COIN: COORDINATING MULTIMODAL UPDATES THROUGH FUNCTIONAL INVARIANCE

Anonymous authors
Paper under double-blind review

ABSTRACT

Multimodal learning must not only train each modality well, but also coordinate how their updates affect the shared prediction. Joint training can underuse informative modalities, while improving each modality separately does not ensure that their updates work well together after fusion. We show that module-wise signals can be misleading: coordinate rescaling can create arbitrarily different learning speeds, and large module responses can cancel while producing little or no fused-output change. To study this interaction directly, we characterize each module by the first-order change it can induce in the fused output and identify a *functional invariant relation*: different module responses can produce the same combined output change. Among such equivalent responses, the minimum-energy allocation removes unnecessary module movement without changing the prediction to first order. Building on this principle, we derive **COIN**, which jointly chooses a task-improving output change and how to distribute it across modules. COIN accounts for both each module's relation to the task and its interaction with other modules through a small linear system. This view also explains why a module that cannot improve the task on its own may still help by correcting another module's response. We further bound total module response and derive the exact cost of ignoring cross-module interactions. Across evaluated benchmarks, COIN achieves the highest mean fusion accuracy among the compared methods.

1 INTRODUCTION

Multimodal learning requires each modality to develop useful information while ultimately contributing to a shared prediction. These goals need not align during training. Modalities can differ substantially in signal strength, learning difficulty, and optimization dynamics, so one modality may reduce the shared loss before another has learned useful features. Modality competition and prolonged unimodal learning phases provide concrete accounts of this behavior (Wang et al., 2020; Huang et al., 2022; Zhang et al., 2024b). Independent supervision can preserve modality-specific learning signals, but it leaves a separate question unresolved: once useful module updates are available, how should their effects be coordinated through the fused prediction (Hua et al., 2024; Wei et al., 2025b)?

Existing approaches address underlearning in several ways. Some adjust modality-dependent gradients or information acquisition (Peng et al., 2022; Li et al., 2023; Huang et al., 2025); others reconcile unimodal and multimodal objectives (Wei & Hu, 2024; Wei et al., 2025a), or separate learning paths and timescales (Zhang et al., 2024a; Wei et al., 2025b; Feng et al., 2026). These interventions address different causes of underlearning. Here, we focus on a complementary question: *how should the available module updates act together on the fused prediction?* A module's learning speed or individual progress does not determine whether the change it induces in the fused output reinforces, overlaps with, or counteracts the change induced by another module. Even updates to disjoint parameter blocks can therefore interact through their effects on the same prediction.

A simple example shows why these interactions matter. Suppose two modules can each induce a first-order change in the *fused output* along the same nonzero direction u. They can produce κu and $-\kappa u$ for any scale $\kappa > 0$. Then

$$\underbrace{\kappa u + (-\kappa u)}_{\text{combined response}} = 0, \qquad \underbrace{\|\kappa u\|_2^2 + \|-\kappa u\|_2^2}_{\text{total squared response magnitude}} = 2\kappa^2 \|u\|_2^2 \xrightarrow[\kappa\to\infty]{} \infty. \tag{1}$$

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.

Thus, response magnitude alone does not reveal how module responses combine. Sec. 2 shows that the same issue persists under near cancellation. It also establishes a separate limitation of learning-speed comparisons: changing parameter units can make the apparent speed ratio arbitrarily large even when the underlying prediction problem and modality information are unchanged. Together, these results show that neither individual response magnitude nor relative learning speed is sufficient to determine how module updates should be coordinated.

This suggests a different starting point: compare module updates by what they do to the fused prediction. We call module m's first-order output change its *response*, v_m. An *allocation* specifies the responses $(v_1, \ldots, v_M)$, and their sum $\delta = \sum_m v_m$ is the *combined response*. The prediction depends on this sum, whereas an update rule must choose the individual responses. This distinction leads to our central question: *which differences between allocations change the combined response, and which leave it unchanged?* Two allocations have the same sum exactly when their difference cancels across modules. We call this output-preserving equivalence the *functional invariant relation*. For each achievable combined response, we identify the unique allocation with the smallest total squared response magnitude. Every other allocation producing that response adds a zero-sum redistribution. The relation therefore lets us distinguish the movement needed to produce a chosen first-order prediction change from extra module movement that adds nothing to it.

This principle determines how a given combined response should be allocated, but not which combined response should be pursued for the current task. From these two requirements, we derive COIN (**CO**ordinating Multimodal Updates through Functional **IN**variance). COIN jointly chooses a task-improving combined response and how that response should be distributed across modules, while discouraging unnecessary module movement. The resulting rule accounts for both how each module can respond to the task and how its response interacts with those of other modules, while remaining invariant to basis choice within fixed response spaces.

This joint view also reveals behavior that cannot be inferred from a module in isolation. A module that cannot improve the task on its own may still need to update if it can correct an unwanted component of another module's response. Whether a module should participate therefore depends not only on what it can do for the task by itself, but also on how its response interacts with those of other modules. We discuss existing approaches and compare them with COIN in Appendix A.

Contributions. We show that module-wise signals are unreliable guides to coordination: rescaling coordinates creates arbitrarily large apparent speed disparities, and large responses can hide behind a small combined response (Sec. 2). We characterize the output-preserving functional invariant relation and use its minimum-energy decomposition to derive COIN, a basis-invariant joint allocation rule. We derive a bound on module energy, an exact cost for ignoring cross-module interactions, and conditions under which an individually inactive module still improves the combined response (Sec. 3–4). To our knowledge, COIN is the first multimodal update formulation derived from the output-preserving relation. Across four benchmarks, COIN achieves the highest accuracy (Sec. 6).

2 What prevents reliable module coordination?

Problem setup and notation. Fix a network state θ and a training batch $\mathcal{B}$. Let θ_m denote one of M participating parameter blocks, such as an encoder or a fusion layer, and let f_θ denote the vector of fused output scores on $\mathcal{B}$. Let $L(\theta; \mathcal{B})$ denote the fused-task loss on this batch. We use $\nabla_f L(\theta; \mathcal{B})$ to denote the gradient of this loss with respect to the fused output f_θ. For a parameter direction d_m of module m, let $J_m = \frac{\partial f_\theta}{\partial \theta_m}$ be the Jacobian of the fused output with respect to that module's parameters. The corresponding first-order change in the fused output is $v_m = J_m d_m$, which we call the module's *response*. For a joint direction $d = (d_1, \ldots, d_M)$ and a small step τ,

$$f_{\theta+\tau d} - f_\theta = \tau\delta + o(\tau), \qquad \delta = \sum_m v_m. \tag{2}$$

Thus, δ is the combined first-order change in the fused output. The module responses lie in the same output space even when the corresponding parameter blocks have different dimensions; the analysis therefore does not require an additive fusion architecture. For classification, we use centered, sample-normalized logits as the output representation (Appendix B.1).

To describe what each module can change locally, let $\mathcal{D}_m$ be a linear subspace of allowed parameter directions for module m. Its image under the Jacobian, $\mathcal{S}_m = J_m \mathcal{D}_m$, is the set of first-order

fused-output changes attainable by that module; we call $\mathcal{S}_m$ its *response space*. The fused-task loss defines a *task direction* $r = -\nabla_f L(\theta; \mathcal{B})$. For any output direction z, $\langle \nabla_f L, z \rangle = -\langle r, z \rangle$. Hence, $\langle r, z \rangle > 0$ means that moving infinitesimally along z decreases the fused-task loss.

Research question 1: what can module-local quantities fail to reveal?

Can a module's learning speed or individual response magnitude determine how much useful learning remains, or how its response will interact with those of other modules?

Coordinate choice can create unbounded speed disparity. Consider two features x_1, x_2 satisfying $\mathbb{E}[x_1^2] = \mathbb{E}[x_2^2] = 1, \mathbb{E}[x_1 x_2] = 0$, with target $y = x_1 + x_2$. Consider the predictor $f_\psi^{(c)} = c\psi_1 x_1 + c^{-1}\psi_2 x_2$ where $\psi = (\psi_1, \psi_2)$ is the parameter vector and $c > 0$ is a scale. Changing c changes only the parameterization: defining $w_1 = c\psi_1$ and $w_2 = c^{-1}\psi_2$ gives the same function class $w_1 x_1 + w_2 x_2$. In particular, at $\psi = 0$, every value of c gives the same prediction and loss.

Proposition 2.1 (Unbounded coordinate-dependent speed disparity)**.** *For the squared loss $L(\psi) = \frac{1}{2}\mathbb{E}\left[(f_\psi^{(c)} - y)^2\right]$, consider gradient flow $\frac{d\psi}{dt} = -\eta \nabla_\psi L(\psi)$, where t denotes optimization time and $\eta > 0$ is a common learning-rate scale. At $\psi = 0$,*

$$\frac{d}{dt}(c\psi_1 x_1) = \eta c^2 x_1, \quad \frac{d}{dt}(c^{-1}\psi_2 x_2) \quad = \eta c^{-2} x_2, \qquad \frac{\left\| \frac{d}{dt}(c\psi_1 x_1) \right\|}{\left\| \frac{d}{dt}(c^{-1}\psi_2 x_2) \right\|} = c^4. \tag{3}$$

The ratio can therefore be made arbitrarily large by changing only the parameter coordinates, while leaving the underlying prediction problem unchanged.

Changing learning rate η cannot remove this disparity because it multiplies both prediction velocities by the same factor. Yet the response spaces remain $\mathrm{span}(x_1)$ and $\mathrm{span}(x_2)$ for every c. What a module *can change* is therefore distinct from how fast its current coordinates move (Appendix C.1).

Early fitting can remove another module's learning signal. The previous example shows that relative learning speed is not intrinsic to the module. Speed differences can nevertheless matter when multiple modules rely on the same fused-task loss. By the chain rule,

$$\nabla_{\theta_m} L(\theta; \mathcal{B}) = J_m^\top \nabla_f L(\theta; \mathcal{B}) = -J_m^\top r, \qquad \|\nabla_{\theta_m} L(\theta; \mathcal{B})\| \le \|J_m\|_2 \|r\|. \tag{4}$$

Here $\|J_m\|_2$ is the induced operator norm of the Jacobian: the largest fused-output change produced by a unit parameter direction. If the Jacobians remain bounded, a small output-space loss gradient $\|r\|$ implies little gradient signal from the fused-task loss for every module. At $r = 0$, all parameter gradients induced by that loss vanish.

Consequently, if one module drives the fused-task objective to a state with a small output gradient before another has developed useful features, the latter may be left with little shared-loss signal from which to learn. In the example above, fitting x_1 still leaves the orthogonal target component x_2, so a learning signal remains. In contrast, Appendix G gives a shared-target example in which the loss tends to zero while a weak branch's contribution remains arbitrarily small. This distinction is consistent with existing analyses of gradient starvation and modality competition (Pezeshki et al., 2021; Huang et al., 2022; Zhang et al., 2024b). Independent supervision can preserve module-specific learning signals; a separate question is how the resulting available responses should interact.

A small combined response can hide large module responses. Even when useful learning signals remain available, the resulting module responses can cancel in the fused output. Eq. (1) gives exact cancellation. The same issue persists under near cancellation. Let $u_1 \in \mathcal{S}_1$ and $u_2 \in \mathcal{S}_2$ be unit response directions with inner product $\langle u_1, u_2 \rangle = \rho \in (0, 1)$. For any scale $\kappa > 0$, consider $v_1 = \kappa u_1, v_2 = -\kappa u_2$. Then $\frac{\|v_1\|^2 + \|v_2\|^2}{\|v_1 + v_2\|^2} = \frac{1}{1-\rho} \xrightarrow[\rho \uparrow 1]{} \infty$. As the two response directions become nearly aligned, oppositely signed responses can therefore produce a small combined response despite large individual changes. Equivalently, if the combined response is normalized to unit norm, the total squared module-response magnitude is $1/(1-\rho)$ and can grow without bound. These are possible response combinations within the current response spaces; the result does not claim that every optimizer follows such a trajectory (Appendix C.2).

Finding 1: signals must remain available, and responses must work together

The fused-task gradient may shrink before every module has learned useful features. Even when useful learning signals remain available, large module responses can partially or completely cancel. Preserving learning signals and coordinating the resulting output changes are distinct requirements.

We next address the second requirement: which differences between available module responses change the fused prediction, and which merely redistribute change across modules?

3 THE INVARIANT RELATION BETWEEN MODULE UPDATES

Different module changes can have the same total effect. Keep the response spaces $\mathcal{S}_m$ fixed at the current network state and batch. Consider an allocation $(v_1, \dots, v_M)$ with $v_m \in \mathcal{S}_m$ and combined response $\delta = \sum_m v_m$; its existence makes δ *achievable*. Different allocations can produce the same δ. We first hold this combined response fixed and ask how the individual responses can change. This separates choosing an output effect from choosing which modules produce it. Let $q_m \in \mathcal{S}_m$ adjust module m's response. Since $\mathcal{S}_m$ is linear, $v_m + q_m$ remains attainable.

Research question 2: what can change between modules without changing the prediction?

Which adjustments to the module responses leave their combined first-order effect unchanged?

Functional invariant relation

For response adjustments $q_m \in \mathcal{S}_m$, the combined response remains δ exactly when the adjustments sum to zero:

$$\sum_m (v_m + q_m) = \delta \quad \Longleftrightarrow \quad \sum_m q_m = 0. \tag{5}$$

Individual module responses can change while their combined response remains the same. We call this output-preserving equivalence between allocations the *functional invariant relation*.

This relation identifies the freedom behind the cancellation in Eq. (1): module responses can change without changing their sum. It is local to the current network state and batch, since the response spaces may change during training. Within fixed response spaces, it does not depend on the particular bases used to represent them. The relation identifies which allocations have the same effect; to choose among them, we must also measure how much each module moves.

For the same effect, remove movement that does nothing. At first order, the fused prediction cannot distinguish allocations with the same combined response. Among them, we prefer the one with the smallest *total squared output movement*, $\sum_m \|v_m\|^2$. We call this quantity the *module energy* of the allocation; it measures prediction changes, not parameter norms. For a fixed achievable δ, the next result shows that this criterion selects a unique allocation and characterizes every other allocation with the same response.

Theorem 3.1 (Minimum-energy decomposition)**.** *Fix any achievable combined response δ. There is a unique allocation $(v_1^{\min}, \dots, v_M^{\min})$ minimizing the module energy. Every other allocation with the same δ can be written as $v_m^{\min} + q_m$, with $q_m \in \mathcal{S}_m$ satisfying $\sum_m q_m = 0$, and*

$$\sum_m \|v_m^{\min} + q_m\|^2 = \sum_m \|v_m^{\min}\|^2 + \sum_m \|q_m\|^2. \tag{6}$$

In words: *for a fixed combined response, any allocation's energy equals the least energy needed for that response, plus extra movement that cancels out*. Dropping the q_m preserves the combined response and removes exactly that extra energy. Because the construction uses only the response spaces and the output-space norm, it is unchanged by choosing different bases for the same $\mathcal{S}_m$. Appendix D.1 provides the proof and an equivalent expression using the pseudoinverse.

Finding 2: preserve the useful effect, remove unnecessary movement

The invariant relation identifies which reallocations leave the combined response unchanged. Choosing the least-motion allocation removes these redundant changes without altering the combined response.

4 FROM THE INVARIANT RELATION TO COIN

The previous section determines how to allocate a given combined response without unnecessary module movement, but not which combined response should be pursued for the current task. We now address both choices jointly and derive COIN.

4.1 FIT THE TASK WHILE LIMITING TOTAL MODULE MOVEMENT

Recall that the task direction $r = -\nabla_f L(\theta; \mathcal{B})$ points toward first-order decrease of the fused-task loss. Rather than choosing each module response independently, we choose the responses v_m jointly:

$$\min_{v_m \in \mathcal{S}_m} \quad \underbrace{\frac{1}{2}\left\| r - \sum_m v_m \right\|^2}_{\text{fit the desired task change}} + \underbrace{\frac{1}{2}\sum_m \|v_m\|^2}_{\text{limit total module movement}} . \tag{7}$$

The objective determines both what combined response to make and how the modules should share that response. For a fixed $\delta = \sum_m v_m$, all allocations have the same fitting error. The energy term therefore selects the allocation in Theorem 3.1. Allowing δ to vary also chooses the task-improving combined response. These are two decisions within one optimization, not successive update stages. Appendix D.1 gives the equivalent output-only formulation.

4.2 GIVE EACH MODULE THE PART IT CAN REALIZE

Let P_m denote the orthogonal projection onto $\mathcal{S}_m$. For any output-space vector x, $P_m x$ is the closest response in $\mathcal{S}_m$ to x. Define $S = \sum_m P_m$; overlapping response spaces mean that S need not be a projection. Denote the optimal responses by v_m^*, with sum $\delta^* = \sum_m v_m^*$. The remainder $e = r - \delta^*$ depends on all these responses; it is not specified in advance. Optimality gives $v_m^* = P_m e$, meaning each module takes the part of the same remainder it can realize. Summing gives $r = e + Se$, which determines e and all module responses together.

Theorem 4.1 (Unique allocation and basis invariance)**.** *Let I denote the identity operator on the output space, Eq. (7) has the unique solution*

$$e = (I + S)^{-1} r, \qquad v_m^* = P_m e, \qquad \delta^* = \sum_m v_m^* = r - e. \tag{8}$$

Using different bases to represent the same fixed response spaces $\mathcal{S}_m$ leaves the optimal responses v_m^* and their combined response δ^* unchanged.

Equivalently, the optimal response of module m can be written as $2v_m^* = P_m(r - \sum_{n \neq m} v_n^*)$. This form makes the interaction explicit: module m responds to the task direction after accounting for the responses of the other modules. Appendices D.2–D.3 provide the proof and the basis-invariance result. The invariance applies to different bases representing the same response spaces; it does not imply invariance to arbitrary network reparameterizations.

Task progress and bounded module movement. Because $r = -\nabla_f L(\theta; \mathcal{B})$, the inner product $\langle r, \delta^* \rangle$ is the predicted first-order decrease in the fused-task loss produced by the optimal δ^*.

Theorem 4.2 (Bounded functional motion)**.** *The optimal responses satisfy*

$$\underbrace{\langle r, \delta^* \rangle}_{\textit{first-order task progress}} = \underbrace{\|\delta^*\|^2}_{\textit{squared combined response}} + \underbrace{\sum_m \|v_m^*\|^2}_{\textit{module energy}} . \tag{9}$$

Moreover, $\sum_m \|v_m^*\|^2 \leq \frac{\|r\|^2}{4}$. The bound holds even when the module response spaces overlap or contain nearly canceling directions.

Eq. (9) ties module energy to first-order loss reduction. If $\delta^* = 0$, then the identity forces $v_m^* = 0$ for every module, so nonzero module responses cannot cancel each other while leaving the prediction unchanged. This rules out the hidden-cancellation failure mode identified in Sec. 2. These guarantees are local and first-order; finite-step conditions are given in Appendices F.1–F.3.

A module may help without improving the task by itself. The preceding result does not imply that every module must be useful on its own. If module m were optimized alone, with all other module responses set to zero, Eq. (7) would give the optimal response $\frac{1}{2}P_m r$. Thus, if $P_m r = 0$, module m would remain inactive on its own. In the joint solution, however, $v_m^* = P_m e$ can still be nonzero because e accounts for the other modules.

Research question 3: can an individually inactive module still help?

Can a module that would remain inactive on its own improve the combined response by compensating for another module?

Proposition 4.3 (Same local information, different coordinated responses)**.** *Consider two one-dimensional response spaces $\mathcal{S}_1 = \mathrm{span}(u_1)$ and $\mathcal{S}_2 = \mathrm{span}(u_2)$, with $u_1 = (1,0)^\top, u_2 = (\chi\rho, \sqrt{1-\rho^2})^\top$, where $0 < \rho < 1$ and $\chi \in \{-1, 1\}$. Let $r = (0, (1-\rho^2)^{-1/2})^\top$. For either value of χ, $\|u_1\| = \|u_2\| = 1$, $u_1^\top r = 0$ and $u_2^\top r = 1$, while $u_1^\top u_2 = \chi\rho$. Writing the joint responses as $v_1 = a_1 u_1$ and $v_2 = a_2 u_2$, the optimal coefficients for Eq. (7) satisfy*

$$\begin{pmatrix} 2 & \chi\rho \\ \chi\rho & 2 \end{pmatrix} \begin{pmatrix} a_1^* \\ a_2^* \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \quad \Longrightarrow \quad a_1^* = -\frac{\chi\rho}{4-\rho^2}, \quad a_2^* = \frac{2}{4-\rho^2}. \tag{10}$$

Thus, although each module has the same norm and task alignment in both cases, module 1's optimal response reverses when the cross-module alignment changes sign. These per-module quantities therefore do not determine the joint optimum.

Module 2 moves toward the task direction but also introduces a horizontal component. Module 1 has no direct task alignment, yet it can reduce this unwanted component. Its value therefore comes from how its response interacts with module 2. Unlike the zero-sum adjustments of Sec. 3, which leave the combined response fixed, module 1 here changes the combined response and improves the joint objective. Appendix E.1 proves the strict improvement over the best module-2-only response, and Appendix E.2 gives the general participation condition.

Finding 3: cooperation depends on what the other modules do

A module can be useful even when it cannot improve the task on its own. It may help by correcting part of another module's response. A module that cannot change the fused output at all, however, cannot contribute.

4.3 Represent directions and amounts in a small matrix

Generalizing the example, let $k_m = \dim \mathcal{S}_m$ be the number of independent response directions of module m. Let $U_m = [u_{m,1} \ \cdots \ u_{m,k_m}]$ have orthonormal columns spanning $\mathcal{S}_m$, and let $a_{m,j}$ specify the signed amount along $u_{m,j}$. Collecting these amounts in a_m, a response and its energy are

$$v_m = \sum_{j=1}^{k_m} a_{m,j} u_{m,j} = U_m a_m, \qquad \|v_m\|^2 = \|U_m a_m\|^2 = \|a_m\|^2. \tag{11}$$

Since $U_m^\top U_m = I_{k_m}$, choosing coefficient amounts measures exactly the same energy as choosing the output responses themselves; no new penalty is introduced. Sec. 5 constructs these bases from the actual parameter candidates.

Stack the module bases and coefficients as $U = [U_1 \ \cdots \ U_M]$ and $a = (a_1^\top, \ldots, a_M^\top)^\top$. Define $k = \sum_m k_m$, $G = U^\top U$, and $b = U^\top r$. The blocks $G_{mn} = U_m^\top U_n$ describe module–module alignment; $b_m = U_m^\top r$ describes module–task alignment. Substituting $v_m = U_m a_m$ into Eq. (7) gives $\phi(a; r) = \frac{1}{2}\left(\|r - Ua\|^2 + \|a\|^2\right)$. Its optimum satisfies $(G + I_k)a^* = b$. Written in module blocks,

$$\underbrace{\begin{bmatrix} 2I_{k_1} & U_1^\top U_2 & \cdots & U_1^\top U_M \\ U_2^\top U_1 & 2I_{k_2} & \cdots & U_2^\top U_M \\ \vdots & \vdots & \ddots & \vdots \\ U_M^\top U_1 & U_M^\top U_2 & \cdots & 2I_{k_M} \end{bmatrix}}_{G+I_k} \begin{bmatrix} a_1^* \\ a_2^* \\ \vdots \\ a_M^* \end{bmatrix} = \begin{bmatrix} U_1^\top r \\ U_2^\top r \\ \vdots \\ U_M^\top r \end{bmatrix}. \tag{12}$$

The right-hand side $U_m^\top r$ gives task alignment, while the off-diagonal blocks $U_m^\top U_n$ capture alignment between module response directions and couple their coefficients. Each diagonal block is $2I_{k_m}$, with one identity coming from response fitting and the other from the energy penalty. The system therefore jointly determines the module responses from task alignment and cross-module interaction.

When does ignoring the interactions hurt? The off-diagonal blocks distinguish the joint solution from treating each module independently. If they are ignored, Eq. (12) reduces to independent coefficients $a^{\mathrm{ind}} = b/2$. To isolate the cross-module interactions, define $E = G - I_k$. Since the diagonal blocks of G are identity matrices, E has zero diagonal blocks and contains only cross-module terms. The next result quantifies the cost of ignoring them.

Theorem 4.4 (Exact cost of ignoring interactions)**.** *For the coefficient objective ϕ, the gap between the independent and joint solutions is*

$$\Xi := \phi(a^{\mathrm{ind}}; r) - \phi(a^*; r) = \frac{1}{8} b^\top E(G + I_k)^{-1} E b. \tag{13}$$

The gap is zero if and only if $Eb = 0$, and is unchanged by choosing different orthonormal bases for the same response spaces.

The result shows that cross-module interaction alone does not make coordination useful. Even when $E \neq 0$, the independent and joint solutions coincide if $Eb = 0$. Coordination matters when the interactions affect the response called for by the current task. Appendix E.3 gives the proof, and Appendix E.4 shows why adjusting a common step size generally cannot reproduce the joint solution. The gap Ξ measures improvement in the local objective of Eq. (7), not necessarily in test accuracy.

5 From the Chosen Output Changes to Parameter Updates

Sec. 4 defines COIN as a joint optimization over module responses in output space. We implement the response allocation by choosing parameter candidates, measuring their output effects, and converting the jointly selected responses back into parameter corrections.

Specifically, COIN starts from a base update $\bar{\theta}_t = \theta_t + d_t^{\mathrm{base}}$, where d_t^{base} is the actual parameter change produced by the base optimizer. The base optimizer trains each encoder with unimodal supervision and the fusion and classification components with the full-input loss (Wei et al., 2025b; Feng et al., 2026). At the updated state $\bar{\theta}_t$, we compute on a training calibration batch $\mathcal{B}$ the fresh fused-task gradient $g_m = \nabla_{\theta_m} L(\bar{\theta}_t; \mathcal{B})$.

For each corrected module, we allow parameter corrections of the form

$$d_m = \alpha_m d_{t,m}^{\mathrm{base}} + \beta_m(-g_m), \qquad \alpha_m, \beta_m \in \mathbb{R}. \tag{14}$$

These directions form $\mathcal{D}_m = \mathrm{span}\{d_{t,m}^{\mathrm{base}}, -g_m\}$. The base direction carries the update proposed by the optimizer, including its supervision and optimizer history, while the fresh gradient reflects the current fused-task objective. The coefficients α_m and β_m are not tuned separately; their effective combination is determined by the COIN joint solve.

We next map this parameter subspace into output space. Let Q_m contain an orthonormal basis of $\mathcal{D}_m$, and evaluate the fused-output Jacobian J_m at $\bar{\theta}_t$ on the same batch $\mathcal{B}$. Jacobian–vector products give $Z_m = J_m Q_m$, where each column is the output response to one parameter-basis direction. Keeping only the positive singular modes gives the compact singular value decomposition

$$Z_m = J_m Q_m = U_m \Sigma_m W_m^\top, \qquad k_m = \mathrm{rank}(Z_m). \tag{15}$$

The columns of U_m form the response-space basis used in Sec. 4.3; Σ_m contains the corresponding response strengths, and W_m gives the parameter-direction combinations in Q_m that produce them. With numerical truncation, k_m denotes the retained numerical rank, and subsequent operations use only the retained modes (Appendix F.2).

Using these response-space bases, form G and b as in Sec. 4.3 and solve Eq. (12) for a^*. The resulting coefficients specify the COIN response assigned to each module, $v_m^* = U_m a_m^*$. To realize this response in parameter space, define

$$d_m^{\mathrm{corr}} = Q_m W_m \Sigma_m^{-1} a_m^*, \qquad \theta_{t+1,m} = \bar{\theta}_{t,m} + \eta_t d_m^{\mathrm{corr}}. \tag{16}$$

where $\eta_t > 0$ is a common correction step size. The factor Σ_m^{-1} accounts for the response strengths, while $Q_m W_m$ converts the resulting coefficients into a parameter direction. Thus $J_m d_m^{\mathrm{corr}} = U_m a_m^* = v_m^*$: the correction realizes the response selected in output space. Among directions in $\mathcal{D}_m$ realizing this response, it has minimum parameter norm (Appendix F.2).

Algorithm 1 summarizes one base update followed by a scheduled COIN correction. During calibration, dropout and normalization states are fixed while the gradients and Jacobian–vector products are computed, then restored afterward. The correction leaves the internal state of the base optimizer unchanged. Here orth returns an orthonormal parameter basis, and svd_+ returns the retained positive singular triplets with right singular vectors stored as columns. Empty response spaces receive zero correction. Theorem F.1 gives sufficient conditions for calibration-loss descent.

Algorithm 1 A base update with a scheduled COIN correction

Require: Parameters θ_t, base optimizer, correction schedule, step η_t
$\bar{\theta}_t, d_t^{\mathrm{base}} \leftarrow \mathrm{BaseStep}(\theta_t)$ ▷ record the actual parameter change
$\theta_{t+1} \leftarrow \bar{\theta}_t$
if a correction is scheduled **then**
  $\mathcal{B} \leftarrow$ training calibration batch; save state and fix calibration mode
  $r \leftarrow -\nabla_f L(\bar{\theta}_t; \mathcal{B})$; $g_m \leftarrow \nabla_{\theta_m} L(\bar{\theta}_t; \mathcal{B})$ for all m
  for each corrected module m **do**
    $Q_m \leftarrow \mathrm{orth}(\mathcal{D}_m)$ where $\mathcal{D}_m \leftarrow \mathrm{span}\{d_{t,m}^{\mathrm{base}}, -g_m\}$
    $(U_m, \Sigma_m, W_m) \leftarrow \mathrm{svd}_+(Z_m)$ where $Z_m \leftarrow J_m Q_m$ ▷ Eq. (15), retained modes
  end for
  $G \leftarrow U^\top U$, $b \leftarrow U^\top r$ where $U \leftarrow [U_1 \ \cdots \ U_M]$
  $a^* \leftarrow \mathrm{solve}(G + I_k, b)$ ▷ Eq. (12), joint response amounts
  for each corrected module m **do**
    $d_m^{\mathrm{corr}} \leftarrow Q_m W_m \Sigma_m^{-1} a_m^*$; $\theta_{t+1,m} \leftarrow \bar{\theta}_{t,m} + \eta_t d_m^{\mathrm{corr}}$ ▷ Eq. (16), realize the response
  end for
  Restore training modes and buffers
end if
return θ_{t+1}

Appendix H gives implementation details for the base step, output centering, numerical rank handling, and correction schedule. If every response space is empty, the joint solve is skipped; blocks without a correction retain their base-step values.

6 Experiments

Setup and baselines. We evaluate on AVE (Tian et al., 2018), AVSBench S4 classification (Zhou et al., 2022), VGGSound50 (Chen et al., 2020), and multimodal FOOD101 (Wang et al., 2015), using audio/video modalities for the first three datasets and text/image modalities for FOOD101. We consider Summation, which adds modality features element-wise, and Concatenation, which joins them along the feature dimension, as vanilla fusion baselines. We further compare methods spanning gradient/loss and information regulation (OGM-GE, AGM, MMCosine, InfoReg, IBML), unimodal–multimodal objective coordination (MMPareto, ARL), and separated or multi-timescale learning (MLA, DGL, MoNet). Training and evaluation details are provided in Appendix J.1.

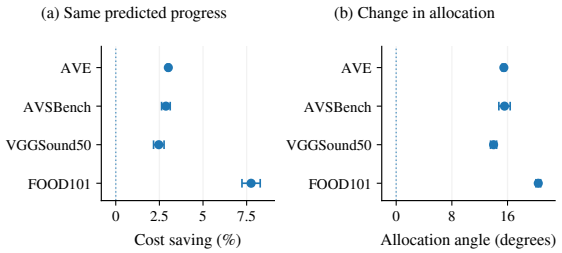


Figure 1: **Less movement for the same predicted task progress.** Each row is a dataset. The left horizontal axis is percentage cost saving over the progress-matched scalar control; farther right means greater saving. The right axis is the angle between joint and scalar allocations; zero means the same direction. Blue dots and horizontal bars are mean ± sample SD across three seed summaries; dotted vertical lines mark zero. Candidates are rebuilt on each fit batch, with 16 fit examples used here.

Table 1: Fusion accuracy (%, mean ± sample SD over three seeds). Full results are in Appendix J.2.

Method	AVE	AVSBench	VGGSound50	FOOD101
Summation	66.33 ± 0.85	87.43 ± 0.90	53.73 ± 0.48	87.58 ± 0.04
Concatenation	66.42 ± 0.54	87.34 ± 0.45	53.82 ± 0.63	87.50 ± 0.10
OGM-GE (Peng et al., 2022)	65.67 ± 1.02	87.50 ± 0.67	55.96 ± 0.21	87.49 ± 0.07
AGM (Li et al., 2023)	68.32 ± 0.94	87.93 ± 0.88	53.94 ± 0.51	87.18 ± 0.15
MMCosine (Xu et al., 2023)	67.49 ± 0.47	89.19 ± 0.69	54.15 ± 0.33	88.73 ± 0.12
MMPareto (Wei & Hu, 2024)	69.73 ± 0.62	91.44 ± 0.93	59.06 ± 0.13	89.24 ± 0.12
MLA (Zhang et al., 2024a)	68.74 ± 0.47	84.59 ± 1.34	58.21 ± 0.77	89.30 ± 0.43
DGL (Wei et al., 2025b)	68.82 ± 2.34	91.98 ± 0.33	57.40 ± 0.22	88.24 ± 0.01
ARL (Wei et al., 2025a)	69.87 ± 0.44	91.94 ± 0.17	60.19 ± 0.79	88.39 ± 0.20
InfoReg (Huang et al., 2025)	69.89 ± 0.19	91.35 ± 0.51	59.46 ± 0.32	89.20 ± 0.15
MoNet (Feng et al., 2026)	71.23 ± 2.17	93.06 ± 0.42	60.90 ± 0.23	90.42 ± 0.16
IBML (Qin et al., 2026)	70.15 ± 1.07	92.34 ± 0.39	60.54 ± 0.57	90.67 ± 0.11
COIN (ours)	$\mathbf{74.38 \pm 1.94}$	$\mathbf{94.14 \pm 0.34}$	$\mathbf{61.52 \pm 0.40}$	$\mathbf{91.96 \pm 0.05}$

Fusion performance. COIN achieves the highest fusion mean on every dataset in Table 1, exceeding the strongest displayed baseline by 3.15, 1.08, 0.62, and 1.29 percentage points, respectively. Appendix J.2 reports the corresponding scores for both modalities.

Joint allocation changes more than the step size. Using the same candidates, we compare COIN with the scalar direction $b = U^\top r$, rescaled to match predicted progress $b^\top a$. The functional cost $C(a) = \|Ua\|^2 + \|a\|^2$ captures both movement in the fused output and total movement across modules. Matching predicted progress allows us to isolate allocation efficiency from step size.

At equal predicted progress, COIN reduces cost by 2.47–7.75%, with allocation angles of 13.99°–20.42°. The nonzero allocation angles show that COIN selects a different update direction, while the positive cost savings show that this direction is more efficient under the functional cost. Thus, the least-motion benefit cannot be explained by a common rescaling of the update.

Modules improve by adapting to one another. At a saved state, *Remove* zeros the correction of one module while keeping the corrections of all other modules fixed; *Re-solve* then recomputes the allocation among the remaining corrections. All modules and inputs remain present in both interventions. We measure intervention cost as the original joint correction's audit cross-entropy (CE) decrease minus the intervention's. Re-solving recovers approximately 70–84% of removal cost across all 12 dataset–fusion-block means. The modules adapt when a participant changes, supporting conditional cooperation (Sec. 4.2). Appendix J.4 reports all encoder and fusion results.

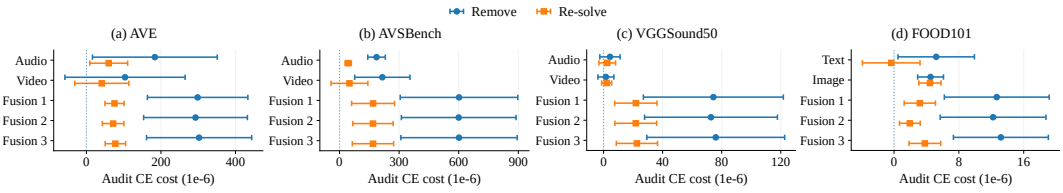


Figure 2: **Re-solving recovers progress after one correction is removed.** Rows identify the affected encoder or fusion block; panels identify datasets. The horizontal axis is lost audit CE decrease in 10^{-6} units. Lower values are better; the dotted zero line matches the original joint update. Blue circles denote Remove, orange squares Re-solve, and bars show sample SD across three seed means. An orange square left of its blue circle indicates recovered progress. Panel scales differ.

The selected response is realized by the network. The predicted and measured CE changes closely agree: active audit-response errors are 0.11–0.44%, and mean progress scales approximately with correction size over the tested range. Appendices J.5–J.7 further report numerical rank retention under basis changes and agreement between predicted and measured finite-step CE changes.

AI USAGE STATEMENT

Generative AI tools were used for language editing, including grammar correction, wording refinement, and improvements to readability.

REFERENCES

Shun-ichi Amari. Natural gradient works efficiently in learning. *Neural Computation*, 10(2):251–276, 1998. doi: 10.1162/089976698300017746.

Ehsan Amid, Rohan Anil, and Manfred K. Warmuth. LocoProp: Enhancing BackProp via local loss optimization. In *International Conference on Artificial Intelligence and Statistics*, volume 151, pp. 9626–9642, 2022. URL https://proceedings.mlr.press/v151/amid22a.html.

Tadas Baltrušaitis, Chaitanya Ahuja, and Louis-Philippe Morency. Multimodal machine learning: A survey and taxonomy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(2): 423–443, 2019. doi: 10.1109/TPAMI.2018.2798607.

Atilim Gunes Baydin, Barak A. Pearlmutter, Alexey Andreyevich Radul, and Jeffrey Mark Siskind. Automatic differentiation in machine learning: A survey. *Journal of Machine Learning Research*, 18(153):1–43, 2018. URL https://jmlr.org/papers/v18/17-468.html.

Ali Behrouz, Meisam Razaviyayn, Peilin Zhong, and Vahab Mirrokni. Nested learning: The illusion of deep learning architectures. In *Advances in Neural Information Processing Systems*, 2025. URL https://arxiv.org/abs/2512.24695.

Adi Ben-Israel and Thomas N. E. Greville. *Generalized Inverses: Theory and Applications*. Springer, second edition, 2003. doi: 10.1007/b97366.

Åke Björck and Gene H. Golub. Numerical methods for computing angles between linear subspaces. *Mathematics of Computation*, 27(123):579–594, 1973. doi: 10.1090/S0025-5718-1973-0348991-3.

Andreas Buja, Trevor Hastie, and Robert Tibshirani. Linear smoothers and additive models. *The Annals of Statistics*, 17(2):453–510, 1989. doi: 10.1214/aos/1176347115.

Peter G. Casazza, Gitta Kutyniok, and Shidong Li. Fusion frames and distributed processing. *Applied and Computational Harmonic Analysis*, 25(1):114–132, 2008. doi: 10.1016/j.acha.2007.10.001.

Honglie Chen, Weidi Xie, Andrea Vedaldi, and Andrew Zisserman. VGGSound: A large-scale audio-visual dataset. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2020. URL https://arxiv.org/abs/2004.14368.

Zhao Chen, Vijay Badrinarayanan, Chen-Yu Lee, and Andrew Rabinovich. GradNorm: Gradient normalization for adaptive loss balancing in deep multitask networks. In *International Conference on Machine Learning*, 2018. URL https://arxiv.org/abs/1711.02257.

Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Conference on Empirical Methods in Natural Language Processing*, pp. 1724–1734, 2014. doi: 10.3115/v1/D14-1179.

Laurent Dinh, Razvan Pascanu, Samy Bengio, and Yoshua Bengio. Sharp minima can generalize for deep nets. In *International Conference on Machine Learning*, 2017. URL https://arxiv.org/abs/1703.04933.

William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23(120):1–39, 2022. URL https://jmlr.org/papers/v23/21-0998.html.

Yanglin Feng, Yang Qin, Dezhong Peng, Rui Wang, Xiaomin Song, and Peng Hu. Multimodal nested learning for decoupled and coordinated optimization. In *International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=8XOncDWc5j.

Gene H. Golub and Charles F. Van Loan. *Matrix Computations*. Johns Hopkins University Press, fourth edition, 2013.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2016. URL https://arxiv.org/abs/1512.03385.

Arthur E. Hoerl and Robert W. Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970. doi: 10.1080/00401706.1970.10488634.

Cong Hua, Qianqian Xu, Shilong Bao, Zhiyong Yang, and Qingming Huang. ReconBoost: Boosting can achieve modality reconcilement. In *International Conference on Machine Learning*, 2024. URL https://arxiv.org/abs/2405.09321.

Chengxiang Huang, Yake Wei, Zequn Yang, and Di Hu. Adaptive unimodal regulation for balanced multimodal information acquisition. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025. URL https://arxiv.org/abs/2503.18595.

Yu Huang, Junyang Lin, Chang Zhou, Hongxia Yang, and Longbo Huang. Modality competition: What makes joint training of multi-modal network fail in deep learning? (provably). In *International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 9226–9259, 2022. URL https://arxiv.org/abs/2203.12221.

Hong Li, Xingyu Li, Pengbo Hu, Yinuo Lei, Chunxiao Li, and Yi Zhou. Boosting multi-modal model performance with adaptive gradient modulation. In *IEEE/CVF International Conference on Computer Vision*, 2023. URL https://arxiv.org/abs/2308.07686.

Bo Liu, Xingchao Liu, Xiaojie Jin, Peter Stone, and Qiang Liu. Conflict-averse gradient descent for multi-task learning. In *Advances in Neural Information Processing Systems*, 2021. URL https://arxiv.org/abs/2110.14048.

Yishun Lu and Wes Armour. Second-order multi-level variance correction for modality competition in multimodal models. *arXiv preprint arXiv:2605.16165*, 2026. URL https://arxiv.org/abs/2605.16165.

James Martens. New insights and perspectives on the natural gradient method. *Journal of Machine Learning Research*, 21(146):1–76, 2020. URL https://jmlr.org/papers/v21/17-678.html.

James Martens and Roger Grosse. Optimizing neural networks with Kronecker-factored approximate curvature. In *International Conference on Machine Learning*, 2015. URL https://arxiv.org/abs/1503.05671.

Basil Mustafa, Carlos Riquelme, Joan Puigcerver, Rodolphe Jenatton, and Neil Houlsby. Multimodal contrastive learning with LIMoE: The language-image mixture of experts. In *Advances in Neural Information Processing Systems*, 2022. URL https://arxiv.org/abs/2206.02770.

Aviv Navon, Aviv Shamsian, Idan Achituve, Haggai Maron, Kenji Kawaguchi, Gal Chechik, and Ethan Fetaya. Multi-task learning as a bargaining game. In *International Conference on Machine Learning*, 2022. URL https://arxiv.org/abs/2202.01017.

Behnam Neyshabur, Ruslan Salakhutdinov, and Nathan Srebro. Path-SGD: Path-normalized optimization in deep neural networks. In *Advances in Neural Information Processing Systems*, 2015. URL https://arxiv.org/abs/1506.02617.

Jiquan Ngiam, Aditya Khosla, Mingyu Kim, Juhan Nam, Honglak Lee, and Andrew Y. Ng. Multimodal deep learning. In *International Conference on Machine Learning*, pp. 689–696, 2011.

Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. Springer, second edition, 2006. doi: 10.1007/978-0-387-40065-5.

Barak A. Pearlmutter. Fast exact multiplication by the hessian. *Neural Computation*, 6(1):147–160, 1994. doi: 10.1162/neco.1994.6.1.147.

Xiaokang Peng, Yake Wei, Andong Deng, Dong Wang, and Di Hu. Balanced multimodal learning via on-the-fly gradient modulation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022. URL https://arxiv.org/abs/2203.15332.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. GloVe: Global vectors for word representation. In *Conference on Empirical Methods in Natural Language Processing*, pp. 1532–1543, 2014. doi: 10.3115/v1/D14-1162.

Mohammad Pezeshki, Sékou-Oumar Kaba, Yoshua Bengio, Aaron Courville, Doina Precup, and Guillaume Lajoie. Gradient starvation: A learning proclivity in neural networks. In *Advances in Neural Information Processing Systems*, 2021. URL https://arxiv.org/abs/2011.09468.

Yang Qin, Yanglin Feng, Yuan Sun, Dezhong Peng, Xi Peng, and Peng Hu. Deep information-balanced multimodal learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 48(8):9384–9396, 2026. doi: 10.1109/TPAMI.2026.3681770. URL https://github.com/QinYang79/IBML.

Andrew M. Saxe, James L. McClelland, and Surya Ganguli. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. In *International Conference on Learning Representations*, 2014. URL https://arxiv.org/abs/1312.6120.

Ozan Sener and Vladlen Koltun. Multi-task learning as multi-objective optimization. In *Advances in Neural Information Processing Systems*, 2018. URL https://arxiv.org/abs/1810.04650.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *International Conference on Learning Representations*, 2017. URL https://arxiv.org/abs/1701.06538.

Yapeng Tian, Jing Shi, Bochen Li, Zhiyao Duan, and Chenliang Xu. Audio-visual event localization in unconstrained videos. In *European Conference on Computer Vision*, pp. 247–263, 2018.

Yao-Hung Hubert Tsai, Shaojie Bai, Paul Pu Liang, J. Zico Kolter, Louis-Philippe Morency, and Ruslan Salakhutdinov. Multimodal transformer for unaligned multimodal language sequences. In *Annual Meeting of the Association for Computational Linguistics*, pp. 6558–6569, 2019. doi: 10.18653/v1/P19-1656. URL https://aclanthology.org/P19-1656/.

Oriol Vinyals and Daniel Povey. Krylov subspace descent for deep learning. In *International Conference on Artificial Intelligence and Statistics*, 2012. URL https://arxiv.org/abs/1111.4259.

Shihao Wang and Xueru Zhang. PRISM: Gauge-invariant tangent-space differentially private LoRA. In *International Conference on Machine Learning*, 2026. URL https://arxiv.org/abs/2606.00944.

Weiyao Wang, Du Tran, and Matt Feiszli. What makes training multi-modal classification networks hard? In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020. URL https://arxiv.org/abs/1905.12681.

Xin Wang, Devinder Kumar, Nicolas Thome, Matthieu Cord, and Frédéric Precioso. Recipe recognition with large multimodal food dataset. In *IEEE International Conference on Multimedia & Expo Workshops*, pp. 1–6, 2015. doi: 10.1109/ICMEW.2015.7169757.

Shicai Wei, Chunbo Luo, and Yang Luo. Improving multimodal learning via imbalanced learning. In *IEEE/CVF International Conference on Computer Vision*, 2025a. URL https://arxiv.org/abs/2507.10203.

Shicai Wei, Chunbo Luo, and Yang Luo. Boosting multimodal learning via disentangled gradient learning. In *IEEE/CVF International Conference on Computer Vision*, 2025b. URL https://arxiv.org/abs/2507.10213.

Yake Wei and Di Hu. MMPareto: Boosting multimodal learning with innocent unimodal assistance. In *International Conference on Machine Learning*, 2024. URL https://arxiv.org/abs/2405.17730.

Ruize Xu, Ruoxuan Feng, Shi-Xiong Zhang, and Di Hu. MMCosine: Multi-modal cosine loss towards balanced audio-visual fine-grained learning. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2023. URL https://arxiv.org/abs/2303.05338.

Jui-Nan Yen, Si Si, Zhao Meng, Felix Yu, Sai Surya Duvvuri, Inderjit S. Dhillon, Cho-Jui Hsieh, and Sanjiv Kumar. LoRA done RITE: Robust invariant transformation equilibration for LoRA optimization. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=VpWki1v2P8.

Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. In *Advances in Neural Information Processing Systems*, 2020. URL https://arxiv.org/abs/2001.06782.

Amir Zadeh, Minghai Chen, Soujanya Poria, Erik Cambria, and Louis-Philippe Morency. Tensor fusion network for multimodal sentiment analysis. In *Conference on Empirical Methods in Natural Language Processing*, pp. 1103–1114, 2017. doi: 10.18653/v1/D17-1115. URL https://aclanthology.org/D17-1115/.

Xiaohui Zhang, Jaehong Yoon, Mohit Bansal, and Huaxiu Yao. Multimodal representation learning by alternating unimodal adaptation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024a. URL https://arxiv.org/abs/2311.10707.

Yedi Zhang, Peter E. Latham, and Andrew M. Saxe. Understanding unimodal bias in multimodal deep linear networks. In *International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 59100–59125, 2024b. URL https://proceedings.mlr.press/v235/zhang24aa.html.

Yupei Zhang, Xiaofei Wang, Fangliangzi Meng, Jin Tang, and Chao Li. Knowledge-driven subspace fusion and gradient coordination for multi-modal learning. In *Medical Image Computing and Computer Assisted Intervention*, pp. 263–273, 2024c. doi: 10.1007/978-3-031-72083-3_25.

Jinxing Zhou, Jianyuan Wang, Jiayi Zhang, Weixuan Sun, Jing Zhang, Stan Birchfield, Dan Guo, Lingpeng Kong, Meng Wang, and Yiran Zhong. Audio-visual segmentation. In *European Conference on Computer Vision*, 2022. URL https://arxiv.org/abs/2207.05042.

A Related work

Modality competition and unequal learning. Multimodal learning seeks complementary information from heterogeneous inputs (Ngiam et al., 2011; Baltrušaitis et al., 2019), but adding modalities does not ensure better optimization. Wang et al. (2020) study different generalization rates, Huang et al. (2022) analyze competition during feature learning, and Zhang et al. (2024b) characterize unimodal learning phases. Gradient starvation provides a related account of how fitting some predictive features can inhibit others (Pezeshki et al., 2021). These works explain important causes of underlearning. We ask what information is required to coordinate the updates available at a given training state.

Balancing and reconciling multimodal objectives. GradNorm balances task-gradient magnitudes (Chen et al., 2018); OGM-GE and AGM adjust modality-dependent gradients (Peng et al., 2022; Li et al., 2023); MMCosine changes the classification loss (Xu et al., 2023); InfoReg regulates unimodal information acquisition (Huang et al., 2025). MMPareto combines unimodal assistance with multimodal optimization (Wei & Hu, 2024), while ARL shows that unequal modality dependence can be beneficial (Wei et al., 2025a). We do not assume that equal speeds or contributions are optimal. Our counterexample instead shows that identical local norms and task alignments can require different joint updates because the responses interact in the fused output.

Independent learning and fusion. Fusion is studied through tensor interactions (Zadeh et al., 2017), cross-modal attention (Tsai et al., 2019), and their optimization. MLA alternates unimodal adaptation (Zhang et al., 2024a); ReconBoost reconciles unimodal learning and cross-modal interactions through a boosting-inspired procedure (Hua et al., 2024); DGL separates encoder and fusion gradient paths (Wei et al., 2025b). MoNet combines independently supervised modules with multi-timescale fusion under nested learning (Feng et al., 2026; Behrouz et al., 2025). Our contribution concerns how updates to such modules are coordinated, rather than a new fusion architecture. We measure their responses in the same output space and solve their simultaneous allocation.

Cross-module information. Multimodal subspace fusion and gradient coordination (Zhang et al., 2024c), and recent second-order variance correction (Lu & Armour, 2026), also look beyond scalar modality strength. Our analysis concerns attainable changes of the current fused predictor: their output-preserving redistribution, the cost of realizing a fused change, and the need for compensating updates. Table 2 summarizes these different coordination objects before the analysis. Technical connections and the scope relative to routed mixture-of-experts models are in Appendix I.

Table 2: Coordination objects. Here g_i is the gradient of objective L_i, and $L_{\text{uni}}, L_{\text{fuse}}$ are unimodal and fused losses. For COIN, d_m is a parameter direction, J_m maps it to an output change v_m, and r is the desired task direction.

Coordination view	Object being coordinated	Information retained
Scalar modulation	$s_m = \|\nabla_{\theta_m} L\|$ Magnitude or contribution score	Relative task/modality strength No response-direction information
Gradient reconciliation	$g_i = \nabla_\theta L_i$ Objective-specific gradients	$\langle g_i, g_j \rangle$ Agreement between objectives
Separated learning	$L_{\text{uni}},\ L_{\text{fuse}}$ Distinct supervision signals	Objective-to-parameter assignment Which loss trains which block
COIN (ours)	$v_m = J_m d_m$ Attainable prediction changes	$\langle v_m, v_n \rangle, \quad \langle v_m, r \rangle$ Interactions and task alignment

B Notation and the local optimization problem

The analysis is conditional on a fixed training calibration batch, a fixed network state, a fixed partition of corrected parameters, and fixed candidate spaces. Conditioning is essential: the candidate spaces can change at the next training step. Unless stated otherwise, all norms and orthogonal projections

in output space use the Euclidean coordinates of Equation (18). Parameter-space norms use the Euclidean norm over the concatenated corrected blocks.

The principal objects are as follows. Module m has a candidate parameter space $\mathcal{D}_m \subseteq \mathbb{R}^{p_m}$, a Jacobian J_m, and a response space $\mathcal{S}_m = J_m\mathcal{D}_m$. The matrix U_m has orthonormal columns spanning $\mathcal{S}_m$; its column count is k_m. The projector onto $\mathcal{S}_m$ is $P_m = U_m U_m^\top$. For the tuple $\boldsymbol{v} = (v_1, \dots, v_M)$ in the product space $\mathcal{H} = \prod_m \mathcal{S}_m$, the sum operator is $T\boldsymbol{v} = \sum_m v_m$, with $\|\boldsymbol{v}\|_{\mathcal{H}}^2 = \sum_m \|v_m\|^2$. For a tuple $\boldsymbol{q} = (q_1, \dots, q_M)$ of achievable response adjustments, the zero-sum space is $\mathcal{N} = \ker T = \{\boldsymbol{q} \in \mathcal{H} : \sum_m q_m = 0\}$. Hence Equation (5) says that all allocations with the same combined response form $\boldsymbol{v} + \mathcal{N}$. Writing T^* for the adjoint of T, the operator on output space is $S = TT^* = \sum_m P_m$. The coefficient-space matrices are $U = [U_1 \ \cdots \ U_M]$, $G = U^\top U$, and $A = I + G$; the task alignment is $b = U^\top r$.

For the classification coordinates below, let $n_f = NC$ be the ambient output dimension, $\ell_m = \dim \mathcal{D}_m$, and $k = \sum_m k_m$. The centered vectors lie in a subspace of $\mathbb{R}^{n_f}$; n_f counts stored coordinates, not their independent degrees of freedom. The table separates spaces, bases, and coordinate vectors.

Table 3: Notation and dimensions. In the exact analysis, $k_m = \dim \mathcal{S}_m$; a numerical rank decision replaces $\mathcal{S}_m$ by its retained subspace.

Symbol	Space or dimensions	Meaning
$\mathcal{D}_m$	$\mathcal{D}_m \subseteq \mathbb{R}^{p_m}$	Candidate parameter-direction space
Q_m	$p_m \times \ell_m$	Orthonormal basis of $\mathcal{D}_m$
$J_m,\ Z_m = J_m Q_m$	$n_f \times p_m,\ n_f \times \ell_m$	Block Jacobian and candidate responses
$\mathcal{S}_m$	$\mathcal{S}_m = J_m\mathcal{D}_m$	Attainable response space
$U_m,\ \Sigma_m,\ W_m$	$n_f \times k_m,\ k_m \times k_m,\ \ell_m \times k_m$	Left vectors, positive singular values, right vectors
$a_m,\ v_m = U_m a_m$	$k_m,\ n_f$	Response coordinates and module response
$\delta = \sum_m v_m,\ e = r - \delta^*$	$n_f,\ n_f$	Combined response and remaining task direction
$P_m,\ S = \sum_m P_m$	$n_f \times n_f$	Module projector and projector sum
$U,\ G,\ A = I_k + G$	$n_f \times k,\ k \times k,\ k \times k$	Joint basis, Gram matrix, coefficient system
$r,\ b = U^\top r$	$n_f,\ k$	Task direction and task alignment
$K_{-m} = I_{n_f} + \sum_{j \neq m} P_j$	$n_f \times n_f$	Output-space system without module m

We use v_m only for a module response and $\delta = \sum_m v_m$ for the combined response. The tuple $\boldsymbol{v} = (v_1, \dots, v_M)$ is used only in the appendix to organize the module responses; it never denotes their sum. The regularized objective can be written in either product or coefficient coordinates:

$$\Phi(\boldsymbol{v}; r) = \frac{1}{2}\|r - T\boldsymbol{v}\|^2 + \frac{1}{2}\|\boldsymbol{v}\|_{\mathcal{H}}^2, \qquad \phi(a; r) = \frac{1}{2}\|r - Ua\|^2 + \frac{1}{2}\|a\|^2. \tag{17}$$

Within each block, $a_m \mapsto U_m a_m$ is an isometry. Hence these are exactly the same problem, not two approximations.

B.1 Why centered, sample-normalized logits give the stated derivative

For a batch of N examples with C classes, let $F_\theta \in \mathbb{R}^{N \times C}$ be the raw fused logits, and Y the one-hot labels. The common output coordinates and task direction are

$$\Pi_C = I_C - \frac{\mathbf{1}\mathbf{1}^\top}{C}, \qquad f_\theta = \frac{\mathrm{vec}(F_\theta \Pi_C)}{\sqrt{N}}, \qquad r = \frac{\mathrm{vec}(Y - \mathrm{softmax}(F_\theta))}{\sqrt{N}}. \tag{18}$$

For example i, regard $p_i = \mathrm{softmax}(F_{\theta,i})$, its one-hot label y_i, and a row perturbation ΔF_i as column vectors. Mean cross-entropy is

$$L(\theta; \mathcal{B}) = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{C} y_{ic} \log p_{ic}. \tag{19}$$

Its differential for a raw logit perturbation ΔF is

$$\mathrm{D}L[\Delta F] = \frac{1}{N}\sum_i (p_i - y_i)^\top \Delta F_i. \tag{20}$$

Because $\mathbf{1}^\top(p_i - y_i) = 0$, replacing ΔF_i with its centered version does not change this expression. Substituting $r = \mathrm{vec}(Y - p)/\sqrt{N}$ and $\Delta f = \mathrm{vec}(\Delta F \Pi_C)/\sqrt{N}$ therefore yields

$$\mathrm{D}L[\Delta f] = -r^\top \Delta f. \tag{21}$$

The factor $1/\sqrt{N}$ makes an ordinary Euclidean inner product equal to the sample-mean inner product on raw response matrices. An implementation can store unnormalized $N \times C$ responses and divide inner products by N, or vectorize with $1/\sqrt{N}$; the two formulations are identical. The target is the ordinary negative cross-entropy output gradient in this geometry, not an inverse-Fisher or Newton target.

The centered-logit metric is a deliberate choice. It is invariant to a common shift of all class logits for an example and to a permutation of the class coordinates. It is not asserted to be invariant to an arbitrary nonlinear reparameterization of probabilities or to changing a softmax temperature. Such a change defines a different coordination metric.

From a local response to an actual network update. Differentiability at the fixed state gives Equation (2): the derivative of the joint network map applied to the concatenated direction is the sum of its block derivatives. The v_m are therefore responses of the same fused logits, not raw encoder features or parameters added across different spaces. For fixed d, the remainder is $o(\tau)$ as $\tau \to 0$. The unbounded-scale examples concern the resulting linear operator, not the validity of a Taylor approximation for arbitrarily large finite steps. In an additive linear model, the response identity is exact at every scale; the nonlinear implementation uses the local response and a finite common correction step.

C Proofs of the problem characterization

C.1 Proof of Proposition 2.1

The predictor used in Proposition 2.1 is

$$f_\psi^{(c)} = c\psi_1 x_1 + c^{-1}\psi_2 x_2, \qquad y = x_1 + x_2, \quad c > 0. \tag{22}$$

Since x_1 and x_2 are orthonormal in L^2, its loss is

$$L(\psi) = \frac{1}{2}(c\psi_1 - 1)^2 + \frac{1}{2}(c^{-1}\psi_2 - 1)^2. \tag{23}$$

At $\psi = 0$, its gradient is $(-c, -c^{-1})^\top$. Gradient flow $\dot\psi = -\eta\nabla L$ therefore gives $\dot\psi_1 = \eta c$ and $\dot\psi_2 = \eta c^{-1}$. The corresponding contributions to the derivative of the prediction are $\dot f_1 = c\dot\psi_1 x_1 = \eta c^2 x_1$ and $\dot f_2 = c^{-1}\dot\psi_2 x_2 = \eta c^{-2} x_2$. Their L^2 norms have ratio c^4. Letting $c \to \infty$ proves the unboundedness claim.

The predictor $f_\psi^{(c)}$ can equivalently be written as $\theta_1 x_1 + \theta_2 x_2$ by taking $\theta_1 = c\psi_1$ and $\theta_2 = c^{-1}\psi_2$. In particular, $\psi = 0$ represents the same zero predictor for every c. A nonzero candidate in the first coordinate generates the response space $\mathrm{span}(x_1)$; the second generates $\mathrm{span}(x_2)$. These spaces do not change with c. With unit orthonormal bases $U_1 = x_1, U_2 = x_2$ and regression residual $r = x_1 + x_2$, the allocation has $G = I_2$, $b = (1,1)^\top$, and $a^* = b/2$. Thus its two functional responses are $x_1/2$ and $x_2/2$, independently of c. □

Scope of the example. It is a statement about an unbounded family of coordinate-dependent local learning rates at a fixed task state. It does not prove unbounded loss, exploding parameters along a particular run, or a universal delay in acquiring a modality. The general role of parameterization in optimization is established in, for example, Amari (1998); Neyshabur et al. (2015); Dinh et al. (2017); Martens (2020). Our use of this example is to show why module-specific speed measurements do not by themselves define the coordination problem.

C.2 UNBOUNDED MOTION AT A FIXED COMBINED RESPONSE

Proposition C.1 (Unbounded motion within a fixed functional description)**.** *If* $\ker T \neq \{0\}$*, every achievable combined response admits collections of module responses with arbitrarily large* $\sum_m \|v_m\|^2$*. Their combined response is exactly the same. The dimension of this ambiguity is*

$$\dim \ker T = \sum_m \dim \mathcal{S}_m - \dim \Big(\sum_m \mathcal{S}_m \Big). \tag{24}$$

Even when T *is injective, a unit product-space vector* $\boldsymbol{q}$ *with* $\|T\boldsymbol{q}\| = \epsilon > 0$ *yields module energy* ϵ^{-2} *for a unit combined response after rescaling by* $1/\epsilon$*.*

Proof. The range of T is $\sum_m \mathcal{S}_m$. Its domain has dimension $\sum_m \dim \mathcal{S}_m$. Rank–nullity immediately gives Equation (24). Suppose $\ker T$ contains a nonzero vector $\boldsymbol{q}$. For any collection $\boldsymbol{v}$ realizing an achievable response $\delta = T\boldsymbol{v}$, the collection $\boldsymbol{v} + t\boldsymbol{q}$ has

$$T(\boldsymbol{v} + t\boldsymbol{q}) = \delta, \qquad \|\boldsymbol{v} + t\boldsymbol{q}\|_{\mathcal{H}}^2 = \|\boldsymbol{v}\|_{\mathcal{H}}^2 + 2t \langle \boldsymbol{v}, \boldsymbol{q} \rangle_{\mathcal{H}} + t^2 \|\boldsymbol{q}\|_{\mathcal{H}}^2 . \tag{25}$$

The quadratic term has a strictly positive coefficient, so the energy tends to infinity as $|t| \to \infty$. This proves the fixed-response claim.

For near cancellation, suppose $\|\boldsymbol{q}\|_{\mathcal{H}} = 1$ and $\|T\boldsymbol{q}\| = \epsilon > 0$. Then $\tilde{\boldsymbol{q}} = \boldsymbol{q}/\epsilon$ satisfies

$$\|T\tilde{\boldsymbol{q}}\| = 1, \qquad \|\tilde{\boldsymbol{q}}\|_{\mathcal{H}}^2 = \epsilon^{-2}. \tag{26}$$

Consequently, across families of response spaces for which the smallest positive singular value of T tends to zero, bounded combined responses do not imply a uniform bound on the allocation energy. Exact non-identifiability is the limiting case, but is not needed for the absence of a uniform bound across such families. □

An explicit injective family. Take $u_1 = (1, 0)^\top$ and $u_2 = (\cos\alpha, \sin\alpha)^\top$ with $0 < \alpha < \pi/2$. Their spaces are distinct, so T is injective. The collection $(u_1/\sqrt{2}, -u_2/\sqrt{2})$ has unit product norm and a fused norm $\sqrt{1 - \cos\alpha}$, which tends to zero as $\alpha \to 0$. This family has no nonzero exact output-invisible redistribution, yet exhibits arbitrarily large energy for a normalized combined response. The diagonal regularization in COIN remains positive in this limit.

Overlap versus interaction. An exact kernel and a nonzero cross-module Gram block are different properties. Two nonorthogonal lines in the plane have a trivial kernel but a nonzero interaction. The minimum-energy decomposition identifies redundant allocations when they exist; the joint objective also accounts for nonorthogonal interactions when the decomposition of each achievable response is unique. Proposition 4.3 concerns precisely this latter case.

Scope of the cancellation example. Equation (1) and the near-cancellation example in Sec. 2 characterize possible allocations in the allowed response spaces. They do not imply that a fresh, unpreconditioned gradient of one fused loss produces a nonzero, exactly canceling update. In fact, if $d_m = J_m^\top r$ for every block, then $v_m = J_m J_m^\top r$ and

$$\left\langle r, \sum_m v_m \right\rangle = \sum_m \|J_m^\top r\|^2. \tag{27}$$

If $\sum_m v_m = 0$, the right side is zero, so all these gradient directions vanish. Our allocation problem is broader: candidates also include base steps driven by independent modality losses, momentum, and weight decay, and the solve can combine them with a fresh task gradient. The counterexample shows why a criterion based only on the net response cannot bound arbitrary such allocations. It is not evidence that every baseline undergoes cancellation or diverges.

For the nearly aligned unit directions in Section 2, $\|tu_1 - tu_2\|^2 = 2t^2(1-\rho)$ and $\|tu_1\|^2 + \|-tu_2\|^2 = 2t^2$, giving the near-cancellation ratio in Sec. 2. At any fixed $\rho < 1$ the ratio is finite; the absence of a uniform bound is over a family with $\rho \uparrow 1$.

D PROOFS OF THE ALLOCATION PRINCIPLE

D.1 ADJOINT, RANGE, AND MINIMUM-ENERGY DECOMPOSITION

Theorem 3.1 has the following explicit expression. Put $P_m = U_m U_m^\top$ and $S = \sum_m P_m$. For $\delta \in \mathrm{range}(S)$,

$$v_m^{\min} = P_m S^\dagger \delta, \qquad \min_{\substack{v_m \in \mathcal{S}_m \\ \sum_m v_m = \delta}} \sum_m \|v_m\|^2 = \delta^\top S^\dagger \delta, \tag{28}$$

where $S^\dagger$ is the Moore–Penrose inverse. This formula is used for analysis; the algorithm does not construct an output-dimensional pseudoinverse.

For z in the output space and $\boldsymbol{v} \in \mathcal{H}$,

$$\langle T\boldsymbol{v}, z\rangle = \sum_m \langle v_m, z\rangle = \sum_m \langle v_m, P_m z\rangle . \tag{29}$$

Thus $T^* z = (P_1 z, \ldots, P_M z)$ and $TT^* = \sum_m P_m = S$. In finite dimensions, $\mathrm{range}(S) = \mathrm{range}(T)$ and $\ker(S) = \ker(T^*)$. One direct verification is

$$z^\top S z = \sum_m \|P_m z\|^2 , \tag{30}$$

which vanishes exactly when z is orthogonal to every $\mathcal{S}_m$.

Fix $\delta \in \mathrm{range}(S)$ and set $\boldsymbol{v}^{\min} = T^* S^\dagger \delta$. This collection is feasible because

$$T\boldsymbol{v}^{\min} = SS^\dagger \delta = \delta. \tag{31}$$

Its energy is

$$\left\|\boldsymbol{v}^{\min}\right\|_{\mathcal{H}}^2 = \delta^\top S^\dagger T T^* S^\dagger \delta = \delta^\top S^\dagger S S^\dagger \delta = \delta^\top S^\dagger \delta. \tag{32}$$

Any other feasible $\boldsymbol{w}$ differs from $\boldsymbol{v}^{\min}$ by a vector $\boldsymbol{q} \in \ker T$. Moreover,

$$\left\langle \boldsymbol{v}^{\min}, \boldsymbol{q}\right\rangle_{\mathcal{H}} = \left\langle S^\dagger \delta, T\boldsymbol{q}\right\rangle = 0. \tag{33}$$

The Pythagorean identity therefore gives Equation (6). Equality of energy occurs only for $\boldsymbol{q} = 0$, proving both optimality and uniqueness in Theorem 3.1. □

Why the functional objective follows. For every fixed combined response, the residual term $\|r - \delta\|^2 /2$ is constant over its equivalence class. Minimizing the module-energy term over that class produces $\delta^\top S^\dagger \delta/2$ by Equation (28). Minimizing next over achievable net responses gives the output-only form

$$\min_{\delta \in \mathrm{range}(S)} \frac{1}{2}\|r - \delta\|^2 + \frac{1}{2}\delta^\top S^\dagger \delta. \tag{34}$$

The first term fits the task direction; the second is the least module energy needed to realize the chosen response. This is an equivalent analytical description, not an extra stage of the implementation. Conversely, for any minimizing combined response, its unique minimum-energy allocation has exactly that objective value in Equation (7). Thus the two formulations have the same minimum and yield the same optimal combined response and module allocation. This derivation uses standard minimum-norm inverse theory (Ben-Israel & Greville, 2003); it introduces no extra parameter-space penalty into the implemented allocation.

The role of the module partition. The invariance results are stated for a fixed module partition and fixed response spaces. Within this setting, they are unchanged by redundant coordinates or alternative basis representations of the same response spaces.

D.2 PROOF OF THEOREM 4.1

On the finite-dimensional product space $\mathcal{H}$, the Hessian of Φ is $T^*T + I$, which is positive definite. Hence there is a unique minimizer. Let $e = r - T\boldsymbol{v}^*$ be its residual. The first-order condition in block m is

$$-P_m e + v_m^* = 0. \tag{35}$$

Summing gives $T\boldsymbol{v}^* = Se$, so $r = e + Se$ and $e = (I+S)^{-1}r$. This proves all response formulas in Equation (8). The inverse exists even when S is singular because its eigenvalues are nonnegative.

At the minimizer,

$$\Phi^* = \frac{1}{2}\|e\|^2 + \frac{1}{2}\sum_m \|P_m e\|^2 = \frac{1}{2}e^\top(I+S)e = \frac{1}{2}r^\top(I+S)^{-1}r. \tag{36}$$

Substituting $v_m = U_m a_m$ into the objective instead gives

$$\phi(a) = \frac{1}{2}r^\top r - b^\top a + \frac{1}{2}a^\top(G+I)a, \qquad \nabla\phi(a) = (G+I)a - b. \tag{37}$$

Consequently $a^* = (G+I)^{-1}b$, proving the small-system representation. The identity $U(G+I)^{-1}U^\top = S(I+S)^{-1}$ follows either from these two expressions for the same solution or from $(I+UU^\top)U = U(I+U^\top U)$. □

Why both terms in the objective matter. The residual term couples all modules through the sum of their responses. The positive energy term makes the minimizer unique, excludes gratuitous output-invisible motion, and prevents the inverse gains associated with almost-canceling responses from becoming unbounded in output space. Replacing the functional energy by a coefficient norm in an arbitrary unnormalized candidate basis would not have the same invariance. The regularizer becomes an ordinary ridge penalty (Hoerl & Kennard, 1970) only *after* the within-module response coordinates have been orthonormalized.

D.3 Representation invariance in Theorem 4.1

Let O_m be a $k_m \times k_m$ orthogonal matrix and $O = \mathrm{diag}(O_1, \ldots, O_M)$. If $U'_m = U_m O_m$, then $P'_m = U_m O_m O_m^\top U_m^\top = P_m$. Thus S and each expression in Equation (8) are unchanged. In coefficient coordinates, $U' = UO$ implies

$$G' = O^\top G O, \qquad b' = O^\top b, \qquad G' + I = O^\top(G+I)O. \tag{38}$$

Taking the inverse and multiplying by b' yields $a'^* = O^\top a^*$. Therefore the individual physical responses satisfy $U'_m a'^*_m = U_m a^*_m$.

An arbitrary full-rank change of spanning columns, a nonzero column scaling, or an addition of redundant columns leaves a response subspace unchanged. Once either spanning set is converted to an orthonormal basis, the two bases are related by an orthogonal matrix, so the preceding proof applies. This argument does not require an invalid general congruence formula for a pseudoinverse; it uses equality of ranges and orthogonal bases.

For the parameter statement, fix the same $\mathcal{D}_m$ and the same assigned response v^*_m. The solution to

$$\operatorname*{arg\,min}_{d \in \mathcal{D}_m} \|d\|^2 \quad \text{subject to} \quad J_m d = v^*_m \tag{39}$$

has a unique minimizer because the objective is strictly convex on an affine feasible set. Changing the basis of $\mathcal{D}_m$ does not change that set or its Euclidean norm, so the solution is unchanged. Appendix F.2 verifies that Equation (16) computes this solution. □

A precise distinction from network reparameterization. Let a differentiable change of parameter coordinates have local Jacobian R_m, so a physical direction is represented by $d'_m = R_m d_m$. The Jacobian becomes $J'_m = J_m R_m^{-1}$; if the candidate space is transported as $\mathcal{D}'_m = R_m \mathcal{D}_m$, then $J'_m \mathcal{D}'_m = J_m \mathcal{D}_m$ and the functional allocation is unchanged. However, a freshly computed Euclidean gradient transforms as $g'_m = R_m^{-\top} g_m$, not as $R_m g_m$ in general. A candidate generator based on Euclidean gradients is therefore not automatically covariant. The theorem applies whenever its response-space premise holds; it does not claim that the entire practical training procedure enforces that premise under every network reparameterization. This is distinct from the parameterization-invariant optimizer questions studied by Martens (2020); Yen et al. (2025).

E PROOFS OF THE COORDINATION CONSEQUENCES

E.1 PROOF OF PROPOSITION 4.3

For completeness, the example uses

$$u_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \qquad u_2^{(\chi)} = \begin{pmatrix} \chi\rho \\ \sqrt{1-\rho^2} \end{pmatrix}, \qquad r = \begin{pmatrix} 0 \\ (1-\rho^2)^{-1/2} \end{pmatrix}, \qquad \chi \in \{-1, 1\}. \tag{40}$$

The bases have unit norm, and their alignments with the target are $u_1^\top r = 0$ and $(u_2^{(\chi)})^\top r = 1$. Their cross inner product is $\chi\rho$. Thus

$$G + I = \begin{pmatrix} 2 & \chi\rho \\ \chi\rho & 2 \end{pmatrix}, \qquad b = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \qquad (G+I)^{-1} = \frac{1}{4-\rho^2} \begin{pmatrix} 2 & -\chi\rho \\ -\chi\rho & 2 \end{pmatrix}. \tag{41}$$

Multiplication gives Equation (10). The coefficient a_1^* is nonzero and changes sign with χ, although the individual statistics $(\|u_1\|, \|u_2\|, b_1, b_2) = (1, 1, 0, 1)$ are identical. Any deterministic rule depending only on those statistics must return the same answer in the two cases; it cannot equal both unique joint solutions. A randomized rule restricted to these statistics likewise cannot equal both distinct optima almost surely. □

The task benefit of the compensating module. For this example, omitting module 1 gives optimum value $\Phi^*_{-1} = \|r\|^2/2 - 1/4$, whereas using both gives $\Phi^* = \|r\|^2/2 - 1/(4-\rho^2)$. Therefore

$$\Delta_1 = \frac{\rho^2}{4(4-\rho^2)} > 0. \tag{42}$$

The improvement is not an artifact of reparameterizing the same individual update or of making its coefficient artificially large. Both response bases have unit norm, and the gain remains after the best module-2-only coefficient has already been chosen.

Role of the existing candidate space. The compensating response lies in an existing nonzero candidate space. If every candidate for module 1 had a zero output response, then $P_1 = 0$ and its optimal correction would vanish. If the full residual vanished, then $b = 0$ and every coefficient would vanish. These are direct consequences of the same linear system, not exceptions handled by a separate policy.

E.2 THE EXACT CONDITIONAL VALUE OF A MODULE

Let $S_{-m} = \sum_{n \neq m} P_n$, $K_{-m} = I + S_{-m}$, and $e_{-m} = K_{-m}^{-1} r$. These objects describe the same optimization problem with module m removed.

Theorem E.1 (Conditional value of a module)**.** *Let Φ^*_{-m} and Φ^* be the minima of Equation (7) without and with module m. Then*

$$\Delta_m := \Phi^*_{-m} - \Phi^* = \frac{1}{2} c_m^\top \big(I + U_m^\top K_{-m}^{-1} U_m\big)^{-1} c_m, \qquad c_m = U_m^\top e_{-m}. \tag{43}$$

In particular, $\Delta_m > 0$ if and only if $P_m e_{-m} \neq 0$. This is also equivalent to $v_m^* \neq 0$ in the full solution.

Proof. Theorem 4.1 gives

$$\Phi^*_{-m} = \frac{1}{2} r^\top K_{-m}^{-1} r, \qquad \Phi^* = \frac{1}{2} r^\top (K_{-m} + U_m U_m^\top)^{-1} r. \tag{44}$$

Because K_{-m} is positive definite, the Woodbury identity gives

$$\begin{aligned} &(K_{-m} + U_m U_m^\top)^{-1} \\ &\quad = K_{-m}^{-1} - K_{-m}^{-1} U_m (I + U_m^\top K_{-m}^{-1} U_m)^{-1} U_m^\top K_{-m}^{-1}. \end{aligned} \tag{45}$$

Subtracting the two minimum values and writing $c_m = U_m^\top K_{-m}^{-1} r$ proves Equation (43). The middle inverse is positive definite, so $\Delta_m > 0$ exactly when $c_m \neq 0$, equivalently $P_m e_{-m} \neq 0$.

To prove equivalence with participation in the full solution, put $H_m = U_m^\top K_{-m}^{-1} U_m$. Multiplying Equation (45) by r and then by $U_m^\top$ yields

$$a_m^* = U_m^\top e = c_m - H_m(I + H_m)^{-1} c_m = (I + H_m)^{-1} c_m. \tag{46}$$

The inverse is nonsingular, and U_m has full column rank. Hence $a_m^* = 0$, $v_m^* = 0$, and $c_m = 0$ are equivalent. Empty response spaces satisfy all zero statements by convention. □

Magnitude bounds. Since $K_{-m} \succeq I$, one has $0 \preceq H_m \preceq I$. Thus

$$\frac{1}{4} \|c_m\|^2 \le \Delta_m \le \frac{1}{2} \|c_m\|^2 . \tag{47}$$

These bounds show that conditional alignment is not merely a sign test. It also controls the exact marginal improvement up to universal constants. We do not use this result to switch modules on and off; a nonzero or zero contribution is already determined by the joint solve.

E.3 Proof of Theorem 4.4

From Equation (37), completing the square around the unique optimum gives, for any a,

$$\phi(a) - \phi(a^*) = \frac{1}{2}(a - a^*)^\top A (a - a^*), \qquad A = G + I. \tag{48}$$

The independent solution is $a^{\mathrm{ind}} = b/2$. As $E = G - I$, its stationarity residual under the joint objective is

$$A a^{\mathrm{ind}} - b = \frac{1}{2} E b. \tag{49}$$

Therefore $a^{\mathrm{ind}} - a^* = A^{-1} E b/2$. Substitution into Equation (48) proves

$$\phi(a^{\mathrm{ind}}) - \phi(a^*) = \frac{1}{8} b^\top E A^{-1} E b. \tag{50}$$

Since A^{-1} is positive definite, this value is zero if and only if $Eb = 0$.

For any coefficient vector a, Cauchy–Schwarz and orthonormality within blocks give

$$a^\top G a = \left\| \sum_m U_m a_m \right\|^2 \le M \sum_m \|U_m a_m\|^2 = M \|a\|^2 . \tag{51}$$

Thus $0 \preceq G \preceq MI$, and $I \preceq A \preceq (M+1)I$. Bounding A^{-1} between $I/(M+1)$ and I gives

$$\frac{\|Eb\|^2}{8(M+1)} \le \Xi \le \frac{\|Eb\|^2}{8}. \tag{52}$$

Under the basis change of Theorem 4.1, $E' = O^\top E O$, $A' = O^\top A O$, and $b' = O^\top b$. The quadratic form defining Ξ is therefore unchanged. □

Why an overlap norm is not enough. A nonzero E can act as zero on the particular vector b. For a concrete example, use one response direction in each of three modules with

$$G = \begin{pmatrix} 1 & \rho & 0 \\ \rho & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \qquad b = (0, 0, 1)^\top, \qquad 0 < \rho < 1. \tag{53}$$

The first two response spaces interact, but the task is aligned only with the third, so $Eb = 0$ and independent allocation is exact. These G, b are realizable, for example with two nonorthogonal directions in the first two coordinates and a third basis vector orthogonal to their plane. This is why the theory conditions on the task direction as well as the geometry.

The leading compensation term. When $\|E\|_2 < 2$, the convergent Neumann series gives

$$a^* = (2I + E)^{-1}b = \frac{1}{2}b - \frac{1}{4}Eb + R_2, \qquad \|R_2\| \le \frac{\|E\|_2^2 \|b\|}{8(1 - \|E\|_2/2)}. \tag{54}$$

The first term is independent allocation. The next term explicitly mixes the task-aligned responses of other modules into each block, and can be nonzero even when that block's own b_m is zero. This expansion is explanatory only; the implemented algorithm solves the full system and does not truncate the series.

E.4 Proof of the best-common-scale result

Let $A = G + I$ and $b \neq 0$. The best common rescaling of the independent direction lies on $\{tb : t \in \mathbb{R}\}$ and has coefficient $a^{\text{ray}} = (b^\top b)/(b^\top Ab)\, b$. Its remaining gap is

$$\phi(a^{\text{ray}}) - \phi(a^*) = \frac{1}{2}\left(b^\top A^{-1}b - \frac{(b^\top b)^2}{b^\top Ab}\right). \tag{55}$$

It vanishes exactly when b is an eigenvector of A. Thus allowing the independent allocation to choose its best common scalar still does not generally give the joint solution.

Proof. For $b \neq 0$, restrict Equation (37) to $a = tb$. The objective becomes

$$\phi(tb) = \frac{1}{2}\|r\|^2 - tb^\top b + \frac{t^2}{2}b^\top Ab. \tag{56}$$

Its unique minimizer is $t^* = (b^\top b)/(b^\top Ab) > 0$, with value $\|r\|^2/2 - (b^\top b)^2/(2b^\top Ab)$. The unrestricted minimum is $\|r\|^2/2 - b^\top A^{-1}b/2$. Their difference is Equation (55).

By Cauchy–Schwarz applied to $A^{-1/2}b$ and $A^{1/2}b$,

$$(b^\top b)^2 \le (b^\top A^{-1}b)(b^\top Ab). \tag{57}$$

Equality holds if and only if those two vectors are collinear, equivalently $Ab = \lambda b$ for some $\lambda > 0$. The result includes any scalar rescaling of $a^{\text{ind}} = b/2$, because its span is the same as that of b. When $b = 0$, both the independent and joint responses are zero. □

F Stability, lifting, and descent

F.1 Proof of Theorem 4.2

In addition to the energy identity and bound stated in Theorem 4.2, the solution obeys the following bounds (the condition number is stated for $k > 0$):

$$\|\delta^*\| \le \frac{M}{M+1}\|r\|, \qquad \kappa_2(G + I) \le M + 1. \tag{58}$$

For fixed response spaces, changing r by Δr changes the product response by at most $\|\Delta r\|/2$ and the combined response by at most $M\|\Delta r\|/(M+1)$.

At the optimum $v_m^* = P_m e$. Since P_m is an orthogonal projector,

$$\|v_m^*\|^2 = \langle e, P_m e\rangle = \langle e, v_m^*\rangle. \tag{59}$$

Summing and using $e = r - \delta^*$ gives

$$\sum_m \|v_m^*\|^2 = \langle r - \delta^*, \delta^*\rangle, \tag{60}$$

which is Equation (9). Completing the square yields

$$\sum_m \|v_m^*\|^2 = \frac{1}{4}\|r\|^2 - \|\delta^* - r/2\|^2 \le \frac{1}{4}\|r\|^2. \tag{61}$$

The bound is attained when $\delta^* = r/2$, for example when r belongs to a single active response space orthogonal to all others.

Each projector satisfies $0 \preceq P_m \preceq I$, hence $0 \preceq S \preceq MI$. The symmetric map $S(I+S)^{-1}$ has eigenvalues $\lambda/(1+\lambda)$ for $\lambda \in [0, M]$. Its norm is at most $M/(M+1)$, proving the combined response bound and the corresponding Lipschitz bound for fixed geometry.

The map from r to the product response is $B = T^*(I+S)^{-1}$. Its squared operator norm is the largest eigenvalue of

$$B^*B = (I+S)^{-1}S(I+S)^{-1}. \tag{62}$$

The scalar function $\lambda/(1+\lambda)^2$ is at most $1/4$ for $\lambda \geq 0$. Thus $\|B\|_2 \leq 1/2$, which proves the product-space perturbation statement.

Finally, for $k > 0$, $0 \preceq G \preceq MI$ as shown in Appendix E.3. The eigenvalues of $G + I$ lie in $[1, M+1]$, so $\kappa_2(G+I) \leq M+1$. When $k = 0$, the solve is skipped and all responses are zero; no empty-matrix condition number is needed. □

Shared directions and inaccessible residuals. With $Sq_j = \lambda_j q_j$ for orthonormal q_j, the spectral form is

$$\delta^* = \sum_j \frac{\lambda_j}{1+\lambda_j}\langle q_j, r\rangle q_j, \qquad e = \sum_j \frac{1}{1+\lambda_j}\langle q_j, r\rangle q_j. \tag{63}$$

If a vector q belongs to h spaces and is orthogonal to all others, then $Sq = hq$. For a target component $r_q = \langle q, r\rangle\, q$ with unit q, the residual component is $r_q/(h+1)$, and each of the h modules contributes that amount. If instead $q \in \ker S$, every module projection of q vanishes, so the target component is unchanged. These are properties of functional reachability, not claims about semantic similarity of modalities.

F.2 The parameter lift is a minimum-norm solution

Let $\ell_m = \dim \mathcal{D}_m$, let $Q_m \in \mathbb{R}^{p_m \times \ell_m}$ have orthonormal columns, and set $Z_m = J_m Q_m \in \mathbb{R}^{n_f \times \ell_m}$. Retaining all $k_m = \operatorname{rank}(Z_m)$ positive singular values gives the compact SVD

$$Z_m = U_m \Sigma_m W_m^\top, \qquad U_m^\top U_m = W_m^\top W_m = I_{k_m}, \qquad \Sigma_m \succ 0. \tag{64}$$

Here $U_m \in \mathbb{R}^{n_f \times k_m}$, $\Sigma_m \in \mathbb{R}^{k_m \times k_m}$, and $W_m \in \mathbb{R}^{\ell_m \times k_m}$. In particular, W_m is a matrix, whereas $\mathcal{D}_m$ is a parameter space. A library that returns the transpose of the right-vector matrix must be transposed before using the lift.

Any candidate direction is $d = Q_m z$ with $\|d\| = \|z\|$. The constraint $Z_m z = U_m a_m^*$ has minimum-norm solution

$$z^{\min} = W_m \Sigma_m^{-1} a_m^*, \qquad d_m^{\text{corr}} = Q_m W_m \Sigma_m^{-1} a_m^*, \qquad J_m d_m^{\text{corr}} = U_m a_m^*. \tag{65}$$

Every other solution differs by a vector in $\ker Z_m$, which is orthogonal to the columns of W_m. Pythagoras proves minimality. This is a minimum parameter norm *after* the functional allocation is fixed, not a parameter penalty used to select that allocation.

Positive-mode inversion versus numerical truncation. A thin SVD can still contain zero singular values when Z_m is rank deficient. Only retained positive values are inverted. If small positive values are also discarded, write the retained triplets as $(\widehat{U}_m, \widehat{\Sigma}_m, \widehat{W}_m)$. Then

$$Z_m \widehat{W}_m = \widehat{U}_m \widehat{\Sigma}_m, \qquad \widehat{\mathcal{S}}_m = \operatorname{range}(\widehat{U}_m) \subseteq \mathcal{S}_m, \tag{66}$$

but $Z_m = \widehat{U}_m \widehat{\Sigma}_m \widehat{W}_m^\top$ is generally false. The allocation uses $\widehat{\mathcal{S}}_m$ and lifts with the retained factors. Its lift still exactly realizes the assigned retained response in the linear model and is its minimum-norm realization in $\mathcal{D}_m$: coefficients of discarded nonzero modes must be zero, while any remaining freedom is in $\ker Z_m$. Subsequent formulas omit hats when referring to the retained system. Rank changes can change its response spaces, so exact invariance is conditional on those spaces remaining the same. If $k_m = 0$, its response and correction are defined as zero without inverting any matrix.

Since $Q_m W_m$ has orthonormal columns and all retained singular values are at least σ_*,

$$\|d_m^{\text{corr}}\|^2 = \left\|\Sigma_m^{-1} a_m^*\right\|^2 \leq \frac{\|a_m^*\|^2}{\sigma_*^2} = \frac{\|v_m^*\|^2}{\sigma_*^2}. \tag{67}$$

Summing over disjoint parameter blocks gives

$$\|d^{\mathrm{corr}}\|^2 \le \frac{\mathcal{E}}{\sigma_*^2}, \qquad \mathcal{E} = \sum_m \|v_m^*\|^2. \tag{68}$$

F.3 Conditional descent for a finite parameter step

Let $\sigma_* > 0$ bound the retained singular values of $J_m Q_m$ from below, and let $L_\theta > 0$ be a parameter-gradient Lipschitz constant along the correction segment. Use $\mathcal{E} = \sum_m \|v_m^*\|^2$.

Theorem F.1 (Conditional descent after lifting)**.** *For the lift d^{corr} in Equation (16), Equation (68) bounds its parameter norm. If $L(\cdot;\mathcal{B})$ is L_θ-smooth along the step and $0 < \eta_t \le \sigma_*^2/L_\theta$, then*

$$L(\bar\theta_t + \eta_t d^{\mathrm{corr}};\mathcal{B}) \le L(\bar\theta_t;\mathcal{B}) - \eta_t \|\delta^*\|^2 - \frac{\eta_t}{2}\mathcal{E}. \tag{69}$$

Proof. All quantities in this subsection are evaluated at the post-base state $\bar\theta_t$ on the fixed calibration map. By Equation (21) and the lift identity,

$$\nabla_\theta L(\bar\theta_t;\mathcal{B})^\top d^{\mathrm{corr}} = -\langle r, \delta^*\rangle = -\|\delta^*\|^2 - \mathcal{E}. \tag{70}$$

If the gradient is L_θ-Lipschitz along the correction segment, the standard descent inequality (Nocedal & Wright, 2006) gives

$$\begin{aligned} L(\bar\theta_t + \eta_t d^{\mathrm{corr}};\mathcal{B}) &\le L(\bar\theta_t;\mathcal{B}) - \eta_t(\|\delta^*\|^2 + \mathcal{E}) + \frac{L_\theta \eta_t^2}{2}\|d^{\mathrm{corr}}\|^2 \\ &\le L(\bar\theta_t;\mathcal{B}) - \eta_t\|\delta^*\|^2 - \eta_t\left(1 - \frac{L_\theta \eta_t}{2\sigma_*^2}\right)\mathcal{E}. \end{aligned} \tag{71}$$

For $0 < \eta_t \le \sigma_*^2/L_\theta$, the last parenthesis is at least $1/2$, proving Equation (69). □

Why the singular-value condition is needed. In the one-dimensional example $J = \epsilon$ and $r = 1$, the response subspace is all of $\mathbb{R}$ and the functional objective is $(1-\delta)^2/2 + \delta^2/2$. Its minimizer is $\delta = 1/2$. The minimum-norm parameter realization is $d = 1/(2\epsilon)$, which diverges as $\epsilon \to 0$. The matrix $G + I$ is still the scalar 2 and is perfectly conditioned. Thus good conditioning of the allocation solve and bounded functional motion do not imply good conditioning of the lift. The minimum singular value and nonlinear step size cannot be dropped from the finite-step guarantee.

For piecewise-smooth networks, the theorem applies on segments where the required smoothness bound holds. At activation-boundary crossings one must control the actual remainder separately; a local Jacobian alone cannot provide a global curvature bound. No loss-checking or backtracking operation is implicitly included in the theorem or the implementation.

F.4 Perturbation of the geometry and fixed-linear-model dynamics

The same resolvent that stabilizes the allocation provides a simple perturbation bound. Let $S, S' \succeq 0$ be sums of projectors for two sets of response spaces. Define $H_S = S(I+S)^{-1} = I - (I+S)^{-1}$. The inverse identity gives

$$H_{S'} - H_S = (I+S')^{-1}(S'-S)(I+S)^{-1}. \tag{72}$$

Both inverse factors have operator norm at most one. Consequently

$$\|H_{S'}r' - H_S r\| \le \|r' - r\| + \|S' - S\|_2 \|r\|, \qquad \|S' - S\|_2 \le \sum_m \|P_m' - P_m\|_2. \tag{73}$$

This controls changes in the actual response spaces, in addition to the exact basis invariance for unchanged spaces. It is not a claim that a rank-threshold crossing produces a small projector perturbation. Subspace geometry and numerical conditioning remain distinct (Björck & Golub, 1973; Golub & Van Loan, 2013).

To isolate the behavior of repeated corrections, consider an exactly linear model with squared output loss, fixed response spaces, no intervening base steps, and update $f_{t+1} = f_t + \eta H_S r_t$, where $r_t = y - f_t$. The residual evolves as

$$r_{t+1} = (I - \eta H_S) r_t. \tag{74}$$

For $0 < \eta \leq 1$, every reachable eigencomponent associated with $\lambda > 0$ is multiplied by $1 - \eta\lambda/(1+\lambda) \in [0, 1)$. If $\lambda^+_{\min}$ is the smallest positive eigenvalue of S, then

$$\left\|\Pi_{\mathrm{range}(S)} r_t\right\| \leq \left(1 - \eta \frac{\lambda^+_{\min}}{1 + \lambda^+_{\min}}\right)^t \left\|\Pi_{\mathrm{range}(S)} r_0\right\|. \tag{75}$$

The component in $\ker S$ remains constant. When $S = 0$, there is no reachable component and the update is identically zero. This exact result gives a useful interpretation of repeated functional corrections, but its fixed-linear-model assumptions are not the nonlinear stochastic training procedure.

Local geometry is not a population guarantee. The correction fits a training calibration batch using geometry estimated on that batch. Conditional descent or a small value of Ξ says nothing by itself about unbiasedness of the stochastic parameter update or improvement of the population risk. The training procedure changes the residual, candidates, and Jacobians over time. Establishing statistical generalization or global stochastic convergence would require further assumptions and analysis; none is claimed here.

F.5 A POSITIVE REGULARIZATION WEIGHT

For completeness, consider the family

$$\Phi_\lambda(\boldsymbol{v}; r) = \frac{1}{2}\|r - T\boldsymbol{v}\|^2 + \frac{\lambda}{2}\|\boldsymbol{v}\|^2_{\mathcal{H}}, \qquad \lambda > 0. \tag{76}$$

Its unique solution is

$$v_m = P_m(\lambda I + S)^{-1} r, \qquad \delta = S(\lambda I + S)^{-1} r, \qquad a = (G + \lambda I)^{-1} b. \tag{77}$$

To verify this, the stationarity condition is $\lambda v_m = P_m(r - \delta)$. If $e = r - \delta$, then $(\lambda I + S)e = \lambda r$, and substitution yields Equation (77). The corresponding identities are

$$\langle r, \delta\rangle = \|\delta\|^2 + \lambda \sum_m \|v_m\|^2, \quad \sum_m \|v_m\|^2 \leq \frac{\|r\|^2}{4\lambda}, \quad \kappa_2(G + \lambda I) \leq 1 + \frac{M}{\lambda} \quad (k > 0). \tag{78}$$

The energy bound follows by completing the square as in Equation (61). The remaining eigenvalues follow directly from $0 \preceq G \preceq MI$.

All results in the main text use $\lambda = 1$, as does our implementation. This extension explains the regularizer's role and is not an additional tuning mechanism or an unreported change to the algorithm. Letting $\lambda \downarrow 0$ removes the uniform condition-number bound and can reintroduce the near-cancellation amplification that motivated the energy cost.

G SIGNAL AVAILABILITY AND A SHARED-RESIDUAL EXAMPLE

This section separates signal availability from coordination. We first prove the shared-gradient bound in Section 2, then give a two-branch model in which one branch remains underdeveloped even as the shared loss vanishes. The result requires the stated shared-target and flow assumptions; unequal learning speeds alone are not sufficient.

G.1 WHAT A SMALL SHARED TASK GRADIENT IMPLIES

For a differentiable shared loss, $r = -\nabla_f L$ and the chain rule give $\nabla_{\theta_m} L = -J_m^\top r$. Submultiplicativity immediately gives Equation (4). Consequently, $r \to 0$ implies $\nabla_{\theta_m} L \to 0$ if $\|J_m\|_2$ stays bounded. Without this bound, a shrinking output gradient does not alone bound its parameter-space amplification.

A corresponding statement for remaining learning uses an integral rather than only an instantaneous limit. Suppose on $[t_0, \infty)$ a shared-loss-only flow satisfies $\dot\theta_m(t) = \eta_m(t) J_m(t)^\top r(t)$, with $\eta_m(t)\|J_m(t)\|_2 \leq C_m$. For every $t_1 \geq t_0$,

$$\|\theta_m(t_1) - \theta_m(t_0)\| \leq \int_{t_0}^{t_1} \|\dot\theta_m(t)\|\, dt \leq C_m \int_{t_0}^{t_1} \|r(t)\|\, dt. \tag{79}$$

Thus a small *remaining integrated signal* limits subsequent parameter movement. Merely knowing that a gradient tends to zero does not imply finite total movement. Momentum, changing data, weight decay, or extra supervision require their own analysis; they are not included in this pure gradient-flow statement.

G.2 A WEAK BRANCH CAN REMAIN SMALL WHILE THE SHARED LOSS VANISHES

Consider two scalar factorized branches, $z_m = \xi_m \zeta_m > 0$ for $m = 1, 2$, predicting $f = (z_1 + z_2)x$ with target $y = x$ and $\mathbb{E}[x^2] = 1$. The scalar z_m measures that branch's effective contribution, not an output-response coordinate of COIN. Under squared loss and gradient flow at fixed branch rates $\eta_m > 0$,

$$\dot{\xi}_m = \eta_m \zeta_m e_{\mathrm{sc}}, \qquad \dot{\zeta}_m = \eta_m \xi_m e_{\mathrm{sc}}, \qquad e_{\mathrm{sc}} = 1 - z_1 - z_2. \tag{80}$$

The scalar residual e_{sc} differs from the remaining task direction e in the main text. The difference $\xi_m^2 - \zeta_m^2$ is conserved. With balanced positive factors, $\xi_m(0) = \zeta_m(0) > 0$, balance persists and

$$\dot{z}_m = 2\eta_m z_m e_{\mathrm{sc}}, \qquad \frac{\log z_1(t)}{\eta_1} - \frac{\log z_2(t)}{\eta_2} = \frac{\log z_1(0)}{\eta_1} - \frac{\log z_2(0)}{\eta_2}. \tag{81}$$

This temporal conservation law belongs to the specified flow. It is *not* the output-preserving allocation relation in Equation (5).

Proposition G.1 (Persistent underlearning in the shared-target model)**.** *Let $z_1(0) = z_{1,0} \in (0, 1)$ and $z_2(0) = \epsilon > 0$, with $z_{1,0} + \epsilon < 1$. Under the fixed-rate balanced flow above, the shared loss tends to zero and, for every $t \geq 0$,*

$$0 < z_2(t) = \epsilon \left(\frac{z_1(t)}{z_{1,0}} \right)^{\eta_2/\eta_1} \leq \epsilon z_{1,0}^{-\eta_2/\eta_1}, \qquad z_1(t) + z_2(t) \longrightarrow 1. \tag{82}$$

For any fixed finite rate ratio and fixed $z_{1,0}$, the second branch's contribution therefore remains arbitrarily small as $\epsilon \to 0$, even after the fused predictor fits the target.

Proof. Differentiating the scalar residual gives

$$\dot{e}_{\mathrm{sc}} = -2(\eta_1 z_1 + \eta_2 z_2) e_{\mathrm{sc}}, \qquad 0 < e_{\mathrm{sc}}(t) \leq e_{\mathrm{sc}}(0) e^{-2\eta_1 z_{1,0} t}. \tag{83}$$

The residual stays positive and both branch contributions increase while their sum stays below one. In particular, $z_1(t) \geq z_{1,0}$ proves the exponential upper bound, so $e_{\mathrm{sc}}(t) \to 0$ and $L(t) = e_{\mathrm{sc}}(t)^2/2 \to 0$. Integrating the logarithmic identity in Equation (81) gives the equality in Equation (82); $z_1(t) \leq 1$ gives the bound. Monotonicity and boundedness yield limiting branch values, with the weak value bounded by the same quantity. □

The weak branch's future growth after t_0 also obeys

$$\log \frac{z_2(\infty)}{z_2(t_0)} = 2\eta_2 \int_{t_0}^{\infty} e_{\mathrm{sc}}(t)\, dt \leq \frac{\eta_2}{\eta_1 z_1(t_0)} e_{\mathrm{sc}}(t_0). \tag{84}$$

To obtain this bound, restart Equation (83) at t_0 using $z_1(t) \geq z_1(t_0)$. Once the shared residual is small, little multiplicative growth remains for the weak branch. For example, equal rates, $z_{1,0} = 1/2$, and $z_2(0) = \epsilon$ give $z_2(\infty) = \epsilon/(1/2 + \epsilon)$. The model learns the shared target but need not develop both branches.

G.3 WHY IMBALANCE ALONE IS NOT ENOUGH, AND WHY THE BASE SUPERVISION MATTERS

The model above has a redundant shared target: either branch can predict x. For orthonormal unit features and $f = \psi_1 x_1 + \psi_2 x_2$, $y = x_1 + x_2$, the loss is $[(\psi_1 - 1)^2 + (\psi_2 - 1)^2]/2$. Setting $\psi_1 = 1$ leaves $\partial L/\partial \psi_2 = \psi_2 - 1$. The first branch's convergence does not remove the second's signal. What matters is the remaining task gradient in directions the second branch can learn, not early convergence by itself. Related feature competition and deep-linear dynamics are studied by Pezeshki et al. (2021); Huang et al. (2022); Zhang et al. (2024b); Saxe et al. (2014); this example clarifies the role of the base losses rather than asserting a new universal starvation law.

In COIN, an encoder's unimodal loss can remain nonzero when the fused loss is already small. Its base step can then supply a candidate missing from the current fused-task gradient alone. The

correction compares the candidates' output effects and allocates them jointly. If the fused task direction is exactly zero, the correction is zero even when base supervision continues to train an encoder. Thus supervision preserves learning signals; functional coordination combines available responses. Neither the invariant relation nor a finite rescaling can create a missing task gradient.

H Implementation details

H.1 Base step and the five corrected modules

The available tasks use ResNet-based audio/visual encoders and, for text, GloVe embeddings with a recurrent encoder (He et al., 2016; Pennington et al., 2014; Cho et al., 2014). Standard orthogonalization and SVD algorithms are used for candidate and response bases (Golub & Van Loan, 2013).

Our implementation retains the two-encoder, three-residual-layer fusion scaffold and masked-input encoder supervision used in the reference nested-learning setup (Feng et al., 2026). Its fusion layers use ordinary full-input gradients in the base step rather than the reference multi-timescale error-filtering rule. The parameters are partitioned into disjoint groups for the two modality encoders, three residual fusion layers, and a readout classifier; the new component is their functional coordination. The two encoder groups receive their corresponding masked-input unimodal cross-entropies with weight 10. The fusion layers and readout receive the full-input cross-entropy with weight 1. The base optimizer is momentum SGD with momentum 0.9 and weight decay 10^{-4}; its learning-rate schedule remains the base schedule. The correction treats only the two encoders and three fusion layers as participants. The readout is updated by the base step and then held fixed during the functional correction.

For a group with base gradient $g_{t,m}^{\mathrm{base}}$, momentum state $h_{t,m}$, weight decay ω, and scheduled learning rate $\gamma_{t,m}$, the standard non-Nesterov base update can be expressed as

$$h_{t,m} = \mu h_{t-1,m} + g_{t,m}^{\mathrm{base}} + \omega\theta_{t,m}, \qquad d_{t,m}^{\mathrm{base}} = -\gamma_{t,m} h_{t,m}. \tag{85}$$

The method uses the actual resulting parameter delta as a candidate. It does not replace this delta with an approximate gradient or apply the base step twice. Optimizer states are maintained by the base optimizer; the functional correction does not create a second momentum process.

H.2 Calibration and candidate construction

The correction is evaluated after the base step. Its calibration examples come only from the training set and exclude the current update examples by sample identity; repeated physical video identities are also excluded where the dataset exposes them. The active fit batch contains 16 examples.

For each corrected module, the candidates are exactly the post-base reuse of its base delta and the negative gradient of the full-input cross-entropy freshly evaluated on the calibration fit batch. The old encoder gradient, an optimizer innovation, or a gradient from a different parameter state is not substituted for the fresh fused-task gradient. This is what makes the task direction, Jacobians, and direct task candidate refer to one common local problem.

The data-loading interface randomizes calibration roles after collation, rather than using sequence-length ordering to choose the fit subset. No validation or test examples are used to construct the geometry. A correction schedule is fixed from the number of update batches: the period is $\max\{1, \lceil B_{\mathrm{epoch}}/8 \rceil\}$, corresponding to a target of roughly eight corrections per epoch. The schedule is not chosen according to a module's measured strength or current loss.

H.3 Fixed-state derivatives and rank handling

For the calibration map, dropout is disabled and batch-normalization buffers are held fixed. The prior module training/evaluation flags and buffers are restored afterward. All candidate responses are derivatives of this same fixed-state fused predictor. The implementation uses a forward-mode Jacobian–vector product where the operator supports it; otherwise it uses an exact reverse-over-reverse product. The fallback changes the differentiation implementation, not the update rule. These

operations follow standard automatic differentiation principles (Pearlmutter, 1994; Baydin et al., 2018).

Candidate parameter vectors are normalized individually and orthogonalized by two passes of modified Gram–Schmidt. The implementation discards zero-norm candidates and uses a relative numerical tolerance of 10^{-6} for the orthogonalized directions; it calls the routine with absolute tolerance zero. Response matrices are then decomposed by SVD in the sample-mean output geometry; singular directions below the relative response-rank tolerance 10^{-6} are discarded. Response matrices and small linear-algebra reductions use double precision where implemented, while model parameters retain their training dtype. Rank decisions near machine precision are numerical approximations to the fixed-space problem, not additional learned policies.

After response bases are formed, the method constructs every cross-module block of $G = U^\top U$, adds I, and performs one Cholesky solve. No extra diagonal damping is selected using a task-specific criterion: the unit diagonal term is the functional-energy regularizer in Equation (7). Empty or all-zero response spaces contribute no columns. If the entire response basis is empty, the correction is the zero direction.

H.4 Applying the correction and its cost

The lifted correction uses a common default multiplier $\eta = 0.1$, scaled with the base learning-rate schedule. This is the implementation's fixed step convention, not an assertion that the sufficient bound in Theorem F.1 has been verified on every batch. Optional post-update diagnostics record quantities such as calibration loss and functional energies; they do not accept, reject, or backtrack the update.

Let $p_{\mathrm{tot}} = \sum_m p_m$ and $\ell_{\max} = \max_m \ell_m \le 2$. Candidate bases require $O(p_{\mathrm{tot}}\ell_{\max})$ storage, and their orthogonalization costs $O(\sum_m p_m \ell_m^2)$; no dense $p_{\mathrm{tot}} \times p_{\mathrm{tot}}$ matrix is formed. The final lift costs $O(\sum_m (p_m \ell_m + \ell_m k_m))$. Raw response matrices require $O(n_f \sum_m \ell_m)$ storage, retained bases $O(n_f k)$, and the Gram matrix $O(k^2)$, where $k \le 10$ here. Forming the Gram matrix costs $O(NCk^2)$ and its Cholesky factorization costs $O(k^3)$. Evaluating up to two Jacobian–vector products per module and a fresh fused-task gradient dominates the additional neural-network computation. These counts describe operations, not measured wall-clock speedups.

Algorithm and analysis. The implemented path consists of the base step, two fixed candidate types, within-module response normalization, full joint allocation, and a common lifted correction. The minimum-energy decomposition, marginal-utility formula, independent-allocation gap, and spectral interpretation analyze properties of this same procedure rather than introducing additional optimization components.

I Technical connections and the scope of the invariant relation

Problem-specific invariance. The invariant relation studied here is $\mathcal{N} = \{\boldsymbol{q} \in \mathcal{H} : \sum_m q_m = 0\}$ for attainable module responses, together with invariance of the allocation to representations of fixed response spaces. It is not a claim that every notion called "invariance" is the same. PRISM (Wang & Zhang, 2026) illustrates a related design lesson in a different problem: it places privacy clipping and noise on intrinsic LoRA updates rather than redundant factor coordinates. Its preserved object is a low-rank matrix update and its randomized privacy mechanism. Here, the preserved object is the fused first-order response of several modules, and the question is how to allocate that change. PRISM is a conceptual connection, not the source of the multimodal compensation criterion.

Background on coordinates, not a substitute for the multimodal problem. Natural-gradient analyses (Amari, 1998; Martens, 2020), K-FAC (Martens & Grosse, 2015), Path-SGD (Neyshabur et al., 2015), reparameterization examples (Dinh et al., 2017), and LoRA-RITE (Yen et al., 2025) explain why optimization measurements depend on parameter coordinates. They provide context for Proposition 2.1 and the transported-space discussion in Appendix D.3. The present allocation is not asserted to make a Euclidean candidate generator covariant under arbitrary network reparameterizations.

Multiple objectives versus multiple responding modules. GradNorm balances task-gradient magnitudes (Chen et al., 2018); PCGrad removes conflicting task-gradient components (Yu et al.,

2020); other methods seek Pareto, conflict-averse, or bargaining solutions (Sener & Koltun, 2018; Liu et al., 2021; Navon et al., 2022). Their usual object is several objectives acting on shared parameters. In our local problem, distinct parameter blocks act on one shared prediction. Moving them into output space exposes cross-module interactions that would be absent from inner products between disjoint parameter blocks.

Subspaces, minimum norm, and regularized fitting. Local target fitting (Amid et al., 2022) and low-dimensional update optimization (Vinyals & Povey, 2012) are established tools. The minimum-energy expression uses standard generalized-inverse theory (Ben-Israel & Greville, 2003); orthogonalized coordinates convert the functional penalty into ridge regularization (Hoerl & Kennard, 1970). Partial-residual fitting connects to additive models (Buja et al., 1989). The projector sum connects to fusion frames (Casazza et al., 2008), and the singular values of $U_m^\top U_n$ are the cosines of principal angles between the response spaces (Björck & Golub, 1973). These foundations support the proofs. The coordination conclusions condition on both those spaces and the current task direction, rather than treating geometric overlap alone as evidence that a module should move.

Routed experts and the current instantiation. Sparse mixture-of-experts systems introduce routing and expert-utilization questions (Shazeer et al., 2017; Fedus et al., 2022); LIMoE studies multimodal routing (Mustafa et al., 2022). A fixed differentiable routing pattern can be included in the Jacobians of our formulation. A change of top-k routing can change the active local map and requires a separate analysis. The evaluated implementation uses modality encoders and residual fusion modules; no router or expert load-balancing result is claimed. This distinction concerns the evaluated architecture, not whether expert updates could also be studied in a common functional space.

J Experimental settings and additional results

This appendix describes the training and reporting procedure, the controls used to test functional allocation, and the measurements behind the figures. All saved-state comparisons evaluate alternative corrections at the same network state.

J.1 Data, training, and measurement units

Data and architecture. We use ResNet audio/visual encoders and a GloVe-based recurrent text encoder (He et al., 2016; Pennington et al., 2014; Cho et al., 2014), with fusion as in Sec. 5. Table 4 gives the splits: AVE training excludes videos shared across splits; VGGSound50 uses class-stratified validation from the training set; FOOD101 uses its released development set for validation.

Table 4: Dataset splits and training budgets used in our experiments. Batch is the number of examples per base update.

Dataset	Classes	Train	Validation	Test	Batch	Epochs
AVE	28	3,320	402	402	32	100
AVSBench	23	3,452	740	740	32	100
VGGSound50	50	23,455	2,499	2,499	32	100
FOOD101	101	62,971	5,000	22,715	24	70

Optimization. We train with momentum SGD (0.9 momentum, 10^{-4} weight decay). Masked-input encoder CE has weight 10; full-input fusion/readout CE has weight 1. Table 5 gives learning rates and correction periods. Rates drop by a factor of 0.1 at the listed epoch. The common correction multiplier starts at 0.1 and follows that rate scale. Parameter orthogonalization and response-rank truncation both use a relative tolerance of 10^{-6}. Response algebra uses double precision where implemented; parameters retain their training precision.

Fit and audit data. Each correction fits 16 training examples that exclude the current base-update identities, with dropout and normalization state fixed. An audit batch is a different set of training examples used only to evaluate the proposed correction. It is disjoint from the fit and temporary base-update examples in that measurement. In fixed-state resampling, each state has 20 repeats, nested fit sizes of 16, 32, and 64, and a fixed 128-example audit batch. Parameter candidates are rebuilt on each fit batch. The main allocation comparison uses 16 fit examples.

Table 5: Learning rates and correction schedules. The learning rate is multiplied by 0.1 at the listed epoch. Correction period is the number of base updates between corrections.

Dataset	Encoder/readout LR	Fusion LR	Decay epoch	Correction period
AVE	0.001	0.003	[70]	13
AVSBench	0.001	0.003	[70]	14
VGGSound50	0.001	0.003	[70]	92
FOOD101	0.001	0.001	[50]	328

Replication and aggregation. All baselines are run with three random seeds $\{0, 1, 2\}$, and results are reported as mean $\pm$ sample standard deviation. For CE effects, we average repeats within state, states within trajectory, and trajectories within seed. Geometry uses state medians followed by averaging within each seed. Means and sample standard deviations give equal weight to the three seed summaries. The earlier removal and finite-step probes use three source trajectories per task; their event measurements are averaged within seed before computing cross-seed summaries.

J.2 Benchmark results

Accuracy is mean $\pm$ sample standard deviation over three seeds. Table 6 evaluates the trained model with the other modality masked. COIN combines the highest fusion means in Table 1 with strong single-modality recognition. Its visual-branch means are higher than every displayed baseline on all four tasks, and its audio means lead on AVE and VGGSound50. The fusion result is therefore accompanied by useful modality-specific predictions, consistent with preserving encoder learning while coordinating the fused update.

Table 6: Single-modality accuracy (%, mean $\pm$ sample SD). Audio/Video applies to the first three datasets; Text/Image to FOOD101.

Method	Input	AVE	AVSBench	VGGSound50	FOOD101
Summation	Audio/Text	53.73 ± 1.42	76.67 ± 0.80	40.55 ± 0.26	82.40 ± 0.09
	Video/Image	19.32 ± 1.73	25.36 ± 0.39	19.04 ± 0.82	29.42 ± 0.16
Concatenation	Audio/Text	53.48 ± 1.81	77.66 ± 1.62	40.78 ± 0.27	87.50 ± 0.10
	Video/Image	18.33 ± 0.59	22.89 ± 2.33	19.33 ± 0.35	30.11 ± 0.38
OGM-GE (Peng et al., 2022)	Audio/Text	50.66 ± 1.19	74.41 ± 2.02	38.64 ± 0.14	81.66 ± 0.13
	Video/Image	19.82 ± 1.73	26.37 ± 2.52	22.21 ± 0.18	30.39 ± 0.24
AGM (Li et al., 2023)	Audio/Text	54.23 ± 2.54	79.28 ± 0.90	42.99 ± 0.23	79.57 ± 0.16
	Video/Image	17.49 ± 1.75	24.68 ± 0.92	20.69 ± 0.64	30.30 ± 0.48
MMCosine (Xu et al., 2023)	Audio/Text	55.72 ± 0.20	81.89 ± 0.80	43.50 ± 0.49	82.89 ± 0.08
	Video/Image	20.98 ± 0.31	35.59 ± 3.89	25.65 ± 2.71	43.59 ± 0.16
MMPareto (Wei & Hu, 2024)	Audio/Text	61.36 ± 1.12	81.80 ± 0.95	45.02 ± 0.09	84.06 ± 0.05
	Video/Image	33.17 ± 1.12	54.05 ± 0.40	36.05 ± 0.69	49.60 ± 0.04
MLA (Zhang et al., 2024a)	Audio/Text	64.10 ± 0.47	75.81 ± 2.65	46.91 ± 0.46	83.53 ± 0.06
	Video/Image	32.18 ± 1.24	40.99 ± 0.17	35.16 ± 0.92	51.70 ± 2.36
DGL (Wei et al., 2025b)	Audio/Text	63.60 ± 1.52	83.97 ± 1.64	48.39 ± 0.57	67.51 ± 0.76
	Video/Image	24.87 ± 1.33	63.54 ± 0.77	39.44 ± 0.47	56.51 ± 0.22
ARL (Wei et al., 2025a)	Audio/Text	62.50 ± 1.61	83.29 ± 0.29	46.31 ± 0.45	67.56 ± 0.84
	Video/Image	37.50 ± 1.39	56.23 ± 1.12	38.91 ± 0.15	56.32 ± 0.10
InfoReg (Huang et al., 2025)	Audio/Text	63.25 ± 0.97	81.80 ± 0.51	46.42 ± 0.36	84.26 ± 0.03
	Video/Image	31.44 ± 0.93	52.88 ± 0.55	35.43 ± 0.72	40.37 ± 0.24
MoNet (Feng et al., 2026)	Audio/Text	63.10 ± 0.85	80.50 ± 0.34	46.14 ± 1.41	84.91 ± 0.09
	Video/Image	39.39 ± 1.19	65.50 ± 1.06	40.88 ± 1.18	54.14 ± 0.46
IBML (Qin et al., 2026)	Audio/Text	59.95 ± 0.71	78.69 ± 1.12	43.74 ± 0.53	84.02 ± 0.05
	Video/Image	35.57 ± 0.93	59.64 ± 1.12	39.83 ± 0.21	55.73 ± 0.46
COIN (ours)	Audio/Text	67.16 ± 0.75	82.43 ± 0.27	48.98 ± 0.28	85.63 ± 0.11
	Video/Image	43.03 ± 1.08	69.19 ± 1.11	42.82 ± 0.88	58.59 ± 0.19

J.3 Matched-progress allocation and response geometry

A control that keeps the same predicted progress. Using the response bases defined in Sec. 4.3, the joint coefficients are $a_J = (G + I)^{-1}b$. We compare them with two controls restricted to the direction b. The first chooses the best common scale for the local objective; the second matches the joint predicted progress:

$$a_R = \frac{b^\top b}{b^\top (G + I)b}\, b, \qquad a_P = \frac{b^\top a_J}{b^\top b}\, b. \tag{86}$$

If $b = 0$, all three corrections are zero. Otherwise a_R and a_P have the same direction, with different scales, and $b^\top a_P = b^\top a_J$. We measure cost saving as $1 - C(a_J)/C(a_P)$, where $C(a) = \|Ua\|^2 + \|a\|^2$, and allocation angle as $\arccos(\langle a_J, a_R\rangle/(\|a_J\|\|a_R\|))$. The angle compares module-response coefficient vectors, not parameter vectors or just the summed output. Table 7 gives the values plotted in Fig. 1.

Table 7: Matched-progress functional allocation with 16 fit examples. State medians are averaged within seed; entries are mean ± sample SD over three seed summaries.

Dataset	Cost saving (%)	Angle (°)
AVE	3.01 ± 0.07	15.47 ± 0.23
AVSBench	2.87 ± 0.25	15.57 ± 0.81
VGGSound50	2.47 ± 0.30	13.99 ± 0.43
FOOD101	7.75 ± 0.52	20.42 ± 0.34

The matched-progress control gives the scalar direction the scale needed to attain the same first-order progress. Positive cost saving therefore isolates the benefit of joint direction choice: COIN uses cross-module information to achieve that progress with less movement.

Where do the interactions occur? For two nonempty module response bases, define the normalized overlap $\|U_m^\top U_n\|_F^2/\min(k_m, k_n)$, where $\|\cdot\|_F$ sums squared matrix entries before taking a square root. This is the average squared canonical correlation and lies between zero and one. Zero means orthogonal response spaces; values near one mean that the modules can change the prediction in strongly aligned directions.

The fusion blocks have pairwise overlaps of approximately 0.92–0.99 despite occupying disjoint parameter blocks. This is precisely the interaction retained by G_{mn} in Eq. (12). Encoder relations are more selective, reinforcing the value of measuring each pair rather than assigning all modules the same scalar description. All five modules retain two numerical response directions each, giving ten directions in total.

J.4 Removing one correction and adapting the others

At each saved scalar-ray trajectory state, we evaluate the joint correction and two interventions. *Remove* zeros the correction for one module. *Re-solve* excludes that module's response space and recomputes the remaining coefficients. If D_J is the audit CE decrease of the original joint update and D_{-m} that of an intervention, its cost is $D_J - D_{-m}$. Zero matches the original progress; a positive value loses progress, and a negative value improves it. *Recovered* progress is the Remove cost minus the Re-solve cost, calculated within each seed before aggregation.

Re-solving lowers mean cost in 19 of 20 dataset–module comparisons, including all 12 fusion-block comparisons. Across the fusion blocks it recovers roughly 70–84% of the removal cost, computed from the displayed means. The adjustment therefore depends on which other modules remain available, as predicted by the conditional-participation analysis in Appendix E.2. All inputs and network modules remain present during these tests; only their correction assignments change.

J.5 Basis changes and numerical response spaces

Theorem 4.1 conditions functional invariance on fixed response spaces. To examine their numerical representation, the basis stress test applies transforms with condition numbers 1, 10^2, 10^4, and 10^8 to 2,828 cached response problems. A larger condition number means a more unevenly scaled, harder-to-resolve basis. The plot records whether every module retains its numerical rank.

All 2,828 cases retain their ranks through condition number 10^2; 2,710 (95.8%) do so at 10^4. At 10^8, the rank threshold changes the retained problem in every case. The stress test thus identifies a

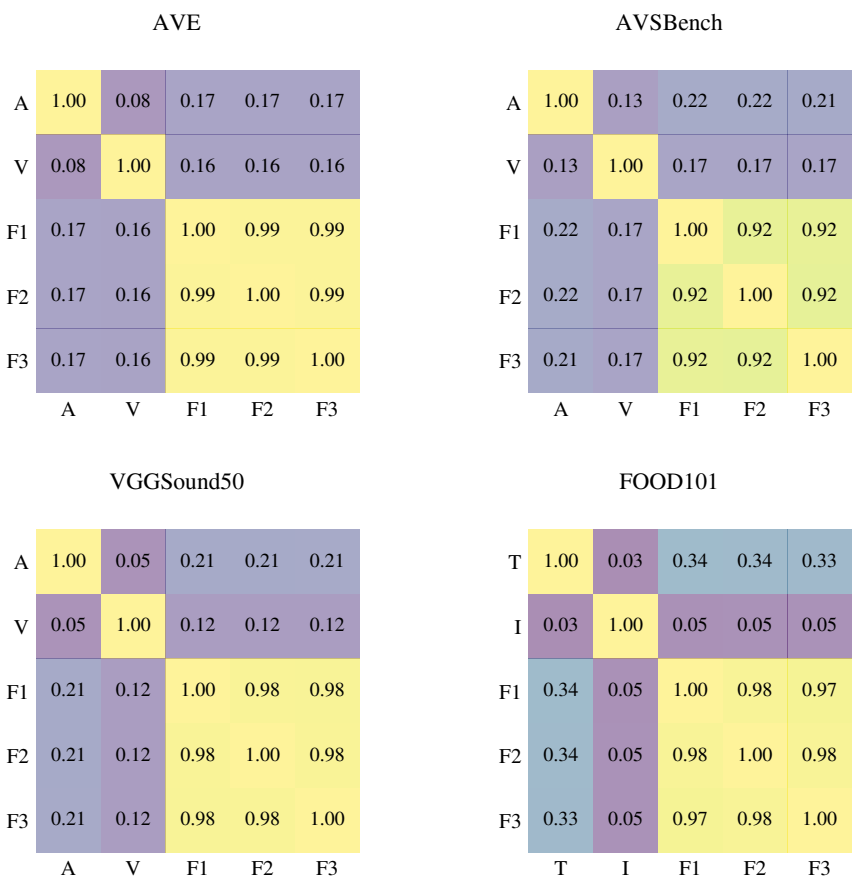


Figure 3: **Distinct parameter blocks have related output effects.** Both axes list modules: A/V are audio/video, T/I text/image, and F1–F3 the fusion blocks. Cell values are mean overlaps across three source seeds; purple/blue shades mark lower overlap and yellow marks higher overlap. Diagonal entries compare a module with itself. The lower-right off-diagonal blocks reveal fusion-to-fusion interaction. The shared scale is $[0, 1]$; higher overlap indicates stronger relations to coordinate, not better accuracy.

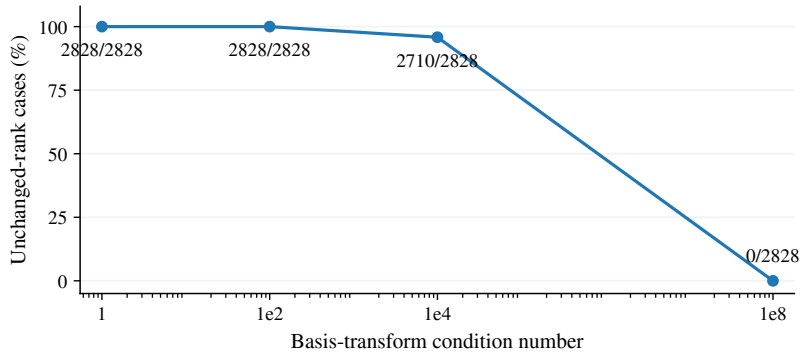


Figure 4: **Rank retention under increasingly ill-conditioned bases.** The horizontal axis is the transform condition number on a logarithmic scale; the vertical axis is the percentage of cached cases with unchanged module ranks. Blue points give the measured fractions, labels give their counts, and the line connects tested settings. Higher values mean more stable rank recovery; 100% retains all dimensions in every case.

Table 8: Audit CE progress lost after a correction is removed, and recovered by re-solving (all in 10^{-6} units). Lower Remove/Re-solve cost and higher recovery are better. Entries are means $\pm$ sample SD across three seeds; recovery is paired within seed.

Dataset	Module	Remove	Re-solve	Recovered
AVE	Audio	183.69 ± 167.50	59.84 ± 50.73	123.85 ± 116.87
	Video	103.56 ± 161.26	41.24 ± 72.64	62.32 ± 88.76
	Fusion 1	298.49 ± 134.81	75.45 ± 25.71	223.05 ± 109.44
	Fusion 2	292.76 ± 139.24	71.70 ± 28.92	221.06 ± 110.39
	Fusion 3	302.39 ± 141.08	77.73 ± 27.45	224.66 ± 113.99
AVSBench	Audio	186.45 ± 44.21	43.89 ± 15.85	142.56 ± 32.10
	Video	215.45 ± 139.08	50.15 ± 92.64	165.30 ± 76.91
	Fusion 1	601.47 ± 295.90	168.94 ± 108.22	432.53 ± 187.75
	Fusion 2	599.90 ± 288.62	168.22 ± 101.74	431.68 ± 187.07
	Fusion 3	601.43 ± 292.63	168.49 ± 103.97	432.93 ± 188.81
VGGSound50	Audio	4.34 ± 6.87	2.50 ± 5.60	1.84 ± 1.48
	Video	1.58 ± 5.41	2.10 ± 3.41	-0.51 ± 2.15
	Fusion 1	74.35 ± 47.39	21.83 ± 14.34	52.52 ± 33.17
	Fusion 2	72.75 ± 44.97	21.81 ± 14.14	50.94 ± 30.94
	Fusion 3	75.97 ± 46.67	22.58 ± 13.95	53.38 ± 33.00
FOOD101	Text	5.20 ± 4.71	-0.35 ± 3.55	5.55 ± 3.00
	Image	4.51 ± 1.59	4.43 ± 1.37	0.08 ± 0.59
	Fusion 1	12.67 ± 6.47	3.18 ± 1.93	9.49 ± 4.55
	Fusion 2	12.22 ± 6.53	1.95 ± 1.28	10.27 ± 5.62
	Fusion 3	13.18 ± 5.86	3.80 ± 1.96	9.38 ± 5.21

well-resolved regime for the fixed-space comparison and makes the effect of numerical truncation explicit. Rank retention tests dimensions; the invariant object in the theorem remains the response space itself.

J.6 Does the parameter step realize the chosen response?

We apply the joint parameter correction at saved states and compare its predicted first-order CE decrease with the decrease measured after the finite step. The fit batch constructs the update; the audit batch supplies different training examples for checking its output effect. In both plots, each point is one recorded probe, horizontal position is predicted decrease, and vertical position is actual decrease. Positive values mean lower loss. The dotted diagonal is exact prediction: points close to it indicate an accurately predicted CE change, while its positive branch also represents task improvement. The symmetric-log scale is linear within $\pm 10^{-7}$ and logarithmic outside, so both signs and near-zero values remain visible.

Across both batches, the points closely follow the equality line over several orders of magnitude. The parameter lift therefore produces CE changes consistent with the local first-order prediction on actual networks, rather than only in the coefficient solve. For a relative-error summary, we use probes with predicted fit CE decrease above 10^{-6}, take the median relative audit-response error within each seed, and then report the three-seed mean and SD. The resulting means are 0.22%, 0.11%, 0.30%, and 0.44% (Table 9); the scatter plots retain all probes.

Table 9: Finite-step response diagnostics at saved scalar-ray trajectory states. Errors are relative fractions (lower is better); overlap measures fusion-direction alignment. Each task has three source seeds.

Dataset	Seeds	Probes	Active audit error	Fusion overlap
AVE	3	75	0.0022 ± 0.0005	0.989 ± 0.001
AVSBench	3	75	0.0011 ± 0.0003	0.914 ± 0.003
VGGSound50	3	75	0.0030 ± 0.0004	0.982 ± 0.001
FOOD101	3	54	0.0044 ± 0.0034	0.975 ± 0.001

Fit/audit batch sizes are 16/32. Probes occur every 32 corrections; each count totals the three source trajectories. Active error uses seed-level medians for probes with predicted fit CE decrease above 10^{-6}. Fusion overlap averages the three unordered fusion-block pairs within seed.

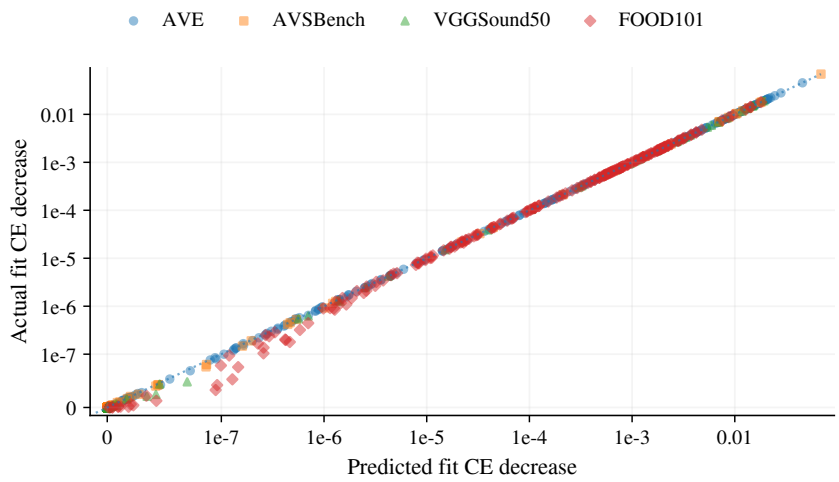


Figure 5: **Predicted and measured change on the fitting batch.** Blue circles, orange squares, green triangles, and red diamonds denote AVE, AVSBench, VGGSound50, and FOOD101. Each point compares predicted fit CE decrease (horizontal) with actual decrease (vertical); closeness to the dotted equality line indicates accurate realization. Points in the upper-right region show positive progress. All recorded joint-correction probes on the three-seed scalar-ray trajectories are shown.

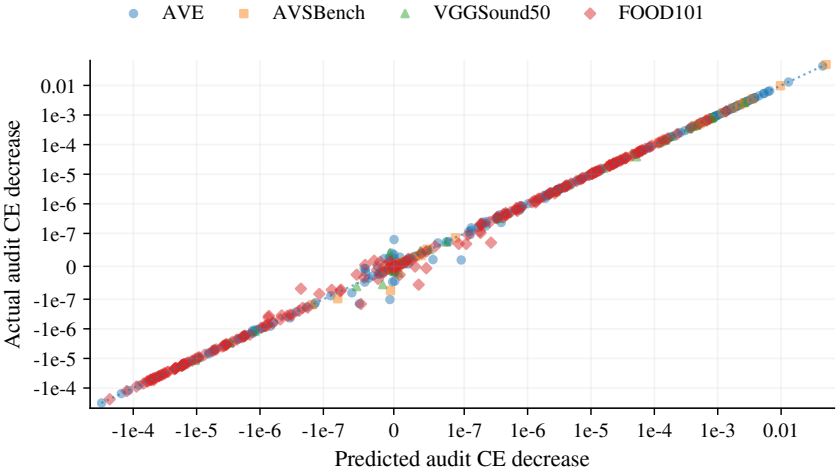


Figure 6: **The same response check on separate audit examples.** Axes, dataset symbols, and the equality line have the same meaning as in Fig. 5, now using audit CE. Near-diagonal points indicate accurate prediction; points on its positive branch also show loss reduction. Negative coordinates denote predicted or measured loss increases, not missing data. The plot keeps all recorded signs and magnitudes.

A separate numerical accounting check decomposes actual CE change into predicted first-order progress, task-projected output-realization error, and CE curvature. Across 11,312 recorded intervention/split combinations, its maximum identity residual is approximately 1.33×10^{-15}, confirming consistency of the recorded CE accounting.

J.7 Response to the correction multiplier

At each saved state, keep the joint direction fixed and multiply the default correction size by $s \in \{0.25, 0.5, 1, 2\}$. Figure 7 plots the measured CE decreases. Moving right increases the step; higher positive values mean a larger reduction in loss. These are alternative finite steps from the same states, so the comparison separates response scale from retraining.

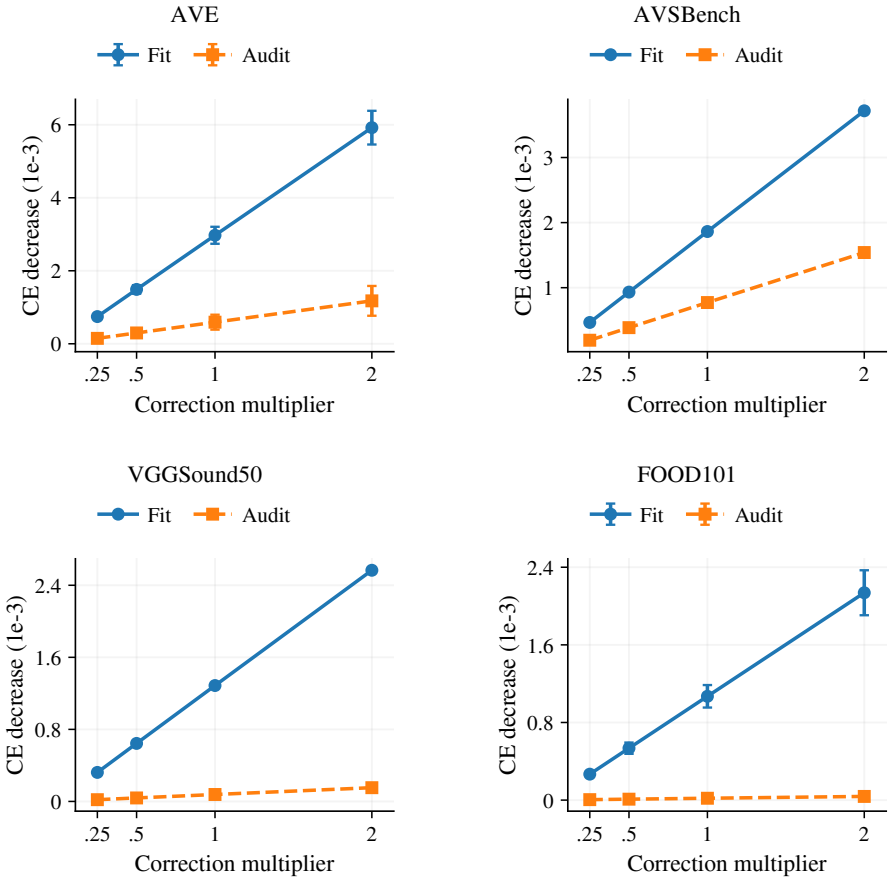


Figure 7: **Measured progress grows with correction size.** The horizontal axis is the multiplier of the default step; the vertical axis is actual CE decrease in 10^{-3} units. Blue circles with solid lines show fit data; orange squares with dashed lines show audit data. Points are three-seed means and bars are sample SD. An upward trend means larger steps yield more average loss reduction over the tested range; panel scales differ.

Both curves remain positive and increase over the tested multipliers. Their different heights reflect progress on the fitted examples versus separate audit examples. To check proportionality, let $D(s)$ be one seed's event-averaged CE decrease and plot $D(s)/(sD(1))$. A linear response equals one; values above or below one indicate more or less progress per unit step than at the default dose.

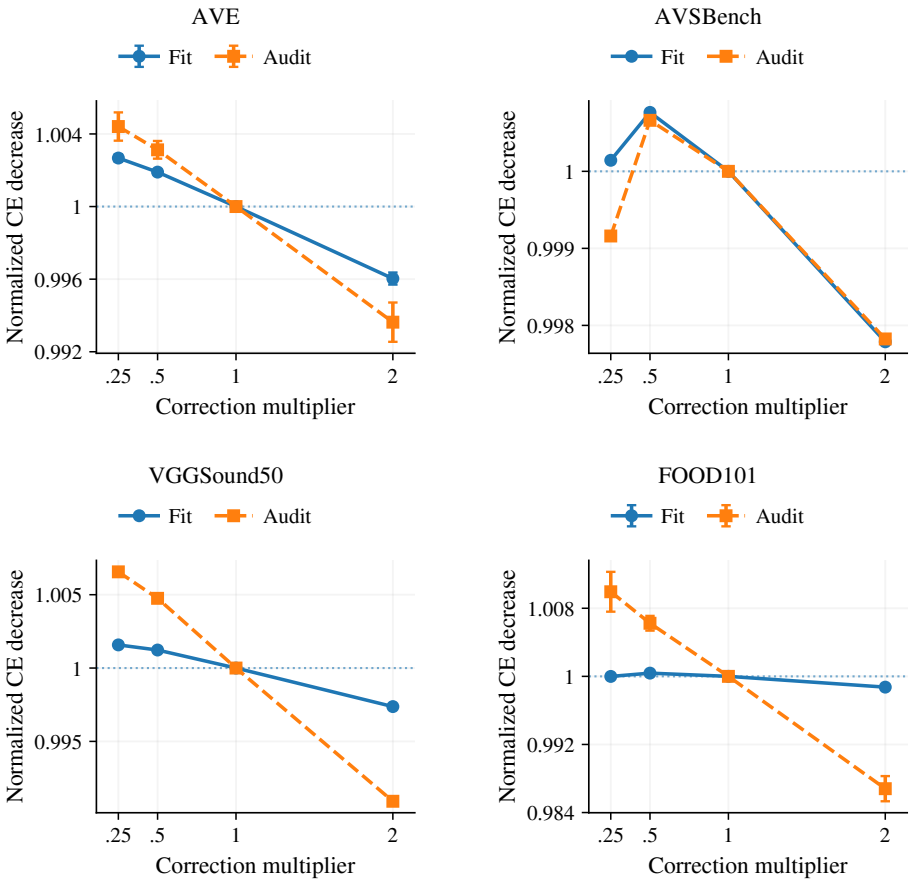


Figure 8: **Progress is close to proportional over the tested step range.** The horizontal axis is correction multiplier; the vertical axis is normalized decrease $D(s)/(sD(1))$. Blue solid circles and orange dashed squares denote fit and audit batches. The dotted line at one is exact proportionality, so closeness to this line—not a higher value—indicates a more linear response. Means and SDs are computed after normalization within each seed; $s = 1$ equals one by construction.

Across task–seed summaries, fit ratios range from 0.9957 to 1.0029 and audit ratios from 0.9852 to 1.0124. Thus the recorded mean responses stay within roughly 1.5% of proportional scaling over $s \in [0.25, 2]$, supporting the local response model used to select and realize the update.