SCALE FIRST, THEN SHAPE: THE PREDICTIVE GEOMETRY OF MERGING MULTIMODAL EXPERTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Model merging combines fine-tuned experts into a single model without training data. For multimodal large language models (MLLMs), where each expert adds one capability such as OCR or grounding to a shared base, it avoids assembling instruction data for every new capability. The merging methods proposed for this setting have grown steadily more elaborate, from sparsification and spectral editing to per-layer optimization, yet each is judged only by its final benchmark score. As a result, a practitioner must evaluate every candidate merge, at over an hour per evaluation, and no existing analysis says in advance which method will work on a given model, or why. In this paper, we show that for MLLM capability merging the answer can be read from the merged weights before any evaluation. (i) We introduce a two-axis geometric account of a merged displacement: an effective scale, how far the merge travels in units of the naive task-vector sum, and a signed depth slope, whether that travel tilts toward deep or shallow layers. Keeping only each method's component along the naive sum and evaluating a plain task-arithmetic curve at its measured scale predicts the benchmark average of three published methods (OptMerge, WUDI and Iso-C) on Qwen2-VL-7B and InternVL2.5-1B within one point in five of the six cases; the depth slope, calibrated on synthetic probes at fixed scale, accounts for the sixth. Re-merging the collapsed Iso-C model from that component alone lifts it from 18 to 47 points, within a point of where the curve placed it. (ii) We show that per-layer optimization reduces to this geometry: 91% of its displacement lies along the naive sum, and a closed-form profile matches the projected optimizer within 0.13 points on both MLLMs at one twentieth of the merging cost. (iii) We find that the component the projection removes adds 0.3 to 1.4 points for the optimizers on capability experts but carries 21 to 47 points when experts have disjoint label spaces, which marks the boundary of the account. Merging multimodal experts is a prediction problem rather than a search problem, and scale is the first quantity to get right.

1 INTRODUCTION

Foundation models are expensive to retrain, while specialized fine-tunes of them appear continuously on public model hubs. Model merging turns that asymmetry into an opportunity: combine several experts that share a base checkpoint into one model that keeps their capabilities, with no training data and no inference-time overhead (Wortsman et al., 2022; Ilharco et al., 2023). For multimodal large language models the appeal is sharper still, because assembling instruction data for a new capability is itself a major cost (Wei et al., 2026).

The methods proposed for this problem have grown steadily more structural. One line sparsifies the task vectors before combining them (Yadav et al., 2023; Yu et al., 2024), a second keeps, rescales or orthogonalizes selected singular directions (Gargiulo et al., 2025; Marczak et al., 2025; Stoica et al., 2025), and a third optimizes the merged vector layer by layer against an interference objective (Cheng et al., 2025; Wei et al., 2026). Each reports gains, and each attributes them to its own mechanism: the right subspace, the right isotropy, the right interference minimum. Because methods are compared only by their end scores, those attributions are never tested, and choosing among them means an evaluation pass of over an hour per candidate.

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.

We take a different object as the unit of analysis: not the procedure, but the displacement it leaves behind, $\theta_m - \theta_0$. Two geometric quantities of that displacement carry most of what there is to know. The effective scale $\bar{a}$ is the average per-layer ratio between the merged displacement and the naive sum of task vectors; the signed depth slope $\hat{g}$ says whether the displacement is tilted toward deep or shallow layers (Figure 1a). They support a strong and falsifiable claim: measure a method's $\bar{a}$, evaluate a plain task-arithmetic reference curve at that value, and the result predicts the method's benchmark average (Figure 1b). On Qwen2-VL-7B and InternVL2.5-1B this holds within one point for five of six projected methods (Table 2), including a method that collapses to 17.98 and whose parallel component the curve places at 46.83; the sixth is deep-tilted, and its residual is the amount the depth slope predicts.

The same measurements say what per-layer optimization learns. Its displacement is 90.7% parallel in energy to the naive task-vector sum. Its per-layer profile is reproduced to 8.4% mean residual by one closed-form expression and one constant (Figure 1c), and that expression, CFS, matches the optimizer at 9 minutes per merge instead of 3.0 hours. Two published depth re-weighting schemes land where the two axes predict, architecture by architecture: on Qwen2-VL, where a deep tilt is a loss, plain task arithmetic at matched scale beats them by 9 to 12 points.

Finally, the axes say when scale is not enough: the orthogonal component carries 21 to 47 points on the classification track against 0.3 to 1.4 for the two optimizers on the vision–language tracks, so the projection doubles as a diagnostic for whether a merge needs more than scale. Our contributions are as follows.

- **A predictive account of merged performance.** A task-arithmetic curve evaluated at a method's measured effective scale lands five of six projected methods inside a one-point band on two vision–language architectures, and the signed depth slope accounts for the sixth; the slope orders shape effects at matched dose the same way in three architectures.
- **A mechanistic decomposition of optimization-based merging** into a scale profile recoverable in closed form, a depth-shape term indistinguishable from zero, and an orthogonal component worth $+0.29$ points: small for capability merging, and the whole score once experts have disjoint label spaces.
- **A closed-form rule and a matched-scale reading of re-weighting.** The rule matches the projected optimizer on both vision–language architectures at a twentieth of the per-merge cost, and the two axes read published depth re-weighting schemes off the merged weights, placing them below task arithmetic on Qwen2-VL and at the curve on InternVL2.5.

2 Related Work

Data-free merging spans linear interpolation of task vectors (Wortsman et al., 2022; Ilharco et al., 2023; Ortiz-Jimenez et al., 2023), sparsification (Yadav et al., 2023; Yu et al., 2024), spectral editing (Gargiulo et al., 2025; Marczak et al., 2025; Stoica et al., 2025), and label-free optimization of the merged vector (Cheng et al., 2025; Wei et al., 2025; 2026); test-time adaptive schemes (Yang et al., 2024) lie outside our scope. Two further families make the coefficient depend on something other than a single scalar: on depth, as a prescribed schedule (Wang et al., 2025) or an energy ratio derived from the task vectors (Section 3), and on the task, as a closed-form per-task solution (Zhou et al., 2024). Such a displacement is a scale profile by construction, which is exactly what our two axes describe, so its shape effect can be measured in isolation. Merging for multimodal models is recent (Sung et al., 2023; Du et al., 2025; Qu et al., 2025), and our tracks follow the capability-partitioned protocol family of Wei et al. (2026).

Closest in spirit is Skorobogat et al. (2025), which predicts degradation from spectral concentration. Our effective scale is a magnitude rather than a spectrum, and predictions hold for the projected merges of methods that differ by $6.7\times$ in parallel energy, so the two accounts complement each other (Appendix L places other recent scaling analyses). What separates this work is its object of study: the mapping from displacement geometry to score, rather than any single method's performance.

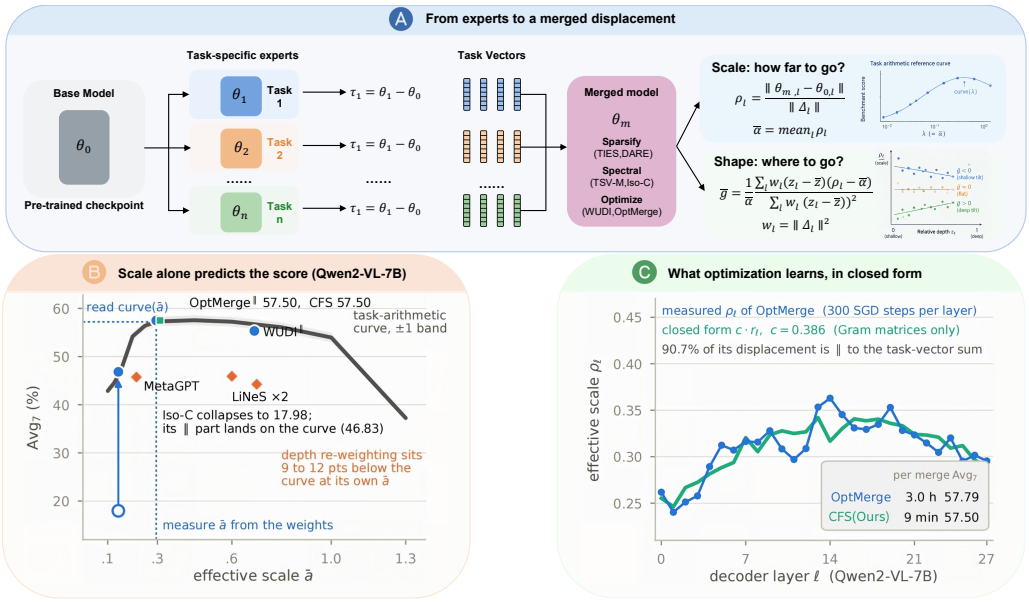

Figure 1: **Merged performance is a prediction problem, and scale is the first quantity to get right.** **(a)** Every merging method leaves a displacement $D = \theta_m - \theta_0$. Per layer we split it into a component along the naive task-vector sum and an orthogonal remainder, and summarize the along-sum profile by two numbers: the effective scale $\bar{a}$ and the signed depth slope $\hat{g}$. **(b)** On Qwen2-VL-7B, task arithmetic evaluated at a method's *measured* $\bar{a}$ predicts the benchmark average of its projected merge (five of six cases within one point on two architectures, Table 2); Iso-C collapses to 17.98, and its parallel component lands on the curve at 46.83. Depth re-weighting baselines fall below the curve at their own $\bar{a}$. **(c)** The per-layer scale that 300 steps of optimization learn is reproduced by a closed form $c \cdot r_\ell$ computed from Gram matrices alone; the resulting rule CFS matches the projected optimizer at 9 minutes per merge instead of 3.0 hours. All three architectures are placed in Figure A1.

3 Two Axes of a Merged Displacement

Setting. Let θ_0 be a base checkpoint and $\theta_1, \ldots, \theta_n$ experts fine-tuned from it on distinct tasks, with task vectors $\tau_i = \theta_i - \theta_0$ and layerwise naive sum $\Delta_\ell = \sum_i \tau_{i,\ell}$ (Ilharco et al., 2023). Every method we study outputs a merged θ_m, and we analyse it through its displacement $\theta_m - \theta_0$ rather than the procedure that produced it: sparsification, spectral truncation and per-layer optimization all leave a displacement behind, and the displacement is what the forward pass sees.

The scale axis. Methods differ in their end scores, but most of that difference turns out to be a question of travel distance. For each parameter tensor ℓ that the merge touches we measure the effective scale

$$\rho_\ell \;=\; \frac{\|\theta_{m,\ell} - \theta_{0,\ell}\|}{\|\Delta_\ell\|}, \qquad \bar{a} \;=\; \mathrm{mean}_\ell\, \rho_\ell, \tag{1}$$

where ℓ indexes individual parameter tensors, one ρ_ℓ per tensor (196, 171 and 152 tensors on Qwen2-VL, InternVL2.5 and ViT-B/32; Appendix A lists them), so $\bar{a}$ answers how far the merge travels, expressed in units of the naive sum. The mean is unweighted because a per-layer rule sets one coefficient per tensor, and $\bar{a}$ averages exactly those (the energy-weighted alternative is tabulated in Appendix C). Task arithmetic with coefficient λ has $\bar{a} = \lambda$ by construction, which gives us a reference family indexed by a single scalar. We sweep λ on each architecture, evaluate the resulting merges on the same benchmarks under the same protocol, and call the resulting function $\mathrm{curve}(\lambda)$. The curve is measured once per expert set from task arithmetic alone and then reused for every method placed on it; nothing about those methods enters it.

The shape axis. Two merges can share $\bar{a}$ and still differ, because $\bar{a}$ is a mean over layers and discards the profile: one may tilt toward deep layers and the other toward shallow ones. We summarize the

profile by a signed depth slope $\hat{g}$: the energy-weighted regression coefficient of ρ_ℓ on relative depth, normalized to unit scale,

$$\hat{g} \;=\; \frac{1}{\bar{\rho}} \frac{\sum_\ell w_\ell (z_\ell - \bar{z})(\rho_\ell - \bar{\rho})}{\sum_\ell w_\ell (z_\ell - \bar{z})^2}, \qquad w_\ell = \|\Delta_\ell\|^2, \tag{2}$$

where $z_\ell = b_\ell/(L-1)$ is the relative depth of the transformer block $b_\ell \in \{0, \ldots, L-1\}$ that contains tensor ℓ, so tensors in one block share a depth, and $\bar{z}$, $\bar{\rho}$ are the w-weighted means. All means in $\hat{g}$ are energy-weighted, so it is a dimensionless slope; tensors outside the blocks enter $\bar{a}$ but not the regression. So $\hat{g} > 0$ means the displacement is tilted toward deep layers and $\hat{g} < 0$ toward shallow ones. To measure shape without confounding it with scale we construct probes: displacements with a prescribed $\hat{g}$ and dose whose $\bar{a}$ is pinned to that of a constant-λ anchor, so a probe and its anchor differ in profile and agree in scale. Probes are pinned at each architecture's own anchor ($\bar{a} = 0.25$, 0.45 and 0.15 for ViT-B/32, Qwen2-VL and InternVL2.5) with $\hat{g}$ varied in both signs at three doses (Appendix D).

Projection surgeries. Both axes are properties of the component of a displacement that lies along the task-vector sum. To test whether that component carries a method's score, we project each method's displacement, layer by layer, onto the span of Δ_ℓ: $D_\ell^{\parallel} = \frac{\langle D_\ell, \Delta_\ell \rangle}{\|\Delta_\ell\|^2} \Delta_\ell$. This retains the layer-varying scalar coefficients $\lambda_\ell^{\parallel} = \langle D_\ell, \Delta_\ell \rangle / \|\Delta_\ell\|^2$ along Δ_ℓ while discarding all orthogonal components. We write the result with a superscript $\parallel$, so OptMerge$^{\parallel}$ is the parallel component of the OptMerge merge. A projected merge has a measured $\bar{a}$ and can be placed on the reference curve, and the difference between a method and its projection measures the value of the orthogonal component.

A closed-form per-layer rule. If scale is what matters, and if the profile the optimizer learns is predictable, then we should be able to skip optimization entirely. We define a per-tensor shrinkage profile from the task-vector Gram structure. Writing $I_\ell = \sum_{i \neq j} \langle \tau_{i,\ell}, \tau_{j,\ell} \rangle$ for the cross-expert inner products, we set

$$r_\ell \;=\; \frac{\|\Delta_\ell\|^2}{\|\Delta_\ell\|^2 + I_\ell}, \tag{3}$$

computable from task vectors with no evaluation data or gradient steps. Since $\|\Delta_\ell\|^2 = \sum_i \|\tau_{i,\ell}\|^2 + I_\ell$, the ratio is 1 for orthogonal experts and decreases toward $1/2$ as the cross terms grow; on our expert sets it lies between 0.53 and 1.02 (Table 3). Section 5.1 shows that r_ℓ reproduces the per-layer profile that optimization-based merging learns. Used as a merging rule, it sets each layer's coefficient to $\lambda_\ell = c \cdot r_\ell$, with c a single global scalar; we call this rule CFS (closed-form scaling). Here c plays the same role as λ in task arithmetic: a scalar the user sets, which fixes the mean scale $\bar{a}$ at which the curve is read. In the runs reported here we set c so that the rule lands at the optimizer's own measured $\bar{a}$, a choice of comparison that leaves the orthogonal component as the only remaining difference.

Parameter-free variants. Two rules fix the mean scale from the task vectors alone. The **median-norm rule** sets $\lambda_\ell = \mathrm{median}_i \|\tau_{i,\ell}\| / \|\Delta_\ell\|$, a typical expert's norm in each tensor; its mean $\bar{r}_2 = \mathrm{mean}_\ell \, \lambda_\ell$ is 0.3726 on Qwen2-VL-7B and 0.3673 on InternVL2.5-1B. **CFS-0** rescales the CFS profile to that mean, $\lambda_\ell = \bar{r}_2 \, r_\ell / \bar{r}$ with $\bar{r} = \mathrm{mean}_\ell \, r_\ell$, so the two rules share $\bar{a}$ by construction and differ only in profile; the constant-λ control of Table 3 sets $\lambda_\ell = \bar{r}_2$ everywhere.

CFS is the closed-form object that Section 5.1 identifies as the content of per-layer optimization. Its relation to other closed-form schemes deserves a precise statement. MetaGPT (Zhou et al., 2024) solves for one coefficient *per task* by minimizing an averaged loss difference under a linearization of the model and the near-orthogonality of task vectors; LiNeS (Wang et al., 2025) prescribes a per-layer schedule fitted to prevent forgetting; CFS derives one coefficient *per layer* from the energy ratio of Equation 3, with no orthogonality assumption. Under exact task-vector orthogonality, $\langle \tau_i, \tau_j \rangle = 0$ for $i \neq j$, the interference term in Equation 3 vanishes and r_ℓ becomes constant across layers, so CFS collapses to task arithmetic at a constant scale. This testable implication distinguishes CFS from MetaGPT: CFS adapts the per-layer profile to measured task-vector geometry and reduces to the simpler form when that geometry becomes orthogonal, whereas MetaGPT's per-task coefficients remain active even under orthogonality.

4 EXPERIMENTAL SETUP

Models, tasks and protocol. We study three merging tracks. The two multimodal tracks merge the five capability experts (VQA, Geometry, Chart, OCR and Grounding) released with OptMerge (Wei et al., 2026): full fine-tunes of InternVL2.5-1B (Chen et al., 2024), and rank-8 LoRA fine-tunes of Qwen2-VL-7B (Wang et al., 2024b) distributed as merged full-weight checkpoints; on every track a task vector is the difference of stored weights. Both tracks are evaluated on seven benchmarks: VizWiz (Gurari et al., 2018), GQA (Hudson & Manning, 2019), MathVista (Lu et al., 2024), MATH-Vision (Wang et al., 2024a), ChartQA (Masry et al., 2022), TextVQA (Singh et al., 2019) and OCR-VQA (Mishra et al., 2019), whose mean is Avg_7. The third track, the eight-task CLIP ViT-B/32 classification suite (Radford et al., 2021; Ilharco et al., 2023) reported as Avg_8, merges experts with mutually exclusive label spaces; it tests whether the account depends on experts sharing the capabilities of one model. Multimodal evaluation runs through VLMEvalKit (Duan et al., 2024) under one fixed configuration, with math benchmarks scored by the official extraction routine. All scores are our own runs under one protocol (Appendix A).

Methods compared. On both vision–language architectures we run weight averaging (Wortsman et al., 2022), task arithmetic (Ilharco et al., 2023), TIES (Yadav et al., 2023), DARE on both (Yu et al., 2024), TSV-M (Gargiulo et al., 2025), Iso-C (Marczak et al., 2025), WUDI (Cheng et al., 2025) and OptMerge (Wei et al., 2026), all under one protocol (Tables 1 and A7). The projection surgeries of Section 3 are applied to OptMerge, Iso-C and WUDI on both vision–language architectures and to Iso-C and TSV-M on ViT-B/32. The depth-dependent layer scaling scheme is represented by LiNeS (Wang et al., 2025) at two schedules; the per-task closed-form weighting is represented by MetaGPT (Zhou et al., 2024). Our own rows are CFS, CFS-0 and the constant-λ control; noise-floor control merges are described in Appendix E.

Decision rule and statistics. All differences are paired per-item bootstrap intervals, with one set of resampled indices applied to every model (Appendix A). The decision band is ± 1.0 point, wider than twice the paired standard error of the benchmark average on every architecture, and verdicts read IN-BAND, ABOVE or BELOW; a method whose $\bar{a}$ falls outside the curve's calibrated range receives no prediction. Equivalence claims use two one-sided tests at a 0.25-point margin, below one paired standard error; intervals too wide to certify either equivalence or a difference are reported as inconclusive. Throughout, $\bar{a}$ is the plain layer mean of Equation 1 (Appendix C).

5 RESULTS

5.1 PER-LAYER OPTIMIZATION LEARNS A SCALE PROFILE

Optimization-based merging solves a per-layer objective over the merged vector, and its output is usually described in terms of that objective. We instead measure the object it returns. Across all 196 weight matrices in the 28 decoder layers of the Qwen2-VL merge, the displacement produced by 300 steps of per-layer optimization lies 90.7% parallel in energy to the naive task-vector sum, and a second optimizer behaves the same way at 87.4% (Figure 2a). Spectral merging does not: isotropic rescaling retains only 13.5%. On InternVL2.5-1B, with full fine-tuned experts, the fraction is 67.8% for OptMerge and 71.7% for WUDI (Table A9).

If the direction is nearly fixed, the content of the optimization must be in the magnitudes. Recovering ρ_ℓ from Equation 1 gives a mean of 0.310 over a range of $[0.218, 0.444]$ for the full displacement, and 0.295 for its parallel component alone: a method whose nominal coefficient is 1.0 in fact travels less than a third of the naive sum, landing close to the hand-tuned optimum of the reference family. The recovered profile is structured: an inverted U in depth with a peak near layer 14 (Figure 1c) and a consistent ordering by projection type (Figure 2b).

This structure is predictable in closed form. Rank-correlating r_ℓ of Equation 3 against the measured profile gives Spearman 0.702, and a single global constant $c = 0.386$ absorbing the scale offset yields $\rho_\ell \approx c \cdot r_\ell$ with a mean absolute residual of 8.4% of $\bar{\rho}$ (Figure 2c). One expression and one number reproduce what 300 gradient steps per layer discovered: c carries the scale and r_ℓ the shape. The decomposition makes a checkable prediction: if a merge's advantage lies in scale and shape, its projection onto the task-vector sum should preserve most of its score.

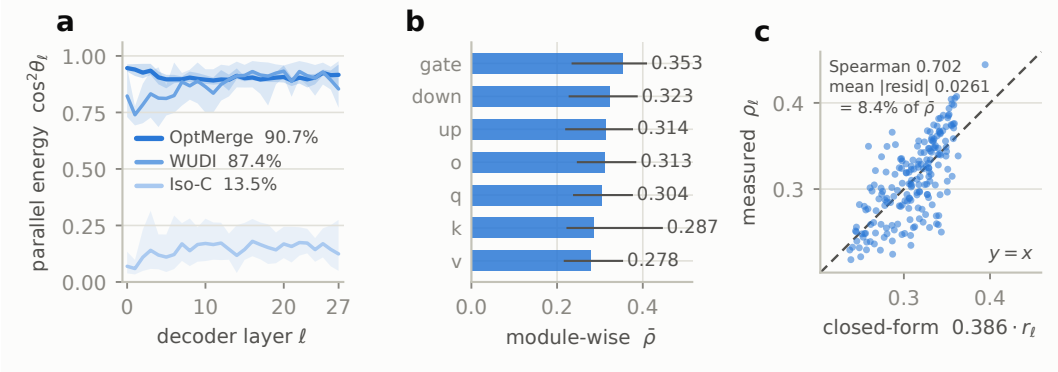

Figure 2: **What 300 steps of optimization actually produce: a per-layer scalar.** Same 196 weight matrices over 28 decoder layers of the Qwen2-VL merge. **(a)** Displacement energy parallel to the naive task-vector sum: 90.7% here, 87.4% for a second optimizer, 13.5% for isotropic rescaling. **(b)** $\bar{\rho}$ by projection type (bar = mean, line = min–max). **(c)** Closed-form prediction against measurement for all 196 matrices: one constant reproduces the profile to 8.4% mean absolute residual at Spearman ρ=0.702.

5.2 The Scale Axis Predicts Merged Performance

Point predictions. We apply the projection surgery to three merging methods on each vision–language architecture, measure each resulting displacement's $\bar{a}$, and compare its benchmark average against $\mathrm{curve}(\bar{a})$. Each projected merge is a per-layer scaling with its own profile, so an in-band verdict says that its mean scale accounts for its score and its profile does not. Five of six land inside the ± 1.0 band (Table 2, Figure A1): residuals of $+0.29$, $+0.77$ and -1.39 on Qwen2-VL and $+0.28$, -0.15 and $+0.13$ on InternVL2.5. The strongest case is isotropic rescaling, which collapses to 17.98 on Qwen2-VL. Its parallel component has $\bar{a} = 0.1427$, on the steep flank of the curve, where the curve reads 46.06; the projected merge scores 46.83, a recovery of $+28.85$ points ($[+27.90, +29.80]$) whose landing point is set by one scalar. The sixth row, WUDI$^{\|}$ on Qwen2-VL at -1.39, is where the second axis enters. Its profile is deep-tilted, $\hat{g} = +0.45$ (Table 2), and the Qwen2-VL probes at $\hat{g} = 0.38$ and 1.15 cost 0.91 and 3.48 points at fixed scale (Section 5.3), bracketing the residual in sign and size. Its orthogonal component adds $+1.41$, so the unprojected method sits on the curve (56.73 against 56.72); the two axes account for all six. On InternVL2.5 the rule variants at $\bar{a} = 0.35$–0.37 land within 0.3 of the curve on its descending flank (Table 2).

A single scalar reaches the optimizer. If scale is the dominant axis, then sweeping the scalar should be competitive with optimizing. It is (Figure A3, Appendix F). On Qwen2-VL the task-arithmetic sweep reaches 57.55 at $\lambda = 0.45$, against 57.79 for 300-step optimization, whose parallel component sits at $\bar{a} = 0.295$; the $+0.24$ difference has a paired interval of $[-0.44, +0.91]$. Under a half-split protocol (λ selected on one half of the items and scored on the other, 200 splits) the optimizer's margin over the best single scalar is $+0.47$ points: under half a point over one well-chosen scalar. The benchmarks disagree about that scalar: reasoning and OCR prefer small scale, perception large, and an oracle that picks one coefficient per benchmark reaches 58.75, above the optimizer (0.49 points under the half-split protocol); that ceiling needs the task identity at test time, which no static merge has (Figure A3b, Appendix F). The top three nodes of the curve are statistically indistinguishable, so its maximum is a plateau over $\lambda \in [0.30, 0.60]$ (Appendix F).

Depth re-weighting is read off the two axes. LiNeS is a per-layer scaling, so its displacement is a scale profile with a deep tilt ($\hat{g} > 0$ on every architecture): the curve at its $\bar{a}$ gives the scale term and Section 5.3 gives the sign of the shape term. The two agree architecture by architecture. On Qwen2-VL, where the deep-tilted branch is a loss, LiNeS lands 11.33 and 12.40 points below the curve at its own $\bar{a}$; on InternVL2.5-1B, where that branch is near zero, its two schedules land $+1.13$ and $+0.96$ from the curve, at the edge of the band; on ViT-B/32, where deep tilt is a gain, they sit 11 points above it (Section 5.5). MetaGPT, whose profile is flat ($\hat{g} \approx 0$), lands 9.12 below the curve on Qwen2-VL and 1.55 below on InternVL2.5-1B. Constant-scale task arithmetic at the same

Table 1: **Qwen2-VL-7B, seven vision–language benchmarks.** Projecting each spectral method onto the task-vector sum reproduces its score, and the closed-form rule CFS matches the projected optimizer.

Method	VQA		Geometry		Chart	OCR		Avg_7
	VizWiz	GQA	MVista	MVis	CQA	TVQA	OCR	
Weight Avg.	40.72	57.35	56.40	24.67	59.64	80.52	59.70	54.14
Task Arithmetic	**41.63**	60.61	57.60	<u>25.00</u>	73.36	81.54	61.33	57.30
TIES	37.55	54.98	51.90	20.72	27.48	70.57	54.13	45.33
DARE-TA	<u>41.59</u>	60.68	57.70	24.67	73.96	<u>81.57</u>	61.43	57.37
DARE-TIES	38.51	55.91	54.10	21.05	41.20	76.24	55.79	48.97
TSV-M	37.99	58.42	54.20	23.36	47.80	76.69	57.26	50.82
Iso-C	11.81	13.69	47.20	18.42	2.56	25.19	6.97	17.98
WUDI	39.24	**62.95**	55.80	20.07	<u>79.16</u>	79.94	59.96	56.73
OptMerge	41.51	61.40	**58.80**	**25.33**	74.40	**81.62**	<u>61.49</u>	**57.79**
TA @ $\lambda^\star{=}0.45$	40.56	62.12	55.70	22.04	**80.08**	81.15	61.20	<u>57.55</u>
OptMerge$^\parallel$	41.21	61.19	<u>58.20</u>	24.34	74.96	81.39	61.20	57.50
Iso-C$^\parallel$	37.94	56.12	52.10	19.41	34.32	74.21	53.71	46.83
WUDI$^\parallel$	38.78	<u>62.63</u>	52.60	17.11	78.72	78.83	58.63	55.33
CFS (ours, closed form)	41.44	61.27	57.90	24.34	74.64	81.32	**61.62**	57.50
Expert: VQA	41.18	62.47	51.10	18.09	67.08	80.24	57.62	53.97
Expert: Geometry	34.80	44.27	49.90	20.39	14.56	59.75	45.18	38.41
Expert: Chart	37.15	25.70	52.90	16.78	61.08	72.78	64.10	47.21
Expert: OCR	28.62	37.65	33.10	8.22	56.92	70.28	64.52	42.76
Expert: Grounding	37.57	32.61	46.60	14.80	17.52	59.70	44.04	36.12
Instruct (mixture upper bd.)	43.84	62.20	56.30	18.75	70.20	78.23	68.29	56.83

Column abbreviations: MVista = MathVista, MVis = MathVision, CQA = ChartQA, TVQA = TextVQA, OCR = OCRVQA. Bold = best, <u>underline</u> = second best, compared within the merging blocks (rows above the expert block); expert and Instruct rows are references. $^\parallel$ denotes the displacement projected parallel to the task-vector sum. All rows are our own runs under one protocol (Section 4); the first block uses $\lambda{=}0.30$ for the eight baselines and $\lambda{=}0.10$ for OptMerge, the coefficients of the protocol we reproduce. Iso-C collapses to 17.98 on this track. The Grounding expert's own skill, referring expression comprehension, is not among the seven benchmarks.

$\bar{a}$ therefore matches or beats both on the vision–language tracks, by 9 to 12 points on Qwen2-VL. Neither scheme was designed for this setting, LiNeS for single-task forgetting and MetaGPT for monolithic language models, and the two axes place both without any adjustment.

5.3 Scale First: Shape Adds at Most One Point

Holding $\bar{a}$ fixed at each architecture's anchor and varying only the depth profile isolates the second axis (Figures A1d–f and A2, Appendix D). The ordering is the same everywhere: tilting toward deep layers beats tilting toward shallow ones at the middle dose in all three architectures, with contrasts of $+10.01$, $+2.18$ and $+1.76$ and intervals bounded away from zero in each case. The shallow-tilted branch degrades monotonically with dose in all three, so the axis behaves as a dose-response relationship.

What differs across architectures is the level of the deep-tilted branch, which forms a graded series: a gain of about three points on classification, near zero on InternVL2.5, and a loss of about three and a half points on Qwen2-VL. This is the result behind the title. On both vision–language architectures, none of the eighteen probes we built exceeds its constant-scale anchor by more than a point; the closest is a block-permuted probe on InternVL2.5 at $+0.83$ with an interval of $[+0.28, +1.38]$. Once scale is set, shape has at most a point to offer a multimodal merge, and the sign of that point depends on the architecture, while the wrong tilt costs 5.65 and 12.77 points at the two larger doses on Qwen2-VL (Table A3). On the classification track several probes clear the band (Section 5.5), and a regression over its twelve probes attributes the effect to the signed slope ($R^2 = 0.76$, against 0.16 for an unsigned dispersion measure). Layer-permutation and alternation probes (Appendix D) are consistent with reading $\hat{g}$ as a summary of coherent depth tilt.

5.4 A Closed-Form Rule Matches the Projected Optimizer

The rule CFS of Section 3 applies the profile recovered in Section 5.1. On Qwen2-VL it reaches $\text{Avg}_7 = 57.50$, the score of the projected optimizer to within 0.006 points, and two one-sided tests certify equivalence at a 0.25-point margin (Table 3). Against the full optimizer it lands 0.29

Table 2: **Point predictions of the one-dimensional account.** On the two vision–language architectures 5 of 6 projected methods land inside the band; on Qwen2-VL all three re-weighting baselines land below it, while on InternVL2.5-1B MetaGPT lands below it, LiNeS β=0.80 lands above it and LiNeS β=1.00 inside it. The WUDI$^{\|}$ outlier and the ViT-B/32 rows are discussed in the text.

Arch.	Method	$\bar{a}$	curve($\bar{a}$)	Measured	Δ	Band	Verdict
ViT-B/32	CFS (c recalibrated)	0.2500	69.98	68.84	-1.14	1.0	BELOW
	Iso-C$^{\|}$	0.3835	62.17	36.45	-25.71	1.0	BELOW
	TSV-M$^{\|}$	0.2283	69.84	62.64	-7.21	1.0	BELOW
	LiNeS β=0.875	0.5395	46.49	57.52	+11.03	1.0	ABOVE
	LiNeS β=1.00	0.5987	41.39	52.74	+11.35	1.0	ABOVE
Qwen2-VL	OptMerge$^{\|}$	0.2952	57.21	57.50	+0.29	1.0	IN-BAND
	Iso-C$^{\|}$	0.1427	46.06	46.83	+0.77	1.0	IN-BAND
	WUDI$^{\|}$	0.6905	56.72	55.33	-1.39	1.0	BELOW
	Constant λ (control)	0.3726	57.42	57.36	-0.06	1.0	IN-BAND
	CFS	0.3105	57.31	57.50	+0.19	1.0	IN-BAND
	Median-norm rule	0.3726	57.42	57.22	-0.20	1.0	IN-BAND
	CFS-0	0.3726	57.42	57.37	-0.05	1.0	IN-BAND
	LiNeS β=0.80	0.6000	57.24	45.91	-11.33	1.0	BELOW
	LiNeS β=1.00	0.7000	56.66	44.26	-12.40	1.0	BELOW
	MetaGPT	0.2154	54.87	45.75	-9.12	1.0	BELOW
InternVL	OptMerge$^{\|}$	0.0720	47.66	47.94	+0.28	1.0	IN-BAND
	Iso-C$^{\|}$	0.2117	46.99	46.85	-0.15	1.0	IN-BAND
	WUDI$^{\|}$	0.0665	47.74	47.87	+0.13	1.0	IN-BAND
	CFS (c recalibrated)	0.0720	47.66	47.81	+0.15	1.0	IN-BAND
	CFS (c from Qwen)	0.3460	45.12	45.39	+0.27	1.0	IN-BAND
	CFS-0	0.3673	44.78	44.74	-0.04	1.0	IN-BAND
	LiNeS β=0.80	0.5930	40.97	42.10	+1.13	1.0	ABOVE
	LiNeS β=1.00	0.6912	38.88	39.84	+0.96	1.0	IN-BAND
	MetaGPT	0.2565	46.43	44.88	-1.55	1.0	BELOW

curve($\bar{a}$) is task arithmetic at the method's measured scale; IN-BAND means accounted for by that axis. Band $\pm$1.0 point. Domains: [0.050, 0.60] (ViT), [0.10, 1.30] (Qwen), [0.025, 0.70] (InternVL). $\bar{a}$ is plain layer mean (Eq. 1); energy-weighted: Table A2. Rule variants: Section 3; constant-λ row shares mean scale of CFS-0 without its depth profile. Signed depth slopes (Eq. 2) of the Qwen2-VL projections: OptMerge$^{\|}$ $+0.29$, Iso-C$^{\|}$ $+0.45$, WUDI$^{\|}$ $+0.45$.

points below (95% interval $[-0.01, +0.62]$), the size of the orthogonal component: the closed form supplies the scale profile, and optimization adds the orthogonal direction. The rule transfers across architectures once its scale constant is set for that architecture: on InternVL2.5, choosing c so that the rule lands at the optimizer's own mean scale puts it in band, within 0.13 points of the projected optimizer (interval $[-0.57, +0.29]$), whereas the Qwen constant carried over unchanged lands at $\bar{a} = 0.35$ on the descending flank, where the curve reads 45.12 and the rule scores 45.39: c carries the scale, r_ℓ is the transferable depth profile, and the curve predicts the outcome of either choice. The parameter-free CFS-0 makes the same point at $\bar{a} = 0.37$, in band at 57.37 on Qwen2-VL and at 44.74 on InternVL2.5.

The ablation shows which part of the rule does the work (Table 3). Starting from constant-coefficient task arithmetic at $\lambda = 0.30$, moving the constant to 0.373 changes Avg_7 by $+0.07$; adding the depth profile r_ℓ at that same mean scale adds $+0.01$ (paired interval $[-0.34, +0.36]$); re-fitting the scale constant to $c = 0.386$ adds $+0.14$ and is what certifies equivalence with the projected optimizer. The scale constant carries the rule, and the depth profile is a refinement, exactly the ordering the shape-axis results predict for this architecture.

The consequence is a $20\times$ cheaper merge (Table A1, Appendix B): 9 minutes in closed form against 3.0 hours for 300-step per-layer optimization at 7B on the same single GPU, a cost that grows from under ten minutes at 1B. Setting a target scale is a cost every method shares through its coefficient; CFS removes the optimization inside each candidate, so with c fixed merging leaves the critical path of a sweep, whose evaluation pass takes a median of 85 minutes.

Table 3: **Ablation of the closed-form rule on Qwen2-VL-7B.** The scale constant, not the depth profile, is what closes the gap to the projected optimizer.

Variant	$\bar{a}$	Avg_7	Δ vs prev.	TOST @ 0.25
TA, λ=0.30 (constant)	0.3000	57.30	—	—
scale only: constant λ=0.373	0.3726	57.36	+0.07	INCONCLUSIVE
+ depth profile r_ℓ, same mean (CFS-0)	0.3726	57.37	+0.01	INCONCLUSIVE
+ fitted constant c=0.386 (CFS)	0.3105	57.50	+0.14	EQUIVALENT
Median-norm rule (alternative, parameter-free)	0.3726	57.22	—	INCONCLUSIVE
Reference: OptMerge$^{\parallel}$ (projected)	0.2952	57.50	—	REFERENCE
OptMerge (300-step SGD, unprojected)	—	57.79	+0.29	DIFFERS

Rows are cumulative from top to bottom. Verdicts are two one-sided tests against the projected optimizer OptMerge$^{\parallel}$ at an equivalence margin of 0.25 points (n_{boot}=4000). Adding the depth profile at matched mean scale is a paired contrast of +0.01 (95% CI $[-0.34, +0.36]$). Unprojected OptMerge gains +0.29 over its projection (95% CI $[+0.04, +0.56]$), the value of the orthogonal component, which is the largest of the three decomposed terms; it also lands +0.29 above CFS (95% CI $[-0.01, +0.62]$), a contrast not resolved at the 0.25 margin. The profile r_ℓ of Eq. 3 ranges over $[0.61, 1.02]$ on Qwen2-VL, $[0.67, 0.95]$ on InternVL2.5 and $[0.53, 0.95]$ on ViT-B/32; one Qwen2-VL tensor has a negative cross term ($r_\ell > 1$), and the cap $\lambda_\ell \leq 1$ is never active.

The orthogonal component is a direction, not a norm: replacing the optimizer's residual by random directions of the same norm leaves the score at the projected optimizer's (three-draw mean $+0.09$, $[-0.04, +0.22]$; Appendix E).

5.5 The Orthogonal Component Marks Where Scale Is Not Enough

The size of the orthogonal component depends on what is being merged, and the projection measurement reads it off. On the vision–language tracks the component adds 0.29 to 1.41 points for the two optimizers and is where isotropic rescaling loses 28.85 on Qwen2-VL; on the classification track it carries 21.07 and 47.05 points for the two surgeries, whose projections land 7 and 26 points below the curve while both LiNeS schedules sit 11 points above it (Table 2; full record in Appendix K). The InternVL controls agree: unprojected OptMerge and WUDI sit above their projections, the optimizer's own residual added to a constant merge beats task arithmetic by $+0.77$ ($[+0.08, +1.47]$), and norm-matched random directions do not (Appendix I). On the grounding expert's own task, referring expression comprehension, projected rows keep the capability and collapsed-scale rows lose it (Appendix J). The salient difference is the experts' task domains: overlapping capabilities of one model on the vision–language tracks, mutually exclusive label spaces on classification, and only there does the orthogonal component carry most of the signal. For capability merging the scalar account is a two-sided predictor for projected methods and, joined with the depth slope, places per-layer scaling schemes as well; for experts with disjoint output spaces the same measurement says the score lives in the orthogonal component.

6 Discussion

The results support a simple operating rule for merging multimodal experts: get the scale right first, then ask whether shape or an orthogonal component has anything further to offer. Shape has a consistent sign; at fixed scale it adds at most a point and, tilted the wrong way, can cost a dozen; the orthogonal component is where the optimizers earn their cost, and the projection reads its size off in advance. Our claims cover data-free static merging of capability experts at 1B and 7B under one protocol, with the classification track at the outer edge of that scope.

7 Conclusion

Merged multimodal performance is largely a question of how far the merge travels, measurable before any benchmark is run. An effective scale and a signed depth slope predict the benchmark average of projected merging methods within one point, order shape effects across three architectures, and locate the value of per-layer optimization in a small orthogonal component; the profile it learns becomes a closed-form rule at a twentieth of the per-merge cost.

AI USE STATEMENT

In this work, we used generative AI tools to edit the manuscript for readability. We have not used generative AI tools to develop the conceptual framework, to propose or refine hypotheses, to design the methodology or experiments, to implement the merging methods, or to interpret results; generating synthetic data, formulating or proving mathematical claims, translation, dataset cleaning and qualitative data analysis are not applicable to this work. We have reviewed all AI-assisted work: every number produced by the typesetting scripts is checked against the stored artefacts by automated consistency checks and by the authors, and all AI-edited text was reviewed by the authors. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

Section 4 states the models, experts, benchmarks, evaluation toolkit and decision rules; Appendix A gives the checkpoint provenance, the evaluation configuration, the bootstrap procedure with its seeds, the numerical precision of the geometric quantities, and the cost accounting. The per-benchmark scores behind every figure and verdict are tabulated in Tables 1, A3, A5, A6, A7 and A11, and every table in the paper is generated by a script from the stored evaluation outputs. Code for merging, projection surgeries, probe construction, the closed-form rule and all statistics, together with the merged-model evaluation outputs, will be released with the camera-ready version.

REFERENCES

Zhe Chen, Weiyun Wang, Yue Cao, Yangzhou Liu, Zhangwei Gao, Erfei Cui, Jinguo Zhu, Shenglong Ye, Hao Tian, Zhaoyang Liu, et al. Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. *arXiv preprint arXiv:2412.05271*, 2024.

Runxi Cheng, Feng Xiong, Yongxian Wei, Wanyun Zhu, and Chun Yuan. Whoever started the interference should end it: Guiding data-free model merging via task vectors. In *ICML*, 2025. arXiv:2503.08099.

Jiho Choi, Donggyun Kim, Chanhyuk Lee, and Seunghoon Hong. Revisiting weight averaging for model merging. *arXiv preprint arXiv:2412.12153*, 2024.

Yiyang Du, Xiaochen Wang, Chi Chen, Jiabo Ye, Yiru Wang, Peng Li, Ming Yan, Ji Zhang, Fei Huang, Zhifang Sui, Maosong Sun, and Yang Liu. AdaMMS: Model merging for heterogeneous multimodal large language models with unsupervised coefficient optimization. In *CVPR*, 2025. doi: 10.1109/CVPR52734.2025.00879. arXiv:2503.23733.

Haodong Duan, Junming Yang, Yuxuan Qiao, Xinyu Fang, Lin Chen, Yuan Liu, Xiaoyi Dong, Yuhang Zang, Pan Zhang, Jiaqi Wang, Dahua Lin, and Kai Chen. VLMEvalKit: An open-source toolkit for evaluating large multi-modality models. In *ACM Multimedia*, pp. 11198–11201, 2024. doi: 10.1145/3664647.3685520. arXiv:2407.11691.

Antonio Andrea Gargiulo, Donato Crisostomi, Maria Sofia Bucarelli, Simone Scardapane, Fabrizio Silvestri, and Emanuele Rodolà. Task singular vectors: Reducing task interference in model merging. In *CVPR*, pp. 18695–18705, 2025. arXiv:2412.00081.

Danna Gurari, Qing Li, Abigale J. Stangl, Anhong Guo, Chi Lin, Kristen Grauman, Jiebo Luo, and Jeffrey P. Bigham. VizWiz grand challenge: Answering visual questions from blind people. In *CVPR*, pp. 3608–3617, 2018. doi: 10.1109/CVPR.2018.00380. arXiv:1802.08218.

Drew A. Hudson and Christopher D. Manning. GQA: A new dataset for real-world visual reasoning and compositional question answering. In *CVPR*, 2019. arXiv:1902.09506.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Suchin Gururangan, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic. In *ICLR*, 2023. arXiv:2212.04089.

Chanhyuk Lee, Jiho Choi, Chanryeol Lee, Donggyun Kim, and Seunghoon Hong. AdaRank: Adaptive rank pruning for enhanced model merging. In *ICLR*, 2026.

Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. MathVista: Evaluating mathematical reasoning of foundation models in visual contexts. In *ICLR*, 2024. arXiv:2310.02255.

Daniel Marczak, Simone Magistri, Sebastian Cygert, Bartłomiej Twardowski, Andrew D. Bagdanov, and Joost van de Weijer. No task left behind: Isotropic model merging with common and task-specific subspaces. In *ICML*, 2025. arXiv:2502.04959.

Ahmed Masry, Do Xuan Long, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. ChartQA: A benchmark for question answering about charts with visual and logical reasoning. In *Findings of ACL*, pp. 2263–2279, 2022. doi: 10.18653/v1/2022.findings-acl.177.

Anand Mishra, Shashank Shekhar, Ajeet Kumar Singh, and Anirban Chakraborty. OCR-VQA: Visual question answering by reading text in images. In *ICDAR*, pp. 947–952, 2019. doi: 10.1109/ICDAR.2019.00156.

Guillermo Ortiz-Jimenez, Alessandro Favero, and Pascal Frossard. Task arithmetic in the tangent space: Improved editing of pre-trained models. In *NeurIPS*, 2023. arXiv:2305.12827.

Huaizhi Qu, Xinyu Zhao, Jie Peng, Kwonjoon Lee, Behzad Dariush, and Tianlong Chen. UQ-Merge: Uncertainty guided multimodal large language model merging. In *Findings of ACL*, pp. 1401–1417, 2025. doi: 10.18653/v1/2025.findings-acl.73.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, pp. 8748–8763, 2021. arXiv:2103.00020.

Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards VQA models that can read. In *CVPR*, 2019. arXiv:1904.08920.

Ronald Skorobogat, Karsten Roth, Mariana-Iuliana Georgescu, and Zeynep Akata. Subspace-boosted model merging. *arXiv preprint arXiv:2506.16506*, 2025.

George Stoica, Pratik Ramesh, Boglarka Ecsedi, Leshem Choshen, and Judy Hoffman. Model merging with SVD to tie the knots. In *ICLR*, 2025. arXiv:2410.19735.

Yi-Lin Sung, Linjie Li, Kevin Lin, Zhe Gan, Mohit Bansal, and Lijuan Wang. An empirical study of multimodal model merging. In *Findings of EMNLP*, pp. 1563–1575, 2023. doi: 10.18653/v1/2023.findings-emnlp.105.

Anke Tang, Li Shen, Yong Luo, Enneng Yang, Han Hu, Lefei Zhang, Bo Du, and Dacheng Tao. FusionBench: A unified library and comprehensive benchmark for deep model fusion. *Journal of Machine Learning Research*, 26:307:1–307:38, 2025. arXiv:2406.03280.

Ke Wang, Junting Pan, Weikang Shi, Zimu Lu, Houxing Ren, Aojun Zhou, Mingjie Zhan, and Hongsheng Li. Measuring multimodal mathematical reasoning with MATH-Vision dataset. In *NeurIPS*, 2024a. arXiv:2402.14804.

Ke Wang, Nikolaos Dimitriadis, Alessandro Favero, Guillermo Ortiz-Jimenez, Francois Fleuret, and Pascal Frossard. LiNeS: Post-training layer scaling prevents forgetting and enhances model merging. In *ICLR*, 2025. arXiv:2410.17146.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfei Du, Xuancheng Ren, Rui Men, Dayiheng Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. Qwen2-VL: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024b.

Yongxian Wei, Anke Tang, Li Shen, Zixuan Hu, Chun Yuan, and Xiaochun Cao. Modeling multi-task model merging as adaptive projective gradient descent. In *ICML*, 2025. arXiv:2501.01230.

Yongxian Wei, Runxi Cheng, Weike Jin, Enneng Yang, Li Shen, Lu Hou, Sinan Du, Chun Yuan, Xiaochun Cao, and Dacheng Tao. OptMerge: Unifying multimodal LLM capabilities and modalities via model merging. In *ICLR*, 2026. arXiv:2505.19892.

Mitchell Wortsman, Gabriel Ilharco, Samir Yitzhak Gadre, Rebecca Roelofs, Raphael Gontijo-Lopes, Ari S. Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, and Ludwig Schmidt. Model soups: Averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In *ICML*, 2022. arXiv:2203.05482.

Prateek Yadav, Derek Tam, Leshem Choshen, Colin Raffel, and Mohit Bansal. TIES-Merging: Resolving interference when merging models. In *NeurIPS*, 2023. arXiv:2306.01708.

Prateek Yadav, Tu Vu, Jonathan Lai, Alexandra Chronopoulou, Manaal Faruqui, Mohit Bansal, and Tsendsuren Munkhdalai. What matters for model merging at scale? *Transactions on Machine Learning Research*, 2025.

Enneng Yang, Zhenyi Wang, Li Shen, Shiwei Liu, Guibing Guo, Xingwei Wang, and Dacheng Tao. AdaMerging: Adaptive model merging for multi-task learning. In *ICLR*, 2024. arXiv:2310.02575.

Le Yu, Bowen Yu, Haiyang Yu, Fei Huang, and Yongbin Li. Language models are super mario: Absorbing abilities from homologous models as a free lunch. In *ICML*, 2024. arXiv:2311.03099.

Yuyan Zhou, Liang Song, Bingning Wang, and Weipeng Chen. MetaGPT: Merging large language models using model exclusive task arithmetic. In *EMNLP*, pp. 1711–1724, 2024. arXiv:2406.11385.

APPENDIX

A PROTOCOL, CHECKPOINTS AND REPRODUCTION

Experts and base checkpoints. The two multimodal tracks merge the five capability experts (VQA, Geometry, Chart, OCR, Grounding) released with OptMerge (Wei et al., 2026). The InternVL2.5-1B experts are full fine-tunes. The Qwen2-VL-7B experts were trained with rank-8 LoRA from the Qwen2-VL-7B base and are distributed as merged full-weight checkpoints, which we use as distributed. On every track the task vector is the difference of stored weights, $\tau_i = \theta_i - \theta_0$, so the projection surgery, the closed-form rule and every baseline act in full weight space. The merge touches 196 parameter tensors on Qwen2-VL-7B (seven weight matrices in each of 28 decoder layers), 171 on InternVL2.5-1B (seven in each of 24 decoder layers plus three in the vision–language projector) and 152 on ViT-B/32 (twelve in each of 12 residual blocks, biases and LayerNorm vectors included, plus eight outside the blocks); $\bar{a}$ averages over all of them and $\hat{g}$ regresses over the block tensors. Section 5.1 reports the parallel-energy fraction for both expert parameterizations, LoRA-derived and full fine-tuned. Every comparison drawn in the main text is between our own runs under one evaluation protocol; the one exception is the reference row in Table A1, which is cited from published work for order-of-magnitude context on different hardware and is labelled as such.

Evaluation. All multimodal evaluation runs through VLMEvalKit under a single fixed configuration, with one model held resident per evaluation and no concurrent instances. Math benchmarks (MathVista, MATH-Vision) are scored with the official answer-extraction routine invoked per item. The LLM judge is Qwen2.5-7B-Instruct-AWQ, served locally with vLLM 0.7.3 behind an OpenAI-compatible endpoint and called through VLMEvalKit's judge interface with the toolkit's built-in MathVista and MATH-Vision extraction prompts, temperature 0, top_p 1.0 and a 2048-token limit; regular-expression scoring of these benchmarks differs from it by tens of points and is not used. Judge-side failures are counted as zero under the official convention and never exceed 10% of items in any reported run (the p8 gate rejects a judge session above that threshold as infrastructure failure); the worst case among the rows of Tables A7, A8 and A9 is 9.2% of MATH-Vision items, and scoring such rows in the worst case does not flip any decision reported in this paper.

Statistics. Every interval is a paired per-item bootstrap: one set of resampled item indices is drawn and applied to all models being compared, so item-difficulty variance cancels. This matters at our effect sizes because the independent standard error of Avg_7 is roughly 0.48 points, which is larger than several differences we need to resolve; the paired standard error is 0.34 on Qwen2-VL and 0.41 on InternVL2.5. All intervals are 95% unless stated otherwise. We use $n_{\mathrm{boot}} = 4000$ except

Table A1: **Cost of merging.** A closed-form recipe reaches the same neighbourhood as 300 steps of per-layer SGD at a small fraction of the compute.

Method	Solving time	Device	Requires (beyond task vectors)
InternVL2.5-1B (32 explicitly timed merges)			
Task Arithmetic	1.1 min	1×GPU	—
TIES	14.8 min	1×GPU	—
WUDI	8.3 min	1×GPU	Gram matrices
OptMerge	9.7 min	1×GPU	Gram + 300 SGD steps/layer
Qwen2-VL-7B			
CFS (closed form)	9 min	1×GPU	Gram matrices
OptMerge (300-step SGD)	3.0 h	1×GPU	Gram + 300 SGD steps/layer
Reference (cited, not our run)			
OptMerge, as reported by Wei et al. (2026)	3.78 h	8×V100	Gram + 300 SGD steps/layer

Timings grouped by architecture, never pooled: optimization cost grows with scale ($\sim$10 min at 1B, 3.0 h at 7B) while closed form stays at I/O cost (9 min), giving up 0.29 Avg_7 points. 1B figures are medians over 32 merges; 7B OptMerge is a dedicated re-timing. Evaluation costs 85 min per model over seven benchmarks (excluded). Cited row (Table 7) is order-of-magnitude context only, different hardware. Our hardware: one NVIDIA RTX 3090 (24GB), CUDA 12.4; float16 weights, per-layer optimization in float32. CFS constant c set once per architecture as a target scale (like λ); requires no gradient steps or evaluation data, only Gram matrices for r_ℓ.

where a table states otherwise and for the Qwen2-VL comparison of task arithmetic at $\lambda = 0.45$ with OptMerge in Section 5.2, which uses 2000. Bootstrap seeds: 20260823 for the ViT-B/32 and InternVL2.5 probe runs, 20260818 for the InternVL2.5 decision statistics, 20260825 for the Qwen2-VL curve-node intervals.

Geometric quantities. Inner products, norms and cosines are accumulated in fp64 after casting the stored fp16 weights, for the OptMerge profile of Figure 2 and for every InternVL2.5 and ViT-B/32 quantity. At these tensor sizes (1.3×10^7 elements per matrix) fp32 reductions are not adequate: a strictly parallel displacement then returns cosines above one by a few parts in 10^3. Two Qwen2-VL profiles, Iso-C$^\parallel$ and WUDI$^\parallel$, come from an earlier fp32 pass, and its precision control is task arithmetic at $\lambda = 1.0$ run through the same code: ρ_ℓ returns exactly 1 in every tensor and the cosine at most 1.0215, which bounds the fp32 error on the parallel coefficient $\rho_\ell \cos\theta_\ell$ at 2.15%. For those two rows the bound moves $\bar{a}$ by at most 0.003 and 0.015 and the curve reading by at most 0.3 and 0.1 points, inside the band in both cases.

Cost accounting. Merging times in Table A1 are grouped by architecture and never pooled across scales, because the optimization-based cost grows steeply with model size while the closed form stays at I/O cost. The 1B figures come from 32 explicitly logged merges and the 7B figure from a dedicated re-timing run. All timing runs are performed on a single NVIDIA RTX 3090 (24GB) GPU (driver 550.54, CUDA 12.4; PyTorch 2.4 for InternVL2.5 and 2.5 for Qwen2-VL); weights are stored and merged in float16, and the per-layer optimization of the 1B merges runs in float32. The system has two such devices but each merge uses only one. Evaluation cost is excluded from that table and stated separately in the main text.

B Cost of Merging

Table A1 gives the timings behind the cost claim in Section 5.4, grouped by architecture and never pooled across scales.

C Effective Scale: Sensitivity to the Estimator

Every $\bar{a}$ in this paper is the plain layer mean of Equation 1, which treats all layers equally. An alternative is an energy-weighted mean $\sum_\ell \|\Delta_\ell\|^2 \rho_\ell / \sum_\ell \|\Delta_\ell\|^2$, which up-weights the layers that move most. The unweighted mean is our default because it matches the object a per-layer rule controls, one coefficient per tensor, whereas the energy-weighted mean is set by the few largest matrices. For probe construction (Appendix D), we pin the energy-weighted $\bar{a}$ to ensure equal total

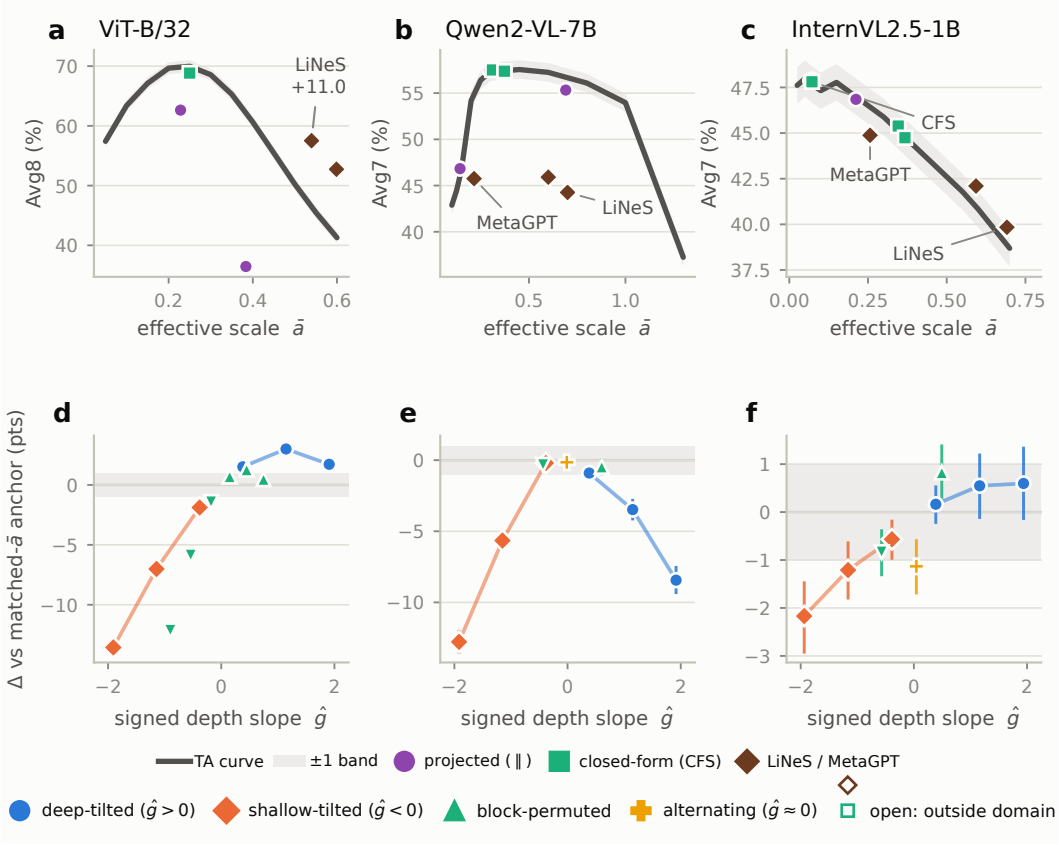

Figure A1: **Both axes on three architectures. Top (a–c), scale axis:** each architecture's reference curve $\mathrm{curve}(\lambda)$ with its ± 1.0 band, every method placed at its *measured* $\bar{a}$; open markers lie outside the calibrated domain, where no verdict is claimed. **Bottom (d–f), shape axis:** Δ is a probe minus a constant-λ anchor at the *same* $\bar{a}$, so only shape varies, and left of zero is shallow-tilted. Vertical axes are Avg_8 in (a) and Avg_7 in (b, c); the two legend rows are independent. Marker shapes in (d–f) are read in Appendix D.

displacement across probes at different slopes, then report both estimators in Table A2 for sensitivity analysis.

Figure A1 places every method on its architecture's reference curve at its measured $\bar{a}$ (top row) and plots the depth-shape probes of Appendix D at matched scale (bottom row); Table A2 tabulates both estimators for every plotted method. The choice matters where the curve is steep. For the isotropic surgery on Qwen2-VL the two estimators differ by only 0.0248 in $\bar{a}$ but 2.03 points in $\mathrm{curve}(\bar{a})$, because that point sits on the steep flank: the secant slope over $\lambda \in [0.1, 0.2]$ is 113 points per unit λ and the steepest adjacent pair reaches 150. Near the plateau the same change moves the prediction by less than 0.2 points; on InternVL2.5 no row moves by more than 0.29, and the weighted estimator carries LiNeS $\beta{=}1.00$ just past the domain end (0.7091 against 0.70). On ViT-B/32 the weighted estimator is systematically larger; under it TSV-M$^{\|}$ and the recalibrated closed-form rule would sit above the curve rather than below it, while both LiNeS residuals remain large and positive.

D Full Depth-Shape Probe Grid

Figure A2 plots the dose response of the two branches for all three architectures, and Table A3 gives every probe with per-benchmark scores, the realized signed slope $\hat{g}$, and paired intervals; the corresponding ViT-B/32 probes, with their intervals, are in Table A11.

Table A2: **The two $\bar{a}$ conventions (appendix).** Where the curve is steep, the choice of estimator moves the prediction by more than the decision band.

Method	Plain mean (Eq. 1, used throughout)		Energy-weighted		Δ curve
	$\bar{a}$	curve($\bar{a}$)	$\bar{a}$	curve($\bar{a}$)	
Qwen2-VL-7B					
OptMerge$^{\parallel}$	0.2952	57.21	0.3106	57.31	+0.10
Iso-C$^{\parallel}$	0.1427	46.06	0.1180	44.03	-2.03
WUDI$^{\parallel}$	0.6905	56.72	0.6765	56.80	+0.08
Constant λ (control)	0.3726	57.42	0.3726	57.42	-0.00
CFS	0.3105	57.31	0.3177	57.33	+0.01
Median-norm rule	0.3726	57.42	0.3798	57.43	+0.01
CFS-0	0.3726	57.42	0.3813	57.43	+0.01
LiNeS β=0.80	0.6000	57.24	0.5828	57.28	+0.04
LiNeS β=1.00	0.7000	56.66	0.6785	56.79	+0.12
MetaGPT	0.2154	54.87	0.2170	54.94	+0.07
InternVL2.5-1B					
OptMerge$^{\parallel}$	0.0720	47.66	0.0702	47.69	+0.03
Iso-C$^{\parallel}$	0.2117	46.99	0.2306	46.75	-0.24
WUDI$^{\parallel}$	0.0665	47.74	0.0622	47.81	+0.06
CFS (c recalibrated)	0.0720	47.66	0.0736	47.64	-0.02
CFS (c from Qwen)	0.3460	45.12	0.3536	45.00	-0.12
CFS-0	0.3673	44.78	0.3755	44.64	-0.13
LiNeS β=0.80	0.5930	40.97	0.6073	40.68	-0.29
LiNeS β=1.00	0.6912	38.88	0.7091	—	—
MetaGPT	0.2565	46.43	0.2604	46.38	-0.05
ViT-B/32					
Iso-C$^{\parallel}$	0.3835	62.17	0.7072	—	—
TSV-M$^{\parallel}$	0.2283	69.84	0.4071	59.89	-9.95
CFS (c recalibrated)	0.2500	69.98	0.3167	67.47	-2.51
LiNeS β=0.875	0.5395	46.49	0.5739	43.48	-3.01
LiNeS β=1.00	0.5987	41.39	0.6381	—	—

Every $\bar{a}$ in the paper is the plain layer mean of Equation 1. This table compares it with an energy-weighted mean (weights $\|\Delta_\ell\|^2$), which up-weights the layers that move most, for every method placed on a reference curve in this paper; Δ curve is the change in the prediction from switching to the weighted estimator, and a dash marks an $\bar{a}$ outside the calibrated domain of the curve. Near the plateau the choice moves the prediction by less than 0.2 points; on the steep flank it does not: for Iso-C$^{\parallel}$ on Qwen2-VL the two estimators differ by only 0.0248 in $\bar{a}$ but 2.03 points in curve($\bar{a}$), since the secant slope over $\lambda \in [0.1, 0.2]$ is 113 points per unit λ and the steepest adjacent pair reaches 150. On ViT-B/32 the energy-weighted estimator is systematically larger, and under it TSV-M$^{\parallel}$, CFS (c recalibrated) would change side relative to the ± 1.0 band, while both LiNeS residuals stay large and positive; the qualitative reading of that track (projected methods far from the curve, LiNeS above it) does not depend on the estimator.

Probe construction. Every probe is a per-tensor coefficient table $\lambda_\ell = \bar{a}\,(1 + a\,\hat{s}_\ell)$ built around the architecture's anchor $\bar{a}$ (0.45, 0.15 and 0.25 for Qwen2-VL, InternVL2.5 and ViT-B/32), a dose $a \in \{0.2, 0.6, 1.0\}$ and a unit shape $\hat{s}$. The shape starts from a block ramp $r_b = 2b/(L-1) - 1$ over the L transformer blocks, shared by the tensors of block b: the ramp itself for the deep-tilted family, its negative for the shallow-tilted family, a random permutation of its L block values for the block-permuted family, and ± 1 by block parity for the alternating family. Each shape is centred under the energy weights $w_\ell = \|\Delta_\ell\|^2$ and scaled to $\max_\ell |\hat{s}_\ell| = 1$, so the energy-weighted mean of λ_ℓ equals $\bar{a}$ exactly at every dose and the smallest coefficient is $\bar{a}(1-a) \geq 0$; tensors outside the blocks (the InternVL2.5 projector, the ViT-B/32 embeddings) keep $\hat{s}_\ell = 0$. On ViT-B/32 all four families run at all three doses with permutation seeds 20260821 and 20260822, twelve probes in all. On the two vision–language tracks the tilted families run at all three doses; the block-permuted and alternating probes run at $a = 0.6$, the permutation seed being the first from 20260823 upward whose realized $\hat{g}$ falls within $[0.35, 0.65]$ times (or minus $[0.35, 0.65]$ times) that of the deep-tilted probe at the same dose, and the parity shape being accepted when $|\hat{g}| \leq 0.05$ and its weighted dispersion is at least 0.85 of the deep-tilted probe's. All schedules are evaluated post hoc through Equation 2 to report the realized $\hat{g}$. On Qwen2-VL the two tilted branches cross at the smallest dose: deep minus shallow is -0.70 with interval $[-1.22, -0.20]$ at $a = 0.2$, against $+2.18$ with $[+1.41, +3.00]$ at $a = 0.6$; both

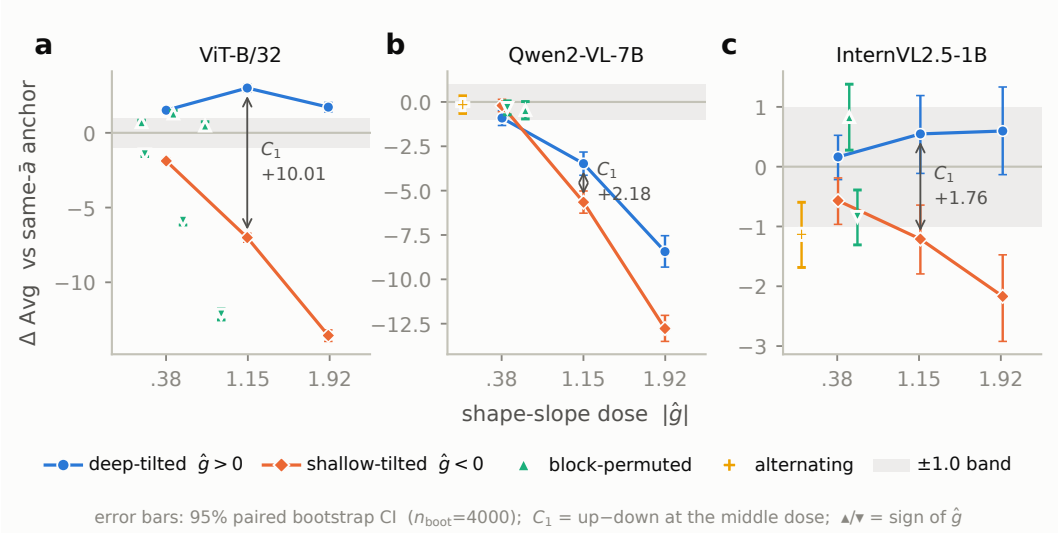

Figure A2: **With scale pinned, the depth-shape branches separate monotonically in dose.** Δ is each probe minus a constant-λ anchor at the same $\bar{a}$, so every displacement is shape rather than scale; grey band is ± 1.0. Axes are independent because the effects differ by an order of magnitude, and the horizontal axis is the *measured* slope $|\hat{g}|$. C_1 is the deep$-$shallow contrast at the middle dose: $+10.01$, $+2.18$, $+1.76$, positive in 3 of 3 architectures; at the lowest Qwen dose the ordering reverses (-0.70 on unrounded scores, interval $[-1.22, -0.20]$), both tilts losing against the anchor. Error bars are 95% paired intervals; on the classification panel their half-widths (0.13–0.38 points) are smaller than the markers against a 17-point span, so they hide inside the symbols. Per-probe scores and intervals: Table A3.

tilts lose against the anchor at every Qwen2-VL dose (Table A3), and InternVL2.5 shows no such crossing.

Three points about the design matter.

First, probes are matched to their anchor in $\bar{a}$, not in the nominal coefficient. Pinning λ on the probe side instead hands it a pure scale advantage of $+0.34$ points on InternVL2.5 before any shape effect enters, so contrasts intended to isolate shape pin $\bar{a}$ on both sides. The probes are constructed with their energy-weighted $\bar{a}$ pinned to the anchor; their plain layer mean deviates from it by at most 0.011, 0.019 and 0.005 on ViT-B/32, Qwen2-VL and InternVL2.5, which moves the curve prediction by at most 0.31, 0.04 and 0.06 points.

Second, the block-permutation probes are not exchangeable controls. Their realized slopes came out with opposite signs by construction, and their residuals track that sign closely, so we stratify them by realized $\hat{g}$ rather than treating them as a single family. This is also why Figure A2 plots realized $|\hat{g}|$ rather than nominal dose.

Third, the shape response is asymmetric on the classification track: the loss from shallow tilt grows monotonically with dose to about 13.6 points, while the gain from deep tilt saturates by a few points. The apparent non-monotonicity of the deep-tilted branch there is saturation rather than noise.

Two finer effects qualify the slope as a summary of shape. On Qwen2-VL, permuting layer order, which changes the coherence of the profile while keeping its slope comparable, is nearly free, whereas coherent monotone tilt is not; on InternVL2.5 the corresponding contrasts include zero. A probe with near-zero slope but extreme layer-to-layer alternation lowers Avg_7 by 1.13 points on InternVL2.5, interval $[-1.69, -0.60]$, while its Qwen2-VL counterpart sits at -0.15 with $[-0.65, +0.35]$: there is a shape effect beyond slope on at least one architecture, and $\hat{g}$ should be read as a summary of coherent depth tilt.

Reading the probe panels. In Figure A1(d–f) and Figure A2, marker shape encodes probe family: circles are deep-tilted, diamonds shallow-tilted, triangles block-permuted (pointing up or down with

Table A3: **Full depth-shape probe grid (appendix).** Nine probes per architecture with per-benchmark scores, realized slope and paired intervals.

Probe	VizWiz	GQA	MVista	MVis	CQA	TVQA	OCR	Avg_7	$\hat{g}$	Δ [95% CI]	
Qwen2-VL-7B (anchor: TA @ λ=0.45)											
deep-tilted @ a=0.2	39.75	62.33	55.90	21.05	79.20	78.79	59.47	56.64	+0.384	-0.91	$[-1.33, -0.49]$
shallow-tilted @ a=0.2	41.14	61.28	55.50	22.04	77.72	82.32	61.43	57.35	-0.384	-0.20	$[-0.55, +0.15]$
deep-tilted @ a=0.6	37.30	62.05	54.40	20.07	76.32	72.36	56.02	54.07	+1.151	-3.48	$[-4.13, -2.82]$
shallow-tilted @ a=0.6	39.53	60.22	54.20	21.71	48.44	78.04	61.13	51.90	-1.151	-5.65	$[-6.27, -5.02]$
block-permuted ($\hat{g}>0$) @ a=0.6	40.32	62.34	56.30	21.05	79.04	79.88	60.68	57.09	+0.603	-0.46	$[-0.96, +0.02]$
block-permuted ($\hat{g}<0$) @ a=0.6	41.47	61.57	55.70	22.37	75.32	82.45	61.65	57.22	-0.434	-0.33	$[-0.72, +0.06]$
alternating ($\hat{g}\approx0$) @ a=0.6	40.00	61.55	55.10	24.34	79.44	80.41	60.94	57.40	-0.014	-0.15	$[-0.65, +0.35]$
deep-tilted @ a=1.0	33.78	61.02	49.70	21.38	67.96	63.21	46.74	49.11	+1.918	-8.44	$[-9.31, -7.54]$
shallow-tilted @ a=1.0	35.15	55.61	53.50	20.72	27.68	63.34	57.42	44.78	-1.918	-12.77	$[-13.50, -12.03]$
anchor TA @ λ=0.45	40.56	62.12	55.70	22.04	80.08	81.15	61.20	57.55	—	*anchor*	
InternVL2.5-1B (anchor: TA @ λ=0.15)											
deep-tilted @ a=0.2	31.21	57.00	44.60	18.42	66.04	74.84	43.46	47.94	+0.388	+0.16	$[-0.22, +0.52]$
shallow-tilted @ a=0.2	31.39	56.66	44.10	17.11	62.52	75.35	43.33	47.21	-0.388	-0.57	$[-0.96, -0.19]$
deep-tilted @ a=0.6	31.17	57.12	45.10	17.76	69.88	74.27	42.97	48.32	+1.163	+0.55	$[-0.11, +1.19]$
shallow-tilted @ a=0.6	30.61	56.38	43.60	17.43	59.64	75.10	43.20	46.57	-1.163	-1.21	$[-1.79, -0.64]$
block-permuted ($\hat{g}>0$) @ a=0.6	31.40	56.77	44.20	21.71	67.60	75.54	43.00	48.60	+0.493	+0.83	$[+0.28, +1.38]$
block-permuted ($\hat{g}<0$) @ a=0.6	31.25	56.58	42.70	15.79	63.68	75.51	43.07	46.94	-0.570	-0.84	$[-1.31, -0.39]$
alternating ($\hat{g}\approx0$) @ a=0.6	30.79	56.37	43.30	15.79	64.08	73.77	42.42	46.64	+0.044	-1.13	$[-1.69, -0.60]$
deep-tilted @ a=1.0	31.15	57.12	44.70	17.43	71.80	74.00	42.42	48.37	+1.938	+0.60	$[-0.13, +1.33]$
shallow-tilted @ a=1.0	29.83	55.28	42.50	17.43	56.88	74.62	42.71	45.61	-1.938	-2.17	$[-2.92, -1.47]$
anchor TA @ λ=0.15	31.38	56.97	45.10	18.42	64.20	75.12	43.23	47.78	—	*anchor*	

Every depth-shape probe is built at a pinned $\bar{a}$, so Δ isolates shape. $\hat{g}$ is the realized signed depth slope, which is what Fig. A2 plots rather than the nominal dose a. Intervals are 95% paired per-item bootstrap (n_{boot}=4000; seed 20260823 for InternVL). The alternating probe disagrees across architectures: outside the band for InternVL, inside and consistent with zero for Qwen. It was not constructed for ViT-B/32, whose 12 probes appear in Table A11.

the sign of the realized $\hat{g}$), and plus signs the alternating probe with $\hat{g} \approx 0$. Error bars are 95% paired per-item bootstrap intervals; on ViT-B/32 their half-widths (0.134–0.384 points) are smaller than the markers against a 17-point vertical span, so they hide inside the symbols rather than being absent. The two legend rows of Figure A1 are independent: the same hue marks projected methods in (a–c) and the deep-tilted family in (d–f).

E Noise Floor and Control Merges

Table A4 reports the control merges used to establish the noise floor for the $+0.29$-point orthogonal component. Each control keeps the optimizer's per-layer scale profile and replaces its orthogonal residual by a random direction of the same norm, drawn either in the full weight space or inside the span of the task vectors. The three independent draws land within $+0.04$ of the projected optimizer in point estimate; their mean, $+0.089$ with interval $[-0.037, +0.218]$, is equivalent to it at the 0.25-point margin, while two single draws are inconclusive on their own because their intervals reach $+0.28$ and $+0.34$. The optimizer's advantage over the same reference is $+0.294$ with $[+0.037, +0.564]$, which excludes zero; contrasted directly against each random-direction merge it is $+0.17$ to $+0.26$ with intervals that include zero, so on Qwen2-VL the noise floor resolves the sign of the orthogonal component but not yet its margin over a random direction of the same norm. The InternVL2.5 counterpart resolves both (Appendix I): the optimizer's residual added to a constant merge gains $+0.77$ with $[+0.08, +1.47]$ where random directions gain nothing. An additivity check here adds the optimizer's actual orthogonal residual to a constant $\lambda = 0.30$ merge; the interaction contrast is -0.11 with interval $[-0.48, +0.28]$ (paired standard error 0.191): the point estimate is small, and the interval includes both zero and values the size of the effect, so the row is reported as inconclusive rather than as a confirmation of additivity. Rows whose intervals are too wide to certify either equivalence or a difference are reported as inconclusive throughout.

F Per-Benchmark Scale Sweep

Table A5 is the twelve-node sweep behind Figure A3, including each benchmark's own optimum. Where a benchmark's optimum is a plateau we take the left edge; MATH-Vision is tied between 0.25 and 0.30. Picking each benchmark's own node is an oracle that needs the task identity at test time, so it is an upper bound on any single static coefficient, not a merge; on this twelve-node grid it reaches 58.75; the half-split analysis of Section 5.2 runs on the eight nodes at $\lambda \in$

Table A4: **Noise floor and control groups (appendix).** Qwen2-VL-7B.

Contrast	Δ	se	95% CI	TOST @ 0.25
Norm-matched random orthogonal directions vs. OptMerge$^{\parallel}$				
Random ⊥ direction, draw 1	+0.105	0.086	$[-0.063, +0.281]$	INCONCLUSIVE
Random ⊥ direction, draw 2	+0.037	0.063	$[-0.097, +0.149]$	EQUIVALENT
Random ⊥ direction, draw 3	+0.125	0.111	$[-0.087, +0.341]$	INCONCLUSIVE
Mean of the three draws	+0.089	0.065	$[-0.037, +0.218]$	EQUIVALENT
Random ⊥ direction (task subspace) − draw 1	+0.035	0.133	$[-0.223, +0.297]$	INCONCLUSIVE
Reference contrast				
OptMerge − OptMerge$^{\parallel}$	+0.294	0.136	$[+0.037, +0.564]$	DIFFERS
OptMerge − (OptMerge$^{\parallel}$ + draw 1)	+0.189	0.145	$[-0.095, +0.472]$	INCONCLUSIVE
OptMerge − (OptMerge$^{\parallel}$ + draw 2)	+0.257	0.147	$[-0.018, +0.555]$	INCONCLUSIVE
OptMerge − (OptMerge$^{\parallel}$ + draw 3)	+0.170	0.161	$[-0.148, +0.492]$	INCONCLUSIVE
Additivity of the orthogonal residual				
(TA@0.30 + OptMerge ⊥ residual) − TA@0.30	+0.182	0.144	$[-0.094, +0.477]$	INCONCLUSIVE
Interaction: above − (OptMerge − OptMerge$^{\parallel}$)	-0.112	0.191	$[-0.483, +0.278]$	INCONCLUSIVE (@0.3)

Control merges that should carry *no* real signal, used to establish the noise floor against which the +0.29-point OptMerge advantage is judged. Each random direction is orthogonal to the task-vector sum in every layer and scaled to the norm of OptMerge's own orthogonal residual, so it matches OptMerge$^{\parallel}$ in scale and differs only in the direction of the orthogonal part; the task-subspace variant draws that direction inside $\mathrm{span}\{\tau_{i,\ell}\}$. The additivity block adds OptMerge's actual orthogonal residual to a constant λ=0.30 merge; the interaction row tests whether that residual adds the same on top of task arithmetic as on top of the optimizer's own profile (margin 0.3). All contrasts are paired per-item bootstrap with n_{boot}=4000. Three independent end-to-end re-evaluations of the first random draw gave Δ values within 0.014 of one another, which bounds evaluation-pipeline noise (LLM-judged math benchmarks) well below every margin used. INCONCLUSIVE is reported as such rather than being read as evidence of equivalence.

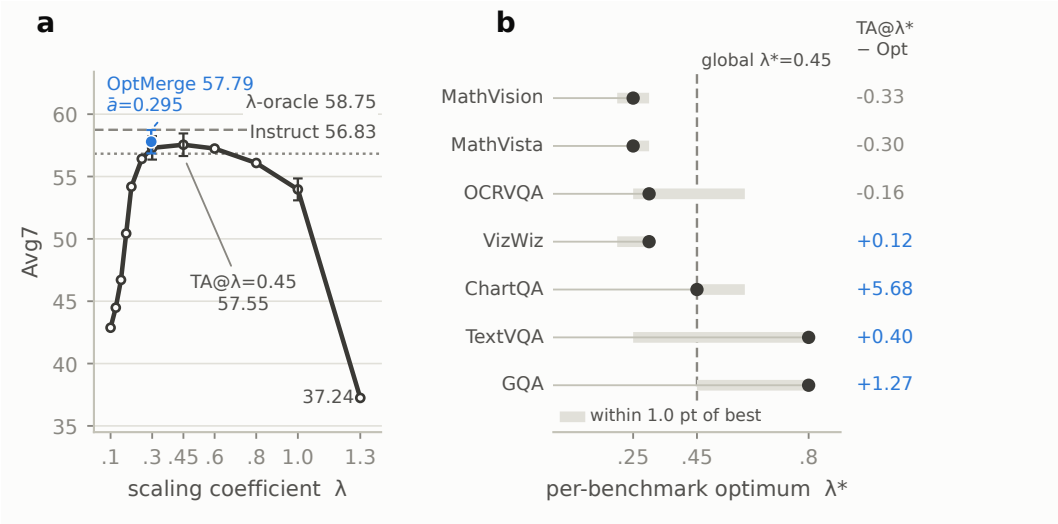

Figure A3: **A single scalar reaches the neighbourhood of optimization-based merging.** Avg_7, Qwen2-VL-7B. **(a)** The sweep peaks at 57.55 (λ=0.45) and collapses to 37.24 by λ=1.3; optimization reaches 57.79; its parallel component sits at $\bar{a}$=0.295. **(b)** Best $\lambda^\star$ per benchmark, with bars giving TA@$\lambda^\star$ minus optimization. The oracle drawn uses all twelve nodes (58.75); the half-split analysis of Section 5.2 uses the eight coarse nodes (58.62).

$\{0.1, 0.2, 0.3, 0.45, 0.6, 0.8, 1.0, 1.3\}$, where the oracle is 58.62. The half-split figures are selection-corrected point estimates with split quantiles, not confidence intervals.

Every node carries a measured paired interval: re-scoring all twelve nodes per item and bootstrapping against the plateau ($n_{\mathrm{boot}} = 4000$) gives per-node standard errors between 0.379 and 0.479 points and paired half-widths between 0.49 and 0.97; both are listed per node in Table A5.

Two consequences bound how finely this curve can be read. First, the location of the peak is not resolved: against $\lambda = 0.45$, the node at 0.30 differs by -0.25 with interval $[-0.90, +0.38]$ and the node at 0.60 by -0.31 with $[-0.80, +0.17]$, so all three are statistically indistinguishable and the maximum should be read as a plateau over $\lambda \in [0.30, 0.60]$ rather than as a point at 0.45. Second,

Table A5: **Per-benchmark λ sweep (appendix).** Qwen2-VL-7B, task arithmetic.

λ	VizWiz	GQA	MVista	MVis	CQA	TVQA	OCR	Avg_7	se	Δ vs λ=0.45 [95% CI]
0.1	37.03	51.85	52.30	20.39	20.48	64.87	53.19	42.87	0.458	-14.67 $[-15.61, -13.79]$
0.125	37.54	53.41	53.00	22.04	22.92	68.23	54.23	44.48	0.462	-13.07 $[-13.94, -12.20]$
0.15	38.33	54.25	54.00	21.71	29.44	73.57	55.66	46.71	0.463	-10.84 $[-11.73, -9.98]$
0.175	39.71	55.50	55.30	22.04	44.72	78.23	57.52	50.43	0.462	-7.12 $[-7.94, -6.31]$
0.2	40.74	57.33	56.80	24.67	59.60	80.49	59.67	54.19	0.476	-3.36 $[-4.14, -2.60]$
0.25	41.29	59.62	**58.50**	**25.00**	68.04	81.46	61.00	56.42	0.479	-1.13 $[-1.84, -0.43]$
0.3	**41.63**	60.61	57.60	25.00	73.36	81.54	**61.33**	57.30	0.479	-0.25 $[-0.90, +0.38]$
0.45	40.56	62.12	55.70	22.04	**80.08**	81.15	61.20	**57.55**	0.463	*ref.*
0.6	40.37	62.45	54.70	20.72	79.92	81.79	60.74	57.24	0.459	-0.31 $[-0.80, +0.17]$
0.8	40.32	**62.67**	52.50	16.78	78.84	**82.02**	59.47	56.09	0.445	-1.46 $[-2.14, -0.79]$
1.0	39.23	60.76	46.50	16.45	77.40	80.96	56.48	53.97	0.445	-3.58 $[-4.41, -2.76]$
1.3	30.47	40.72	31.20	6.25	47.64	67.02	37.40	37.24	0.379	-20.31 $[-21.29, -19.34]$
λ-oracle	41.63	62.67	58.50	25.00	80.08	82.02	61.33	58.75	—	—
best $\lambda^\star$	0.3	0.8	0.25	0.25	0.45	0.8	0.3	—	—	—

Column abbreviations: MVista = MathVista, MVis = MathVision, CQA = ChartQA, TVQA = TextVQA, OCR = OCRVQA. The full 12-node sweep behind Fig. A3. Bold marks each column's own optimum; where a column has a plateau the left edge is taken. The oracle row picks each benchmark's best node independently and averages, giving 58.75 on the 12-node grid (the half-split analysis of the main text uses the eight coarse nodes). Perception benchmarks peak at large λ and reasoning/OCR at small, which is why the aggregate optimum is a compromise rather than anyone's best. Every node is re-scored per item and bootstrapped jointly ($n_{\text{boot}} = 4000$): se is the per-node standard error of Avg_7 (0.379–0.479 points) and Δ the paired difference from the plateau node λ=0.45 with its 95% interval. The top three nodes are mutually indistinguishable, so the maximum is a plateau rather than a point.

Table A6: **Qwen2-VL-7B: rule variants and re-weighting baselines (appendix).** The per-benchmark record behind the remaining Qwen2-VL rows of Table 2.

Method	VQA		Geometry		Chart	OCR		Avg_7	$\bar{a}$
	VizWiz	GQA	MVista	MVis	CQA	TVQA	OCR		
Constant λ (control)	41.09	61.70	56.90	21.71	78.00	81.13	61.00	57.36	0.3726
CFS-0	40.92	61.77	57.20	22.37	77.12	80.89	61.30	57.37	0.3726
Median-norm rule	40.90	61.81	57.10	21.38	76.96	81.12	61.26	57.22	0.3726
LiNeS β=0.80	32.22	60.11	47.30	13.16	55.80	59.80	52.99	45.91	0.6000
LiNeS β=1.00	31.04	59.24	45.80	14.14	51.92	58.26	49.45	44.26	0.7000
MetaGPT	38.51	60.09	52.70	20.07	26.44	65.47	57.00	45.75	0.2154

Column abbreviations: MVista = MathVista, MVis = MathVision, CQA = ChartQA, TVQA = TextVQA, OCR = OCRVQA. Per-benchmark scores for the Qwen2-VL-7B rows that appear only as Avg_7 in Tables 2 and 3: the constant-λ control and the two rule variants of Section 3, and the depth re-weighting baselines. Same evaluation protocol as Table 1; $\bar{a}$ is the plain layer mean of Equation 1.

the paired half-width at the nodes that matter for in-band verdicts is of the same order as the ± 1.0 decision band itself. The band therefore separates methods that differ by more than about a point, which is what we use it for, and it cannot adjudicate differences of a few tenths; the equivalence tests at a 0.25-point margin, which are paired against a specific reference rather than against the curve, remain the instrument for those.

G QWEN2-VL-7B: RULE VARIANTS AND RE-WEIGHTING BASELINES

Table A6 gives the per-benchmark scores of the Qwen2-VL-7B rows that Tables 2 and 3 report only as Avg_7: the constant-λ control, CFS-0 and the median-norm rule of Section 3, and the depth re-weighting baselines.

Parameter-free rule definitions. The median-norm rule sets $\lambda_\ell = \text{median}_i \|\tau_{i,\ell}\| / \|\Delta_\ell\|$ in every tensor, balancing the merged displacement against a typical expert; its mean scale $\bar{r}_2 = \text{mean}_\ell \, \lambda_\ell$ is 0.3726 on Qwen2-VL-7B and 0.3673 on InternVL2.5-1B. CFS-0 rescales the CFS profile of Equation 3 to that same mean, $\lambda_\ell = \bar{r}_2 \, r_\ell / \bar{r}$ with $\bar{r} = \text{mean}_\ell \, r_\ell$, so the two rules share $\bar{a}$ by construction and differ only in profile; the constant-λ control sets $\lambda_\ell = \bar{r}_2$ everywhere. All three derive their scale from the task vectors alone.

The three rule variants differ from one another by at most 0.15 points on Avg_7 and by less than 1.1 points on any single benchmark. The two LiNeS schedules and MetaGPT lose most of their Avg_7 deficit on ChartQA, where they score between 26 and 56 against 74 to 80 for the rows of Table 1 that sit on the curve.

H INTERNVL2.5-1B: FULL TABLE

Table A7 gives the complete InternVL2.5-1B record: the same nine merging methods as Table 1 at their default coefficients, the reference curve, the three projections, the closed-form rule, the depth re-weighting baselines, and the five experts with the base. The ten task-arithmetic rows double as the numerical form of the reference curve for that architecture, whose peak is 47.99 at $\lambda = 0.050$, with a secondary local maximum of 47.78 at $\lambda = 0.15$. The three nodes at $\lambda = 0.55$, 0.60 and 0.70 extend the calibrated domain to $[0.025, 0.70]$ so that the two LiNeS schedules, whose $\bar{a}$ of 0.593 and 0.691 lie beyond the default coefficients, are read against constant-scale task arithmetic at their own scale in Table 2; methods whose $\bar{a}$ falls outside the domain are marked OUT-OF-DOMAIN rather than scored against an extrapolation.

The orthogonal component on this architecture is read off the same way as on Qwen2-VL, as the unprojected method minus its projection under the paired bootstrap of Appendix A: $+0.60$ $[-0.06, +1.25]$ for OptMerge at $\lambda = 0.1$ and $+2.26$ $[+1.51, +3.02]$ for Iso-C at $\lambda = 0.3$, the two methods whose unprojected and projected merges share one coefficient. Both projections land inside the band (Table 2), so the scale axis accounts for the projected merge in each case and the orthogonal component enters as an additive term on top of it, as on Qwen2-VL (Appendix E). The two architectures differ on Iso-C for a reason the geometry makes plain. Iso-C replaces every singular value of the summed task vector by their mean. On Qwen2-VL the five rank-8 experts give a sum of rank at most 40 per matrix, so flattening the spectrum pours most of the energy into directions the experts never touched: 13.5% of the displacement stays parallel to the sum (Figure 2a) and the merge collapses to 17.98. The full fine-tuned InternVL2.5 experts give a full-rank sum, the flattened spectrum stays close to it (78.7% parallel, Table A9), and the orthogonal remainder is small and positive.

I INTERNVL2.5-1B: SCALE- AND SHAPE-CONTROLLED ROWS

Table A8 records the rows that isolate scale from shape on this architecture, after the closed-form rule of Section 5.4: a constant-λ merge at each of the two rule scales, the parameter-free median-norm rule, norm-matched random directions orthogonal to the task-vector sum, and the additivity row that adds OptMerge$\|$'s own orthogonal residual to a task-arithmetic merge. All eight are our own runs under the evaluation protocol of Appendix A.

The pattern is the same as on Qwen2-VL, and it is scale rather than shape. At the recalibrated rule's mean scale ($\bar{a} = 0.0720$) the constant-λ control scores 47.63, within 0.03 of the curve and within 0.18 of the rule itself: within resolution, the depth profile adds nothing. The three random-$\perp$ directions, which share the mean scale of OptMerge$\|$ by construction, land at 47.47–47.78, all inside the band and within 0.19 of the curve. Increasing the mean scale to the $\bar{a} = 0.3673$ shared by CFS-0 moves all three rows with that scale (CFS-0, the median-norm rule and the corresponding constant-λ control) to 44.74–44.77, on the curve at that scale ($\mathrm{curve}(0.3673) = 44.78$), and the median-norm rule again matches its constant-λ counterpart to within 0.06 (Table A8). What sets the score is where the merge lands on the axis; among the profiles we tested, the one that reaches that point does not change it.

The orthogonal component has the same sign on this architecture (Table A9). Unprojected OptMerge sits $+0.60$ above its projection ($[-0.06, +1.25]$) and WUDI $+0.18$ ($[-0.46, +0.80]$); these direct contrasts and the interaction row ($+0.17$, $[-0.45, +0.81]$) have intervals that include zero. The resolved evidence is the additivity row: adding the optimizer's actual orthogonal residual to a constant $\lambda = 0.075$ merge beats task arithmetic by $+0.77$ ($[+0.08, +1.47]$), while the three random directions of the same norm, which by construction carry no direction from the optimizer, sit at or below the curve (Table A8). The gain belongs to the optimizer's direction, not to the norm it adds. As on Qwen2-VL, the effect is small relative to scale: at 0.2–0.8 points it is of the same order as the ± 1.0 band. The paired intervals are about $2.5\times$ as wide here as on Qwen2-VL (paired standard error 0.34 against 0.14 for the OptMerge contrast, Tables A9 and A4): the two merges disagree on more items on this architecture, which is why the same-signed OptMerge contrast reads DIFFERS on Qwen2-VL and INCONCLUSIVE here.

Table A7: **InternVL2.5-1B, same seven benchmarks.** Baselines, the reference curve, the projections, the closed-form rule and the depth re-weighting baselines under one protocol: all three projections land inside the band; the depth re-weighting baselines are read against the extended curve in Table 2.

Method	VQA		Geometry		Chart	OCR		Avg_7
	VizWiz	GQA	MVista	MVis	CQA	TVQA	OCR	
Weight Avg.	30.81	56.14	42.70	**18.42**	65.52	74.08	42.61	47.18
Task Arithmetic	30.03	54.14	39.80	17.11	67.48	72.13	40.46	45.88
TIES	31.40	<u>57.60</u>	42.90	14.14	64.20	75.72	44.76	47.25
DARE-TA	29.88	54.17	40.50	16.78	67.08	72.07	40.36	45.83
DARE-TIES	31.27	57.43	**45.10**	17.76	62.96	75.41	44.63	47.80
TSV-M	31.34	57.14	43.80	15.13	69.64	75.99	<u>45.83</u>	48.41
Iso-C	<u>31.63</u>	57.42	43.40	17.11	69.12	75.81	**49.28**	**49.11**
WUDI	31.45	57.06	42.30	15.79	68.72	75.59	44.86	47.97
OptMerge	31.06	57.11	44.40	14.80	<u>70.60</u>	76.02	45.77	<u>48.54</u>
TA λ=0.025	30.23	56.27	40.20	14.80	**71.24**	75.68	44.86	47.61
TA λ=0.05	30.91	56.92	44.10	14.14	68.96	75.98	44.89	47.99
TA λ=0.075	31.30	57.32	43.70	13.49	66.52	<u>76.17</u>	44.82	47.62
TA λ=0.10	31.33	57.22	43.90	13.49	65.12	76.15	44.08	47.33
TA λ=0.15	31.38	56.97	<u>45.10</u>	<u>18.42</u>	64.20	75.12	43.23	47.78
TA λ=0.20	30.76	56.12	42.70	18.09	65.60	74.06	42.64	47.14
TA λ=0.55	28.05	49.40	30.30	15.13	65.88	66.61	37.14	41.79
TA λ=0.60	27.59	48.43	28.90	13.82	64.92	65.36	36.82	40.83
TA λ=0.70	27.05	47.03	25.30	10.53	63.08	62.53	35.32	38.69
OptMerge$^{\parallel}$	31.26	57.35	43.70	14.80	67.40	76.08	44.99	47.94
Iso-C$^{\parallel}$	31.04	55.64	42.70	15.79	66.72	73.82	42.22	46.85
WUDI$^{\parallel}$	31.03	57.20	44.40	13.49	68.12	76.03	44.86	47.87
CFS (c recalibrated)	31.25	57.30	44.10	14.14	66.72	**76.19**	44.95	47.81
CFS (c from Qwen)	29.73	53.20	39.00	16.78	67.80	71.28	39.91	45.39
CFS-0	29.42	52.72	37.90	15.13	67.60	70.79	39.65	44.74
LiNeS β=0.80	28.48	51.32	35.50	15.46	65.08	63.81	35.06	42.10
LiNeS β=1.00	27.22	49.71	31.90	15.46	62.08	59.65	32.85	39.84
MetaGPT	**32.12**	**58.74**	42.20	15.79	50.56	73.34	41.44	44.88
Expert: VQA	30.25	60.88	31.40	15.46	48.68	63.18	35.16	40.71
Expert: Geometry	13.55	32.74	27.10	14.80	51.88	56.88	35.25	33.17
Expert: Chart	20.06	40.48	33.00	15.79	69.72	54.47	34.99	38.36
Expert: OCR	12.55	22.44	27.40	12.50	39.60	72.91	54.79	34.60
Expert: Grounding	18.46	0.00	19.20	14.80	42.92	49.78	0.88	20.86
InternVL2.5-1B (base, no merge)	28.86	54.88	39.60	12.83	71.04	74.16	40.85	46.03

Column abbreviations: MVista = MathVista, MVis = MathVision, CQA = ChartQA, TVQA = TextVQA, OCR = OCRVQA. First block: same methods as Table 1 at λ=0.30 (eight baselines) and λ=0.10 (OptMerge); projections from these. Ten task-arithmetic rows (the λ=0.30 row of the first block and the nine TA λ rows) form the reference curve of Fig. A1(c), peak 47.99 at λ=0.050. Decision statistics: $\text{se}^{\star}$=0.413, n_{boot}=2000, seed 20260818. Bold/underline mark best/second over merging rows. Shaded rows: CFS (c recalibrated) re-fits c here, CFS (c from Qwen) reuses Qwen2-VL constant, CFS-0 is parameter-free. Grounding expert's skill (referring expression comprehension) not among benchmarks.

J REFCOCO REFERRING EXPRESSION COMPREHENSION

Table A10 reports the referring expression comprehension result (Precision@1 at IoU 0.5) on the RefCOCO family for the re-merged Stage-II models of Table A8 and the first-round models of Table A7, using the InternVL native evaluator. This benchmark is the Grounding expert's own skill and is not among the seven of Avg_7, so it is an out-of-domain test of the account rather than a place where the reference curve was fitted. The projected rows that keep the grounding expert (OptMerge$\|$ at 73.66 and WUDI$\|$ at 73.58) are within its spread of the projected optimizer's unprojected score (73.39), so the projection surgery preserves the expert's capability. The WUDI rows separate scale from projection directly: unprojected WUDI at its default $\lambda = 0.30$ scores 56.98, while the same unprojected method at $\lambda = 0.10$ scores 73.39, within 0.19 of its projection WUDI$\|$ at the same coefficient. The 16-point drop belongs to the coefficient, not to the orthogonal component the projection removes. The scale collapse dominates again: the rule rows whose $\bar{a}$ sat at 0.367 in Table A8 (CFS-0 at 38.53 and the c-from-Qwen variant at 43.12) fall to the same low scores, far below the 71.15 of the unmerged base model, exactly as they fall together on Avg_7. The base model already grounds at 71.15 and the grounding expert adds 1.6 points on top, so a merge that keeps its scale small inherits the base's capability and the projected rows keep the expert's increment as well; the fall to 38–43 at $\bar{a} \approx 0.37$ is the other experts' task vectors, applied at that scale, destroying a

Table A8: **InternVL2.5-1B: scale- and shape-controlled rows (appendix).** At matched mean scale the constant-λ control reproduces the recalibrated rule, and nothing about the depth profile moves Avg_7 beyond resolution: mean scale, not profile, sets the score.

Model	Benchmark							Avg_7	$\bar{a}$	Δ	Verdict
	VizWiz	GQA	MVista	MVis	CQA	TVQA	OCR				
References (Table A7)											
OptMerge$^{\parallel}$	31.26	57.35	43.70	14.80	67.40	76.08	44.99	47.94	—	—	—
OptMerge	31.06	57.11	44.40	14.80	70.60	76.02	45.77	48.54	—	—	—
Scale- and shape-controlled rows (this run)											
WUDI (λ=0.1, unprojected)	30.98	57.08	44.50	11.51	70.84	75.95	45.51	48.05	0.0665	+0.31	IN-BAND
Constant λ=0.0720 (scale of CFS recalc.)	31.26	57.31	44.10	13.16	66.52	76.10	44.99	47.63	0.0720	-0.03	IN-BAND
Constant λ=0.3673 (scale of CFS-0)	29.48	52.72	37.90	15.46	67.44	70.87	39.52	44.77	0.3673	-0.01	IN-BAND
Median-norm rule	29.47	52.90	38.10	15.13	67.00	70.84	39.52	44.71	0.3673	-0.07	IN-BAND
Random $\perp$ direction, draw 1	31.25	57.26	43.60	12.17	66.96	76.20	44.86	47.47	0.0720	-0.19	IN-BAND
Random $\perp$ direction, draw 2	31.18	57.21	43.60	14.47	66.96	76.12	44.92	47.78	0.0720	+0.12	IN-BAND
Random $\perp$ direction, draw 3	31.18	57.29	44.10	12.50	67.80	76.17	44.92	47.71	0.0720	+0.05	IN-BAND
TA@0.075 + OptMerge $\perp$ residual	31.15	57.12	43.50	14.80	70.04	76.14	45.93	48.38	0.0750	+0.76	IN-BAND

Column abbreviations: MVista = MathVista, MVis = MathVision, CQA = ChartQA, TVQA = TextVQA, OCR = OCRVQA. The shape- and scale-controlled InternVL2.5-1B rows added after Table A7, all evaluated under the same protocol. $\bar{a}$ is the plain layer mean of Equation 1; Δ is the measured Avg_7 minus $\text{curve}(\bar{a})$, and IN-BAND means the method is accounted for by where it lands on the reference curve (band ± 1.0 point). The calibrated domain is $[0.025, 0.70]$, so rows whose $\bar{a}$ falls outside it are marked OUT-OF-DOMAIN and get no prediction. The constant-λ rows share the mean scale of the recalibrated rule and of CFS-0 respectively; at matched mean scale the depth profile adds nothing within resolution, and the median-norm rule, CFS-0 and the constant row at the same (larger) scale land at the same Avg_7. Decision statistics as in Table A7 (n_{boot}=2000, seed 20260818, $\text{se}^{\star}$=0.413).

Table A9: **InternVL2.5-1B: noise floor and control merges (appendix).** The orthogonal component is a real effect on this architecture too, and random $\perp$ directions — unlike the residual of the optimizer — sit at or below $\text{curve}(\bar{a})$.

Model	Δ	se	95% CI	TOST @ 0.25	Qwen2-VL mirror		
					Δ	95% CI	TOST @ 0.25
Orthogonal component (unprojected − projected)							
OptMerge − OptMerge$^{\parallel}$	+0.598	0.338	$[-0.060, +1.246]$	INCONCLUSIVE	+0.294	$[+0.037, +0.564]$	DIFFERS
WUDI − WUDI$^{\parallel}$	+0.178	0.323	$[-0.459, +0.799]$	INCONCLUSIVE	+1.405	$[+0.740, +2.091]$	DIFFERS
Noise floor: norm-matched random $\perp$ directions							
Random $\perp$ direction, draw 1 − OptMerge$^{\parallel}$	-0.468	0.220	$[-0.911, -0.045]$	DIFFERS	—	—	—
Random $\perp$ direction, draw 2 − OptMerge$^{\parallel}$	-0.159	0.182	$[-0.518, +0.205]$	INCONCLUSIVE	—	—	—
Random $\perp$ direction, draw 3 − OptMerge$^{\parallel}$	-0.230	0.198	$[-0.639, +0.137]$	INCONCLUSIVE	—	—	—
Mean of the three draws − OptMerge$^{\parallel}$	-0.286	0.155	$[-0.613, +0.011]$	INCONCLUSIVE	+0.089	$[-0.037, +0.218]$	EQUIVALENT
Matched-scale profile controls							
Constant λ=0.0720 − OptMerge$^{\parallel}$	-0.306	0.174	$[-0.666, +0.013]$	INCONCLUSIVE	-0.136	$[-0.658, +0.409]$	INCONCLUSIVE
CFS (c recalibrated) − constant λ=0.0720	+0.175	0.185	$[-0.169, +0.559]$	INCONCLUSIVE	+0.006	$[-0.212, +0.219]$	EQUIVALENT
Median-norm rule − CFS-0	-0.034	0.098	$[-0.228, +0.152]$	EQUIVALENT	-0.280	$[-0.773, +0.219]$	INCONCLUSIVE
Additivity of the orthogonal residual							
TA@0.075 + res. − TA@0.075	+0.765	0.354	$[+0.084, +1.472]$	DIFFERS	+0.182	$[-0.094, +0.477]$	INCONCLUSIVE
as above − (OptMerge − OptMerge$^{\parallel}$)	+0.167	0.320	$[-0.452, +0.810]$	INCONCLUSIVE	-0.112	$[-0.483, +0.278]$	INCONCLUSIVE

InternVL2.5-1B counterpart of Table A4, from the same paired per-item bootstrap (n_{boot}=2000, seed 20260818). Random directions are orthogonal to the task-vector sum in every layer and scaled to the norm of the orthogonal residual of OptMerge$^{\parallel}$, so they match it in scale and differ only in the direction of the orthogonal part. The reference contrast OptMerge − OptMerge$^{\parallel}$ is the value of the orthogonal component; the additivity block adds that residual to a constant λ=0.075 merge (the node matching its scale) and tests whether it adds the same on top of task arithmetic as on top of the profile of the optimizer; the interaction row uses margin 0.3. The last three columns give the same contrast on Qwen2-VL for side-by-side reading, from the stored decision artefacts. INCONCLUSIVE is reported as such rather than read as evidence of equivalence. Parallel-energy fractions of the unprojected InternVL2.5-1B merges (Section 5.1): OptMerge 67.8%, WUDI 71.7%, Iso-C 78.7%; the experts here are full fine-tunes.

capability the base model had. Where the scale is preserved the grounded score is preserved; where the scale collapses the grounded score collapses with it.

K ViT-B/32: Full Table

Table A11 is the complete eight-task classification record: the zero-shot and fine-tuned gates, the twelve curve nodes defining $\text{curve}(\lambda)$ on $[0.05, 0.60]$, both unprojected methods at their best coefficient with their projections, four recipe and baseline rows, and all twelve depth-shape probes with paired intervals. This is the track on which the orthogonal component dominates, and we present it in full. Both projected methods fall far below the curve at their own measured $\bar{a}$, both

Table A10: **RefCOCO referring expression comprehension (appendix).** Scale-first holds on the grounding expert's own benchmark: the rows that keep the expert's scale keep its capability.

Model	RefCOCO family			Avg_3
	RefCOCO	RefCOCO+	RefCOCOg	
InternVL2.5-1B (base, no merge)	74.40	67.91	71.14	71.15
Expert: Chart	1.92	1.57	2.23	1.91
Expert: Geometry	24.57	19.53	23.73	22.61
Expert: Grounding	76.76	71.41	70.06	72.74
Expert: OCR	73.76	68.00	68.93	70.23
Expert: VQA	0.00	0.00	0.00	0.00
DARE-TA	51.50	44.17	54.02	49.90
DARE-TIES	72.12	64.99	69.71	68.94
Iso-C	48.99	44.03	46.20	46.41
OptMerge	76.00	69.81	74.37	73.39
CFS (c recalibrated)	76.33	69.73	74.26	73.44
CFS (c from Qwen)	43.87	37.75	47.73	43.12
CFS-0	39.48	33.42	42.69	38.53
Iso-C$^{\parallel}$	51.47	44.99	52.70	49.72
OptMerge$^{\parallel}$	76.41	69.94	74.63	73.66
WUDI$^{\parallel}$	76.19	70.26	74.29	73.58
TA @ λ=0.10	75.10	68.32	73.20	72.21
TA @ λ=0.025	75.42	69.22	73.41	72.68
TA @ λ=0.05	76.25	70.12	73.63	73.33
TA @ λ=0.075	76.25	69.70	74.41	73.45
TA @ λ=0.15	62.37	55.23	58.39	58.67
TA @ λ=0.20	54.38	46.89	53.45	51.57
TA @ λ=0.30	51.21	44.19	53.15	49.51
TIES @ λ=0.30	75.02	68.16	72.86	72.01
TSV-M	67.41	60.24	61.76	63.14
Weight Avg.	54.64	46.96	53.53	51.71
WUDI @ λ=0.10 (unprojected)	76.28	70.02	73.86	73.39
WUDI	61.30	53.92	55.72	56.98

Referring expression comprehension (Precision@1 at IoU 0.5, %) over the three RefCOCO variants, for the InternVL2.5-1B models of Tables A7 and A8 (28 models); bold marks the best Avg_3 among the merging rows (experts and the base model are references, not ranked). This benchmark is the grounding expert's own skill and is not among the seven of Avg_7, so it is an out-of-domain check on the claim: the projected rows that retain the grounding expert (OptMerge$^{\parallel}$ at 73.66, WUDI$^{\parallel}$ at 73.58) keep that expert's capability, while the two rows at the collapsed scale (CFS-0 and CFS (c from Qwen)) drop to 38.53 and 43.12.

LiNeS schedules sit about 11 points above it, and the orthogonal component, unprojected minus projected and computed from unrounded scores, accounts for $+21.07$ $[+20.64, +21.49]$ and $+47.05$ $[+46.59, +47.54]$ points for the two surgeries, against $+0.29$ to $+1.41$ for the two optimizers on the vision–language tracks.

The projected merges on this track also carry unusually dispersed per-layer profiles. Their energy-weighted scale is close to twice the plain layer mean (Table A2: 0.7072 against 0.3835 for Iso-C$^{\parallel}$, 0.4071 against 0.2283 for TSV-M$^{\parallel}$), and the curve is steep past its peak. Read at the weighted scale, TSV-M$^{\parallel}$ moves to within 2.75 points of the curve and Iso-C$^{\parallel}$ leaves the calibrated domain on the collapse side, where its low score belongs. A single layer mean summarizes a profile well when the profile is coherent, as it is on both vision–language tracks; on this track the dispersion of the profile is a second quantity, measurable before evaluation, that places the classification track at the edge of the account's scope.

The LiNeS residual is read against the curve, not against the peak. The increasing depth schedule at $\beta = 0.875$ sits $+11.03$ points above $\mathrm{curve}(\bar{a} = 0.540)$, a reference evaluated at a scale deep in the collapse region; its absolute score is 57.52, below our curve peak of 69.98, so the residual measures how much a deep-tilted profile recovers at large scale. At the anchor scale the deep-tilted probe reaches 72.98, above the curve peak; the residual and the absolute score answer different questions.

L Extended Related Work

Data-free merging. Weight averaging (Wortsman et al., 2022) and task arithmetic (Ilharco et al., 2023) established that a single scalar over a sum of task vectors is a serviceable merge, and the tangent-space analysis of Ortiz-Jimenez et al. (2023) gave the linearity conditions under which this should be expected to work. Sparsification methods reduce redundancy before summing, by trimming

Table A11: **ViT-B/32, eight classification tasks (appendix).** Every measured model behind Fig. A1(a) and (d).

Model	SUN	Cars	RES	Euro	SVHN	GTSRB	MNIST	DTD	Avg_8	$\bar{a}$ / $\hat{g}$	Δ [95% CI]
Zero-shot (no merge)	63.1	59.6	60.2	45.0	31.6	32.6	48.2	44.4	48.11	—	—
Fine-tuned (diagonal)	75.3	78.2	95.9	99.8	97.6	99.1	99.7	79.0	90.58	—	—
TA λ=0.05	64.3	62.0	66.7	61.3	46.7	41.0	69.8	47.5	57.41	—	—
TA λ=0.1	65.1	62.9	70.5	69.5	58.3	48.4	83.2	49.1	63.39	—	—
TA λ=0.15	64.9	63.0	72.4	73.0	66.0	56.9	89.8	51.0	67.13	—	—
TA λ=0.2	63.8	62.3	72.5	76.5	71.8	64.4	93.6	52.5	69.67	—	—
TA λ=0.25	60.4	59.7	70.4	77.4	75.8	68.5	95.9	51.5	69.98	—	—
TA λ=0.3	55.0	55.6	67.2	75.1	78.1	70.1	97.2	50.1	68.56	—	—
TA λ=0.35	46.6	49.8	62.0	70.7	78.9	69.8	97.7	46.8	65.29	—	—
TA λ=0.4	36.6	41.9	54.9	64.7	78.6	67.9	98.0	42.4	60.62	—	—
TA λ=0.45	26.2	33.6	46.4	58.7	77.6	64.2	97.9	38.9	55.45	—	—
TA λ=0.5	17.9	24.6	38.8	53.0	76.0	59.7	97.8	34.0	50.22	—	—
TA λ=0.55	12.1	17.1	31.9	47.5	73.7	54.0	97.5	30.2	45.49	—	—
TA λ=0.6	8.0	10.7	25.7	43.0	71.0	48.7	96.9	26.3	41.28	—	—
Iso-C ($\alpha^\star$=1.5, unprojected)	67.7	70.0	87.6	93.7	88.3	92.3	99.2	69.3	83.51	0.3835	—
Iso-C$^\parallel$	3.8	3.7	20.2	41.1	68.5	37.6	95.7	21.0	36.45	0.3835	—
TSV-M ($\alpha^\star$=1.0, unprojected)	67.2	70.8	86.1	94.3	91.0	92.3	99.3	68.7	83.71	0.2283	—
TSV-M$^\parallel$	32.0	44.0	56.0	70.1	83.6	74.1	98.4	43.0	62.64	0.2283	—
CFS (c recalibrated)	52.5	54.7	66.2	76.7	80.6	73.0	97.8	49.2	68.84	0.2500	—
LiNeS β=0.875	17.2	30.3	53.9	59.7	87.2	68.7	98.9	44.1	57.52	0.5395	—
LiNeS β=1.00	11.8	22.7	48.5	53.7	86.9	60.4	98.8	39.1	52.74	0.5987	—
MetaGPT	66.2	64.3	71.5	67.5	64.1	48.6	86.6	49.4	64.77	—	—
deep-tilted @ A=0.05	60.4	61.7	71.5	81.0	77.7	70.1	96.5	53.0	71.49	+0.382	+1.51 $[+1.35, +1.68]$
shallow-tilted @ A=0.05	60.0	58.0	69.3	73.1	73.3	66.0	94.9	50.0	68.10	-0.382	-1.88 $[-2.06, -1.70]$
block-permuted ($\hat{g}$<0) @ A=0.05	60.4	59.1	70.0	72.7	72.9	67.2	94.4	51.6	68.55	-0.180	-1.43 $[-1.59, -1.27]$
block-permuted ($\hat{g}$>0) @ A=0.05	60.3	60.2	70.4	79.3	77.9	68.9	96.6	51.8	70.68	+0.149	+0.71 $[+0.57, +0.84]$
deep-tilted @ A=0.15	59.3	62.8	72.9	86.4	79.1	71.1	97.0	55.2	72.98	+1.144	+3.00 $[+2.73, +3.25]$
shallow-tilted @ A=0.15	58.2	52.8	64.3	64.1	67.0	58.1	92.4	46.9	62.97	-1.144	-7.01 $[-7.32, -6.72]$
block-permuted ($\hat{g}$<0) @ A=0.15	59.7	56.6	67.2	61.6	65.6	62.5	90.3	49.3	64.10	-0.540	-5.88 $[-6.17, -5.59]$
block-permuted ($\hat{g}$>0) @ A=0.15	59.7	60.1	70.2	82.7	80.6	68.5	97.4	50.9	71.26	+0.449	+1.28 $[+1.06, +1.49]$
deep-tilted @ A=0.25	56.3	61.1	71.6	88.0	76.2	68.6	96.5	55.2	71.70	+1.907	+1.72 $[+1.40, +2.03]$
shallow-tilted @ A=0.25	54.3	47.2	57.2	54.7	58.0	50.5	87.6	41.9	56.42	-1.907	-13.56 $[-13.95, -13.19]$
block-permuted ($\hat{g}$<0) @ A=0.25	57.3	52.8	61.6	48.6	56.8	55.8	83.4	46.4	57.83	-0.899	-12.15 $[-12.52, -11.78]$
block-permuted ($\hat{g}$>0) @ A=0.25	58.1	58.5	69.0	82.9	81.5	67.0	97.8	49.0	70.46	+0.747	+0.48 $[+0.21, +0.74]$

Task abbreviations: SUN = SUN397, RES = RESISC45, Euro = EuroSAT. Complete ViT-B/32 record: the two gate rows, the 12 curve nodes that define $\text{curve}(\lambda)$ on $[0.05, 0.60]$, the two unprojected parents at their best coefficient $\alpha^\star$ with their projections ($^\parallel$), four recipes/baselines, and all 12 depth-shape probes. $\bar{a}$ is the effective scale (plain layer mean) of a merge and, for an unprojected parent, that of its parallel component, which is the row beneath it; for a probe the column gives its realized signed depth slope $\hat{g}$ (Equation 2), the nominal amplitude A being named in the row and $\bar{a}$ pinned to the anchor 0.25. Δ is given for probes: score minus the anchor TA λ=0.25 (69.98), with 95% paired per-item bootstrap intervals (n_{boot}=4000, seed 20260823). Under the same bootstrap the parent−projection differences are +47.05 $[+46.59, +47.54]$ for Iso-C and +21.07 $[+20.64, +21.49]$ for TSV-M. Decision statistics for this architecture use $\text{se}^\star$=0.2511. MetaGPT re-weights experts rather than layers; its $\bar{a}$ is that of its parallel component, which was not recorded on this track. Block-permuted probes are listed by the sign of their realized slope.

and sign election (Yadav et al., 2023) or by random drops with rescaling (Yu et al., 2024). We evaluate both under our protocol (Table 1) but do not place them on the reference curve.

Spectral and optimization-based merging. A second line treats the task vectors as matrices and edits their spectra, whether by keeping task-specific singular directions (Gargiulo et al., 2025), equalizing the spectrum over common and task-specific subspaces (Marczak et al., 2025), or aligning bases before merging (Stoica et al., 2025). A third optimizes the merged vector against an interference objective without labels (Cheng et al., 2025; Wei et al., 2025; 2026). Both lines are usually evaluated by end score; we instead measure the displacement each produces and ask how much of its score is set by scale alone, which also locates the remaining value: for optimization-based merging, a small orthogonal component rather than a better profile. Adaptive schemes using unlabeled test data (Yang et al., 2024) sit outside our data-free scope.

Layer-dependent scaling. Several methods make the merging coefficient depend on depth, either by a fixed increasing schedule to counter forgetting (Wang et al., 2025) or by a closed-form per-layer solution (Zhou et al., 2024). This family is exactly the one our two axes describe, since its displacement is a scale profile by construction and its shape effect is therefore measurable in isolation. On our multimodal tracks these schedules land below the reference curve at their own measured scale, which we report as a dominance result rather than as a general judgment: the same increasing schedule is a gain on the classification track, and the level of the deep-tilted branch is architecture-dependent.

Rank of the task-vector space versus magnitude of the displacement. A separate line studies the *spectrum* of the merged task-vector space. Skorobogat et al. (2025) argue that as more experts are merged the shared component comes to dominate the task-specific one, driving what they call rank collapse, and recover accuracy by restoring rank in SVD space. Because both that account and ours predict merged performance from displacement geometry, it is necessary to say precisely how they differ, and why one does not reduce to the other.

The quantities are different in kind. Rank collapse is a statement about how energy is distributed across singular directions, which is irreducibly multi-dimensional; $\bar{a}$ is a single magnitude ratio and says nothing about how the displacement is spread across directions. Two merges with identical $\bar{a}$ can have entirely different spectra. So neither claim implies the other, and the question is whether the spectrum is what our predictions are secretly tracking.

Our measurements separate the two, because the methods we predict span a wide range of spectral concentration. The optimization-based merge sits 90.7% parallel in energy to the naive task-vector sum, close to the dominant-shared-direction regime that rank collapse describes. Isotropic rescaling is the opposite case by construction: it deliberately flattens the spectrum and retains only 13.5% parallel energy before projection, a factor of 6.7 less. The scalar account predicts the along-sum component of both. Evaluated at its own measured $\bar{a}$, the projected isotropic surgery lands at $+0.77$ points on Qwen2-VL and -0.15 on InternVL2.5, both inside the band, even though the profile it projects was produced by a spectral procedure and sits at a very different scale from the optimizer's. What the spectral edit changes beyond that profile is the orthogonal component, and the projection measures it directly: -28.85 points on Qwen2-VL, where the curve recovers a collapsed score from 17.98 to 46.83, $+2.26$ on InternVL2.5, and $+47.05$ on the classification track. Spectral structure therefore enters the account through one measured term, while the scale term is read from the along-sum profile alone.

Two limits on this cut apply. First, rank collapse is a claim about growth in the number of experts, tested up to twenty; our expert count is fixed within each track (five on the multimodal tracks, eight on classification), so we do not test it and our results neither confirm nor contradict it. Spectral structure may still matter for performance in general, and the literature does not agree on its sign: Choi et al. (2024) report that task-specific knowledge concentrates in the top singular directions and that keeping them helps, while Lee et al. (2026) report that dominant components are a principal source of interference and prune them. Our own claim is narrower: whatever the spectrum does, a single magnitude already accounts for benchmark averages to within a point on the tracks we calibrated, so a spectral explanation is not needed to predict them.

Merging at scale in the number of experts. Yadav et al. (2025) study how merging behaves as model size and expert count grow, finding that larger base models merge more readily and tolerate more experts, and that different merging methods converge in behaviour at scale. That study is empirical about scale factors rather than about displacement geometry, and it varies the expert count, which we hold fixed within each track. It bounds our scope: the one-dimensional account is calibrated at a fixed expert count per track, and how $\bar{a}$ behaves as experts accumulate is open.

Merging for multimodal models. Benchmarks for merging multimodal models are recent: Sung et al. (2023) merged modality-specific models, Du et al. (2025) addressed heterogeneous pairs through unsupervised coefficient selection, Qu et al. (2025) used prediction uncertainty to order merges, and Wei et al. (2026) introduced the capability-partitioned benchmark with expert checkpoints whose protocol family our multimodal tracks follow; broader fusion benchmarks (Tang et al., 2025) cover our third track. Relative to these, our object of study is not another method's score but the mapping from displacement geometry to score, which is what lets a prediction be made before evaluation.