du, \qquad r \in \{1, 2\}.$$

Direct integration gives

$$\mu_1(t) = t + (1-t)\log(1-t),$$
$$\mu_2(t) = 2t - t^2 + 2(1-t)\log(1-t).$$

Substitution into the path expectation and integration over t give

$$\begin{aligned}
\mathsf{ARA}_1(\Psi) &= \int_0^1 \int_x^1 \frac{x^2 - 2x\mu_1(t) + \mu_2(t)}{1-x}\, dt\, dx \\
&= \int_0^1 \left[(1-x) - \frac{5}{6}(1-x)^2 + (1-x)^2 \log(1-x)\right] dx \\
&= \frac{1}{9} > 0.
\end{aligned}$$

Thus synchronized restoration assigns zero attribution to g_1, whereas ARA assigns a strictly positive attribution.