PROTECT WHAT PREDICTIONS RELY ON: DECISION-AWARE PRESERVATION FOR MASSIVE MODEL EDITING

Anonymous authors
Paper under double-blind review

ABSTRACT

Editing thousands of facts in a language model requires deciding which aspects of its existing behavior to protect. Closed-form editors commonly make this decision using background activation statistics, which measure representation usage but do not directly measure predictive sensitivity. We propose **Decision-Aware Preservation** (DAP), a reusable preservation metric that augments activation statistics with prediction-gradient information. DAP weights Wikipedia probe keys by squared loss-gradient norms and adds the resulting matrix to the host editor's preservation term. This construction follows from an upper bound on loss derivatives at selected probe positions and retains the host's closed-form solution without changing target optimization. Experiments on GPT2-XL, GPT-J (6B), and Qwen2.5-7B across two benchmarks show improved locality at comparable edit success with up to $10{,}000$ edits. The evaluated compositions gain up to 9.4 percentage points in locality success and 4.3 points in overall editing score over their hosts. Matched-probe controls on Qwen/PRUNE and GPT-J/RECT at $10{,}000$ edits show that correct gradient–key pairing improves preservation beyond uniform or shuffled weights, with gains persisting after independent strength tuning. Fixed-spectrum interventions further characterize the role of protection orientation. Together, these results identify sensitivity-guided allocation as a practical way to improve preservation in closed-form model editing. Code is available at `https://anonymous.4open.science/r/dap-editing-46F4/`.

1 INTRODUCTION

Factual knowledge in pretrained LLMs can be wrong or become outdated. Full fine-tuning is costly and can disturb unrelated capabilities. Model editing instead updates selected facts directly in the weights while seeking to preserve the rest (Yao et al., 2023; Cao et al., 2021).

Locate-then-edit (LTE) methods identify a factual association, then solve for a weight update that writes a target representation (Meng et al., 2022; 2023) (Figure 1). Because edited and unrelated facts share weights, large edit batches can also alter untargeted predictions. Preserving those predictions is measured by *locality*, or *specificity* (Yao et al., 2023; Meng et al., 2022; Liu et al., 2026a).

Covariance-based LTE editors protect unrelated behavior using keys collected from Wikipedia text. Their background second-moment matrix, which we call Wiki-C, penalizes updates along directions with high activation energy. It is precomputed once and reused across edit batches.

Activation energy measures how much a representation direction is used; predictive sensitivity measures how strongly a change at that representation affects the loss. These quantities need not agree. Figure 2 illustrates different leading axes for Wiki-C and a gradient-weighted metric built from $4{,}262$ held-out zsRE probes. A single preservation coefficient rescales the activation metric without changing its relative allocation of protection. This motivates a complementary question: can prediction sensitivity guide that allocation?

Decision-Aware Preservation (DAP) answers this question with a simple addition to the closed-form solve. For each Wikipedia probe, DAP measures the next-token loss gradient at the edited layer and weights the probe key by its squared norm. A Cauchy–Schwarz bound converts local

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

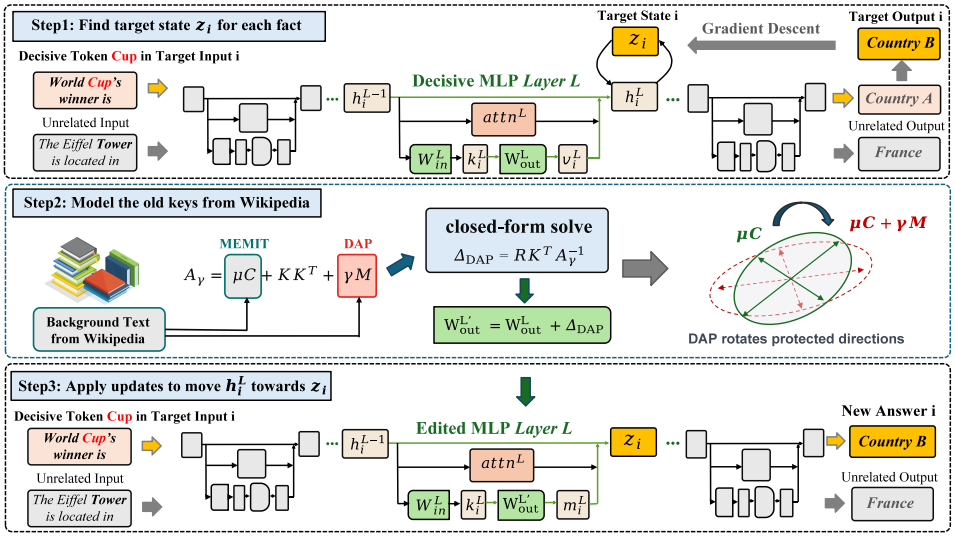

Figure 1: DAP instantiated on MEMIT. Step 1: For each edit, gradient-based optimization finds a target state z_i that elicits the desired answer when substituted for h_i^L. Step 2: Wikipedia supplies activation statistics (Wiki-C) and gradient-weighted probe statistics (Wiki-M). Their combination allocates protection using both representation usage and local prediction sensitivity; the closed-form solve produces Δ_{DAP} (Eq. (7)). Step 3: Applying Δ_{DAP} to W_{out}^L induces the target answer while better preserving unrelated behavior.

loss responses into a reusable key-space penalty, Wiki-M. The gradient determines which probe associations receive more protection; their keys determine where that protection acts.

DAP adds Wiki-M to the host editor's Wiki-C term while retaining target optimization and the closed-form editing procedure. Both matrices are precomputed from Wikipedia and reused across edit batches; setting the DAP coefficient to zero recovers the host at the same configuration. We examine the construction and its geometry separately: matched-probe controls test the assignment of sensitivity weights to keys, and fixed-spectrum interventions test the orientation of the resulting protection.

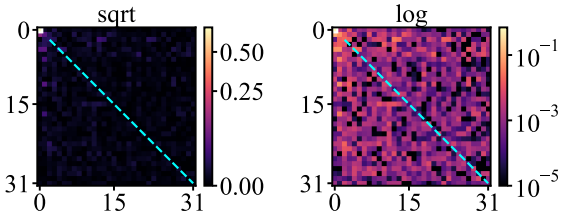

Figure 2: Activation energy and sensitivity weighting emphasize different directions. Squared-cosine alignment between the top-32 eigenvectors of Wiki-C (columns) and a gradient-weighted metric built from 4,262 held-out zsRE probes (rows) on GPT-J, shown with square-root (left) and log (right) color scales. Individual leading axes have weak alignment.

Our contributions are (1) a sensitivity-guided preservation metric derived from a local loss-response bound and compatible with closed-form editors; (2) evidence of improved locality across three models, two benchmarks, and edit batches up to 10,000; and (3) controlled analyses connecting gradient–key assignment, protection geometry, and realized loss changes.

2 Related Work

Knowledge editing and closed-form editors. Model editing changes factual behavior by updating weights or attaching memory (Yao et al., 2023; Cao et al., 2021; Mitchell et al., 2022a;b; Hartvigsen et al., 2023; Huang et al., 2023; Wang et al., 2024). ROME introduced a rank-one located update; MEMIT extends closed-form editing to batches (Meng et al., 2022; 2023). Their

shared preservation–memorization view (Gupta et al., 2024) makes the preservation metric central to which directions an update avoids.

Methods that modify the preservation metric. MEMIT builds its metric from Wikipedia activations (Meng et al., 2023). ENCORE and RoSE redistribute protection across directions (Gupta et al., 2025; Liu et al., 2026c); AlphaEdit and EMMET impose hard constraints (Fang et al., 2025; Gupta et al., 2024). DAP instead adds gradient-weighted Wikipedia statistics to the metric while retaining the closed-form solve.

Intervention points outside the solve. PMET, EAMET, and MEMIT-Merge alter targets or their distribution across layers (Li et al., 2024; Dai et al., 2026; Dong et al., 2025; Li et al., 2025); PRUNE and RECT postprocess the update (Ma et al., 2025; Gu et al., 2024) and can therefore compose with DAP. DAP shares the sensitivity-based motivation of Fisher regularization and model merging (Kirkpatrick et al., 2017; Matena & Raffel, 2022). Its distinction is a gradient-norm relaxation into a precomputed key-space metric that fits the host solve (Section B).

3 PRELIMINARIES

Notation and the LTE paradigm. For a batch of edit requests $\{(x_i, y_i^*)\}_{i=1}^n$, causal tracing locates a decisive token and MLP layer L (Meng et al., 2022). LTE treats the output projection $W_{\text{out}}^L \in \mathbb{R}^{d' \times d}$ as a key–value memory: key $k_i \in \mathbb{R}^d$ produces value $v_i = W_{\text{out}}^L k_i \in \mathbb{R}^{d'}$. It optimizes a target state z_i so replacing the hidden state h_i^L yields y_i^* (Figure 1; Section A).

Closed-form solve with a preservation term. The target state z_i implies a desired output m_i and residual $r_i = m_i - v_i$. Many LTE methods then solve (Meng et al., 2023):

$$\Delta^* = \arg\min_{\Delta} \sum_{i=1}^{n} \|\Delta k_i - r_i\|^2 + \frac{\mu}{u} \sum_{j=n+1}^{n+u} \|\Delta k_j\|^2, \tag{1}$$

The first term installs the requested residuals; the second penalizes changes to u background keys from Wikipedia, with μ setting the trade-off. Section A of the supplementary material traces one edit end to end.

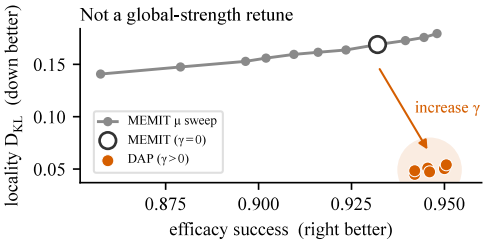

Figure 3: Tested efficacy–locality frontier on GPT-J/zsRE-2k. Locality is measured by D_{KL}, the KL divergence between the pre- and post-edit output distributions (lower is better). No global-μ setting in the sweep matches DAP on both axes.

Eq. (1) admits a closed-form solution (Meng et al., 2023):

$$\Delta^* = R K^\top (K K^\top + \mu C)^{-1}, \tag{2}$$

Here $K = [k_1, \ldots, k_n]$ and $R = [r_1, \ldots, r_n]$ stack edit keys and residuals, and $C = \frac{1}{u} \sum_{j=n+1}^{n+u} k_j k_j^\top$ is the background key second moment (Wiki-C). The edited weight is $W_{\text{out}}^{L'} = W_{\text{out}}^L + \Delta^*$.

Preservation geometry. Eigendecomposing Wiki-C as $C = \sum_{t=1}^{d} \lambda_t q_t q_t^\top$ with orthonormal q_t splits the preservation term across directions:

$$\mu \sum_{j=n+1}^{n+u} \frac{\|\Delta k_j\|^2}{u} = \mu \sum_{t=1}^{d} \lambda_t \|\Delta q_t\|^2, \tag{3}$$

Here $\lambda_t = \frac{1}{u} \sum_j (q_t^\top k_j)^2$ is background activation energy along q_t. Thus μ changes overall strength but not relative axis protection; Figure 3 shows DAP outside the tested global-μ frontier.

4 DECISION-AWARE PRESERVATION

DAP augments the host's activation-based protection with sensitivity-weighted probe associations. We derive the local response score, relax it into a reusable key-space metric, and insert that metric into the closed-form solve. Section B gives the full derivations.

4.1 PREDICTION SENSITIVITY AT PROBE POSITIONS

Wiki-C records background activation energy. To obtain a complementary sensitivity signal, we use Wikipedia prefixes and their observed next tokens as prediction probes, requiring no editing-benchmark labels.

From the same Wikipedia corpus behind Wiki-C, we draw $N = 2{,}000$ prefix–next-token pairs (x_p, y_p), at most one per article, chosen once, without looking at the model's predictions or the editing benchmarks (Section C).

For probe p, the unedited model incurs loss $\ell_p = -\log P(y_p \mid x_p)$. At the final prefix position s_p of layer L, we collect key k_p, value $v_p = W_{\text{out}}^L k_p$, and gradient $g_p = \partial \ell_p / \partial v_p$. Consider an output intervention at this position, $v_p \mapsto v_p + \varepsilon \Delta k_p$, with weights fixed. Its loss derivative at $\varepsilon = 0$ is $g_p^\top \Delta k_p$, including all downstream effects of that intervention. Averaging the squared derivatives defines

$$\mathcal{D}_{\text{loc}}(\Delta) = \frac{1}{N} \sum_{p=1}^{N} \left(g_p^\top \Delta k_p \right)^2, \tag{4}$$

a sensitivity objective at the sampled positions. A shared-weight update acts at every prefix position: its full first-order response at layer L is $\sum_t g_{p,t}^\top \Delta k_{p,t}$, evaluated on the unedited model. DAP uses the selected-position contribution to guide protection; Section B relates it to the full response.

4.2 RELAXATION INTO A REUSABLE PENALTY

Directly penalizing $\mathcal{D}_{\text{loc}}$ couples each input-side key k_p to an output-side gradient g_p. This quadratic penalty is solvable, but in general cannot be represented by a fixed key-space matrix in the host's existing solve (Section B). We seek a relaxation that preserves that reusable interface.

Writing $w_p = \|g_p\|^2$, Cauchy–Schwarz removes the dependence on output-gradient direction while retaining its magnitude:

$$\mathcal{D}_{\text{loc}}(\Delta) \;\le\; \sum_{p=1}^{N} \frac{w_p \left\| \Delta k_p \right\|^2}{N}. \tag{5}$$

The right-hand side has the same form as the host's background penalty, with each probe key weighted by its squared gradient norm. Keys associated with larger local sensitivity incur a higher cost for the same output displacement. Both keys and weights are computed on the unedited model, so the penalty is reusable across edit batches.

4.3 THE DAP UPDATE

The weighted penalty and the background penalty are not yet on a common scale. The weights w_p inherit the arbitrary scale of gradient norms. We therefore rescale all weights w_p into calibrated weights $\bar{w}_p$ by one shared constant, chosen to match the traces of their covariance matrices (Section C gives the constant). DAP then appends the calibrated penalty, with a single new hyperparameter $\gamma \ge 0$, to the objective of Eq. (1),

$$\begin{aligned} \Delta_{\text{DAP}} = \; & \arg\min_{\Delta} \sum_{i=1}^{n} \left\| \Delta k_i - r_i \right\|^2 \\ & + \mu \sum_{j=n+1}^{n+u} \frac{\left\| \Delta k_j \right\|^2}{u} + \gamma \sum_{p=1}^{N} \frac{\bar{w}_p \left\| \Delta k_p \right\|^2}{N}, \end{aligned} \tag{6}$$

Table 1: Adding DAP to three locate-then-edit hosts on zsRE at 10,000 edits across three LLMs. Each host+DAP row is followed by its host; the better value in each host pair is shown in bold. Ties at the reported precision are bolded for both methods.

Methods	**Score**	**Eff.**	**Gen.**	**Locality**			
	S↑	ES↑	GS↑	LS_{10k} ↑	top1↑	top5↑	top10↑
GPT2-XL							
MEMIT+DAP	**66.7**	**71.9**	**68.8**	**26.6**	**60.5**	**56.8**	**57.5**
MEMIT	65.1	**71.9**	68.0	25.5	57.3	54.4	55.4
RECT+DAP	**66.0**	**63.7**	**60.9**	**26.5**	**74.9**	**69.9**	**71.8**
RECT	64.9	63.3	60.3	25.6	72.2	67.0	68.1
PRUNE+DAP	**70.8**	**69.2**	**65.4**	**25.9**	**79.2**	**73.6**	**74.7**
PRUNE	70.0	68.9	65.0	25.5	77.1	71.2	72.6
GPT-J (6B)							
MEMIT+DAP	**82.7**	**91.6**	**80.4**	**27.4**	**77.4**	**68.4**	**70.4**
MEMIT	82.4	91.1	**80.4**	26.5	77.0	68.3	70.3
RECT+DAP	**74.3**	**76.9**	**67.5**	**27.2**	**79.2**	**71.7**	**72.7**
RECT	73.5	76.8	67.3	26.7	77.3	68.6	70.3
PRUNE+DAP	**82.6**	**90.4**	**80.0**	**26.4**	**78.3**	**69.5**	**71.5**
PRUNE	81.9	90.3	**80.0**	25.8	76.7	68.2	70.2
Qwen2.5-7B							
MEMIT+DAP	**63.0**	**88.8**	**81.8**	**33.2**	**41.5**	**39.6**	**44.3**
MEMIT	62.2	**88.8**	81.5	31.3	40.5	39.0	43.9
RECT+DAP	**68.9**	**85.7**	**77.7**	**42.3**	**52.7**	**49.7**	**53.4**
RECT	67.9	85.3	77.2	42.2	51.2	48.2	51.8
PRUNE+DAP	**70.8**	87.7	**82.9**	**46.2**	**52.9**	**49.3**	**52.5**
PRUNE	66.5	**89.7**	81.5	36.8	46.1	43.6	47.9

so the host objective is kept verbatim and one term is added. The solution keeps the closed form of Eq. (2) with one more matrix inside the inverse (Step 2 of Figure 1),

$$\Delta_{\mathrm{DAP}} = R\,K^\top A_\gamma^{-1}, \qquad A_\gamma = KK^\top + \mu C + \gamma M, \tag{7}$$

where $M = \frac{1}{N}\sum_{p=1}^{N} \bar{w}_p\, k_p k_p^\top$ is Wiki-M. Trace matching, $\mathrm{tr}(M) = \mathrm{tr}(C)$, calibrates its scale before the coefficients μ and γ are applied. Wiki-C supplies broad activation-based protection; Wiki-M allocates additional protection using gradient-weighted keys. At the default probe budget, $\mathrm{rank}(M) \le N < d$ (Table 6). Adding this positive-semidefinite term preserves invertibility whenever the host solve is positive definite. Setting $\gamma = 0$ recovers the host at the same configuration.

DAP changes the preservation term within each solve and retains the host's target optimization, key extraction, and subsequent processing. This permits composition with post-solve methods such as PRUNE and RECT. Building Wiki-M requires forward–backward evaluation of the probes; the matrix for each edited layer is then reused across edit batches and γ settings (Sections C and D).

4.4 Orientation Beyond Reweighting

Does adding Wiki-M merely reweight the axes q_t that Wiki-C already protects (Eq. (3)), or does it change which directions are protected? If M shares the eigenvectors of C, adding it only reweights those axes. If it does not, the protected axes themselves turn toward new directions. The commutator $[C, M] = CM - MC$ separates the two cases.

Table 2: Adding DAP to three locate-then-edit hosts on Multi-CounterFact at 10,000 edits across three LLMs. Each host+DAP row is followed by its host; the better value in each host pair is shown in bold. Ties at the reported precision are bolded for both methods.

Methods	**Score**	**Eff.**	**Gen.**	**Locality**			
	S↑	ES↑	GS↑	LS_{10k} ↑	top1↑	top5↑	top10↑
GPT2-XL							
MEMIT+DAP	**66.3**	**80.9**	**68.3**	**70.7**	**54.8**	**53.6**	**52.7**
MEMIT	65.3	80.4	67.8	69.4	53.4	52.0	51.2
RECT+DAP	**59.7**	**63.4**	**50.4**	**74.4**	**68.5**	**64.6**	**63.8**
RECT	59.1	62.4	50.0	74.3	67.9	63.8	62.9
PRUNE+DAP	**67.0**	73.9	**60.7**	**72.3**	**67.7**	**64.0**	**63.1**
PRUNE	66.7	**74.7**	60.1	72.1	66.8	63.4	62.5
GPT-J (6B)							
MEMIT+DAP	79.3	**99.0**	88.1	**74.2**	**61.1**	**57.8**	**57.3**
MEMIT	**79.4**	**99.0**	**88.2**	74.1	**61.1**	57.6	57.1
RECT+DAP	**73.2**	**89.1**	**64.6**	**79.5**	**69.9**	**65.5**	**66.3**
RECT	71.4	88.8	64.1	76.1	66.0	62.7	62.2
PRUNE+DAP	**80.9**	**98.9**	**84.3**	**76.5**	**66.2**	**63.0**	**63.6**
PRUNE	80.7	98.7	**84.3**	75.7	65.8	62.6	62.3
Qwen2.5-7B							
MEMIT+DAP	**65.2**	**99.2**	**97.3**	**58.5**	**39.0**	**40.9**	**39.0**
MEMIT	64.9	**99.2**	97.1	57.8	38.7	40.2	38.2
RECT+DAP	**75.7**	98.0	92.1	**71.2**	**53.9**	**54.6**	**53.5**
RECT	72.7	**98.7**	**93.0**	67.6	49.1	49.7	48.3
PRUNE+DAP	**71.8**	**99.4**	**96.8**	**65.4**	**46.7**	**48.2**	**46.9**
PRUNE	68.1	99.2	96.7	60.8	42.3	43.1	41.2

Proposition 1 *Let $C, M \succeq 0$ and $\mu, \gamma > 0$. If $[C, M] \neq 0$, then no reweighting of the background axes reproduces the DAP metric, that is, for all weights $\tilde{\lambda}_1, \ldots, \tilde{\lambda}_d$,*

$$\mu C + \gamma M \;\neq\; \sum\nolimits_{t=1}^{d} \tilde{\lambda}_t \, q_t q_t^{\top}. \tag{8}$$

(Proof in Section B.)

Fixed-spectrum interventions test the geometric freedom in Proposition 1 (Section 5.4). Matched-probe controls test gradient–key assignment, which can change both orientation and spectrum (Section 5.3).

5 Experiments

5.1 Setup

Baselines. Code for DAP is available at `https://anonymous.4open.science/r/dap-editing-46F4/`. DAP is built upon the standard MEMIT framework, and MEMIT serves as the direct baseline. We also compare with WISE (Wang et al., 2024), RECT (Gu et al., 2024), PRUNE (Ma et al., 2025), AlphaEdit (Fang et al., 2025), and EMMET (Gupta et al., 2024). In each composition, DAP augments the host solve with a calibrated preservation penalty.

Models, data, and metrics. We evaluate GPT2-XL, GPT-J (6B), and Qwen2.5-7B on zsRE (Levy et al., 2017) and Multi-CounterFact (MCF) (Meng et al., 2022) using teacher-forced efficacy and

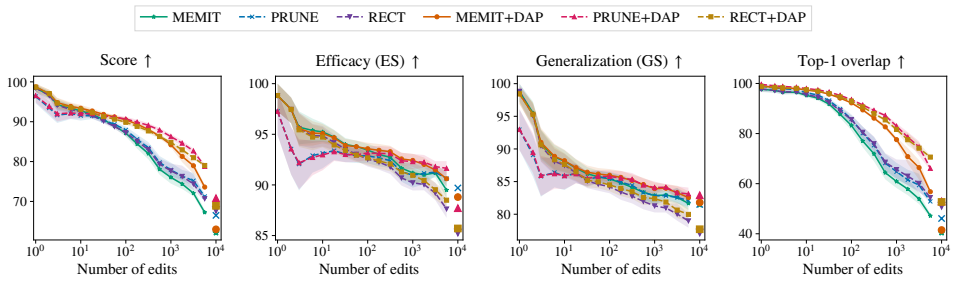

Figure 4: Qwen2.5-7B/zsRE scaling. Curves and 95% bootstrap bands: 1–5,623 edits; isolated markers: main-table operating points at 10,000. Locality is top-1 overlap (protocols in Section D).

generalization Success (ES and GS) (Meng et al., 2023). We report reference-answer locality success (LS_{10k}) over all 10,000 cases and label-free top-k overlap: the intersection of pre- and post-edit prediction sets divided by $k \in \{1, 5, 10\}$ (Liu et al., 2026b). Diagnostic sweeps also report full-vocabulary $D_{KL}(p^{\text{pre}} \| p^{\text{post}})$ to detect changes that preserve the top-k set. Following Meng et al. (2022), Score is the harmonic mean $S = 3/(\text{ES}^{-1} + \text{GS}^{-1} + \text{top1}^{-1})$. Rows within each table share the edit manifest and evaluation pipeline. Each +DAP variant retains its host's editing procedure and largely follows its hyperparameters, including RECT/PRUNE post-processing settings (Table 7), adding a preservation objective weighted by γ. We select γ under the same development-set protocol across methods and fix configurations for testing. Benchmark tables evaluate compositions; controlled interventions isolate metric properties. Operating points and the selection protocol are provided in Section D; code and selected configurations are available in our repository.

5.2 Results on Standard Benchmarks

Editing 10k facts. DAP compositions improve locality across the three models and both benchmarks (Tables 1 and 2). On Qwen2.5-7B, each composition improves LS and all three top-k overlaps on both datasets. PRUNE+DAP reaches 46.2 LS on zsRE, 9.4 points above PRUNE, while Score rises from 66.5 to 70.8, with ES decreasing by 2.0 points. On GPT-J, RECT+DAP raises Score by 0.8 points on zsRE and 3.1 on MCF, combining stronger edit success with better preservation. The effect depends on the host and dataset: MEMIT on GPT-J/MCF remains close to its original operating point. Table 8 provides additional comparisons with standalone editors on Qwen2.5-7B.

Scaling and answer retention. Figure 4 combines Qwen2.5-7B/zsRE scaling curves (1–5,623 edits) with main-table operating points at 10,000. DAP improves top-1 overlap in both evaluations and Score at the main-table points. On 500 MQuAKE-derived multi-hop questions (Zhong et al., 2023) with subjects disjoint from the edit batch, MEMIT+DAP retains more original answers at 10,000 edits (22.0% vs. 18.4%; paired McNemar $p = 9.1 \times 10^{-4}$). At 2,000 edits, retention is similar (32.2% vs. 32.8%; $p = 0.55$). This provides an answer-level complement to the next-token locality metrics.

5.3 Does Gradient–Key Assignment Improve Preservation?

We fix the host configuration, probe keys, edit requests, and evaluation prompts. Uniform weights give $M_{\text{unif}} \propto \sum_p k_p k_p^\top$, paired weights give $M_{\text{grad}} \propto \sum_p \|g_p\|^2 k_p k_p^\top$, and permutation gives $M_{\text{perm}} \propto \sum_p \|g_{\pi(p)}\|^2 k_p k_p^\top$. Each added matrix is trace-matched to Wiki-C and uses the same γ. Uniform weighting tests additional probe protection; permutation tests gradient–key correspondence. These fixed-host controls are separate from the main benchmark compositions.

Uniform and permuted protection improve locality over the host; on Qwen this trades 2.7–2.8 ES points for preservation. Correct pairing adds further locality gains (Table 3). Relative to uniform weights, DAP improves LS by 1.2/1.1 points and top-10 by 1.4/1.3 points on Qwen/PRUNE and GPT-J/RECT. ES/GS are unchanged on Qwen and decrease by 0.2 points on GPT-J. DAP also exceeds permuted weights on every locality metric, with ES/GS within 0.1 points. Sensitivity assignment therefore adds value beyond ordinary probe protection.

Table 3: Matched-probe controls at 10,000 edits. Host configurations are fixed; added operators share probe keys, trace, and γ. Bold marks the best locality per block.

Preservation	ES↑	GS↑	LS↑	top1↑	top10↑
Qwen2.5-7B / PRUNE / zsRE					
Host	89.7	81.5	36.8	46.1	47.9
Uniform weights	87.0	81.7	39.8	48.2	50.0
Permuted weights	86.9	81.6	39.5	48.0	49.8
Paired weights (DAP)	87.0	81.7	**41.0**	**49.1**	**51.4**
GPT-J / RECT / Multi-CounterFact					
Host	88.8	64.1	76.1	66.0	62.2
Uniform weights	90.2	65.7	76.9	66.9	63.7
Permuted weights	90.1	65.6	76.7	66.7	63.5
Paired weights (DAP)	90.0	65.5	**78.0**	**68.0**	**65.0**

Table 4: Independent strength tuning at 10,000 edits with equal search budgets and fixed host configurations. DAP retains higher locality at comparable ES/GS.

Preservation	ES↑	GS↑	LS↑	top1↑	top10↑
Qwen2.5-7B / PRUNE / zsRE					
Uniform, independently tuned	87.1	81.8	40.2	48.5	50.4
DAP, independently tuned	87.0	81.7	**41.0**	**49.1**	**51.4**
GPT-J / RECT / Multi-CounterFact					
Uniform, independently tuned	90.2	65.7	77.1	67.0	64.0
DAP, independently tuned	90.0	65.5	**78.0**	**68.0**	**65.0**

Does independent strength tuning close the gap? Uniform weighting and DAP then select strengths independently with equal tuning budgets and fixed host configurations. DAP retains LS gains of 0.8/0.9 points and top-10 gains of 1.0 point on both settings, with ES/GS lower by 0.1–0.2 points (Table 4). Thus the locality advantage persists after strength selection at comparable editing performance. Sections F and G provide additional GPT-J/zsRE-2k controls and local-response diagnostics.

5.4 Protection Geometry and Calibration

The matched-probe controls test the information used to construct Wiki-M. We now examine properties of the resulting operator: its orientation, its strength, and its probe budget.

Does orientation affect the editing–locality trade-off? Table 5 compares preservation operators on GPT2-XL and GPT-J (Section H). The C-axis, rotated, and random-frame operators share rank, trace, and spectrum. Relative to the C-axis control, targeted rotation improves top-10 overlap by 0.63 points on GPT2-XL and 0.11 on GPT-J, with ES changing by -0.25 and 0.00 points. The full Wiki-M achieves the highest top-k overlaps on both models. Diagonal reweighting also

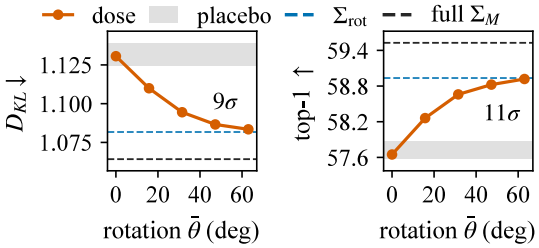

Figure 5: Rotation dose–response on GPT2-XL/zsRE at 10,000 edits, with the added operator's spectrum fixed. Both locality measures improve monotonically with the angle; edit success moves by 0.20 points. The band is a 5-seed energy-matched random-rotation placebo.

Table 5: Preservation-geometry controls on zsRE at 2,000 edits for GPT2-XL and GPT-J (6B). Σ_C: C-axis control; Σ_{diag}: reweighting null; Σ_{rand}: placebo (mean over 5 random-frame seeds); Σ_{rot}: pure rotation; Σ_M: full M. Boldface marks the best value in each column.

GPT2-XL

Operator	**Score**	**Eff.**	**Gen.**	**Locality**			
	S↑	ES↑	GS↑	LS↑	top1↑	top5↑	top10↑
MEMIT (γ=0)	74.24	77.40	73.15	25.48	72.37	67.05	68.53
Σ_C	74.55	77.60	**73.50**	25.45	72.73	67.67	69.10
Σ_{diag}	72.27	74.60	68.95	**25.49**	73.51	68.37	69.85
Σ_{rand}	74.46	**77.69**	73.17	25.41	72.72	67.47	68.95
Σ_{rot}	74.57	77.35	73.00	25.45	73.52	68.42	69.73
Σ_M	**74.59**	77.00	72.80	25.48	**74.09**	**68.76**	**70.12**

GPT-J 6B

Operator	**Score**	**Eff.**	**Gen.**	**Locality**			
	S↑	ES↑	GS↑	LS↑	top1↑	top5↑	top10↑
MEMIT (γ=0)	86.36	93.20	81.55	27.69	85.13	75.91	78.18
Σ_C	86.53	93.20	**81.70**	27.73	85.48	76.13	78.40
Σ_{diag}	84.67	88.00	80.15	28.24	86.27	77.91	80.02
Σ_{rand}	86.46	**93.22**	81.60	27.72	85.34	76.05	78.30
Σ_{rot}	86.51	93.20	81.60	27.76	85.53	76.25	78.51
Σ_M	**86.81**	93.00	81.55	**28.86**	**86.62**	**78.40**	**80.67**

improves top-k preservation, but at lower ES. Thus orientation affects the measured trade-off, while the full metric benefits from its joint allocation of protection across directions.

How sensitive is DAP to strength and probe count? On GPT2-XL, all four locality readouts improve monotonically across the tested γ values (Section I); the largest value trades 0.6 ES points and 1.2 GS points for stronger preservation. Trace matching fixes $\text{tr}(M) = \text{tr}(C)$ as the probe count N varies, so more probes redistribute the same added trace budget. Across the three Qwen2.5-7B/zsRE compositions, increasing N from 2,000 to 10,000 improves LS_{10k} by 0.9–2.0 points, with ES decreases of 0.07–0.62 points (Figure 6). The default $N = 2{,}000$ provides a smaller precomputation budget with effective preservation.

How does locality change along a rotation path? Interpolating the added operator's frame from the leading axes of C toward those of M at fixed eigenvalues improves locality monotonically on GPT2-XL, while ES moves by 0.20 points (Figure 5). This demonstrates the empirical value of the geometric freedom in Proposition 1 without increasing trace or eigenvalues. The global-μ sweep does not match DAP on both efficacy and locality (Figure 3).

6 Conclusion

DAP augments activation-based preservation with a reusable metric derived from selected-position loss sensitivity. Across three models and two benchmarks, its compositions improve locality with up to 10,000 edits. Matched-probe controls support the value of gradient–key assignment, and geometric interventions characterize the resulting protection.

Limitations. The bound covers selected-position responses. Shared-weight edits also introduce other position/layer contributions, cross terms, and finite-update nonlinearities; the loss diagnostic does not separate their effects. Actual-token loss differs from distributional preservation. Matched-probe evidence covers selected model–host settings, and orientation effects vary by model. Broader capability retention and amortized precomputation cost remain to be quantified.

AI USE STATEMENT

In this work, we have not used generative AI tools for any prohibited or undisclosed research tasks listed in the ICLR 2027 AI Policy for Authors. We used generative AI tools to assist with software development and to edit the paper for readability. The proposed method was implemented by the authors. We have reviewed all AI-assisted work and take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work studies decision-aware preservation in locate-then-edit knowledge editing. It involves no human-subject experiments or collection of personal data: experiments use publicly released models, public benchmarks including ZSRE and Multi-CounterFact, and Wikipedia text. Like other knowledge editing methods, DAP could be misused to insert false or harmful facts into a language model, so edited models should be evaluated before they are deployed.

REPRODUCIBILITY STATEMENT

The code and configurations for reproducing the main experiments are available at `https://anonymous.4open.science/r/dap-editing-46F4/`.

REFERENCES

Nicola De Cao, Wilker Aziz, and Ivan Titov. Editing factual knowledge in language models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 6491–6506. Association for Computational Linguistics, 2021. doi: 10.18653/v1/2021.emnlp-main.522.

Yanbo Dai, Zhenlan Ji, Zongjie Li, and Shuai Wang. EAMET: Robust massive model editing via embedding alignment optimization. In *The Fourteenth International Conference on Learning Representations (ICLR)*, 2026. URL `https://openreview.net/forum?id=VJGVfJJ2wW`.

Zilu Dong, Xiangqing Shen, and Rui Xia. MEMIT-Merge: Addressing MEMIT's key-value conflicts in same-subject batch editing for LLMs. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 7952–7960, 2025. doi: 10.18653/v1/2025.findings-acl.415. URL `https://aclanthology.org/2025.findings-acl.415/`.

Junfeng Fang, Houcheng Jiang, Kun Wang, Yunshan Ma, Jie Shi, Xiang Wang, Xiangnan He, and Tat-Seng Chua. Alphaedit: Null-space constrained knowledge editing for language models. In *The Thirteenth International Conference on Learning Representations (ICLR)*, 2025. URL `https://openreview.net/forum?id=HvSytvg3Jh`.

Jia-Chen Gu, Hao-Xiang Xu, Jun-Yu Ma, Pan Lu, Zhen-Hua Ling, Kai-Wei Chang, and Nanyun Peng. Model editing harms general abilities of large language models: Regularization to the rescue. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 16801–16819, 2024. doi: 10.18653/v1/2024.emnlp-main.934.

Akshat Gupta, Dev Sajnani, and Gopala Anumanchipalli. A unified framework for model editing. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 15403–15418, 2024.

Akshat Gupta, Phudish Prateepamornkul, Maochuan Lu, Ahmed Alaa, Thomas Hartvigsen, and Gopala Anumanchipalli. Lifelong knowledge editing requires better regularization, 2025. URL `https://arxiv.org/abs/2502.01636`.

Tom Hartvigsen, Swami Sankaranarayanan, Hamid Palangi, Yoon Kim, and Marzyeh Ghassemi. Aging with grace: Lifelong model editing with discrete key-value adaptors. *Advances in Neural Information Processing Systems*, 36:47934–47959, 2023.

Zeyu Huang, Yikang Shen, Xiaofeng Zhang, Jie Zhou, Wenge Rong, and Zhang Xiong. Transformer-patcher: One mistake worth one neuron. In *The Eleventh International Conference on Learning Representations (ICLR)*, 2023.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114(13):3521–3526, 2017. doi: 10.1073/pnas.1611835114.

Omer Levy, Minjoon Seo, Eunsol Choi, and Luke Zettlemoyer. Zero-shot relation extraction via reading comprehension. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL)*, pp. 333–342. Association for Computational Linguistics, 2017. doi: 10.18653/v1/K17-1034.

Xiaopeng Li, Shasha Li, Shezheng Song, Jing Yang, Jun Ma, and Jie Yu. PMET: precise model editing in a transformer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 18564–18572, 2024. doi: 10.1609/aaai.v38i17.29818.

Xiaopeng Li, Sheng Wang, Siyuan Li, Shuang Song, Bin Ji, Min Jun, and Jun Yu. Rethinking residual distribution in locate-then-edit model editing. In *Advances in Neural Information Processing Systems*, 2025.

Wei Liu, Hongkai Liu, Zhiying Deng, Yee Whye Teh, and Wee Sun Lee. From backward spreading to forward replay: Revisiting target construction in LLM parameter editing, 2026a. URL https://arxiv.org/abs/2605.00358.

Wei Liu, Haomei Xu, Hongkai Liu, Zhiying Deng, Ruixuan Li, Heng Huang, Yee Whye Teh, and Wee Sun Lee. Are we evaluating the edit locality of LLM model editing properly?, 2026b. URL https://arxiv.org/abs/2601.17343.

Xiyu Liu, Qingyi Si, Zhengxiao Liu, Chenxu Yang, Naibin Gu, and Zheng Lin. Beyond the covariance trap: Unlocking generalization in same-subject knowledge editing for large language models, 2026c. URL https://arxiv.org/abs/2603.15518.

Jun-Yu Ma, Hong Wang, Hao-Xiang Xu, Zhen-Hua Ling, and Jia-Chen Gu. Perturbation-restrained sequential model editing. In *The Thirteenth International Conference on Learning Representations (ICLR)*, 2025. URL https://openreview.net/forum?id=bfI8cp8qmk.

James Martens and Roger B. Grosse. Optimizing neural networks with kronecker-factored approximate curvature. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015*, pp. 2408–2417. JMLR.org, 2015.

Michael Matena and Colin Raffel. Merging models with fisher-weighted averaging. In *Advances in Neural Information Processing Systems*, volume 35, pp. 17703–17716, 2022.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT. In *Advances in Neural Information Processing Systems 35 (NeurIPS)*, 2022. URL http://papers.nips.cc/paper_files/paper/2022/hash/6f1d43d5a82a37e89b0665b33bf3a182-Abstract-Conference.html.

Kevin Meng, Arnab Sen Sharma, Alex J. Andonian, Yonatan Belinkov, and David Bau. Mass-editing memory in a transformer. In *The Eleventh International Conference on Learning Representations (ICLR)*, 2023. URL https://openreview.net/forum?id=MkbcAHIYgyS.

Eric Mitchell, Charles Lin, Antoine Bosselut, Chelsea Finn, and Christopher D. Manning. Fast model editing at scale. In *The Tenth International Conference on Learning Representations (ICLR)*, 2022a. URL https://openreview.net/forum?id=0DcZxeWfOPt.

Eric Mitchell, Charles Lin, Antoine Bosselut, Christopher D. Manning, and Chelsea Finn. Memory-based model editing at scale. In *International Conference on Machine Learning (ICML)*, pp. 15817–15831. PMLR, 2022b.

Peng Wang, Zexi Li, Ningyu Zhang, Ziwen Xu, Yunzhi Yao, Yong Jiang, Pengjun Xie, Fei Huang, and Huajun Chen. WISE: Rethinking the knowledge memory for lifelong model editing of large language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Yunzhi Yao, Peng Wang, Bozhong Tian, Siyuan Cheng, Zhoubo Li, Shumin Deng, Huajun Chen, and Ningyu Zhang. Editing large language models: Problems, methods, and opportunities. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 10222–10240. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.emnlp-main.632.

Zexuan Zhong, Zhengxuan Wu, Christopher D. Manning, Christopher Potts, and Danqi Chen. MQuAKE: Assessing knowledge editing in language models via multi-hop questions. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 15686–15702, 2023. doi: 10.18653/v1/2023.emnlp-main.971. URL https://aclanthology.org/2023.emnlp-main.971/.

SUPPLEMENTARY MATERIAL

This supplement provides derivations, construction details for Wiki-M, experimental settings, and additional comparisons.

A COMPUTING z_i AND THE RESIDUAL r_i

For the i-th edit request (x_i, y_i^*), all model parameters are frozen and z_i is parameterized as the original hidden state h_i^L plus an optimizable offset δ_i, which is optimized by gradient descent so that, after the replacement, the model assigns high probability to the target answer:

$$z_i = h_i^L + \arg\min_{\delta_i} \frac{1}{P} \sum_{j=1}^{P} -\log P_{f_i(\delta_i)}\big[\, y_i^* \mid p_j \oplus x_i \,\big], \tag{9}$$

where $f_i(\delta_i)$ denotes the forward pass in which h_i^L is replaced by $h_i^L + \delta_i$, and p_j are P random prefixes concatenated before x_i so that z_i remains effective across different contexts.

One edit end to end. We trace the running example of Figure 1 of the main paper, where x_i = "World Cup's winner is" and the new answer is y_i^* = "Country B". First, a forward pass on x_i records three quantities at the decisive layer L and the decisive token, the key k_i entering W_{out}^L, the stored value $v_i = W_{\text{out}}^L k_i$, and the hidden state h_i^L. The unedited model still answers "Country A". Second, the optimization above freezes all weights and finds the offset δ_i^*, giving the target hidden state $z_i = h_i^L + \delta_i^*$. Replacing h_i^L with z_i during the forward pass already makes the model answer "Country B", but no weight has changed yet, so z_i only records where the hidden state should go. Third, the edited layer writes its output into the residual stream additively, so shifting that output by δ_i^* shifts the hidden state by the same amount. The stored value must therefore move to the target value $m_i = v_i + \delta_i^*$, and the residual of Eq. (1) of the main paper is $r_i = m_i - v_i = z_i - h_i^L$, exactly the displacement found in the previous step. Finally, the closed-form solve of Eq. (2) of the main paper returns a weight update Δ with $\Delta k_i \approx r_i$. After the update the same forward pass stores m_i at k_i, the hidden state lands near z_i, and the model answers "Country B" on its own. When several consecutive layers are edited, each layer absorbs a fraction of the remaining displacement instead of all of it (Meng et al., 2023).

B CLOSED-FORM DERIVATIONS

Let $K \in \mathbb{R}^{d \times n}$ collect edit keys and $R \in \mathbb{R}^{d' \times n}$ collect the host's desired output residuals for the current layer. Let the fixed host metric be $P_0 \succeq 0$, with $M \succeq 0$, $P_\gamma = P_0 + \gamma M$, and $A_\gamma = KK^\top + P_\gamma$. For MEMIT, $P_0 = \mu C$, where $C = S^{-1} K_0 K_0^\top = S^{-1} \sum_s b_s b_s^\top$ is the normalized background second moment.

Host closed form. The objective in Eq. (1) is $J_0(\Delta) = \|\Delta K - R\|_F^2 + \operatorname{tr}(\Delta P_0 \Delta^\top)$, with gradient

$$\nabla_\Delta J_0 = 2(\Delta K - R)K^\top + 2\Delta P_0. \tag{10}$$

Thus $\Delta(KK^\top + P_0) = RK^\top$. If $KK^\top + P_0 \succ 0$—equivalently, $\ker P_0 \cap \ker K^\top = \{0\}$—the unique minimizer is

$$\Delta(P_0) = RK^\top (KK^\top + P_0)^{-1}. \tag{11}$$

For MEMIT, $(\mu/S)\|\Delta K_0\|_F^2 = \mu \operatorname{tr}(\Delta C \Delta^\top)$ and $P_0 = \mu C$, recovering Eq. (2). The implementation uses a linear solve.

Local response and shared weights. Let s_p be the final prefix position and let $k_{p,t}^L$ and $v_{p,t}^L$ denote the input and output of the edited projection at position t. All activations and loss gradients $g_{p,t}^L = \partial \ell_p / \partial v_{p,t}^L$ below are evaluated on the unedited model. If only v_{p,s_p}^L is replaced by $v_{p,s_p}^L + \varepsilon \Delta^L k_{p,s_p}^L$ during the forward pass, with model weights fixed, the resulting loss ℓ_p^{site} satisfies

$$\left. \frac{d\ell_p^{\text{site}}(\varepsilon)}{d\varepsilon} \right|_{\varepsilon=0} = (g_{p,s_p}^L)^\top \Delta^L k_{p,s_p}^L. \tag{12}$$

This gradient includes all downstream paths from the selected output to the final prediction. Equation (4) averages the squares of these derivatives, suppressing the fixed layer and position indices.

For a shared-weight edit, let $\theta(\varepsilon)$ replace each edited projection W^L by $W^L + \varepsilon\Delta^L$, for L in the edited layer set $\mathcal{L}$. The chain rule gives

$$\left.\frac{d\ell_p(\theta(\varepsilon))}{d\varepsilon}\right|_{\varepsilon=0} = \sum_{L\in\mathcal{L}} \langle \nabla_{W^L}\ell_p, \Delta^L\rangle_F = \sum_{L\in\mathcal{L}}\sum_t (g^L_{p,t})^\top \Delta^L k^L_{p,t}. \tag{13}$$

The single-layer case sets $\mathcal{L} = \{L\}$ but still sums over prefix positions. At multiple layers, each gradient includes downstream dependence, and the sum accounts for the distinct parameter perturbations.

Writing the selected contribution as a_p and all remaining first-order contributions as b_p, the full squared first-order response is $(a_p + b_p)^2$, whereas DAP bounds a_p^2. The terms $2a_pb_p + b_p^2$ need not vanish. Thus the local bound is not a bound on the full shared-weight response, even in the infinitesimal limit. Finite edits additionally introduce higher-order effects. DAP uses local sensitivity to weight background associations within the host's preservation objective; Section G reports empirical agreement with observed loss changes.

Diagnostic intervention. In Section G, A_p is measured by comparing the unedited model with the model after applying the complete multi-layer edit: $A_p = [\ell_p(\theta(1)) - \ell_p(\theta(0))]^2$. By contrast, E_p and B_p use the selected layer's update and the unedited model's key and gradient at the selected position. Their correlations with A_p assess whether local sensitivity ranks end-to-end loss changes. The comparison includes contributions from other positions and layers, their cross terms, and finite-update nonlinearities; it does not isolate any one of these effects. The ratio E_p/B_p concerns only the selected-position bound.

The closed-form-compatible relaxation. Write $\delta v_p = \Delta k_p$ for the output shift at the probed position. With column-stacking vec, $g_p^\top \Delta k_p = (k_p \otimes g_p)^\top \mathrm{vec}(\Delta)$. Omitting the common $1/N$ factor, the squared local response is

$$\begin{aligned}\sum_p (g_p^\top \delta v_p)^2 &= \sum_p \mathrm{tr}\big(\Delta\, k_p k_p^\top\, \Delta^\top\, g_p g_p^\top\big) \\ &= \mathrm{vec}(\Delta)^\top \Big[\textstyle\sum_p (k_p k_p^\top) \otimes (g_p g_p^\top)\Big] \mathrm{vec}(\Delta),\end{aligned} \tag{14}$$

an empirical-Fisher-style sum of per-probe Kronecker outer products, with the key factor first because vec stacks columns. Its activation–gradient structure is related to K-FAC (Martens & Grosse, 2015), but K-FAC specifically approximates such an expected Kronecker product by a product of expectations. In general the exact sum is *not* $\mathrm{tr}(\Delta\Sigma\Delta^\top)$ for one key-space Σ: that form corresponds to $\Sigma \otimes I_{d'}$, already impossible for a nondegenerate single probe when $d' > 1$ because rank-one $gg^\top$ cannot be a multiple of $I_{d'}$. A PMCF key-space core therefore cannot absorb the exact object. Relaxing $g_p g_p^\top \preceq \|g_p\|_2^2 I = w_p I$ gives

$$\begin{aligned}\sum_p (g_p^\top \delta v_p)^2 \;&\le\; \sum_p w_p\, \mathrm{tr}\big(\Delta\, k_p k_p^\top\, \Delta^\top\big) \\ &= \mathrm{tr}\big(\Delta\, \big(\textstyle\sum_p w_p k_p k_p^\top\big)\, \Delta^\top\big),\end{aligned} \tag{15}$$

Restoring the $1/N$ factor gives Eq. (5), with matrix form $\mathrm{tr}(\Delta M_{\mathrm{raw}}\Delta^\top)$ and $M_{\mathrm{raw}} = \frac{1}{N}\sum_p w_p k_p k_p^\top$. Retaining squared gradient norms while removing output-direction dependence yields a precomputable key-space metric.

DAP closed form. Adding $\gamma\,\mathrm{tr}(\Delta M\Delta^\top)$ to J_0 contributes $2\gamma\Delta M$, so $\Delta(KK^\top + P_0 + \gamma M) = RK^\top$. Since $M \succeq 0$, the unique solution whenever $A_\gamma \succ 0$ is

$$\Delta_{\mathrm{DAP}} = RK^\top (KK^\top + P_0 + \gamma M)^{-1}, \tag{16}$$

recovering Eq. (7); $\gamma = 0$ returns the host. When $\mathrm{tr}(P_\gamma) > 0$, the added trace share is $\rho_{\mathrm{tr}} = \gamma\mathrm{tr}(M)/[\mathrm{tr}(P_0) + \gamma\mathrm{tr}(M)]$. For $P_0 = \mu C$ and $\mathrm{tr}(M) = \mathrm{tr}(C)$,

$$P_\gamma = (\mu+\gamma)\big[(1-\rho)C + \rho M\big], \qquad \rho = \frac{\gamma}{\mu+\gamma}, \tag{17}$$

so raising γ changes both strength and geometry. More generally, if a host exposes a covariance slot as αC with $\alpha > 0$, returning $C' = C + (\gamma/\alpha)M$ gives $\alpha C' = \alpha C + \gamma M$; α is the coefficient in the final solve, including any outer multiplier. Thus an implementation with $P = \varepsilon G$ and $G = \mu C$ must use $\alpha = \varepsilon\mu$, not μ.

Remark 1 (Weighted preservation of background associations) *Let τ denote the ratio $\mathrm{tr}(C)/\mathrm{tr}(M_{\mathrm{raw}})$ (if the denominator is zero, set $M = 0$), and let $\beta_p = (\gamma\tau/N)w_p \geq 0$. Then*

$$\gamma\,\mathrm{tr}\big(\Delta M \Delta^\top\big) = \textstyle\sum_{p=1}^{N} \beta_p \left\|(W_0{+}\Delta)k_p - W_0 k_p\right\|_2^2, \tag{18}$$

Thus DAP augments the host objective with N weighted pseudo-associations $(k_p, W_0 k_p)$, and Eq. (7) is their joint minimizer.

Proof and scope. Since $\gamma M = \sum_p \beta_p k_p k_p^\top$, the left side of Eq. (18) is $\sum_p \beta_p \|\Delta k_p\|_2^2$, and $\Delta k_p = (W_0{+}\Delta)k_p - W_0 k_p$ identically. The equivalence needs no fit assumption on W_0, holds per layer, and vanishes at $\gamma = 0$. For a covariance host, $(\mu/S)\|\Delta K_0\|_F^2$ similarly rehearses background associations at uniform prices; DAP differs in its held-out sampling and weights. Other factorizations $\gamma M = FF^\top$ give equivalent pseudo-associations, but the probe factorization uses realized activations and per-probe prices. □

Solving the unrelaxed local penalty. The quadratic penalty $\mathcal{D}_{\mathrm{loc}}$ admits a matrix-inversion-lemma solve using a $d \times d$ system with $A_0 = KK^\top + P_0$ and an $N \times N$ system in $a_p = g_p^\top \Delta k_p$. Its minimizer includes the batch-dependent correction $-(\gamma/N)\sum_p a_p g_p k_p^\top A_0^{-1}$. DAP's relaxation instead preserves the host form $RK^\top(\text{key-space matrix})^{-1}$ with a fixed additional metric. The Cauchy–Schwarz bound is tight per probe when g_p and Δk_p are linearly dependent, including zero cases.

Proof of Proposition 1 of the main paper. If $[C, P_0] = 0$, then $[C, P_\gamma] = [C, P_0 + \gamma M] = \gamma[C, M]$. Real symmetric matrices are simultaneously diagonalizable exactly when they commute. Hence $[C, M] = 0$ preserves every distinct eigenspace of C (while possibly refining prices inside a repeated one), whereas $[C, M] \neq 0$ forces cross-eigenspace mixing. Under a simple spectrum this is axis rotation, unattainable by global scaling or another commuting metric. For the MEMIT specialization $P_0 = \mu C$, trace-matched $M = C$ gives $P_\gamma = (\mu + \gamma)C$. Sequential history terms may themselves fail to commute with C; the proposition identifies DAP's explicit probe-conditioned contribution rather than excluding incidental rotation elsewhere. □

C Constructing Wiki-M

Wiki-M is constructed from Wikipedia without using editing-benchmark labels or model-dependent probe selection.

Default probe pool. For the default $N = 2{,}000$ construction, probes are drawn from the same Wikipedia dump that builds Wiki-C. Articles shorter than 400 characters are discarded, leaving 4,944,120 eligible articles. A reservoir sample with a fixed seed retains 6,000 articles, which are then shuffled; the first 3,000 are kept and each is truncated to its first 1,500 characters. The resulting corpus is frozen and content-addressed, and the same Wikipedia source is used for every model.

Default pair selection. Each article contributes at most one prefix–next-token pair. Scanning token positions 20–79, we take the first token that decodes to a non-empty string, contains a letter, is not a stopword, is not a byte fragment, and re-encodes to the same single token. The text before it is the prefix x_p; the token itself is the target y_p, i.e. the continuation that actually occurs in the corpus. Articles are consumed in their frozen order until N pairs are collected.

Larger probe-count pool. For the probe-count sweep up to $N = 10{,}000$, we return to the same filtered Wikipedia dump and preprocessing rather than restricting sampling to the 3,000 articles retained for the default construction. Candidate prefix–next-token pairs are sampled randomly from eligible articles, subject to a fixed maximum number of pairs per article so that long articles or a few

topics do not dominate. Sampling continues until the target probe count is reached. The resulting probes are fixed and reused across edit scales and experimental settings.

Both selection procedures use only the corpus and tokenizer, not gradients, loss, or editing-benchmark labels. The probe sets are fixed before sensitivity is computed. Because tokenizers differ, each model derives its own pair list from the shared Wikipedia source and reuses it across benchmarks and hosts.

From pairs to the matrix. For each pair we take the key k_p at the edited layer and the gradient g_p of the next-token loss with respect to that layer's output, set $w_p = \|g_p\|_2^2$, and form $M_{\text{raw}} = \frac{1}{N}\sum_p w_p k_p k_p^\top$. The result is symmetrized, and the weights are calibrated by one shared constant, $\bar{w}_p = \tau w_p$ with $\tau = \text{tr}(C)/\text{tr}(M_{\text{raw}})$, equivalently the ratio of the background keys' mean squared norm to the probes' weighted mean squared norm. This calibration is exactly trace matching to Wiki-C, $\text{tr}(M) = \text{tr}(C)$. In the degenerate case $\text{tr}(M_{\text{raw}}) = 0$, which never occurred in any of our builds, M is set to zero and the solve reduces to the host editor.

Rank at the probe budget. Since M_{raw} sums N rank-one terms, $\text{rank}(M) \le \min(N, d)$ (Table 6). At the default budget it is a low-rank correction. Since $M \succeq 0$, adding it preserves positive definiteness of the host solve matrix.

Table 6: Algebraic upper bounds on Wiki-M's rank relative to the key dimension: $\text{rank}(M) \le \min(N, d)$. Percentages are rounded upward; a 100% bound permits but does not establish full rank.

Model	d	$\text{rank}(M)/d$ at	
		N=2,000	N=8,000
GPT2-XL	6,400	$\le 31.3\%$	$\le 100\%$
GPT-J (6B)	16,384	$\le 12.3\%$	$\le 48.9\%$
Qwen2.5-7B	18,944	$\le 10.6\%$	$\le 42.3\%$

D Setup Details

Editing configuration. All experiments are run on NVIDIA L20X GPUs. The MEMIT-host runs edit layers 13–17 on GPT2-XL, 3–8 on GPT-J, and 4–8 on Qwen2.5-7B, with $\mu = 2\times10^4$ on GPT2-XL and $\mu = 1.5\times10^4$ on GPT-J and Qwen2.5-7B. The factorial runs of Table 5 of the main paper use the same GPT2-XL and GPT-J settings. The sweep of Table 11 holds $N = 2{,}000$ probes and covers $\mu \in \{1, 2, 2.5, 3, 4, 6, 8\}\times10^4$ at $\gamma = 0$ and $\gamma \in \{10^3, 10^4, 3\times10^4\}$ at $\mu = 2\times10^4$. PRUNE (Ma et al., 2025) and RECT (Gu et al., 2024) are applied only after their underlying solve.

Operating points. Table 7 lists the host defaults and the +DAP settings for the 10,000-edit results. The +DAP operating points were selected by Score on a 2,000-edit slice disjoint from the 10,000-edit evaluation set; hosts use their default configurations.

Probe sets. Probe counts are experiment-specific and independent of edit/evaluation case counts. The zsRE-2k factorials of Table 5 use $N = 2{,}000$ held-out probes on both models; the GPT-J analysis in Figure 2 uses a gradient-weighted metric built from 4,262 held-out zsRE probes. This diagnostic metric is distinct from the Wikipedia-based Wiki-M used for DAP editing. All probe sets are disjoint from the corresponding edit and evaluation data, computed once, and cached.

Scaling-curve protocol. Figure 4 reports two complementary evaluations. The curves cover 16 log-spaced edit scales from 1 to 5,623. At scale m, we pool $K = \min(\lfloor 10{,}000/m \rfloor, 256)$ disjoint edit chunks and restore base weights between chunks. Bands are 95% bootstrap intervals using 10,000 resamples: chunk-cluster bootstrap for $K > 1$ and case-level bootstrap at $m = 5{,}623$, where $K = 1$. The last panel measures prompt-pooled top-1 overlap, distinct from reference-answer LS_{10k}. The isolated 10,000-edit markers use the operating points and values in Table 1, with Score

Table 7: Operating points for the main comparisons. Entries apply to both benchmarks unless specified. Compositions inherit the host's μ unless overridden. PRUNE uses $\alpha = 20$ and clip fraction 0.5 by default; RECT and RECT+DAP use a 40% keep fraction.

Model / layers	Host μ	MEMIT+DAP	RECT+DAP	PRUNE+DAP
GPT2-XL 13–17	2×10^4	$\gamma = 10^4$	$\gamma = 10^4$	$\gamma = 10^4$
GPT-J 3–8	1.5×10^4	$\gamma = 3\times10^4$	$\gamma = 3\times10^4$	$\gamma = 3\times10^4$; zsRE clip 0.75 (host: 0.5)
Qwen2.5-7B 4–8	1.5×10^4	$\gamma = 3\times10^4$	zsRE: $\gamma = 3\times10^4$; MCF: $\mu = 3\times10^4$, $\gamma = 9\times10^5$	$\mu = 3\times10^4$; zsRE: $\gamma = 3\times10^5$; MCF: $\gamma = 9\times10^5$

defined from ES, GS, and top1. They are not connected to the scaling-study curves because the operating points differ; no intervals are assigned to these aggregate-only markers. Coverage varies with m because the chunk cap and removal of incomplete chunks change the evaluated population.

Significance testing. Per-metric significance uses a two-stage test: an independent-standard-error z screen, then a per-case paired bootstrap. Reported table values are means over edited cases.

Probe-count sensitivity. Figure 6 sweeps N from 2,000 to 10,000 for all three compositions at their main-table operating points. Both locality readouts improve overall, with LS_{10k} gains of 0.9–2.0 points; intermediate values include dips of up to 0.4 points. Generalization stays within 0.3 points, while ES decreases by 0.07 (MEMIT), 0.44 (RECT), and 0.62 (PRUNE) points. Increasing the probe budget therefore trades additional precomputation and small efficacy changes for stronger locality. We use $N = 2{,}000$ by default.

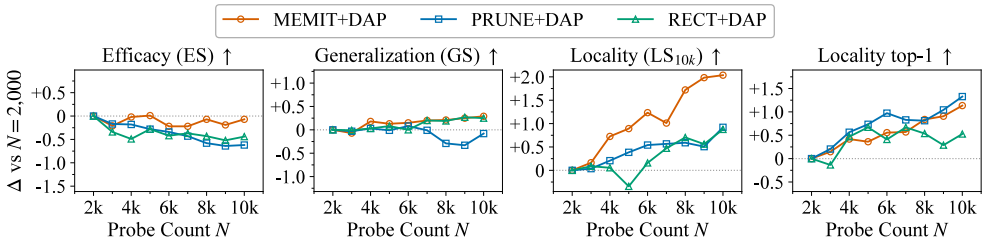

Figure 6: Effect of the Wiki-M probe count N on three +DAP compositions (Qwen2.5-7B/zsRE, 10,000 edits), as the change from each arm's own N=2,000. All panels span 2.6 points, so slopes are comparable.

E Comparison with Standalone Editors

Table 8 compares DAP compositions with standalone editors on Qwen2.5-7B at 10,000 edits.

Table 8: Comparison with standalone editors on Qwen2.5-7B at 10,000 edits. For each dataset, the DAP row is the highest-Score composition from Tables 1 and 2; paired host comparisons are reported there. Bold values mark column maxima among the methods shown.

zsRE

Methods	**Score**	**Eff.**	**Gen.**	**Locality**			
	S↑	ES↑	GS↑	LS_{10k} ↑	top1↑	top5↑	top10↑
AlphaEdit	10.7	67.2	62.9	2.98	4.0	7.2	10.9
EMMET	37.2	73.1	65.6	10.21	19.3	21.8	27.0
WISE	11.8	62.2	62.1	2.83	4.5	7.8	11.7
PRUNE+DAP	**70.8**	**87.7**	**82.9**	**46.2**	**52.9**	**49.3**	**52.5**

Multi-CounterFact

Methods	**Score**	**Eff.**	**Gen.**	**Locality**			
	S↑	ES↑	GS↑	LS_{10k} ↑	top1↑	top5↑	top10↑
AlphaEdit	51.1	85.0	53.0	61.68	35.6	35.9	33.9
EMMET	56.1	90.6	87.5	55.55	32.2	32.3	29.7
WISE	0.0	56.6	55.2	45.03	0.0	0.2	0.2
RECT+DAP	**75.7**	**98.0**	**92.1**	**71.2**	**53.9**	**54.6**	**53.5**

F Additional Matched-Probe Controls

On GPT-J/zsRE at 2,000 edits, we fix the same $N = 2{,}000$ probe keys and compare uniform, permuted, and correctly paired gradient weights. All three matrices are trace-matched to Wiki-C and enter the same MEMIT host solve with identical γ, edit requests, and evaluation prompts.

Table 9: Matched-probe controls (GPT-J/zsRE, 2,000 edits). Probe keys, trace, and γ are fixed.

Operator	ES↑	LS↑	top10↑
MEMIT ($\gamma = 0$)	93.20	27.69	78.18
Uniform weights	93.15	28.05	78.75
Permuted weights	93.10	27.90	78.55
Paired weights (DAP)	93.00	**28.86**	**80.67**

Correctly paired weights achieve the strongest preservation in Table 9: DAP exceeds uniform weighting by 0.81 LS points and 1.92 top-10 points, with a 0.15-point decrease in ES. Relative to permuted weights, the gains are 0.96 and 2.12 points, with ES differing by 0.10 points. Uniform and permuted operators also improve on MEMIT, but neither matches the correctly paired operator. With probe positions and added trace budget fixed, this comparison supports sensitivity-guided assignment on GPT-J/zsRE-2k. We next examine whether the underlying response bound tracks realized loss changes.

G Local Sensitivity and Realized Loss Changes

We distinguish bound tightness from empirical agreement with observed loss changes. The diagnostic uses Qwen2.5-7B with MEMIT as the host and its edited MLP layers (4–8), as in the main experiments. At each edit scale, we evaluate 2,000 held-out Wikipedia prefix–next-token pairs, disjoint from the 2,000 Wiki-M construction pairs and all editing requests. For a held-out position

p and the selected layer's MEMIT update Δ, let $E_p = (g_p^\top \Delta k_p)^2$ and $B_p = \|g_p\|^2 \|\Delta k_p\|^2$. Let $A_p = (\ell_p^{\text{post}} - \ell_p^{\text{pre}})^2$ denote the squared loss difference between the unedited model and the model after applying the complete multi-layer edit. The ratio E/B measures the tightness of the bound on the selected-position response. Spearman correlations with A measure empirical rank agreement, rather than an upper-bound guarantee for the evaluated edit. Both local scores have positive rank agreement with end-to-end loss changes (Table 10). Across the tested scales, $\rho(B, A)$ decreases from 0.51 to 0.30, while retaining gradient direction gives $\rho(E, A) = 0.62$ to 0.38. Median E/B decreases from 0.18 to 0.10, measuring the slack of the local bound. The diagnostic therefore supports the bound's use as a coarse sensitivity signal; the matched-probe controls test its value for allocating protection in the editor.

Table 10: Local-response bound and realized loss (Qwen2.5-7B/MEMIT; $2{,}000$ held-out positions). A uses the complete multi-layer edit.

Edits	Med. E/B	$\rho(E, A)$	$\rho(B, A)$
100	0.18	0.62	0.51
500	0.15	0.58	0.47
2,000	0.12	0.50	0.39
10,000	0.10	0.38	0.30

H Factorial Operator Construction

The five operators in Table 5 are built per model and edited layer in float64 from cached C and calibrated M, with $T = \text{tr}(C) = \text{tr}(M)$. Each replaces M in Eq. (7) at the model's reference γ ($3{\times}10^4$ on GPT-J), with all other settings fixed. The implementation preserves each operator's trace on load and checks the injected solve against an explicit $(\mu C + \gamma\Sigma + KK^\top)^{-1}K$.

Eigendecompose $C = Q\Lambda Q^\top$ and $M = U\Theta U^\top$ with eigenvalues descending. In the zsRE-2k factorial, $N = 2{,}000$, with $d = 16{,}384$ on GPT-J and $d = 6{,}400$ on GPT2-XL. The per-layer effective ranks $\#\{\theta_j > 10^{-6}\theta_1\}$ on GPT-J are 1,752–1,762. The synthetic controls use rank $r = 256$: Q_r and U_r are the top-r eigenvector frames, and $\hat{\Lambda}_r = \Lambda_r \cdot T/\text{tr}(\Lambda_r)$ rescales C's leading eigenvalues to trace T. The five operators are

$$\begin{aligned} \Sigma_C &= Q_r \hat{\Lambda}_r Q_r^\top, \qquad \Sigma_{\text{rot}} = U_r \hat{\Lambda}_r U_r^\top, \qquad \Sigma_M = M, \\ \Sigma_{\text{diag}} &= Q\,\text{diag}(Q^\top M Q)\,Q^\top, \qquad \Sigma_{\text{rand}} = Q_{\text{rnd}} \hat{\Lambda}_r Q_{\text{rnd}}^\top. \end{aligned} \tag{19}$$

The synthetic operators Σ_C, Σ_{rot}, and Σ_{rand} share rank 256, trace T, and spectrum $\hat{\Lambda}_r$. Their comparison isolates orientation at a fixed added-operator spectrum. Construction checks enforce positive semidefiniteness, orthonormal frames, and relative trace error at most 10^{-4}. The diagonal control uses the full C eigenbasis and preserves trace T; numerical negative entries are clamped, with negative mass below $10^{-6}T$. Full M retains its original spectrum and rank, bounded by N, so its comparison with the synthetic controls includes these differences.

The placebo frame Q_{rnd} is sampled within C's top-k' high-energy subspace. We increase $k' \geq r$ to target a relative mean-C-energy gap within 10%. Five fixed seeds give the reported spread (D_{KL} standard deviation ≤ 0.0025). Dose-response operators interpolate the frame from Q_r toward U_r along a Grassmann geodesic with $\hat{\Lambda}_r$ fixed. The added operator's spectrum and the total metric's trace remain constant along this path; the spectrum of $\mu C + \gamma\Sigma$ can change with orientation. The global-μ sweep in Section I separately tests scalar rescaling of the host metric.

I Global-μ Frontier and γ Response

Table 11 sweeps the host's global strength μ and DAP's γ separately on GPT2-XL and zsRE at $2{,}000$ edits, with Wiki-M built as in Section C. Raising μ slides MEMIT along a trade-off curve. The smallest μ wins Score, edit success, and generalization, the largest wins all four locality metrics, and no single setting wins both sides.

Table 11: Global-μ frontier and γ response on GPT2-XL and zsRE at 2,000 edits. The MEMIT rows vary the host's global preservation strength μ at γ=0. The DAP rows hold μ at the host default 2×10^4 and vary γ. Entries are single-run means. No entry is marked best, because the two sweeps trace a trade-off rather than a ranking.

Point	**Score**	**Efficacy**	**Generalization**	**Locality**			
	S↑	ES↑	GS↑	$D_{KL}\downarrow$	top1↑	top5↑	top10↑
MEMIT μ=1×10^4	76.2	82.2	77.2	0.617	70.1	65.3	66.7
MEMIT μ=2×10^4	74.2	77.4	73.1	0.562	72.4	67.0	68.5
MEMIT μ=2.5×10^4	73.1	75.8	70.7	0.541	72.9	67.9	69.3
MEMIT μ=3×10^4	72.3	74.3	69.1	0.521	73.7	68.4	70.0
MEMIT μ=4×10^4	70.6	71.2	66.4	0.485	74.8	69.6	71.1
MEMIT μ=6×10^4	67.9	66.0	62.6	0.425	76.7	71.7	73.1
MEMIT μ=8×10^4	65.8	62.2	59.8	0.376	78.3	73.3	74.8
DAP γ=1×10^3	74.4	77.2	73.2	0.544	72.9	67.8	69.2
DAP γ=1×10^4	74.5	76.8	72.6	0.513	74.1	68.8	70.1
DAP γ=3×10^4	74.3	76.8	71.9	0.497	74.5	69.2	70.7

At fixed $\mu = 2\times10^4$, increasing γ improves all four locality readouts. DAP at $\gamma = 3\times10^4$ exceeds MEMIT at $\mu = 3\times10^4$ on ES (76.8 vs. 74.3), GS (71.9 vs. 69.1), and every locality readout, including lower D_{KL} (0.497 vs. 0.521). None of the seven tested global-μ settings matches a DAP point on both ES and D_{KL}. This supplies empirical evidence beyond scalar tuning on GPT2-XL, complementing the GPT-J sweep in Figure 3.

Relative to the default host, $\gamma = 3\times10^4$ exchanges 0.6 ES points and 1.2 GS points for a 0.065 reduction in D_{KL} and a 2.2-point gain in top-10 overlap. Equation (17) shows that this sweep changes both total trace and normalized geometry; the fixed-spectrum interventions isolate orientation separately.