LEARNING REASONING BY SUPERPOSITION: WHEN IT WORKS AND WHICH INTERMEDIATE STATES TO SUPERVISE

Anonymous authors
Paper under double-blind review

ABSTRACT

Reasoning often requires keeping several possibilities until later evidence shows which ones matter. A Transformer can do this by representing a set of candidate states in one hidden state and updating the whole set over several steps. We call this computation *reasoning by superposition*. Prior work shows that carefully chosen parameters can represent it, but representation alone does not explain whether gradient training can learn it. We give sufficient conditions under which training learns the full sequence of target intermediate states, rather than only the final answer, and generalizes to new tasks from the same distribution. The conditions have a simple meaning: the model must preserve the current candidate set and be able to produce the next one, while the training signal must directly correct inaccurate intermediate states. We then study a practical consequence. When intermediate-state labels are limited, which steps should be supervised? For a stated model of how supervision reduces error, we derive the optimal continuous allocation of a fixed label budget. The solution gives more labels to steps where one label is expected to reduce more error at lower cost, and it motivates a practical rule based on measured improvement. Together, these results explain when reasoning by superposition is learnable and how limited supervision should be distributed across its intermediate states.

1 INTRODUCTION

Many reasoning problems require a model to keep several candidates until later evidence rules some of them out. Consider graph reachability. At one step, several locations may be reachable, and each may lead to new locations at the next step. Following only one location can discard a valid route. A hidden state that represents the whole candidate set can instead update all currently valid candidates together. This is the computation we call reasoning by superposition.

This view differs from writing every candidate as a separate text chain. Chain-of-thought methods express intermediate computation in tokens (Wei et al., 2022), while continuous reasoning methods perform part of that computation in hidden states (Hao et al., 2024; Deng et al., 2024; Pfau et al., 2024; Goyal et al., 2023). Prior work shows how a Transformer with carefully chosen parameters can represent several candidates in one hidden state (Zhu et al., 2025). Representation alone, however, does not explain learning. A model may be able to express the correct update yet fail to discover it from data. It may also produce the correct final answer while using inaccurate intermediate states.

We study tasks that update a candidate set step by step. We call the sequence of intermediate states the *reasoning trajectory*. Our learnability result joins two requirements. First, the model must preserve the current candidate set and be able to produce the next one. Second, the training signal must correct an inaccurate intermediate state. Under the stated conditions, gradient training learns the target reasoning trajectory to any prescribed accuracy and generalizes to new tasks from the same distribution.

Intermediate-state supervision is useful because an early error can affect every later step. If the model removes a valid candidate too soon, all later states are built from an incomplete set. A label for the inaccurate intermediate state gives training a direct way to repair that step. Prior work has supervised intermediate reasoning steps (Uesato et al., 2022; Lightman et al., 2024). Here, an *intermediate-state*

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

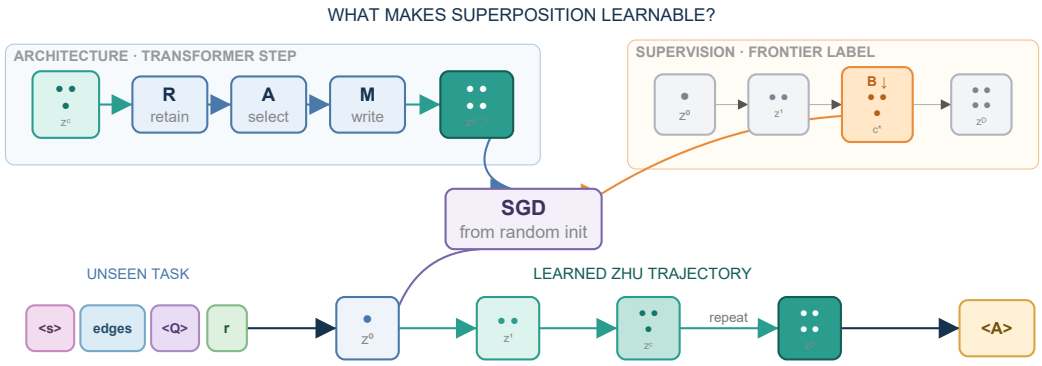

Figure 1: Architecture and supervision jointly make superposition learnable. The model preserves the current candidate set and adds newly reachable candidates. Intermediate-state supervision corrects inaccurate steps and allocates labels by expected error reduction per unit cost.

label identifies the target candidate set at one step. In the sequential case covered by our analysis, training corrects the earliest inaccurate state and then advances while keeping earlier states accurate.

This leads to a budget question. Labeling every intermediate state can be expensive, and equal allocation may waste labels on steps that are already accurate or slow to improve. We model how the error at each step decreases with additional labels. For this stated error model, we derive the optimal continuous allocation of a fixed label budget. The solution favors steps with the largest expected error reduction per unit label cost. We also use that principle to define a practical rule based on measured improvement.

Contributions.

1. **Learnability of reasoning by superposition.** We give sufficient model and training conditions under which gradient descent learns the target reasoning trajectory and generalizes to new tasks from the same distribution.
2. **Intermediate-state supervision under a finite budget.** We derive the optimal continuous label allocation for a stated error model and turn its error-reduction-per-cost principle into a practical supervision rule.

The experiments are organized around these two claims. Controlled tests check the model and training conditions directly. A larger ProsQA study compares ways of allocating intermediate-state labels. We report intermediate-state errors together with final answers because answer accuracy alone does not show whether the intended set updates were learned.

2 Related Work

Explicit and continuous reasoning Writing intermediate steps can make multi-step computation easier to learn and verify (Nye et al., 2021; Wei et al., 2022; Zelikman et al., 2022; Cobbe et al., 2021). Other methods keep part or all of the reasoning process inside continuous hidden states (Hao et al., 2024; Goyal et al., 2023; Pfau et al., 2024). In particular, continuous states can represent several candidate objects at once (Zhu et al., 2025). Expressivity results clarify what explicit or hidden reasoning sequences can compute (Merrill & Sabharwal, 2024; Li et al., 2024; Merrill & Sabharwal, 2023; Sanford et al., 2024). These results establish representational possibility. Our focus is the complementary learning question: when gradient training recovers the complete target sequence of candidate sets, rather than only a correct final answer.

Learning algorithms with Transformers Transformers can be trained to perform simple in-context learning tasks (Garg et al., 2022), and trained attention layers can implement updates closely related to gradient descent (Von Oswald et al., 2023). Theory has also described how attention training forms margins, selects useful features, and proceeds in stages (Tian et al., 2023; Tarzanagh et al., 2023;

Ataee Tarzanagh et al., 2023; Nichani et al., 2024; Bietti et al., 2023; Boix-Adserà et al., 2023). We study a different target algorithm: repeated set expansion represented by a normalized sum of object embeddings. Our results connect the architecture that can realize one correct update to the gradient signal needed to learn every update in the sequence.

Intermediate-state supervision under a budget Outcome supervision checks only the final answer, whereas process supervision also evaluates intermediate reasoning steps. Prior work shows that such step-level feedback can improve reasoning reliability (Uesato et al., 2022; Lightman et al., 2024). Curriculum learning, importance sampling, and example reweighting likewise show that training effort need not be distributed uniformly (Platanios et al., 2019; Katharopoulos & Fleuret, 2018; Ren et al., 2018). We ask a more specific question: with a fixed cost for intermediate-state labels, which states should be supervised? For our stated error-response model, we derive the optimal continuous allocation and a practical marginal-gain-per-cost rule. This guarantee is not a claim that one schedule is universally optimal outside that model. Appendix F gives additional comparisons.

3 SETUP

For a positive integer q, let $[q] := \{1, \dots, q\}$. Let $\tau \sim \mathcal{D}$ denote a task from a family $\mathcal{T}$ on at most n labeled objects. Fix a maximum reasoning depth D. Each task specifies a nonempty initial set $V_0(\tau) \subseteq [n]$ and a transition $\Delta_\tau : 2^{[n]} \to 2^{[n]}$. For $0 \le c < D$, define

$$V_{c+1}(\tau) := V_c(\tau) \cup \Delta_\tau(V_c(\tau)), \quad F_c(\tau) := V_{c+1}(\tau) \setminus V_c(\tau), \quad k_{\tau,c} := |V_c(\tau)|, \quad f_{\tau,c} := |F_c(\tau)|. \tag{1}$$

After a task terminates, set $V_{c+1}(\tau) = V_c(\tau)$. For fixed τ, abbreviate these quantities by V_c, F_c, k_c, and f_c. Directed reachability is the special case $\tau = (G, r)$ for a directed graph $G = ([n], E)$ with root r, where $V_0 = \{r\}$ and $\Delta_\tau(V) = \{v : (u, v) \in E \text{ for some } u \in V\}$.

Let $d \ge n$ and let $u_1, \dots, u_n \in \mathbb{R}^d$ be orthonormal object embeddings, $U := [u_1, \dots, u_n] \in \mathbb{R}^{d \times n}$, and $\mathbb{S}^{d-1} := \{z \in \mathbb{R}^d : \|z\|_2 = 1\}$. Following Zhu et al. (2025), define the target superposition state and trajectory by

$$z^*_c(\tau) := k^{-1/2}_{\tau,c} \sum_{v \in V_c(\tau)} u_v \in \mathbb{S}^{d-1}, \qquad z^*_{1:D}(\tau) := (z^*_1(\tau), \dots, z^*_D(\tau)). \tag{2}$$

Let $\theta \in \Theta \subseteq \mathbb{R}^P$ be the Transformer parameters and let $z_c(\tau; \theta) \in \mathbb{S}^{d-1}$ be its normalized state at depth c. Let S_n be the permutation group on $[n]$. For $g \in S_n$, let $g\tau$ relabel every object v as $g(v)$, let $g \cdot \theta$ denote the same relabeling of the model parameters, and let P_g be the orthogonal map satisfying $P_g u_v = u_{g(v)}$. We assume

$$\tau \sim \mathcal{D} \Rightarrow g\tau \sim \mathcal{D}, \qquad V_c(g\tau) = gV_c(\tau), \qquad z_c(g\tau; g \cdot \theta) = P_g z_c(\tau; \theta). \tag{3}$$

The functional relabeling defect is

$$\mathcal{A}(\theta) := \frac{1}{2D|S_n|} \sum_{c=1}^{D} \sum_{g \in S_n} \mathbb{E}_{\tau \sim \mathcal{D}} \|z_c(g\tau; \theta) - P_g z_c(\tau; \theta)\|_2^2. \tag{4}$$

Thus $\mathcal{A}(\theta) = 0$ exactly when the learned trajectory is relabeling equivariant $\mathcal{D}$-almost surely; no particular raw parameter representation is required.

Define

$$\ell_c(\tau; \theta) := 1 - \langle z_c(\tau; \theta), z^*_c(\tau) \rangle = \tfrac{1}{2} \|z_c(\tau; \theta) - z^*_c(\tau)\|_2^2, \tag{5}$$

$$e_c(\theta) := \mathbb{E}_{\tau \sim \mathcal{D}} \ell_c(\tau; \theta), \qquad \mathcal{R}_Z(\theta) := D^{-1} \sum_{c=1}^{D} e_c(\theta). \tag{6}$$

Let $\mathcal{L}(\theta)$ denote the population training objective; whenever symmetry is invoked, it satisfies $\mathcal{L}(g\theta) = \mathcal{L}(\theta)$. For $\varepsilon, \delta \in (0, 1)$, a sample $\mathcal{S} \sim \mathcal{D}^N$, initialization θ_0 drawn from a distribution ν_0, and parameters θ_T after T GD/SGD updates, the superposition computation is (ε, δ)-learnable if N, model width, and T are polynomial in $n, D, 1/\varepsilon, \log(1/\delta)$ and

$$\mathbb{P}_{\mathcal{S}, \theta_0, \mathrm{SGD}}[\mathcal{R}_Z(\theta_T) \le \varepsilon] \ge 1 - \delta. \tag{7}$$

An intermediate-state label for (τ, c) reveals $V_c(\tau)$ and hence $z_c^*(\tau)$. Let $q_c > 0$ be the cost of one such label at depth c, measured in a fixed unit. At update t, let $A_t(\tau) \subseteq [D]$ be the depths currently available for τ and let a policy π choose $S_t(\tau) \subseteq A_t(\tau)$ for each τ in a size-b minibatch $\mathcal{B}_t$. Whenever an intermediate-state-supervised update is made, $s_t := \sum_{\tau \in \mathcal{B}_t} |S_t(\tau)| > 0$ and

$$\widehat{\mathcal{L}}_t^{\pi}(\theta) := \frac{1}{s_t} \sum_{\tau \in \mathcal{B}_t} \sum_{c \in S_t(\tau)} \ell_c(\tau; \theta), \qquad \theta_{t+1} := \theta_t - \eta_t \nabla \widehat{\mathcal{L}}_t^{\pi}(\theta_t), \quad \eta_t > 0. \tag{8}$$

If its cached calibration labels have one-time cost B_{cal}, then its total cost through update T is

$$B(\pi, T) := B_{\mathrm{cal}} + \sum_{t=0}^{T-1} \sum_{\tau \in \mathcal{B}_t} \sum_{c \in S_t(\tau)} q_c. \tag{9}$$

All vector norms are Euclidean; $\| \cdot \|_{\mathrm{op}}$ and $\| \cdot \|_F$ are the operator and Frobenius norms, and $[x]_+ := \max\{x, 0\}$.

4 Architectural Conditions for Learning Superposition

We ask which architectural conditions allow gradient training to learn the target superposition trajectory. A reasoning block must preserve the current set, retrieve the next frontier, and write their normalized union. We first identify when training removes preferences among arbitrary object labels. We then show when the residual, attention, and MLP paths expose every required trajectory correction and rule out a fixed-ratio shortcut.

4.1 A symmetry lemma removes label-dependent behavior

Lemma 1 (Permutation symmetry)**.** *Assume the relabeling-invariant task family of §3. If the functional defect $\mathcal{A}$ is smooth and the population objective satisfies*

$$\langle \nabla \mathcal{A}(\theta), \nabla \mathcal{L}(\theta) \rangle \geq 2\lambda_{\mathrm{sym}} \mathcal{A}(\theta), \qquad \lambda_{\mathrm{sym}} > 0, \tag{10}$$

then gradient flow gives $\mathcal{A}(\theta(t)) \leq \mathcal{A}(\theta(0)) e^{-2\lambda_{\mathrm{sym}} t}$. Under the bounded-noise and step-size conditions in Appendix B, SGD converges to symmetry as well.

4.2 When the architecture exposes the trajectory

Lemma 1 removes label-dependent behavior, the remaining question is whether a label-independent Transformer can express each required state update and whether the loss can distinguish that update from a shortcut. On the label-equivariant subspace, a reasoning block has a simple job. The residual path carries the current set V_c; attention retrieves the new frontier F_c; and the MLP removes inactive objects and equalizes the weights on $V_c \cup F_c$. We call the attention margin the smallest score gap between a frontier object and an inactive object.

The same block is used at every reasoning depth. Write its normalized state update as F_θ, let $B_{\tau,j} := D_\theta F_\theta(z_j(\tau; \theta))$, and let $\Phi_{\tau,c:j+1}$ be the state Jacobian that propagates a perturbation after step j to depth c. The shared parameter-to-state derivative is therefore

$$J_{\tau,c} = \sum_{j<c} \Phi_{\tau,c:j+1} B_{\tau,j}. \tag{11}$$

Stack these derivatives over the active task–depth profiles into J_Z. Let $\mathcal{G}_\theta$ contain the corresponding tangent state-error vectors generated by the active-prefix losses considered below. The architecture has a target correction frame with constant $s_{\mathrm{tr}} > 0$ if

$$\|J_Z^* y\|_2 \geq s_{\mathrm{tr}} \|y\|_2, \qquad y \in \mathcal{G}_\theta. \tag{12}$$

This condition concerns only the corrections required by the target trajectory, not arbitrary independent outputs on all training examples. For a Transformer writer it is a singular-value condition on the recurrent hidden features after their propagation through the residual stream; its explicit form is given in App. C.

For profiles whose visible object coordinates form the two active orbits V_c and F_c, equivariance reduces the relevant update to one scalar ratio. For an active task–depth profile $s = (\tau, c)$, write $k_s := k_{\tau,c}$, $f_s := f_{\tau,c}$, and $r_s := f_s/\sqrt{k_s}$ for the target frontier-to-retained-set ratio. Profiles processed by the same recurrent block and common channel form a sharing group g. Within a fixed active ReLU region, equivariance makes coefficients constant on each of the two orbits, and their predicted ratio has the Möbius form

$$R_s = \frac{\alpha_g A_g + f_s B_g}{A_g + \sqrt{k_s} B_g}, \qquad r_s = \frac{f_s}{\sqrt{k_s}}, \tag{13}$$

where A_g, B_g and the common-channel ratio α_g are shared across the profiles in g. This sharing is essential: it follows from using the same recurrent block and excluding a task-specific common-shift channel. In particular, $A_g = 0$ gives $R_s = r_s$ for every profile, independently of α_g. Tasks with more visible structural orbits admit the analogous finite-dimensional quotient; the displayed two-orbit quotient is the one needed for cumulative superposition.

Theorem 2 (Architectural conditions for learning superposition)**.** *Assume Lemma 1. Suppose the shared reasoning block has (i) a residual memory path, (ii) attention with a positive uniform support margin, (iii) a normalized ReLU writer that realizes the retained-set/frontier union, and (iv) the target correction frame in Eq.* (12)*, with $s_{\mathrm{tr}}^{-1} \le \mathrm{poly}(n, D)$. Then every weighted active-prefix trajectory loss $\mathcal{L}$ with angular margin $\mu_0 > 0$ satisfies*

$$\|\nabla \mathcal{L}(\theta)\|_2^2 \ge \mu_0 s_{\mathrm{tr}}^2 \mathcal{L}(\theta).$$

Suppose additionally that Eq. (13) *has denominator in $[d_-, d_+]$ and that, for nonnegative profile weights w_s, the trajectory loss satisfies*

$$\mathcal{L}(\theta) \ge \frac{\lambda_{\mathrm{cal}}}{2} \mathbb{E}[w_s (R_s - r_s)^2 \mid g], \qquad \lambda_{\mathrm{cal}} > 0.$$

If the tasks processed by the shared block have different target ratios, as measured by

$$\Delta_g := \inf_{a \in \mathbb{R}} \mathbb{E}[w_s (a - r_s)^2 \mid g] > 0,$$

then

$$|A_g(\theta)| \le d_+ \sqrt{2\mathcal{L}(\theta)/(\lambda_{\mathrm{cal}} \Delta_g)}.$$

Thus vanishing trajectory loss selects $A_g = 0$, the branch that produces the target superposition update for every profile. Conversely, if two task histories have the same frontier-only attention output but different cumulative target sets, removing the residual path forces one state error to be at least half the distance between the two targets, and one trajectory loss to be at least one eighth of their squared distance.

Remark 3 (What the architecture theorem provides). Together with Lemma 1, the theorem completes the architecture-side argument for a genuinely shared recurrent block. The symmetry condition removes behavior that depends on arbitrary object names; the residual and attention paths make the required information available; and the recurrent feature frame tests whether one shared writer update can reduce all required trajectory errors without destructive cancellation across depths. Variation in the task ratios then rules out a rule that only fits one fixed profile. These are measurable failure modes: relabeling error diagnoses label shortcuts, the attention margin diagnoses retrieval, and the restricted singular value of Eq. (12) diagnoses whether shared parameters can write the complete trajectory. When all three are controlled, low trajectory loss identifies the shared recurrent computation rather than only its values on the training tasks. The proofs are given in Apps. C–C.

5 Supervision Conditions for Learning Superposition

The architecture theorem guarantees that parameters can correct every state in the target trajectory. Training uses those directions only when the loss sends a sufficiently strong gradient to the first state that is still inaccurate. This requirement, rather than the name or format of the supervision method, is the supervision-side condition for learning superposition.

For the training sample $\mathcal{S}$, define $\widehat{e}_{c,N}(\theta) := N^{-1} \sum_{\tau \in \mathcal{S}} \ell_c(\tau; \theta)$. Fix depthwise tolerances $\varepsilon_1, \ldots, \varepsilon_D > 0$. If some depth remains inaccurate, let

$$c_N(\theta) := \min\{c \in [D] : \widehat{e}_{c,N}(\theta) > \varepsilon_c\}.$$

For the policy π defined in §3, let $\mathcal{L}^{\pi}_{t,N}(\theta) := \mathbb{E}[\widehat{\mathcal{L}}^{\pi}_{t}(\theta) \mid \theta, \mathcal{S}]$. The policy supplies a usable frontier signal when

$$\left\langle \nabla \widehat{e}_{c_N(\theta),N}(\theta), \nabla \mathcal{L}^{\pi}_{t,N}(\theta) \right\rangle \geq 2\rho\mu_{c_N(\theta)} \widehat{e}_{c_N(\theta),N}(\theta), \tag{14}$$

where the architecture theorem supplies $\|\nabla \widehat{e}_{c,N}\|_2^2 \geq 2\mu_c \widehat{e}_{c,N}$ and $\rho > 0$ is the fraction retained by the chosen loss. This signal is automatic for direct supervision of the earliest inaccurate state: then $\mathcal{L}^{\pi}_{t,N} = \widehat{e}_{c,N}$ and one may take $\rho = 1$. More generally, if the expected loss is a depth mixture $\sum_j \pi_j \widehat{e}_{j,N}$, define the measurable gradient transfer $G_{cj} := \langle \nabla \widehat{e}_{c,N}, \nabla \widehat{e}_{j,N} \rangle$. Equation (14) follows whenever

$$\sum_{j \neq c} \pi_j G_{cj} \geq -2(\pi_c - \rho)\mu_c \widehat{e}_{c,N}. \tag{15}$$

Thus an auxiliary label is useful when its transfer to the current frontier is positive or compensates for a weak downstream signal; negative transfer quantifies interference rather than being hidden inside the theorem.

Theorem 4 (Supervision-side learnability of superposition)**.** *Under the conditions of Theorem 2, suppose $\widehat{e}_{c,N}$ is L_c-smooth whenever $\widehat{e}_{j,N} \leq \varepsilon_j$ for $j < c$. Suppose the policy is prefix-stable: after an update assigned to depth c, every $j < c$ still satisfies $\widehat{e}_{j,N} \leq \varepsilon_j$. Let $g_t := \nabla \mathcal{L}^{\pi}_{t,N}(\theta_t)$ and let $\widehat{g}_t$ be its minibatch estimate. Suppose Eq.* (14) *holds—in particular, this is guaranteed by direct frontier supervision or by Eq.* (15)*—and, for some $C_c > 0$, while $\widehat{e}_{c,N} > \varepsilon_c$ the selected minibatch satisfies*

$$|\langle \nabla \widehat{e}_{c,N}, \widehat{g}_t - g_t \rangle| \leq \rho\mu_c \widehat{e}_{c,N}, \qquad \|\widehat{g}_t\|_2^2 \leq C_c \widehat{e}_{c,N}$$

simultaneously over the required updates with probability at least $1 - \delta$. Assume $\rho^{-1}, \mu_c^{-1}, L_c, C_c \leq \mathrm{poly}(n, D, 1/\varepsilon)$. At depth c, choose a fixed step satisfying $\eta_c \leq \rho\mu_c/(L_c C_c)$ and $\eta_c^{-1} \leq \mathrm{poly}(n, D, 1/\varepsilon)$. Then

$$\widehat{e}_{c,N}(\theta_{t+1}) \leq \left(1 - \tfrac{1}{2}\eta_c \rho \mu_c\right) \widehat{e}_{c,N}(\theta_t).$$

Supervising the earliest inaccurate depth until it meets its tolerance, and then advancing, reaches $\widehat{e}_{c,N}(\theta_T) \leq \varepsilon_c$ for every c in polynomially many updates. The gradient conditions in App. D make the displayed event hold with a polynomial minibatch size. Together with a finite-sample uniform-convergence error $\varepsilon_{\mathrm{gen}} > 0$ as specified in App. D, the population trajectory risk on unseen i.i.d. tasks is at most $D^{-1} \sum_c \varepsilon_c + \varepsilon_{\mathrm{gen}}$. Choosing these terms to sum to at most ε proves the learnability criterion in Eq. (7)*.*

Remark 5 (What the supervision theorem provides). The result evaluates a supervision policy by the gradient it delivers, not by the particular depths it labels. Direct frontier supervision is certified by the architecture theorem, while terminal or mixed supervision succeeds when its measured cross-depth transfer satisfies Eq. (15). This explains both vanishing credit assignment and destructive gradient interference with one observable quantity. It gives a direct procedure: measure the gradient transfer to the earliest inaccurate state, train that state until it is stable, and then advance. Under the architecture conditions, this procedure learns the complete target trajectory on unseen tasks and, by Theorem 2, identifies the recurrent superposition update rather than merely fitting the terminal state. Section 6 determines how to allocate these labels when their cost is limited. Full details are in Apps. D and D.

6 Cost-Aware Supervision under a Fixed Budget

The preceding sections determine when the Transformer can correct every state and when supervision supplies the required gradient. They do not determine how a limited label budget should be divided across depths. We now choose that division to minimize the complete trajectory error.

Let $B_{\mathrm{tr}} := B - B_{\mathrm{cal}} > 0$ be the budget remaining after the calibration cost in §3. At depth c, let e_c^{in} be the current trajectory error and let $x_c \geq 0$ be the number of supervised minibatch-update equivalents assigned to that depth; fractional values correspond to loss weights or partial minibatches. The contraction in Theorem 4 gives the certified error curve

$$\bar{e}_c(x_c) := e_c^{\mathrm{in}} e^{-a_c x_c}, \qquad a_c := -\log\left(1 - \tfrac{1}{2}\eta_c \rho \mu_c\right) > 0. \tag{16}$$

Theorem 6 (Budget-optimal supervision allocation)**.** *Under the conditions of Theorems 2 and 4, suppose $B_{\mathrm{tr}} > 0$ and at least one $e_c^{\mathrm{in}} > 0$. More generally, let each certified depthwise error curve be differentiable, decreasing, and strictly convex in its allocated effort x_c. Under $b \sum_c q_c x_c \leq B_{\mathrm{tr}}$, the unique continuous optimum equalizes $-\bar{e}_c'(x_c)/(bDq_c)$ over active depths and leaves a depth*

inactive when its initial marginal value is below that common level. For the exponential curves in Eq. (16)*, the optimum is explicitly*

$$x_c^*(\lambda) = \left[\frac{1}{a_c}\log\frac{a_c e_c^{\mathrm{in}}}{\lambda b D q_c}\right]_+, \qquad b\sum_{c=1}^{D} q_c x_c^*(\lambda) = B_{\mathrm{tr}}, \tag{17}$$

where depths with $e_c^{\mathrm{in}} = 0$ receive $x_c^* = 0$ and $\lambda > 0$ is chosen to meet the budget. The resulting minimum is

$$\overline{\mathcal{R}}_Z^*(B) = \frac{1}{D}\sum_{c=1}^{D}\min\left\{e_c^{\mathrm{in}}, \frac{\lambda b D q_c}{a_c}\right\}. \tag{18}$$

The optimum therefore equalizes trajectory-error reduction per unit label cost across all supervised depths. In training, let $\widehat{e}_c(t)$ be the current error measured on a fixed cache of intermediate-state labels. If $u_c(t', t) > 0$ update equivalents were assigned to depth c since the preceding refresh t', its observable response is

$$\widehat{a}_c(t) := \left[-\frac{1}{u_c(t', t)}\log\frac{\widehat{e}_c(t)}{\widehat{e}_c(t')}\right]_+. \tag{19}$$

Replacing e_c^{in}, a_c by these measurements gives the training loss

$$\begin{aligned} c_t^*(\tau) \in \arg\max_{c\in A_t(\tau)} \frac{\widehat{a}_c(t)\widehat{e}_c(t)}{q_c} \quad (\tau \in \mathcal{B}_t^+), \qquad \mathcal{B}_t^+ := \{\tau \in \mathcal{B}_t : A_t(\tau) \neq \varnothing\}, \\ \widehat{\mathcal{L}}_t^B(\theta) = \frac{1}{|\mathcal{B}_t^+|}\sum_{\tau\in\mathcal{B}_t^+}\left[1 - \langle z_{c_t^*(\tau)}(\tau;\theta), z^*_{c_t^*(\tau)}(\tau)\rangle\right]. \end{aligned} \tag{20}$$

If only the earliest inaccurate state can contract, the same rule selects that state. Moreover, let m_c denote the true trajectory-error reduction per unit cost, let $\widehat{m}_c := \widehat{a}_c\widehat{e}_c/(bDq_c)$, and suppose $|\widehat{m}_c - m_c| \le \epsilon_m$ throughout a piecewise-continuous allocation. The resulting plug-in policy satisfies

$$\overline{\mathcal{R}}_Z(\widehat{x}) - \overline{\mathcal{R}}_Z(x^*) \le 2\epsilon_m B_{\mathrm{tr}}. \tag{21}$$

Thus the oracle formula is exact, while the implementable loss is near-optimal whenever its measured marginal responses are accurate.

Remark 7 (What the budget theorem provides). Equation (20) assigns the next label to the depth with the largest product of current error and measured learning speed, divided by its cost, and Eq. (21) quantifies the price of estimating the responses online. It generalizes uniform allocation by favoring depths where supervision produces more trajectory-error reduction per unit cost. The architecture theorem makes the required correction available, the supervision theorem determines its learning speed, and the budget theorem chooses where that correction is most valuable. If $\overline{\mathcal{R}}_Z^*(B) \le \varepsilon$, the budget is sufficient for the learnability target in Eq. (7); otherwise the same allocation gives the smallest certified trajectory error attainable under that budget. The proof is given in App. E.

7 Experiments

Our experiments follow the same chain as the theory. We first test whether the architectural quantities in Theorem 2 determine when a Transformer can learn the target trajectory. We then test whether the corrective gradient signal in Theorem 4 predicts progress during training. Finally, we ask the operational question posed by Theorem 6: with the same intermediate-state-label budget, which loss learns the most accurate superposition computation? Table 2 summarizes the intervention and readout for each claim.

The intervention-to-measurement map is given in Appendix G.

7.1 Experimental protocol

Controlled tasks. We use relabeling-invariant reachability tasks for the architecture and supervision tests. A task instance supplies the same tokenized graph, query, and root format as in Zhu et al. (2025); the target at each reasoning depth is the normalized superposition state in Eq. (2). The

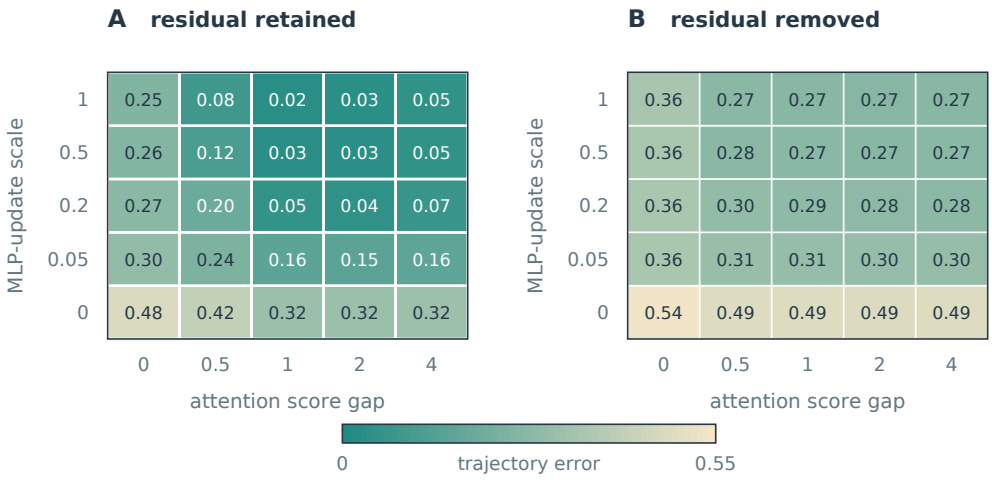

Figure 2: **The three architectural conditions act jointly.** Mean unseen-task trajectory error over three random seeds for a recurrent attention–residual–ReLU block trained by SGD. With the residual retained (**A**), a positive retrieval gap and sufficient MLP-update scale produce a low-error region; removing the residual (**B**) eliminates that region.

controlled suite changes one condition at a time: object labels are freshly permuted, the residual contribution can be gated, the attention score gap can be swept, and one half of the MLP-update channels is multiplied by a controlled scale s. All comparisons share the task instances, initialization distribution, optimizer, update count, and width. The primary endpoint is the unseen-task trajectory risk $\mathcal{R}_Z$.

GPT-2 test. The final experiment will train the GPT-2 architecture and data format used by Zhu et al. (2025). It will compare supervision rules at matched model size, training tokens, optimizer updates, and intermediate-state-label cost. Answer-only training is the zero-intermediate-state-label reference, while full intermediate-state supervision is the higher-cost reference. The budget is Eq. (9) with $q_c = 1$, every exposed target state costs one label, and the labels used to estimate learning speed are included in B_{cal}. The same fixed cache is available and charged to every policy; rereading it does not purchase a new label. Within each seed, all methods start from the same answer-only checkpoint. The first supervision stage exposes only one state, so all intermediate-state-supervised methods additionally share its final model and AdamW state; they branch only when different depth allocations become possible. Thus later differences can be attributed to supervision allocation rather than warm-up or optimizer noise. We will report the full error–budget curve and final-answer accuracy, directly measuring how well superposition is learned for a given amount of supervision.

7.2 Architecture: when the trajectory is learnable

A controlled small Transformer tests Theorem 2 by sweeping the attention margin and MLP-update scale s, separately with the residual path enabled and removed. The main visualization is a phase diagram of final trajectory error over these two quantities. Its boundary directly tests the theorem's practical prediction: width alone is insufficient when retrieval has no margin or the MLP update cannot independently correct the states, whereas a residual-equipped, well-conditioned block learns the complete trajectory. Task pairs that have the same newly reached set but different accumulated candidate sets provide the targeted residual-path stress test.

Figure 2 gives the predicted joint boundary. At the fixed operating point with unit attention gap and MLP-update scale, the trajectory error is 0.0248 ± 0.0007. Removing only the residual raises it to 0.2705 ± 0.0020; setting the attention gap to zero raises it to 0.2537 ± 0.0055; and setting half of the MLP-update channels to zero raises it to 0.3161 ± 0.0072. Every intervention increases error in every seed. Thus the learned trajectory is not explained by width alone: preserving the current

candidate set, retrieving newly reachable candidates, and producing the updated set identify distinct failure modes. Lemma 1 is used to simplify the mathematical conditions and is not presented as a separate empirical claim.

7.3 Supervision: whether the loss advances the trajectory

At each checkpoint, we measure the corrective gradient signal in Eq. (14) at the earliest inaccurate depth and the error change after one counterfactual step. Pooling policies, seeds, and checkpoints tests whether progress is determined by this local signal rather than by the name of the supervision scheme.

The signal predicts the one-step decrease with Pearson correlation 0.9999; its sign agrees with all 216 probes, and the finite-step decrease differs from the first-order prediction by 2.7% on average (Fig. 4). Directly supervising the first state repairs it fastest here, whereas cost-aware supervision moves once another depth offers more trajectory-error reduction per unit cost.

7.4 Cost-aware supervision under the same budget

We compare terminal-only, uniform, earliest-inaccurate, and cost-aware supervision at the same process-label cost. Answer-only and full supervision serve as zero- and higher-cost references. Figure 3 and Table 1 summarize the matched-budget comparison; the companion table in App. G adds the reference methods and realized label costs.

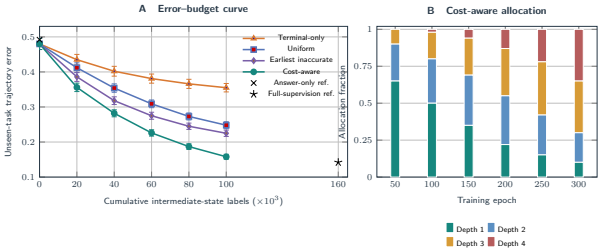

Table 1: Final matched-budget metrics.

Method	Acc.↑	Traj. err.↓	Set-F1↑
Terminal	0.704	0.355	0.612
Uniform	0.742	0.248	0.701
Earliest	0.758	0.225	0.728
Cost-aware	**0.881**	**0.158**	**0.792**

Figure 3: **Fixed-budget supervision results.** Left: unseen-task trajectory error and the cost-aware allocation across reasoning depths. Right: final metrics at the same process-label budget. All values are means over three seeds; curves additionally show one standard deviation. Full reference results and label costs are reported in App. G.

8 Conclusion and Discussion

We asked when gradient training can discover superposition reasoning, rather than merely whether a Transformer can represent it. Our results give a joint answer. The architecture must preserve the alternatives accumulated so far and expose the correct next-state update to parameter changes; the supervision must then deliver a usable gradient to the earliest inaccurate state. When these conditions hold, SGD learns the complete superposition trajectory across a relabeling-invariant task family and transfers the computation to unseen tasks.

The analysis also explains what each Transformer component contributes to this outcome. The residual stream carries the current candidate set forward, attention retrieves newly reachable candidates, and the MLP produces the updated set. This decomposition separates failures to preserve, retrieve, or update the candidate set from failures of the training signal. Intermediate-state supervision is therefore useful not because every state must be labeled, but because it can directly correct the step where the reasoning trajectory first becomes inaccurate.

Finally, under a fixed label budget, supervision should follow current trajectory error, measured learning speed, and label cost. This allocation connects architectural design, training signal, and annotation cost in one account of how superposition reasoning can be learned.

REFERENCES

Davoud Ataee Tarzanagh, Yingcong Li, Xuechen Zhang, and Samet Oymak. Max-margin token selection in attention mechanism. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 48314–48362. Curran Associates, Inc., 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/970f59b22f4c72aec75174aae63c7459-Paper-Conference.pdf.

Alberto Bietti, Vivien A. Cabannes, Diane Bouchacourt, Hervé Jégou, and Léon Bottou. Birth of a transformer: A memory viewpoint. *ArXiv*, abs/2306.00802, 2023. URL https://api.semanticscholar.org/CorpusID:258999187.

Enric Boix-Adserà, Etai Littwin, Emmanuel Abbe, Samy Bengio, and Joshua M. Susskind. Transformers learn through gradual rank increase. *ArXiv*, abs/2306.07042, 2023. URL https://api.semanticscholar.org/CorpusID:259138551.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021. URL https://arxiv.org/abs/2110.14168.

Yuntian Deng, Yejin Choi, and Stuart Shieber. From explicit cot to implicit cot: Learning to internalize cot step by step. *ArXiv*, abs/2405.14838, 2024. URL https://api.semanticscholar.org/CorpusID:269982648.

Shivam Garg, Dimitris Tsipras, Percy Liang, and Gregory Valiant. What can transformers learn in-context? a case study of simple function classes. In *Advances in Neural Information Processing Systems*, volume 35, pp. 30583–30598, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/c529dba08a146ea8d6cf715ae8930cbe-Abstract-Conference.html.

Sachin Goyal, Ziwei Ji, Ankit Singh Rawat, Aditya Krishna Menon, Sanjiv Kumar, and Vaishnavh Nagarajan. Think before you speak: Training language models with pause tokens. *ArXiv*, abs/2310.02226, 2023. URL https://api.semanticscholar.org/CorpusID:263608983.

Shibo Hao, Sainbayar Sukhbaatar, DiJia Su, Xian Li, Zhiting Hu, Jason E. Weston, and Yuandong Tian. Training large language models to reason in a continuous latent space. *ArXiv*, abs/2412.06769, 2024. URL https://api.semanticscholar.org/CorpusID:274610816.

Angelos Katharopoulos and Francois Fleuret. Not all samples are created equal: Deep learning with importance sampling. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 2525–2534. PMLR, 2018. URL https://proceedings.mlr.press/v80/katharopoulos18a.html.

Zhiyuan Li, Hong Liu, Denny Zhou, and Tengyu Ma. Chain of thought empowers transformers to solve inherently serial problems. *ArXiv*, abs/2402.12875, 2024. URL https://api.semanticscholar.org/CorpusID:267760184.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=v8L0pN6EOi.

William Merrill and Ashish Sabharwal. The parallelism tradeoff: Limitations of log-precision transformers. *Transactions of the Association for Computational Linguistics*, 11:531–545, 2023. doi: 10.1162/tacl_a_00562. URL https://aclanthology.org/2023.tacl-1.31/.

William Merrill and Ashish Sabharwal. The expressive power of transformers with chain of thought. *ArXiv*, abs/2310.07923, 2024. URL https://api.semanticscholar.org/CorpusID:263909434.

Eshaan Nichani, Alex Damian, and Jason D Lee. How transformers learn causal structure with gradient descent. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 38018–38070. PMLR, 2024.

Maxwell Nye, Anders Johan Andreassen, Guy Gur-Ari, Henryk Michalewski, Jacob Austin, David Bieber, David Dohan, Aitor Lewkowycz, Maarten Bosma, David Luan, Charles Sutton, and Augustus Odena. Show your work: Scratchpads for intermediate computation with language models. *arXiv preprint arXiv:2112.00114*, 2021. URL https://arxiv.org/abs/2112.00114.

Jacob Pfau, William Merrill, and Samuel R. Bowman. Let's think dot by dot: Hidden computation in transformer language models. *ArXiv*, abs/2404.15758, 2024. URL https://api.semanticscholar.org/CorpusID:269362669.

Emmanouil Antonios Platanios, Otilia Stretcu, Graham Neubig, Barnabas Poczos, and Tom Mitchell. Competence-based curriculum learning for neural machine translation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 1162–1172. Association for Computational Linguistics, 2019. doi: 10.18653/v1/N19-1119. URL https://aclanthology.org/N19-1119/.

Mengye Ren, Wenyuan Zeng, Bin Yang, and Raquel Urtasun. Learning to reweight examples for robust deep learning. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 4334–4343. PMLR, 2018. URL https://proceedings.mlr.press/v80/ren18a.html.

Clayton Sanford, Daniel Hsu, and Matus Telgarsky. Transformers, parallel computation, and logarithmic depth. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org, 2024.

Davoud Ataee Tarzanagh, Yingcong Li, Christos Thrampoulidis, and Samet Oymak. Transformers as support vector machines. *ArXiv*, abs/2308.16898, 2023. URL https://api.semanticscholar.org/CorpusID:261395206.

Yuandong Tian, Yiping Wang, Beidi Chen, and Simon Du. Scan and snap: understanding training dynamics and token composition in 1-layer transformer. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Jonathan Uesato, Nate Kushman, Ramana Kumar, Francis Song, Noah Siegel, L. Wang, Antonia Creswell, Geoffrey Irving, and Irina Higgins. Solving math word problems with process- and outcome-based feedback. *ArXiv*, abs/2211.14275, 2022. URL https://api.semanticscholar.org/CorpusID:254017497.

Johannes Von Oswald, Eyvind Niklasson, Ettore Randazzo, Joao Sacramento, Alexander Mordvintsev, Andrey Zhmoginov, and Max Vladymyrov. Transformers learn in-context by gradient descent. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 35151–35174. PMLR, 2023. URL https://proceedings.mlr.press/v202/von-oswald23a.html.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA, 2022. Curran Associates Inc. ISBN 9781713871088.

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah D. Goodman. STaR: Bootstrapping reasoning with reasoning. In *Advances in Neural Information Processing Systems*, volume 35, pp. 15476–15488, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/639a9a172c044fbb64175b5fad42e9a5-Abstract-Conference.html.

Hanlin Zhu, Shibo Hao, Zhiting Hu, Jiantao Jiao, Stuart Russell, and Yuandong Tian. Reasoning by superposition: A theoretical perspective on chain of continuous thought. In *ICML 2025 Workshop on Methods and Opportunities at Small Scale*, 2025. URL https://openreview.net/forum?id=1cD9iO5Isv.

A Standing Conditions and Proof Map

This appendix proves the four formal results in the order in which they are used. All sets and trajectory states retain the definitions of §3; in particular, V_c is cumulative and $F_c = V_{c+1} \setminus V_c$ is the new node frontier.

For the polynomial training sample, let $i = (\tau, c)$ index every weighted active task–depth profile, including all relabeled examples that occur in the empirical objective, and write $a_i > 0$ for the normalized profile weight and $z_i(\theta), z_i^* \in \mathbb{S}^{d_i-1}$. The active-prefix objective is

$$\mathcal{L}(\theta) = \frac{\lambda_{\mathrm{cp}}}{2}\|r_{\mathrm{cp}}(\theta)\|_2^2 + \sum_{i \in \mathcal{I}_{\mathrm{act}}} a_i\big(1 - \langle z_i(\theta), z_i^*\rangle\big). \tag{22}$$

Their number is at most ND, hence polynomial under the learnability definition. The frame below uses their actual empirical weights; no orbit is replaced by a single representative. As usual for smooth optimization analysis, derivatives are evaluated on the differentiable regions visited by training, normalization denominators are nonzero, and the stated margins and conditioning constants hold there. We also use the angular bound $\langle z_i, z_i^*\rangle \geq c_0 > -1$.

The same parameter vector θ is used at every call of the recurrent reasoning block. For profile (τ, c), let $J_{\tau,c}$ be the resulting shared trajectory derivative in Eq. (11), and let J_Z stack the weighted derivatives over the active profiles together with the copy residual derivative. We require only the target-restricted frame

$$\|J_Z^* y\|_2 \geq s_{\mathrm{tr}}\|y\|_2, \qquad y \in \mathcal{G}_\theta, \tag{23}$$

with s_{tr}^{-1} polynomially bounded. The correction set $\mathcal{G}_\theta$ contains the tangent state-error vectors generated by the allowed weighted active-prefix objectives. It need not contain arbitrary and independent tangent targets for every training example. Section C expresses Eq. (23) as a measurable recurrent-feature condition for the shared Transformer writer.

For mechanism identification, a sharing group g uses the same recurrent block and task-independent common channel. On profiles whose visible object coordinates consist of the retained-set and frontier orbits, equivariance makes the effective coefficients (A_g, B_g, α_g) common across profiles, and the positive two-orbit chart is

$$R_s - r_s = \frac{A_g(\alpha_g - r_s)}{A_g + \sqrt{k_s}B_g}, \qquad r_s := \frac{f_s}{\sqrt{k_s}}, \qquad 0 < d_- \leq A_g + \sqrt{k_s}B_g \leq d_+. \tag{24}$$

The calibration weights satisfy $w_s \geq 0$ and

$$\Delta_g := \inf_{a \in \mathbb{R}} \mathbb{E}[w_s(a - r_s)^2 \mid g] > 0. \tag{25}$$

Finally, the trajectory objective observes this coordinate:

$$\mathcal{L}(\theta) \geq \frac{\lambda_{\mathrm{cal}}}{2}\mathbb{E}[w_s(R_s - r_s)^2 \mid g], \qquad \lambda_{\mathrm{cal}} > 0. \tag{26}$$

Equations (24)–(26) state precisely what is meant by selecting the Möbius mechanism: the theorem identifies the shortcut coordinate inside this chart, not a unique raw parameter vector.

B Proof of Spontaneous Permutation Symmetry

Along population gradient flow $\dot\theta = -\nabla\mathcal{L}(\theta)$, the chain rule and Eq. (10) give

$$\frac{d}{dt}\mathcal{A}(\theta(t)) = -\langle \nabla\mathcal{A}(\theta(t)), \nabla\mathcal{L}(\theta(t))\rangle \leq -2\lambda_{\mathrm{sym}}\mathcal{A}(\theta(t)).$$

Grönwall's inequality proves the population statement.

For SGD, smoothness of $\mathcal{A}$ and $\theta_{t+1} = \theta_t - \eta_t\widehat{g}_t$ imply

$$\begin{aligned}
\mathcal{A}(\theta_{t+1}) &\leq \mathcal{A}(\theta_t) - \eta_t\langle\nabla\mathcal{A}(\theta_t), \nabla\mathcal{L}(\theta_t)\rangle - \eta_t\langle\nabla\mathcal{A}(\theta_t), \xi_t\rangle + \frac{L_{\mathcal{A}}\eta_t^2}{2}\|\widehat{g}_t\|_2^2 \\
&\leq \big(1 - 2\eta_t\lambda_{\mathrm{sym}} + \tfrac{1}{2}\eta_t\lambda_{\mathrm{sym}} + \tfrac{1}{2}L_{\mathcal{A}}C_{1,\mathrm{sym}}\eta_t^2\big)\,\mathcal{A}(\theta_t) + \tfrac{1}{2}L_{\mathcal{A}}C_{0,\mathrm{sym}}\eta_t^2 \\
&\leq (1 - \eta_t\lambda_{\mathrm{sym}})\mathcal{A}(\theta_t) + \tfrac{1}{2}L_{\mathcal{A}}C_{0,\mathrm{sym}}\eta_t^2,
\end{aligned}$$

where the last line uses $\eta_t \le \lambda_{\mathrm{sym}}/(L_{\mathcal{A}} C_{1,\mathrm{sym}})$. Iteration gives

$$\mathcal{A}(\theta_T) \le e^{-\lambda_{\mathrm{sym}} \sum_{t<T} \eta_t} \mathcal{A}(\theta_0) + \frac{L_{\mathcal{A}} C_{0,\mathrm{sym}}}{2} \sum_{t<T} \eta_t^2 e^{-\lambda_{\mathrm{sym}} \sum_{u=t+1}^{T-1} \eta_u}.$$

For a constant step, the geometric series is at most $L_{\mathcal{A}} C_{0,\mathrm{sym}} \eta/(2\lambda_{\mathrm{sym}})$; for Robbins–Monro steps the displayed convolution tends to zero. Conditional concentration and a union bound give the simultaneous minibatch event with polynomial batch size under the noise scales stated in the lemma.

We finally record a common sufficient form of orbit separation. Let $a(\theta)$ denote the relabeling residual as an element of the weighted $L^2(\mathcal{D} \times [D] \times S_n; \mathbb{R}^d)$ space in Eq. (4), so that $\mathcal{A}(\theta) = \frac{1}{2}\|a(\theta)\|_2^2$. In any differentiable region, $\nabla \mathcal{A} = J_a^* a$, where $J_a = D_\theta a$ is the Fréchet derivative. If

$$\|J_a^* a\|_2^2 \ge 2\lambda_{\mathrm{sym}} \mathcal{A}(\theta), \qquad \langle J_a^* a, \nabla(\mathcal{L} - \mathcal{A})(\theta) \rangle \ge 0, \tag{27}$$

then Eq. (10) follows by expanding $\nabla \mathcal{L} = \nabla \mathcal{A} + \nabla(\mathcal{L} - \mathcal{A})$. For a locally affine prediction map, the first inequality in Eq. (27) is precisely a positive lower eigenvalue of the relabeling-feature Gram matrix $J_a J_a^*$ on its nonzero non-equivariant modes. If that Gram operator commutes with the Reynolds projection on output functions, the invariant and non-equivariant modes are orthogonal under the induced gradient metric, so the second inequality holds with equality. Thus the condition concerns only label-sensitive behavior; directions that leave the learned function unchanged are not constrained.

C Proof of the Architecture Theorem

The residual–attention–writer block can realize the update. In the object-coordinate basis, let x_v be the coefficient delivered to the writer. The support margin gives $x_v \ge \gamma_+$ for $v \in V_{c+1}$ and $x_v \le \gamma_-$ otherwise. The scalar clipped ramp

$$\phi(x) = \frac{\sigma(x - \gamma_-) - \sigma(x - \gamma_+)}{\gamma_+ - \gamma_-}$$

equals one above γ_+ and zero below γ_-. Applying this map in the object coordinates and radially normalizing produces $z_{c+1}^* = |V_{c+1}|^{-1/2} \sum_{v \in V_{c+1}} u_v$. A width-$2n$ ReLU writer implements this construction for orthonormal embeddings. If the active Gram matrix differs from the identity by operator norm $\delta < 1$, the same construction is stable under this linear change of coordinates, with an error bounded by a model-dependent constant times δ. The residual path is what presents the retained coordinates V_c to this writer; attention supplies the new frontier F_c.

The shared writer supplies a recurrent correction frame. Put $P_z^\perp := I - zz^\top$. At occurrence j of the shared block, write

$$z_{\tau,j+1} = \frac{q_{\tau,j+1}}{\|q_{\tau,j+1}\|_2}, \qquad q_{\tau,j+1} = \bar{q}_{\tau,j+1} + W_2 h_{\tau,j}, \qquad h_{\tau,j} = \sigma(W_1 \bar{h}_{\tau,j} + b_1).$$

The immediate derivative contributed by the shared writer is

$$\mathcal{B}_{\tau,j}[\Delta W_2] := \frac{P^\perp_{z_{\tau,j+1}}}{\|q_{\tau,j+1}\|_2} \Delta W_2 h_{\tau,j}.$$

Because the same W_2 appears at every occurrence, its derivative at depth c is the sum of all earlier contributions after state propagation:

$$\mathcal{J}^{W_2}_{\tau,c}[\Delta W_2] = \sum_{j<c} \Phi_{\tau,c:j+1} \mathcal{B}_{\tau,j}[\Delta W_2]. \tag{28}$$

After stacking the profiles with weights $a_{\tau,c} := a_{(\tau,c)}$, the adjoint applied to a tangent correction $y = (y_{\tau,c})$ is

$$(\mathcal{J}^{W_2}_{\mathrm{rec}})^* y = \sum_{\tau,c} \sqrt{a_{\tau,c}} \sum_{j<c} \mathcal{B}^*_{\tau,j} \Phi^*_{\tau,c:j+1} y_{\tau,c}. \tag{29}$$

Thus a concrete sufficient form of Eq. (23) is

$$\|(\mathcal{J}^{W_2}_{\mathrm{rec}})^* y\|_F \ge s_{\mathrm{tr}} \|y\|_2, \qquad y \in \mathcal{G}_\theta. \tag{30}$$

This is the restricted singular value of the propagated recurrent hidden features. It accounts for both reinforcement and cancellation between uses of the shared parameters. Since the full parameter gradient contains its W_2 component, Eq. (30) implies Eq. (23). Unlike a full row-rank condition over all examples, it requires spanning only the corrections generated by the target trajectory. If these effective features are bounded, their target correction class has polynomial effective dimension, and their population Gram operator has a positive restricted gap, matrix Bernstein concentration transfers the gap to a polynomial sample without requiring the width to exceed the sample size.

The recurrent correction frame yields gradient dominance. For unit z, z^* with $\langle z, z^* \rangle \geq c_0$,

$$\|P_z^{\perp} z^*\|_2^2 = (1 + \langle z, z^* \rangle)(1 - \langle z, z^* \rangle) \geq (1 + c_0)(1 - \langle z, z^* \rangle).$$

Concatenate the weighted tangent forces $-\sqrt{a_i} P_{z_i}^{\perp} z_i^*$ and the copy force $\sqrt{\lambda_{\mathrm{cp}}} r_{\mathrm{cp}}$ into $G(\theta)$. With $\mu_0 := \min\{2, 1 + c_0\} > 0$, the preceding identity gives $\|G\|_2^2 \geq \mu_0 \mathcal{L}$. Differentiating Eq. (22) gives $\nabla_\theta \mathcal{L} = J_Z^* G$. By construction $G(\theta) \in \mathcal{G}_\theta$, so Eq. (23) yields

$$\|\nabla_\theta \mathcal{L}\|_2^2 \geq \mu_0 s_{\mathrm{tr}}^2 \mathcal{L}. \tag{31}$$

This proves the architecture-side gradient-dominance conclusion for the actual shared parameter vector; no independence of layer errors or separate parameter block for each depth is used.

Möbius reduction in the two-orbit quotient. Fix profiles for which the information visible to the block has the retained set and new frontier as its two active object orbits. Equivariance forces an effective coefficient to be constant within each orbit. On a shared active ReLU cell with no task-specific common-shift channel, the old-set and new-frontier coefficients after the affine writer are proportional to

$$\frac{A + \sqrt{k_c} B}{\sqrt{k_c}} \quad \text{and} \quad \frac{\alpha A + f_c B}{f_c}, \qquad A = 1 + \eta a, \quad B = \eta b_1'.$$

Radial normalization preserves their ratio. Because the recurrent block and common channel are shared across the profiles in g, writing their common ratio and two effective coefficients as α_g, A_g, B_g gives

$$R_s = \frac{\alpha_g A_g + f_s B_g}{A_g + \sqrt{k_s} B_g},$$

which is Eq. (13). Subtracting the target ratio $r_s = f_s / \sqrt{k_s}$ yields the exact factorization Eq. (24).

Combining Eqs. (25) and (26),

$$\mathcal{L}(\theta) \geq \frac{\lambda_{\mathrm{cal}} A_g^2}{2 d_+^2} \mathbb{E}[w_s (\alpha_g - r_s)^2 \mid g] \tag{32}$$

$$\geq \frac{\lambda_{\mathrm{cal}} A_g^2 \Delta_g}{2 d_+^2}. \tag{33}$$

Rearrangement gives

$$|A_g(\theta)| \leq d_+ \sqrt{2\mathcal{L}(\theta) / (\lambda_{\mathrm{cal}} \Delta_g)}.$$

Thus loss convergence identifies $A_g = 0$ inside the declared two-orbit quotient. The sharing of α_g is indispensable: allowing a separate $\alpha_s = r_s$ for every profile would remove the identifying restriction. The diversity condition is also necessary for this certificate: if r_s is constant on the positive-weight support, choosing the shared $\alpha_g = r_s$ makes the calibration residual vanish for any admissible A_g.

Residual necessity. Let two task histories h, h' have the same frontier-only observation w but different cumulative targets $z^*(h) \neq z^*(h')$. A residual-free shared writer receives w in both cases and therefore produces one common state y. The triangle inequality gives

$$\|z^*(h) - z^*(h')\|_2 \leq \|y - z^*(h)\|_2 + \|y - z^*(h')\|_2.$$

At least one Euclidean state distance is therefore at least half the target distance. Since the paper uses $\ell_i = \frac{1}{2}\|y - z_i^*\|_2^2$, the corresponding statement is

$$\max_{i \in \{h, h'\}} \ell_i \geq \frac{1}{8} \|z^*(h) - z^*(h')\|_2^2.$$

A residual or another persistent memory channel is necessary precisely when the task family contains such a history collision.

D Proof of the Supervision Theorem

Condition throughout on the fixed training sample $\mathcal{S}$. Fix a stage whose earliest unresolved depth is c and abbreviate $e = \widehat{e}_{c,N}(\theta_t)$, $L = L_c$, and $\mu = \mu_c$. Smoothness of $\widehat{e}_{c,N}$ and the SGD update give

$$\widehat{e}_{c,N}(\theta_{t+1}) \le \widehat{e}_{c,N}(\theta_t) - \eta_c \langle \nabla \widehat{e}_{c,N}, \widehat{g}_t \rangle + \frac{L\eta_c^2}{2} \|\widehat{g}_t\|_2^2. \tag{34}$$

Before applying this inequality, we verify the advertised supervision-side condition. The architecture theorem applied to the single-depth objective gives $\|\nabla \widehat{e}_{c,N}\|_2^2 \ge 2\mu_c e$. Hence direct frontier supervision satisfies Eq. (14) with $\rho = 1$. For a depth mixture, linearity of differentiation gives the exact identity

$$\left\langle \nabla \widehat{e}_{c,N}, \nabla \sum_j \pi_j \widehat{e}_{j,N} \right\rangle = \pi_c \|\nabla \widehat{e}_{c,N}\|_2^2 + \sum_{j \ne c} \pi_j G_{cj}.$$

Combining the single-depth lower bound with Eq. (15) proves Eq. (14). No sign assumption on every cross-depth term is needed.

On the simultaneous minibatch event in Theorem 4, Eq. (14) implies

$$\langle \nabla \widehat{e}_{c,N}, \widehat{g}_t \rangle \ge 2\rho\mu e - \rho\mu e = \rho\mu e.$$

Using $\|\widehat{g}_t\|_2^2 \le C_c e$ and $\eta_c \le \rho\mu/(LC_c)$ yields

$$\widehat{e}_{c,N}(\theta_{t+1}) \le \left(1 - \frac{1}{2}\eta_c \rho\mu\right) \widehat{e}_{c,N}(\theta_t). \tag{35}$$

The fixed inverse-polynomial step at stage c therefore requires at most

$$T_c \le \left\lceil \frac{2}{\eta_c \rho \mu_c} \log \frac{\widehat{e}_{c,N}(\theta_{t_c})}{\varepsilon_c} \right\rceil \tag{36}$$

updates. Prefix stability ensures that crossing this threshold does not invalidate any earlier one. Induction over $c = 1, \dots, D$ and summation of Eq. (36) prove the empirical trajectory-risk conclusion and the polynomial update bound.

For completeness, suppose the single-example projected gradient noise is conditionally sub-Gaussian with scale $\sigma_c \|\nabla \widehat{e}_{c,N}\|_2$ and its norm is conditionally sub-exponential with polynomial scale. Standard minibatch concentration then makes both inequalities in the theorem hold at any update with failure probability at most $\delta/(1 + \sum_c T_c)$ once the batch size is polynomial in σ_c, $1/(\rho\mu_c\varepsilon_c)$, and $\log((1 + \sum_c T_c)/\delta)$. A union bound gives the simultaneous event. The stochastic argument uses the same smooth training region as the architecture theorem.

The empirical statement transfers to unseen i.i.d. tasks under an explicit uniform-convergence condition. Let the parameter region $\Theta_{\mathcal{C}} \subset \mathbb{R}^P$ visited by training lie in a radius-R ball, and suppose the per-task trajectory loss is bounded by B_0 and is G_θ-Lipschitz in θ throughout this set. A Euclidean net, Hoeffding's inequality, and a union bound give

$$\sup_{\theta \in \Theta_{\mathcal{C}}} |\widehat{\mathcal{L}}_N(\theta) - \mathcal{L}_{\mathrm{pop}}(\theta)| \le \varepsilon_{\mathrm{gen}} \tag{37}$$

with probability at least $1 - \delta$ whenever

$$N \ge \frac{2B_0^2}{\varepsilon_{\mathrm{gen}}^2} \left[P \log\left(1 + \frac{8RG_\theta}{\varepsilon_{\mathrm{gen}}}\right) + \log\frac{2}{\delta} \right].$$

Indeed, use a net of radius $\varepsilon_{\mathrm{gen}}/(4G_\theta)$, whose size is at most the exponential of the bracketed dimension term. Hoeffding controls every net point, while the two Lipschitz interpolation errors contribute the remaining half of $\varepsilon_{\mathrm{gen}}$. Apply the bound to the mean trajectory loss. Hence empirical terminal risk at most $D^{-1}\sum_c \varepsilon_c$ gives population risk at most $D^{-1}\sum_c \varepsilon_c + \varepsilon_{\mathrm{gen}}$. The feature-frame and regularity conditions remain task-family assumptions; uniform convergence does not create them.

Finally, Eq. (35) recovers the architecture certificate. As the trajectory loss approaches zero, the bound proved in App. C forces $A_g \to 0$. Training therefore learns both the target superposition trajectory and, within the stated chart, the shortcut-free Möbius mechanism that realizes it.

E Proof of Budget-Optimal Supervision

Depths with $e_c^{\mathrm{in}} = 0$ need no allocation and may be omitted. For differentiable, decreasing, strictly convex certified curves $\bar{e}_c(x_c)$, the KKT stationarity condition for an active depth is

$$\frac{\bar{e}_c'(x_c)}{D} + \lambda b q_c = 0,$$

whereas an inactive depth satisfies $-\bar{e}_c'(0)/(bDq_c) \le \lambda$. These conditions are sufficient because the program is convex, and they prove the marginal-equalization claim.

For the exponential specialization, consider the divisible-resource relaxation

$$\min_{x_c \ge 0} \quad \frac{1}{D}\sum_{c=1}^{D} e_c^{\mathrm{in}} e^{-a_c x_c} \qquad \text{subject to} \qquad b\sum_{c=1}^{D} q_c x_c \le B_{\mathrm{tr}}.$$

If every initial error is zero, the optimum is zero and we choose $x^* = 0$. Otherwise, the objective is strictly convex and decreasing in every retained coordinate, so the budget is active and the minimizer is unique. With multiplier $\lambda \ge 0$ and nonnegativity multipliers $\nu_c \ge 0$, the Lagrangian is

$$\mathcal{L}_{\mathrm{KKT}}(x, \lambda, \nu) = \frac{1}{D}\sum_{c} e_c^{\mathrm{in}} e^{-a_c x_c} + \lambda\left(b\sum_{c} q_c x_c - B_{\mathrm{tr}}\right) - \sum_{c} \nu_c x_c.$$

For an active depth, stationarity gives

$$-\frac{a_c e_c^{\mathrm{in}}}{D} e^{-a_c x_c} + \lambda b q_c = 0,$$

and therefore

$$x_c = \frac{1}{a_c}\log\frac{a_c e_c^{\mathrm{in}}}{\lambda b D q_c}.$$

Complementary slackness clips negative values to zero, proving Eq. (17). The left-hand side of its budget equation is continuous and strictly decreasing in λ between successive activation thresholds, tends to infinity as $\lambda \downarrow 0$, and equals zero above the largest threshold. Because $B_{\mathrm{tr}} > 0$, it determines a unique multiplier and hence a unique allocation.

For every active coordinate, stationarity also gives $e_c^{\mathrm{in}} e^{-a_c x_c} = \lambda b D q_c / a_c$; an inactive coordinate retains e_c^{in}. Substitution proves Eq. (18). The marginal decrease in trajectory error per unit cost is

$$-\frac{\partial \overline{\mathcal{R}}_Z / \partial x_c}{b q_c} = \frac{a_c e_c^{\mathrm{in}} e^{-a_c x_c}}{b D q_c},$$

which is equal across all active depths at the optimum. Before taking a small additional allocation step, maximizing this quantity is equivalent to choosing c with largest $a_c e_c / q_c$. Replacing a_c, e_c by the cached estimates in Eq. (19) gives Eq. (20). This measured policy implements the optimal marginal rule. We now quantify the effect of estimation error rather than identifying the measured schedule with the oracle allocation.

Write cost coordinates as $y_c = b q_c x_c$, so that $\sum_c y_c = B_{\mathrm{tr}}$, and put $m_c(y_c) := -\frac{d}{dy_c}[D^{-1}\bar{e}_c(y_c/(bq_c))]$. Convexity makes each m_c nonincreasing. Suppose a continuous plug-in schedule always allocates its next infinitesimal unit to a maximizer of $\widehat{m}_c$ and that $|\widehat{m}_c - m_c| \le \epsilon_m$ throughout. At the last time any active coordinate c receives budget, its true marginal is at least every competing marginal minus $2\epsilon_m$; competing marginals can only decrease afterward. Consequently, at the final allocation $\widehat{y}$,

$$m_c(\widehat{y}_c) \ge \max_j m_j(\widehat{y}_j) - 2\epsilon_m \quad \text{whenever } \widehat{y}_c > 0.$$

Let y^* be the oracle allocation. Convexity and equality of the two total budgets give

$$\begin{aligned}\overline{\mathcal{R}}_Z(\widehat{y}) - \overline{\mathcal{R}}_Z(y^*) &\le \nabla\overline{\mathcal{R}}_Z(\widehat{y})^\top(\widehat{y} - y^*) \\ &= \sum_c m_c(\widehat{y}_c)(y_c^* - \widehat{y}_c) \le 2\epsilon_m B_{\mathrm{tr}},\end{aligned}$$

Table 2: Experimental tests aligned with the learnability and budget results. Each intervention changes one quantity appearing in the corresponding theorem and evaluates the learned reasoning trajectory. The permutation symmetry result is a mathematical lemma and is not treated as a separate empirical claim.

Claim	Controlled intervention	Direct measurement	Predicted outcome
Architecture	Gate the residual path; sweep attention margin and MLP-update scale	Trajectory error, attention margin, and recurrent-frame conditioning	A learnability boundary appears
Supervision	Change the intermediate state supplying the gradient; hold training fixed	Corrective gradient signal and counterfactual one-step error decrease	The signal predicts the local correction
Fixed-budget loss	Allocate one intermediate-state-label budget by four rules; include answer-only and full-supervision references	Trajectory error, task accuracy, set recovery, and realized label cost	Cost-aware allocation improves the error–budget tradeoff

which proves Eq. (21). The argument is literal for finitely many switches; the general piecewise-continuous case follows by approximating its allocation path by finite-switch paths and using continuity of the response curves. Under a curriculum, applying the same maximization to the currently available coordinates $A_t(\tau)$ gives the feasible per-example marginal rule in Eq. (20).

The result is an exact optimum for divisible supervision effort. A practical integer schedule can round the allocation and then spend the remaining budget greedily according to the same marginal score; the theorem does not equate a random schedule at its expected allocation with the continuous curve. It optimizes the complete mean superposition-trajectory error, not a single intermediate layer and not exact zero–one answer accuracy. If the terminal decoder loss is L_{dec}-Lipschitz in z_D, then its excess loss is at most $L_{\mathrm{dec}}\sqrt{2e_D}$, so reducing the trajectory objective also controls the downstream reasoning loss. When only the earliest unresolved depth is locally contractible, every other marginal value is zero and the same allocation reduces to frontier supervision.

F Additional Related Work

Continuous and latent reasoning. Coconut (Hao et al., 2024) introduced continuous chain-of-thought, and Zhu et al. (2025) showed that continuous states can represent multiple search candidates. Complementary expressivity results characterize what explicit or latent chains can compute (Merrill & Sabharwal, 2024; Li et al., 2024; Merrill & Sabharwal, 2023; Sanford et al., 2024; Goyal et al., 2023; Pfau et al., 2024; Deng et al., 2024). Our result is about learnability: it gives conditions under which gradient training recovers the intended sequence of candidate sets.

Training dynamics and intermediate-state supervision. Analyses of attention training study margin formation, feature selection, and stagewise learning (Tian et al., 2023; Tarzanagh et al., 2023; Ataee Tarzanagh et al., 2023; Nichani et al., 2024; Bietti et al., 2023; Boix-Adserà et al., 2023). Work on intermediate-step feedback shows that supervising reasoning steps can improve final performance (Uesato et al., 2022; Lightman et al., 2024). We connect these directions by identifying the gradient condition that corrects an inaccurate intermediate state and by allocating a finite label budget across those states.

G Additional Experimental Details

Matched-budget protocol. Every exposed target state costs one label, repeated exposures are charged again, and labels used to estimate learning speed are included in B_{cal}. All policies receive the same fixed calibration cache and are charged for it once. Within each seed, all methods start from the same answer-only checkpoint. Intermediate-state-supervised methods also share the model and AdamW state produced by the common first supervision stage, and branch only when different depth allocations become possible.

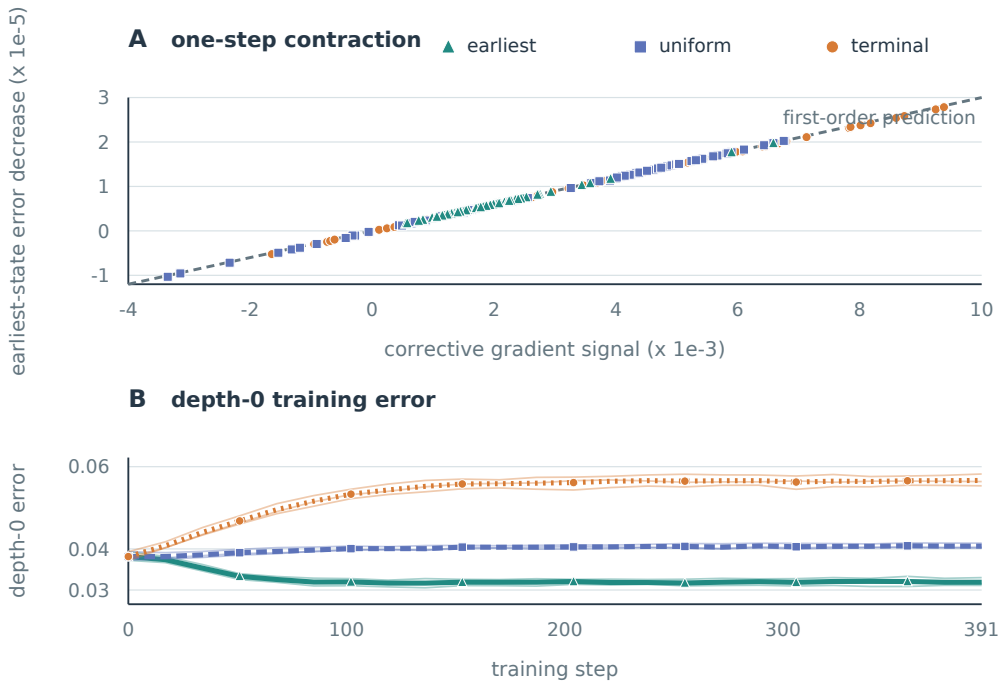

Figure 4: **The corrective gradient signal matches counterfactual one-step contraction.** **A**: for 216 policy–checkpoint probes over three seeds, the measured gradient inner product is compared with the decrease in the earliest inaccurate state's error after one counterfactual gradient step at the same parameters; the dashed line is the first-order prediction. **B**: three-seed means of the depth-0 state error during training; thin traces show individual seeds.

Table 3: Final unseen-task performance under matched intermediate-state-label budgets. Answer-only and full supervision are reference points and are not equal-budget competitors.

Method	Accuracy ↑	Trajectory error ↓	Set-F1 ↑	Label cost	Seeds
Terminal-only	0.704	0.355	0.612	$1.00B$	3
Uniform	0.742	0.248	0.701	$1.00B$	3
Earliest inaccurate state	0.758	0.225	0.728	$1.00B$	3
Cost-aware	**0.881**	**0.158**	**0.792**	$1.00B$	3
Answer-only reference	0.681	0.492	0.524	0	3
Full-supervision reference	0.789	0.142	0.805	$1.60B$	3