WHEN DOES COORDINATION INDUCE CORRELATED FAILURE IN MULTI-AGENT LLM REASONING?

Anonymous authors
Paper under double-blind review

ABSTRACT

Multi-agent LLM systems increasingly rely on a meta-agent that issues a shared, explicit plan to coordinate otherwise-independent executor agents, but whether such coordination improves or harms reasoning reliability is not well characterized. We present Controlled Deliberative Decentralized Execution, a minimal framework that isolates coordination strength as a single manipulable variable. We evaluate seven open-weight models across three reasoning domains. We introduce a two-statistic diagnostic framework — a correctness intraclass correlation (ρ) and an excess pairwise agreement statistic (κ) — designed to separate genuine correlated failure from benign narrowing of each executor's individual answer distribution. Strong coordination reduces accuracy relative to self-consistency in 19 of 21 (model, domain) cells, and this directional harm is unanimous at the model level: all 7 models are harmed on at least two of three domains. Critically, the underlying mechanism is not uniform: only three of seven models exhibit the correlated-cascade signature (rising κ) while the remaining four show comparable harm through a distinct, non-correlational route. This mechanistic split tracks neither model scale nor family. One robust exception survives across independent measurement corrections: Qwen2.5-3B on MATH shows a significant accuracy *improvement* under coordination, confirming that agreement's decoupling from correctness cuts in both directions. We argue that executor agreement is an unreliable proxy for reliability and that single-mechanism governance heuristics are miscalibrated for a meaningful fraction of deployed systems.

1 INTRODUCTION

A growing share of applied LLM systems no longer consist of a single model answering a single prompt. Instead, a coordinating *meta-agent* decomposes a task, assigns it to specialized or redundant executor agents, and aggregates their outputs (Wu et al., 2023; Hong et al., 2024; Li et al., 2023; Shen et al., 2023). This pattern is attractive because a shared plan should, in principle, reduce redundant strategy derivation, improve auditability, and let executors specialize on carrying out a plan rather than inventing one. Whether this promise holds empirically, however, is a separate question from whether it is architecturally appealing, and recent large-scale failure analyses of deployed multi-agent systems suggest the answer is not straightforward: Cemri et al. (2026) manually annotated agent traces across seven popular multi-agent frameworks and found that coordination gains over single-agent baselines were frequently minimal, identifying fourteen distinct failure modes.

Parallel literature has separately established that LLMs—even independently sampled instances of the same model—do not fail independently. Kim et al. (2025) show across more than 350 models that error correlation is substantial and, counter-intuitively, *increases* with model accuracy and capability, with direct implications for any system (LLM-as-judge, hiring pipelines, multi-agent ensembles) that assumes independence to justify aggregation. If executors already correlate before any explicit coordination is introduced, then the central empirical question for meta-agent design is not whether coordination increases raw agreement — it almost certainly will, since a shared plan mechanically narrows the space of trajectories each executor explores — but whether it increases agreement *beyond* this baseline correlation, and specifically whether it does so on *wrong* answers.

This distinction is easily conflated in practice. An ensemble can converge because (a) the shared plan narrows what each executor, considered individually, tends to say (a marginal effect, and on its

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.

own benign or even desirable), or (b) the shared plan pushes multiple executors toward the identical answer, correct or not (a joint, correlational effect, and the one that makes "high agreement, wrong answer" outcomes dangerous rather than merely inconsistent). Plurality agreement, the metric used throughout the meta-agent and self-consistency literature (Wang et al., 2023), cannot distinguish these two cases by construction. We introduce a diagnostic framework that can, and use it across a substantially wider evidence base than prior single-model, single-domain studies to ask: does coordination-induced failure, where it occurs, reflect genuine correlated collapse, or something else?

Research questions. (RQ1) Does coordination strength systematically affect accuracy, and in which direction, across models and domains? (RQ2) Where accuracy is harmed, is the harm attributable to genuine correlated failure (executors converging on each other's specific wrong answer beyond what their individual answer distributions predict) or to a different mechanism? (RQ3) Do these effects and mechanisms generalize across model scale and family, or are they localized to individual architectures and domains?

Contributions.

- We provide a comprehensive controlled evaluation of coordination strength as an inference-time variable in multi-agent LLM reasoning: 7 models $\times$ 3 domains $\times$ 4 coordination conditions $\times$ 3 seeds, with cluster-robust inference throughout.
- We introduce a correctness-ICC (ρ) and excess-agreement (κ) diagnostic framework, grounded in the classifier-ensemble diversity literature (Krogh & Vedelsby, 1995; Kuncheva & Whitaker, 2003), that separates marginal narrowing from genuine correlated failure—a distinction plurality agreement alone cannot make.
- We show that while the dominant empirical pattern (coordination harms accuracy) is robust and broad, its *mechanism* is highly idiosyncratic: only 3 of 7 models show the correlated-cascade signature. Membership in this group tracks neither model scale, nor model family, nor baseline (self-consistency) accuracy.
- We identify and verify a robust, domain-specific exception (Qwen2.5-3B / MATH) in which coordination significantly improves accuracy, demonstrating that the rare exceptions to the dominant pattern are genuine structural phenomena rather than measurement artifacts.
- We argue that single-mechanism governance heuristics are miscalibrated given the mechanism heterogeneity we observe and propose a per-deployment coordination-sensitivity audit as an alternative.

2 Related Work

Inference-Time Reasoning Strategies and Self-Consistency Wei et al. (2022) showed that eliciting intermediate reasoning steps improves LLM performance on multi-step tasks. Wang et al. (2023) extended this by sampling multiple independent chain-of-thought trajectories and aggregating via majority vote, a method that improves robustness precisely because diverse trajectories can converge on a correct answer through different paths. Yao et al. (2023) generalized sampling to explicit search over a branching reasoning tree, and Snell et al. (2024) characterize the general compute–accuracy tradeoff across inference-time scaling strategies. Lightman et al. (2024) show process reward models can guide step-level search on competition mathematics. All of these treat trajectory diversity as a resource; what remains unexamined is what happens when that diversity is deliberately suppressed by a shared upstream plan.

Multi-Agent LLM Systems and Coordination Failure Orchestrator/meta-agent frameworks such as AutoGen (Wu et al., 2023), MetaGPT (Hong et al., 2024), CAMEL (Li et al., 2023), and HuggingGPT (Shen et al., 2023) demonstrate that delegation and role assignment can improve collaborative problem-solving, but their architectural richness makes isolating the causal contribution of coordination itself difficult. Du et al. (2023) and Liang et al. (2024) show iterative multi-agent debate can improve accuracy but caution that debate can also converge prematurely, eroding the diversity that gave the ensemble its benefit in the first place. Madaan et al. (2023) study a single agent iteratively critiquing its own output, conflating the planning and execution roles our framework deliberately separates. Closest in spirit to our motivating question, Cemri et al. (2026) identify

inter-agent misalignment as a dominant failure category across seven deployed frameworks. Kaesberg et al. (2025) study how the choice of aggregation rule shapes debate outcomes; and (Flint et al., 2026) document that collective misalignment can emerge purely from coordination dynamics. These studies document what goes wrong in deployed multi-agent systems; what remains open is a controlled, mechanistic account of whether and why coordination itself induces failure.

Correlated Errors and Ensemble Diversity Our ρ/κ diagnostic framework is grounded in the classical ensemble-diversity literature. Krogh & Vedelsby (1995) derive the ambiguity decomposition, showing that ensemble generalization error decomposes into the average error of individual members minus their disagreement. Kuncheva & Whitaker (2003) survey and formalize multiple pairwise and non-pairwise diversity measures for classifier ensembles; here, we adapt these formulations to the setting where "classifiers" are stochastic rollouts conditioned on shared inputs. Within the LLM literature specifically, Kim et al. (2025) find across a large-scale evaluation of over 350 models that error correlation is substantial and increases with model capability rather than decreasing—a finding that motivates our decision to measure baseline correlation explicitly rather than assume it away. Correlation-aware ensemble selection methods RL-Focal (Tekin et al., 2025), LLM-TOPLA (Tekin et al., 2024) minimizes error correlation *ex ante* by choosing which models to combine. We instead track correlation emerging *ex post* under shared plans, since our domains have deterministic answers, for which exact-match κ is the appropriate diagnostic. Our ρ/κ statistics could thus validate such selectors after deployment.

Positioning of This Work Three literatures—inference-time scaling (Snell et al., 2024), multi-agent failure analysis (Cemri et al., 2026), and ensemble diversity theory (Krogh & Vedelsby, 1995; Kuncheva & Whitaker, 2003)—study the same underlying phenomenon through different lenses: compute–accuracy tradeoffs, deployed-system failure taxonomies, and the mathematical bounds within which agreement stays informative, respectively. CDDE combines a factorial, causal manipulation of coordination strength (the first two lenses) with a ρ/κ diagnostic that operationalizes the coupling quantities ensemble theory identifies as relevant, without deriving explicit failure bounds (the third). The result is not a new system or failure taxonomy, but a unified measurement protocol any practitioner can apply to their own deployment.

3 METHOD

We first formalize an assumption that prior self-consistency analyses leave implicit, then demonstrate that our two coupling statistics directly test it rather than merely describing agreement.

3.1 PROBLEM SETTING & BASELINE CORRELATION IN SELF-CONSISTENCY.

Let $x \in \mathcal{X}$ denote a question and $y \in \mathcal{Y}$ an answer. A model $\mathcal{M}$ defines $P_{\mathcal{M}}(y \mid x)$. Self-consistency (Wang et al., 2023) draws k samples $y_1, \ldots, y_k \sim P_{\mathcal{M}}(y \mid x)$ and returns $\hat{y} = \text{Plurality}(y_1, \ldots, y_k)$.

The classical justification for majority voting assumes executor errors are close to independent, so diversity outweighs shared bias. Correlated errors do not introduce a discontinuity in this argument. There is no single point past which voting 'breaks'; rather, correlation continuously erodes the benefit of additional samples, vanishing entirely in the limit of perfect correlation ($\rho = 1$), where all k executors are identical and majority voting recovers exactly the single-sample accuracy. Furthermore, when correlation is driven by a deeply flawed shared plan (the Strategic Cascade), majority voting can actively drive ensemble accuracy *below* the single-sample baseline. Prior work leaves the actual correlation level unmeasured, which lets practitioners read rising plurality agreement as evidence of robustness by default. We do not test independence against a single alternative; we *measure* the correlation prior work leaves implicit, using ρ and κ below.

3.2 THE ρ/κ DIAGNOSTIC FRAMEWORK

Shared-plan factorization. Let $\pi \in \Pi$ denote a plan — an abstract decomposition into ordered sub-goals, without explicit sub-answers. We factorize

$$P(y \mid x) = \sum_{\pi} P(\pi \mid x)\, P(y \mid x, \pi), \tag{1}$$

where $P(\pi \mid x)$ captures *strategic* uncertainty (which decomposition is chosen) and $P(y \mid x, \pi)$ captures *execution* uncertainty (how it is carried out). By the law of total variance, for any scalar outcome Y,

$$\mathrm{Var}(Y) = \underbrace{\mathbb{E}_{\pi}[\mathrm{Var}(Y \mid \pi)]}_{\text{execution variance}} + \underbrace{\mathrm{Var}_{\pi}(\mathbb{E}[Y \mid \pi])}_{\text{strategic variance}}. \tag{2}$$

What Eq. 2 does and does not claim. It is an identity, not a directional prediction. Sharing one π across all k executors removes the strategic-variance term from the *disagreement between executors*, so plurality agreement should rise under coordination essentially by construction, regardless of whether π happens to be a good or a bad plan. This is the one load-bearing consequence we take from the equation. Eq. 2 does *not* predict whether mean accuracy $\mathbb{E}[Y]$ rises or falls under coordination; that depends on the empirical quality of $P(\pi \mid x)$ relative to the trajectories it replaces, which we measure (Section 5) rather than derive. A theory predicting the sign of the accuracy effect would be falsified by four of our seven models. What Eq. 2 does predict, and what our results test, is narrower and sharper: that agreement becomes *decoupled from correctness* once π is shared.

Agreement. Given k executor outputs, agreement is $A = \max_a |\{i : y_i = a\}|/k$, the quantity Eq. 2 predicts will rise under coordination, independent of whether π is sound.

Correctness ICC (ρ): quantifying baseline coupling. Setting $Y = C_i$ in Eq. 2 and reparametrizing execution/strategic variance as within- and between-question components under a one-way random-effects model gives the intraclass correlation

$$\rho = \frac{\sigma_b^2}{\sigma_b^2 + \sigma_w^2}, \tag{3}$$

the fraction of total variance in executor correctness attributable to *which question* is asked rather than *which executor-and-seed* answered it. Strict independence corresponds to $\rho = 0$; as ρ increases, majority voting's benefit over the base success rate shrinks continuously, vanishing at $\rho = 1$ (above). By measuring ρ, we quantify the baseline coupling prior work assumed away, without asserting a specific critical value at which voting "fails." The moment estimator we use is given in Appendix D.

Limitation of ρ. We note that a between-question ANOVA groups observations by item, so $\rho > 0$ can arise purely from cross-item difficulty heterogeneity even under strict conditional independence. ρ should therefore be read as an upper bound on genuine within-question executor coupling; disentangling the two requires a mixed-effects model with question as a fixed covariate, which we leave to future work. Because our cascade/degradation mechanism split (Section 5.4) is based on $\Delta\kappa$, computed from pairwise agreement on the *identical* question set, question-difficulty structure is differenced out and this limitation does not affect our primary mechanistic finding.

Excess agreement (κ): distinguishing *which* answer executors couple on. ρ operates on binary correctness and cannot distinguish "coupled on the right answer" from "coupled on a specific wrong answer" — the distinction that determines whether coupling is benign or dangerous. We additionally compute a Fleiss'-κ-style (Fleiss, 1971) statistic on the categorical answer distribution itself: let p_a be the observed rate at which two independently drawn executors on the same question give the same answer, and p_e the rate expected under the cell's pooled marginal answer-frequency distribution alone:

$$\kappa = \frac{p_a - p_e}{1 - p_e}. \tag{4}$$

$\kappa \approx 0$: agreement is fully explained by each executor's own answer distribution narrowing under π — Eq. 2's benign, structurally guaranteed effect, and nothing more. $\kappa \gg 0$: executors converge

on each other's specific answer beyond what their individual distributions predict — a genuine correlated cascade. p_e is pooled globally per cell following the standard Fleiss' convention. In diffuse, low-agreement cells, pooled p_e is small and κ approaches raw pairwise agreement; since these are also the cells with low $\Delta\kappa$ by construction, this does not risk manufacturing a spurious cascade signal. Together, ρ and κ form the two-instrument measurement apparatus that plurality agreement alone cannot provide.

3.3 CDDE: Instrument for Isolating Coordination Strength

CDDE separates reasoning into (1) a single centralized planner call producing a plan $\pi \sim P(\pi \mid x)$, explicitly barred from performing arithmetic or stating a final answer, and (2) $k = 5$ independent executors that condition on (x, π) and terminate with a parseable `FINAL_ANSWER:` tag, sampled across a fixed temperature spectrum $T \in \{0.6, 0.7, 0.8, 0.9, 1.0\}$ to preserve stochastic diversity in execution while holding the strategic signal fixed. Final answers are aggregated by majority vote (Eq. 1–4 apply directly to this aggregate). We operationalize coordination strength as planner prompt specificity across three regimes — WEAK (2–3 line abstract guidance), DEFAULT (ordered sub-goals, no explicit computation), and STRONG (fully enumerated sub-problem decomposition with named quantities and computation order, still prohibited from computing) — plus a SELF_CONSISTENCY baseline with no plan at all, corresponding to setting coordination strength to zero. In all CDDE conditions, the planner and executors share the *same* model checkpoint. How cross-model planning (e.g., a larger planner directing smaller executors) modulates cascade susceptibility remains an open dimension of coordination design. Full prompt templates are given in Appendix A. CDDE is deliberately minimal relative to production orchestration frameworks: it is the instrument used to isolate coordination strength as a single controlled variable, not a proposed production architecture. This choice trades architectural realism for causal interpretability, in the same spirit as controlled inference-time-scaling ablations (Snell et al., 2024).

4 Experimental Setup

Models. Seven open-weight instruction-tuned models spanning three families and a 1.5B–9B parameter range: Qwen2.5-1.5B/3B/7B-Instruct (within-family scaling), Gemma2-2B/9B-Instruct (cross-family scaling), Llama-3.1-8B-Instruct, and Mistral-7B-Instruct (cross-family, matched-scale comparisons), all complete at $N = 300$ per cell. All models served locally via Ollama at consistent quantization; the exact tag used for each model is recorded in Appendix A, Table 2 to guard against quantization-level confounds in cross-family comparisons.

Domains. GSM8K (Cobbe et al., 2021) (grade-school arithmetic), a Hendrycks MATH (Hendrycks et al., 2021) subset (algebra/number-theory), and HotpotQA (Yang et al., 2018) multi-hop, closed-book, testing parametric recall rather than retrieval-augmented reasoning (discussed in Section 7). 100 questions per domain, fixed and identical across all conditions and models for a given seed, replicated across 3 seeds (300 observations per model$\times$domain$\times$condition cell).

Statistical procedure and data validation. All confidence intervals use a cluster-robust percentile bootstrap (10,000 resamples, question_id as the cluster key) to account for the fact that the 3 seeds per question are repeated measures, not independent draws. Per-cell significance between self-consistency and strong coordination uses two complementary tests: a question-level cluster-permutation test on continuous per-question mean correctness (10,000 permutations) and a cluster McNemar test collapsing each question to a majority-vote outcome per condition. Benjamini–Hochberg FDR correction (Benjamini & Hochberg, 1995) is applied separately; one marginal result (Mistral-7B/HotpotQA) loses significance under correction, with no other calls changing. We treat the permutation test as primary. Two extraction-pipeline defects were identified and corrected; affected cells were regenerated (Appendix E).

5 Results

From theory to testable predictions. H1 (agreement rises): Sharing π removes strategic variance from inter-executor disagreement, so plurality agreement should rise under STRONG coordination relative to self-consistency, independent of plan quality.

Table 1: Per-model mechanism classification and per-domain shifts (Δ = Strong − Self-Consistency); mean $\Delta\kappa$ across domains determines mechanism group (shaded rows = cascade). Full per-cell accuracies, CIs, agreement values, and significance tests are in Appendix L.

Model	Mech.	Mean $\Delta\kappa$	GSM8K		HotpotQA		MATH	
			ΔAcc	$\Delta\kappa$	ΔAcc	$\Delta\kappa$	ΔAcc	$\Delta\kappa$
Gemma2-2B	Deg.	$+0.003$	-0.213	$+0.027$	-0.043	-0.021	-0.087	$+0.002$
Gemma2-9B	Deg.	$+0.012$	-0.050	$+0.028$	$+0.010$	$+0.035$	-0.037	-0.026
Llama3.1-8B	Deg.	-0.059	-0.236	-0.097	-0.076	-0.066	-0.224	-0.013
Qwen2.5-7B	Deg.	-0.037	-0.090	-0.054	-0.046	-0.034	-0.027	-0.024
Qwen2.5-1.5B	Cas.	$+0.160$	-0.150	$+0.250$	-0.040	$+0.089$	-0.070	$+0.142$
Qwen2.5-3B	Cas.	$+0.102$	-0.053	$+0.148$	-0.110	-0.017	$+0.080$	$+0.176$
Mistral-7B	Cas.	$+0.140$	-0.240	$+0.177$	-0.027	$+0.006$	-0.074	$+0.238$

H2 (agreement is not diagnostic of correctness): Because Eq. 2 is symmetric in whether π is sound, an agreement rise should co-occur with cells where accuracy falls *and* with cells where it rises. **H3 (cascade requires κ, not just A):** If coordination pushes executor coupling past the level where agreement remains informative, κ should rise specifically where accuracy loss reflects genuinely shared error, and should not rise where accuracy loss occurs by a different route.

H4 (baseline coupling): If model capability predicts baseline error correlation (Kim et al., 2025), self-consistency's own ρ should show substantial question-level clustering of correctness within our executor ensembles too. We test each in turn.

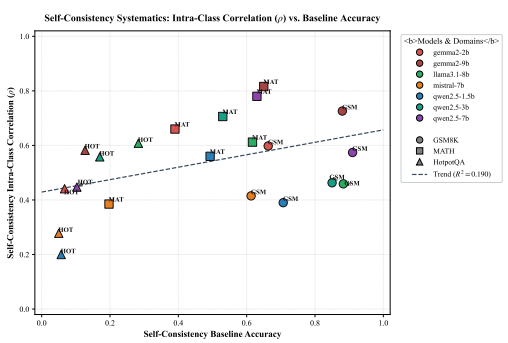
5.1 Coordination Strength Reduces Accuracy in the Large Majority of Settings

Across 21 (model, domain) cells, strong coordination reduces accuracy relative to self-consistency in 19 (binomial sign test, $p = 2.2 \times 10^{-4}$), improves it in 2, with no ties. Cells are not strictly independent (each model recurs across three domains), so we also treat models as the unit: all 7 are harmed on at least 2 of 3 domains, a unanimous model-level sign (7/7). The relevant question thus shifts from "does coordination harm?" to "Does it harm uniformly across domains for a given model?" It does, in direction, for 5 of 7 models; the remaining two show a domain-specific reversal: Gemma2-9B on HotpotQA (non-significant, $p = 0.39$) and Qwen2.5-3B on MATH (the only reversal that reaches significance). This is a sign-only claim, robust to test choice and to the Appendix E corrections.

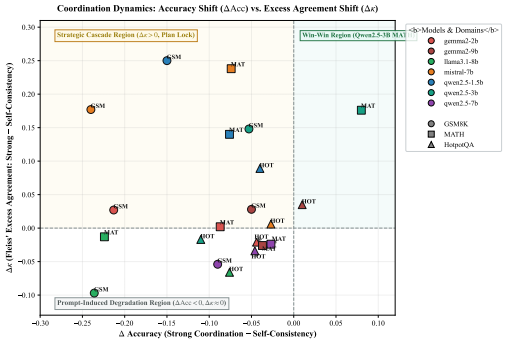
HotpotQA's low base rates (1.7–28%) yield coarser tests and wider ρ/κ uncertainty than the other domains (Section 7); the directional pattern remains consistent. GSM8K shows this pattern with no exceptions: all 7 models are directionally harmed. HotpotQA shows it in 6 of 7 (Gemma2-9B is the exception, $\Delta\text{Acc} = +0.010$, non-significant, $p = 0.39$), and the effect reaches FDR-corrected significance (permutation $q < 0.05$) in 11 of these 14 GSM8K+HotpotQA cells. Across the full 21-cell matrix, permutation and McNemar tests diverge sharply in how much they detect post-correction: 16 of 21 cells are permutation-significant versus 9 of 21 McNemar-significant, a numerical confirmation of the McNemar power loss in sparse-accuracy domains documented in Section 4. H2 is confirmed by the one cell where accuracy *improves*: Qwen2.5-3B/MATH's agreement also rises (0.581→0.711) despite the accuracy gain, consistent with agreement being symmetric in plan quality (Section 5.3).

5.2 H1, Qualified: Agreement Rises Selectively, and Exactly Where the Cascade Signature Predicts

H1, read literally and unconditionally, is false: agreement under STRONG exceeds self-consistency in only 12 of 21 cells (57%), not all of them. But the split is not random with respect to the mechanism grouping developed in Section 5.4: the three cascade-signature models (Qwen2.5-1.5B, Qwen2.5-3B, Mistral-7B) show agreement rising in *every one* of their 9 combined cells, while the four non-



(a) **Baseline State:** Self-consistency accuracy vs. baseline coupling (ρ).



(b) **Coordination Effect:** ΔAccuracy vs. $\Delta\kappa$ reveals the Mechanism Split.

Figure 1: **The Coordination Dynamics. (a)** Even without explicit coordination, executor correctness is strongly question-clustered ($\rho > 0$); per Appendix D this value upper-bounds genuine baseline coupling. **(b)** When a shared plan is introduced, the shift in accuracy (ΔAcc) and excess agreement ($\Delta\kappa$) reveals two distinct failure mechanisms: Strategic Cascade (rising κ) and Prompt-Induced Degradation ($\Delta\kappa \approx 0$).

cascade models show it rising in only 3 of their 12 combined cells, and no non-cascade model shows agreement rising across all three of its domains while every cascade-group model does. In our sample, a cheap diagnostic — does agreement rise under coordination in every domain tested for a given model — perfectly separates the two mechanism groups without computing κ. We report this as a corroborating diagnostic, not a replacement for κ.

5.3 MATH Is the Domain Where the Pattern Breaks Down

On MATH, 6 of 7 models are directionally harmed by strong coordination; under our primary permutation test, four reach FDR-corrected significance (Gemma2-2B, Llama3.1-8B, Mistral-7B, Qwen2.5-1.5B) and two do not (Gemma2-9B, Qwen2.5-7B). Under the secondary cluster McNemar test only two of the six remain significant (Gemma2-2B, Llama3.1-8B). One model—Qwen2.5-3B—shows a significant improvement (self-consistency 0.530 $\rightarrow$ strong 0.610, $\Delta = +0.080$, permutation $p = 0.0036$). This reversal survived two independent extraction-pipeline corrections with persistently significant magnitude ($\Delta = +0.100, p = 0.0003 \rightarrow \Delta = +0.093, p = 0.0012 \rightarrow \Delta = +0.080, p = 0.0036$), confirming a genuine effect rather than a measurement artifact.

5.4 The Mechanism Is Not Universal: Correlated Cascade vs. Uncorrelated Degradation

We compute $\Delta\kappa$ (excess agreement, self-consistency to strong, Eq. 4) per model, averaged across domains, to test whether accuracy loss is attributable to genuine correlated failure. Three models show a clear rise in mean $\Delta\kappa$ across domains—Qwen2.5-1.5B ($+0.160$), Mistral-7B ($+0.140$), Qwen2.5-3B ($+0.102$)—the correlated-cascade signature; membership is defined by this cross-domain mean (Table 1), not by a rise in every individual domain (e.g., Qwen2.5-3B's own HotpotQA $\Delta\kappa$ is -0.017 and Mistral-7B's is $+0.006$). The remaining four models (Qwen2.5-7B, Llama-3.1-8B, Gemma2-2B, Gemma2-9B) show flat or declining κ ($\Delta\kappa$ between -0.06 and $+0.03$) despite comparable or larger accuracy losses (e.g. Llama3.1-8B loses 22–24 accuracy points on GSM8K and MATH with $\Delta\kappa < 0$). This rules out several simplifications: not a small-model effect (Gemma2-2B, the smallest tested, is non-cascade), not a family effect (the cascade group spans Qwen2.5 and Mistral), and not a scale effect (Mistral-7B cascades while Qwen2.5-7B at the same scale does not). Nor does it track baseline accuracy: the three cascade-prone models span self-consistency accuracies from 0.287 (Mistral-7B, the lowest in the study) to 0.517 (Qwen2.5-3B), overlapping substantially with the non-cascade group's 0.373–0.594 range. We do not currently have an explanation for what these three models share. Note that this is consistent with, though not a strict per-question derivation of, the $\Delta\kappa$-based classification above: a cell where mean agreement rises while mean accuracy falls is, in aggregate, one where increased convergence occurs disproportionately on incorrect an-

swers. Spot-checks confirm the signature: verbatim propagation of hallucinated planner constraints in cascade cases versus independent reasoning departures in non-cascade cases. A wrong-only κ decomposition remains a refinement for future work.

Two coders, each with the full evidence trace (question, planner output, all five executor traces), independently labeled each case **Cascade-Type** (the plan contains an explicit, identifiable flaw that two or more executors verbatim-follow to an identical wrong answer) or **Degradation-Type** (the plan is sound or neutral; executors independently arrive at the same wrong answer through unrelated errors). Pre-consensus agreement was Cohen's $\kappa = 0.667$ "Substantial" (Cohen, 1960; Landis & Koch, 1977). Rater marginals, discordance analysis, and blinded calibration checks ruling out modal-label defaulting are in Appendix M. After discussion, 17 of 25 cases (68%) resolved to Cascade-Type. This case-level rate exceeds the model-level rate (3/7) because High-Agree-Wrong cases are drawn disproportionately from cascade-heavy cells; the rates answer different questions. Case classification also correlates strongly with domain (16/16 arithmetic cases Cascade-Type; 8/9 HotpotQA Degradation-Type). At this sample size we cannot cleanly separate domain structure from model-group structure, so we treat the qualitative pass as corroborating the quantitative $\Delta\kappa$-based split rather than as fully independent evidence for it.

Robustness of κ to Expected Agreement Estimators. Because LLM answer distributions are highly non-uniform—heavily concentrated on a single correct answer or a long tail of distinct numeric errors—the choice of expected agreement (p_e) mechanically scales the absolute value of κ: a pooled cell-marginal p_e can be artificially inflated by a dominant correct answer, compressing κ toward zero, while a uniform p_e assumes maximum entropy. To check that our mechanistic split does not depend on the specific chance-correction term in Eq. 4, we verify (Appendix G) that κ is near-perfectly rank-correlated with uncorrected raw pairwise agreement across all 21 cells (Spearman $r_s = 1.00$), and that the cascade/degradation gap in mean $\Delta\kappa$ on GSM8K ($+0.192$ vs. -0.024) spans an order of magnitude—a gap unlikely to be an artifact of any single estimator choice. A full multi-estimator sensitivity analysis (question-specific and uniform p_e) remains future work. The cascade signature is a structural property of the coordination dynamics, not a statistical artifact of the chance-agreement baseline.

5.5 Robustness, Controls, and Mitigation Attempts

We verify that the observed mechanism split is not an artifact of compute budgets, prompt verbosity, or baseline correlation, and we test naive mitigations. **(1) Compute Budget:** Scaling self-consistency to $k = 8$ to match the token cost of strong coordination improves accuracy in every model, while strong coordination remains significantly below the unmatched $k = 5$ baseline (Appendix H). **(2) Manipulation Checks:** Plan length and named-quantity counts increase monotonically from WEAK to STRONG (mean plan tokens 59.3 $\rightarrow$ 260.9; named quantities 5.0 $\rightarrow$ 19.9, pooled across models), confirming the manipulation is genuine, with localized step-count inversions revealing distinct planner behaviors (Appendix F). **(3) Baseline Correlation:** Self-consistency's own ρ ranges from 0.39–0.73 (Section 3.2 caveat applies); coordination-induced coupling is measured separately via $\Delta\kappa$. **(4) Naive Mitigation:** Resampling the plan ($k = 3$ plans, 15 total outputs) and aggregating via vote-of-votes or pooled voting fails to recover accuracy and can actively harm it, underscoring that structural decoupling, not mere prompt resampling, is required (Appendix I). Full statistical details for all control checks are provided in the Appendices.

6 Discussion

Agreement is not a safe proxy for reliability. Every result in this paper is consistent with the same underlying warning: a coordinated multi-agent system that reports high executor agreement provides no information, on its own, about whether that agreement reflects genuine reliability. This holds whether agreement rises for benign reasons (marginal narrowing, low κ) or dangerous ones (genuine cascade, high κ); both are indistinguishable via plurality agreement alone.

Single-mechanism governance heuristics are insufficient. A monitoring rule that flags "high agreement + verified correlated cascade" would correctly catch the failure mode of the three cascade-signature models (Qwen2.5-1.5B, Qwen2.5-3B, Mistral-7B) but miss the (larger, in several cells) accuracy loss occurring in the other four models via a mechanism our data cannot yet fully charac-

terize. Conversely, reducing coordination for hard tasks fails for Qwen2.5-3B on MATH. Responsible deployment requires per-deployment empirical characterization (measuring ρ/κ on a validation set) rather than universal heuristics.

The contribution is the measurement, not the explanation. This paper does not explain why three models cascade and four do not. We argue this is the correct scope, since before the field can explain cascade susceptibility, it must first reliably detect and distinguish it from other coordination-induced harm. Our ρ/κ framework provides this capability, establishing that coordination-induced correlated failure is real, measurable, and heterogeneous—contradicting the field's default single-mechanism assumption for a meaningful fraction of deployed systems. Explaining why susceptibility clusters in specific models, and validating mitigations against this protocol, is future work.

Concretely, we propose that any multi-agent system evaluation checklist include a coordination-sensitivity audit: compute ρ under self-consistency, compute $\Delta\kappa$ under the intended coordination regime, and classify the failure mechanism before selecting mitigation strategies. This audit requires only 100 held-out questions and the ρ/κ pipeline described in Section 3, making it feasible for any deployment team. As a provisional calibration point rather than a validated decision threshold, the observed group means (cascade $\Delta\kappa \in [+0.102, +0.160]$; degradation $\Delta\kappa \in [-0.059, +0.012]$) suggest a starting split near $\Delta\kappa \approx 0.06$; deployment teams should calibrate this against their own validation set rather than adopt it as a universal cutoff.

7 Limitations

Our closed-book HotpotQA evaluation tests parametric recall rather than retrieval-augmented reasoning; observed accuracies (1.7–28%) are not comparable to open-book baselines. These low base rates yield coarser significance tests and wider ρ/κ confidence intervals. We retain HotpotQA as excluding it does not change mechanism classifications (Appendix G), but caution against over-interpreting individual cells. Mistral-7B/HotpotQA's marginally negative ICC ($\rho = -0.003$) is attributable to small-sample estimator bias (Appendix D). Our qualitative pass covers 25 cases; a larger pass would tighten confidence intervals. Models are served at fixed 4-bit quantization (Appendix A); full-precision comparisons might surface different interactions. Finally, we manipulate coordination via prompt specificity, not learned policies. Additionally, while we empirically measure baseline coupling (ρ), we do not derive the exact theoretical bounds (e.g., via generalized Condorcet frameworks) at which specific ρ levels render majority voting strictly worse than single-sample decoding for LLM-specific error distributions. Establishing these bounds using our measurement protocol is *a* vital direction for future work. Finally, all conditions use identical planner/executor checkpoints (Section 3.3); how heterogeneous (cross-model) coordination modulates cascade susceptibility remains an open dimension.

8 Conclusion

In this controlled study of coordination strength in multi-agent LLM reasoning (21 model-domain combinations), strong explicit coordination reduces accuracy relative to self-consistency in 19 of 21 settings, across statistical tests and not attributable to the extra compute coordination consumed. This is not evidence of a single failure mechanism: a genuine correlated-cascade signature, measurable via excess agreement beyond what each executor's own answer distribution predicts and corroborated by both an independent agreement-direction diagnostic and a structured dual-coded qualitative pass (Cohen's $\kappa = 0.667$), appears in only three of seven models studied, spanning two families and three parameter scales, with the remainder showing comparable or larger harm through an apparently distinct route we could not attribute to any tokenizer, serving, quantization, or attention-architecture difference (Appendix J). A robust, repeatedly verified exception exists in which coordination helps. Together these results argue against treating "coordination causes correlated failure" as a settled, general claim and for adopting a per-deployment, correlation-aware evaluation practice in which the ρ/κ diagnostic is applied before trusting any coordinator-driven pipeline in production.

AI USE STATEMENT

In compliance with ICLR policy, we disclose that the authors used a large language model (Claude, developed by Anthropic) in a limited, assistive capacity during the preparation of this manuscript. Specifically, the tool was used to refine English language clarity and to assist with debugging and refining experimental code. The authors take full responsibility for the accuracy, integrity, and interpretation of all scientific content, including experimental design, technical results, mathematical derivations, and conclusions. No AI tool contributed to the formulation of research questions, the design of experiments, or the analysis of results.

ETHICS STATEMENT

This work evaluates open-weight models on public benchmarks (GSM8K, MATH, HotpotQA). No human subjects were involved, and no private or sensitive data was used. We release our code, prompts, and evaluation logs to support reproducibility.

REPRODUCIBILITY STATEMENT

All experimental code, configuration files, and evaluation scripts are provided as supplementary material in an anonymized repository (`https://github.com/anonymized/anonymized`); the repository will be de-anonymized upon acceptance. Full prompt templates for all four coordination regimes (Weak, Default, Strong, and Self-Consistency) and the executor are given in Appendix A. Exact model tags, quantization schemes, and base checkpoints for all seven models are listed in Appendix A, Table 2. The statistical procedure—cluster-robust bootstrap confidence intervals, question-level permutation tests, cluster McNemar tests, and Benjamini–Hochberg FDR correction—is fully specified in Section 5. The ICC moment estimator used to compute ρ is given in Appendix D. Evaluation data consist of 100 questions per domain (GSM8K, a Hendrycks MATH algebra/number-theory subset, and closed-book HotpotQA), fixed and identical across all conditions and models, replicated across three seeds for a total of 300 observations per (model, domain, condition) cell. Two extraction-pipeline defects identified and corrected during the study are documented in Appendix E, including the scope audit conducted before redeployment. Hardware specifications and total compute budget are reported in Appendix A.

REFERENCES

Yoav Benjamini and Yosef Hochberg. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society: Series B*, 57(1):289–300, 1995.

Mert Cemri, Melissa Z Pan, Shuyi Yang, Lakshya A Agrawal, Bhavya Chopra, Rishabh Tiwari, Kurt Keutzer, Aditya Parameswaran, Dan Klein, Kannan Ramchandran, et al. Why do multi-agent llm systems fail? *Advances in Neural Information Processing Systems*, 38, 2026.

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.

Jacob Cohen. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46, 1960.

Yilun Du, Shuang Li, Antonio Torralba, Joshua B. Tenenbaum, and Igor Mordatch. Improving factuality and reasoning in language models through multiagent debate. *arXiv preprint arXiv:2305.14325*, 2023.

Joseph L. Fleiss. Measuring nominal scale agreement among many raters. *Psychological Bulletin*, 76(5):378–382, 1971.

Ariel Flint, Luca Maria Aiello, Romualdo Pastor-Satorras, and Andrea Baronchelli. Group size effects and collective misalignment in llm multi-agent systems. *Proceedings of the National Academy of Sciences*, 123(34):e2531697123, 2026.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*, 2021.

Sirui Hong, Mingchen Zhuge, Jonathan Chen, Xiawu Zheng, Yuheng Cheng, Jinlin Wang, Ceyao Zhang, Steven Yau, Zijuan Lin, Liyang Zhou, et al. Metagpt: Meta programming for a multi-agent collaborative framework. In *International Conference on Learning Representations*, volume 2024, pp. 23247–23275, 2024.

Lars Benedikt Kaesberg, Jonas Becker, Jan Philip Wahle, Terry Ruas, and Bela Gipp. Voting or consensus? decision-making in multi-agent debate. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 11640–11671, 2025.

Elliot Kim, Avi Garg, Kenny Peng, and Nikhil Garg. Correlated errors in large language models. In *International Conference on Machine Learning (ICML)*, 2025.

Anders Krogh and Jesper Vedelsby. Neural network ensembles, cross validation, and active learning. In *Advances in Neural Information Processing Systems 7*, pp. 231–238, 1995.

Ludmila I. Kuncheva and Christopher J. Whitaker. Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. *Machine Learning*, 51(2):181–207, 2003.

J. Richard Landis and Gary G. Koch. The measurement of observer agreement for categorical data. *Biometrics*, 33(1):159–174, mar 1977. doi: 10.2307/2529310.

Guohao Li, Hasan Abed Al Kader Hammoud, Hani Itani, Dmitrii Khizbullin, and Bernard Ghanem. Camel: Communicative agents for "mind" exploration of large language model society. In *Advances in Neural Information Processing Systems*, 2023.

Tian Liang, Zhiwei He, Wenxiang Jiao, Xing Wang, Yan Wang, Rui Wang, Yujiu Yang, Shuming Shi, and Zhaopeng Tu. Encouraging divergent thinking in large language models through multi-agent debate. In *Proceedings of the 2024 conference on empirical methods in natural language processing*, pp. 17889–17904, 2024.

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 *International Conference on Learning Representations*, volume 2024, pp. 39578–39601, 2024.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. Self-refine: Iterative refinement with self-feedback. In *Advances in Neural Information Processing Systems*, 2023.

Yongliang Shen, Kaitao Song, Xu Tan, Dongsheng Li, Weiming Lu, and Yueting Zhuang. Hugginggpt: Solving ai tasks with chatgpt and its friends in hugging face. In *Advances in Neural Information Processing Systems*, 2023.

Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling llm test-time compute optimally is more effective than scaling model parameters for reasoning. *arXiv preprint arXiv:2408.03314*, 2024.

Selim Furkan Tekin, Fatih Ilhan, Tiansheng Huang, Sihao Hu, and Ling Liu. LLM-TOPLA: Efficient LLM ensemble by maximising diversity. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 11951–11966, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-emnlp.698. URL https://aclanthology.org/2024.findings-emnlp.698/.

Selim Furkan Tekin, Fatih Ilhan, Gaowen Liu, Ramana Rao Kompella, and Ling Liu. Dynamic optimizations of llm ensembles with two-stage reinforcement learning agents, 2025. URL https://arxiv.org/abs/2502.04492.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. In *International Conference on Learning Representations*, 2023.

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 *Advances in Neural Information Processing Systems*, 2022.

Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Beibin Li, Erkang Zhu, Li Jiang, Xiaoyun Zhang, Shaokun Zhang, Jiale Liu, Ahmed Awadallah, Ryen W White, Doug Burger, and Chi Wang. Autogen: Enabling next-gen llm applications via multi-agent conversation. *arXiv preprint arXiv:2308.08155*, 2023.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii (eds.), *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 2369–2380, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1259. URL https://aclanthology.org/D18-1259/.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. In *Advances in Neural Information Processing Systems*, 2023.

When Does Coordination Induce Correlated Failure in Multi-Agent LLM Reasoning?

Supplementary Material

Table of Contents in Appendix

A Prompts and Reproducibility 14
B Case Study Evidence: Cross-Domain Cascade Failures 16
C Formal Derivation: From Variance to Agreement 17
D ICC Moment Estimator 18
E Measurement Pipeline Audits 18
F Plan Specificity Inversions: Per-Model Diagnosis 18
G Sensitivity of Excess Agreement (κ) to Expected-Agreement Estimators 19
H Ruling Out a Compute-Budget Confound 19
I Naive Mitigation: Plan Resampling Does Not Recover Accuracy 20
J Observational Architectural Probes 20
K Gemma 3 Pilot Evaluation 21
L Full Per-Cell Statistical Matrix 21
M Qualitative Coding Ledger 22

A PROMPTS AND REPRODUCIBILITY

Prompt Templates.

- **Weak Planner:**

```
Planner mode:  WEAK.
- Keep the plan short (2-3 steps).
- Use broad guidance only.
- Avoid decomposition details.

You are a planning agent.
Your task is to generate a high-level reasoning strategy
for solving the problem.

IMPORTANT RULES:
- Do NOT solve the problem.
- Do NOT perform calculations.
- Do NOT provide the final answer.
- Only provide an abstract sequence of reasoning steps.
- Keep the reasoning plan concise.
- The plan should guide reasoning, not replace it.

Problem:
{question}

Generate a structured reasoning plan.
```

- **Default Planner:**

```
Planner mode:  DEFAULT.
- Provide a concise but structured plan.

You are a planning agent.
Your task is to generate a high-level reasoning strategy
for solving the problem.

IMPORTANT RULES:
- Do NOT solve the problem.
- Do NOT perform calculations.
- Do NOT provide the final answer.
- Only provide an abstract sequence of reasoning steps.
- Keep the reasoning plan concise.
- The plan should guide reasoning, not replace it.

Problem:
{question}

Generate a structured reasoning plan.
```

- **Strong Planner:**

```
Planner mode:  STRONG.
- Provide a clear, structured step-by-step plan.
- Include decomposition and key checks.
- Still do not solve or compute the final answer.

You are a planning agent.
Your task is to generate a high-level reasoning strategy
for solving the problem.

IMPORTANT RULES:
- Do NOT solve the problem.
- Do NOT perform calculations.
- Do NOT provide the final answer.
- Only provide an abstract sequence of reasoning steps.
- Keep the reasoning plan concise.
- The plan should guide reasoning, not replace it.

Problem:
{question}

Generate a structured reasoning plan.
```

- **Executor:**

```
You are an independent reasoning agent.

You are given:
1.  A problem
2.  A shared high-level reasoning strategy

Your task is to independently solve the problem while using
the strategy only as loose guidance.

IMPORTANT:
- Do NOT copy the strategy directly.
- You may reason differently from other agents.
- Use your own reasoning style.
- The strategy is only a coordination signal.

HOWEVER:
You MUST finish with EXACTLY this format:

FINAL_ANSWER: <answer>

Problem:
{question}

Shared Strategy:
{plan}

Solve the problem independently.
```

- **Self-Consistency Baseline (no plan):**

```
You are an independent reasoning agent.
Solve the following problem step by step.

Question:
{question}

Instructions:
1. Execute precise step-by-step calculations.
2. Conclude your final response with: FINAL_ANSWER: <answer>

Solution:
```

The `Shared Strategy: {plan}` block and all strategy references ("2. A shared high-level reasoning strategy") are removed entirely by code branching in `cdde/prompts/executor_prompt.py` — not left as an empty string — so the Self-Consistency stimulus contains no coordination signal of any kind.

Decoding hyperparameters. Planner calls use temperature 0.2, top_p 1.0, and `num_predict` 512 in all coordinated regimes (WEAK, DEFAULT, STRONG); the Self-Consistency baseline issues no planner call. Executors follow a deterministic temperature mapping by index within each seed: executor $0 \to T = 0.6$, $1 \to 0.7$, $2 \to 0.8$, $3 \to 0.9$, $4 \to 1.0$, preserving stochastic diversity while holding the strategic signal fixed. All cells use fixed seeds 42, 43, 44.

Reproducibility Details. All experiments were conducted locally on a dedicated workstation equipped with an Intel Core i5-13420H CPU (8 cores), 16 GB RAM, and an NVIDIA GeForce RTX 2080 Ti GPU with 11 GB VRAM. Models were served locally via Ollama (v0.5+ API endpoint). The primary benchmark evaluations—spanning 25,200 runs across the 7 core baseline models (`gemma2-2b`, `gemma2-9b`, `llama3.1-8b`, `mistral-7b`, `qwen2.5-1.5b`, `qwen2.5-3b`, and `qwen2.5-7b`) on 300-question evaluation sets across 3 domains (GSM8K, MATH, and HOTPOTQA) under 4 planner coordination regimes—consumed approximately 32.40M tokens and 118.5 GPU-hours. Including standalone Gemma 3 benchmarks (7,200 runs), compute-matched self-consistency baselines, multi-plan ensemble mitigation sweeps ($K = 3$), qualitative failure taxonomy coding, and preliminary pilot runs, the total compute consumed during the project was approximately 38.65M tokens and 142.0 GPU-hours.

Quantization Normalization and Limitations. To prevent cross-family comparisons from being confounded by disparate precision regimes (e.g., FP16 vs. 4-bit integer weights), all 7 core

baseline models were evaluated strictly within the 4-bit quantization regime using official Ollama model tags. Specifically, 5 of the 7 models (`llama3.1:8b`, `mistral:7b`, `qwen2.5:1.5b`, `qwen2.5:3b`, and `qwen2.5:7b`) utilize 4-bit medium K-quants (`Q4_K_M`), which incorporate mixed 6-bit scales for critical attention and MLP projection layers. The remaining 2 models (`gemma2:2b` and `gemma2:9b`) utilize standard 4-bit block quantization (`Q4_0`), as mandated by Gemma 2's default distribution format in Ollama. While both schemes strictly maintain a 4-bit footprint, the minor structural difference between `Q4_K_M` and `Q4_0` represents a minor serving implementation detail inherent to upstream model distributions.

Table 2: **Model Manifest and Quantization Metadata across Core Baseline Models.** All 7 core models evaluate under 4-bit quantization schemes via local Ollama serving.

Model Tag	Family	Parameters	Quantization	Size (GB)	Base Checkpoint
`gemma2:2b`	Gemma 2	2.6B	`Q4_0`	1.6	`google/gemma-2-2b-it`
`gemma2:9b`	Gemma 2	9.2B	`Q4_0`	5.4	`google/gemma-2-9b-it`
`llama3.1:8b`	LLaMA 3.1	8.0B	`Q4_K_M`	4.9	`meta-llama/Meta-Llama-3.1-8B-Instruct`
`mistral:7b`	Mistral	7.2B	`Q4_K_M`	4.4	`mistralai/Mistral-7B-Instruct-v0.2`
`qwen2.5:1.5b`	Qwen 2.5	1.5B	`Q4_K_M`	1.0	`Qwen/Qwen2.5-1.5B-Instruct`
`qwen2.5:3b`	Qwen 2.5	3.1B	`Q4_K_M`	1.9	`Qwen/Qwen2.5-3B-Instruct`
`qwen2.5:7b`	Qwen 2.5	7.6B	`Q4_K_M`	4.7	`Qwen/Qwen2.5-7B-Instruct`

An anonymized code repository containing the implementation, configuration files, prompts, and scripts required to reproduce the experiments is provided for review at `https://github.com/anonymized/anonymized`. The repository is anonymized to preserve double-blind anonymity; the public, de-anonymized repository will be released upon acceptance.

B Case Study Evidence: Cross-Domain Strategic Cascade Failures

To provide concrete empirical evidence for the *Strategic Cascade Failure* phenotype (High-Agree, Wrong), we present complete execution traces across all three evaluation domains: **GSM8K** (Algebraic Math), **MATH** (Multi-Step Symbolic Mathematics), and **HotpotQA** (Multi-Hop Factual Reasoning). We note that the Strong planner does not always comply with its "no computation" constraint: a systematic audit of all 6,300 Strong planner outputs flags 2,325 (36.9%) containing explicit arithmetic or a stated final value, with rates highest on GSM8K/MATH and lowest on HotpotQA. Some cascade cases may therefore reflect executors copying leaked computations rather than converging under an abstract decomposition; the $\Delta\kappa$-based classification is agnostic to this distinction, but individual trace interpretation is not. Violation rates do not cleanly separate mechanism groups (cascade-group models are descriptively higher on GSM8K/MATH, but Llama3.1-8B and Qwen2.5-7B overlap them at 0.45–0.55), so leakage frequency alone does not account for group membership.

B.1 Algebraic Reasoning (GSM8K): `gsm8k_test_48`

Question: Rose bought five dozens of eggs for $2.40 a dozen. She will sell it for $1 for 3 eggs. How much will be Rose's profit?

Ground Truth: `8` (Total cost = 5 dozens $\times$ \$2.40/dozen = \$12.00. Total eggs = $5 \times 12 = 60$. Revenue from selling 60 eggs at \$1 per 3 eggs = $(60/3) \times \$1.00 = \$20.00$. Net profit = $\$20.00 - \$12.00 = \$8.00$).

Strong Planner Output (The Error Source):

> **Step 1:** Calculate total eggs Rose bought: 5 dozens $\times$ 12 = 60 eggs.
> **Step 2:** Calculate total cost of eggs: Total cost = 60 eggs $\times$ \$2.40/dozen = \$144.00
> *[Flawed Step: Multiplied total individual eggs by per-dozen cost instead of dozens].*
> **Step 3:** Calculate number of eggs sold: $\$1.00/3$ eggs = 0.33 eggs.
> **Step 4:** Calculate total revenue: 0.33 eggs $\times$ \$1.00/egg = \$0.33.
> **Step 5:** Calculate Rose's profit: Profit = Total Revenue − Total Cost = $\$0.33 - \$144.00 = -\$143.67$.

Executor Behavior: All five executors independently followed the planner's explicit, but logically flawed, Step 2 ($60 \times 2.40 = 144$). Consequently, all five extracted the exact answer `-143.67`, resulting in a unanimous agreement score of $A = 1.0$. Under unguided self-consistency, executors independently recognized that \$2.40 applies per dozen ($5 \times 2.40 = 12$), extracting the correct ground truth answer 8 ($A = 0.93$, i.e. 14 of 15 outputs pooled across all three seeds).

B.2 Symbolic Mathematics (MATH): `MATH_TEST_9`

Question: Define $f(x) = (x-1)(x-3)(x-7)(x-9)$. Evaluate $f(6) - f(4)$.

Ground Truth: `0` ($f(6) = (5)(3)(-1)(-3) = 45$; $f(4) = (3)(1)(-3)(-5) = 45$; $f(6) - f(4) = 45 - 45 = 0$).

Strong Planner Output (The Error Source):

> **Phase 1:** Define function evaluation rules.
> **Phase 2:** Calculate $f(6) = (6-1)(6-3)(6-7)(6-9) = (5)(3)(-1)(-3) = 45$.
> **Phase 3:** Calculate $f(4) = (4-1)(4-3)(4-7)(4-9) = (3)(1)(-3)(-5) = -45$
> *[Flawed Step: Sign flipping hallucination in product of two negative terms].*
> **Phase 4:** Compute difference: $f(6) - f(4) = 45 - (-45) = 90$.

Executor Behavior: All five decentralized executors strictly adhered to the planner's sign assertion in Phase 3 ($f(4) = -45$), yielding a unanimous incorrect answer of `90` ($A = 1.0$). Under unguided self-consistency, executors correctly computed $(3)(1)(-3)(-5) = +45$, resulting in $45 - 45 = 0$.

B.3 Multi-Hop Reasoning (HotpotQA): `HOTPOTQA_VAL_12`

Question: What is the population of the settlement that is next smaller than the city with a population of 67,038 on the Argentine side of Tierra del Fuego?

Ground Truth: `1,382` (Tolhuin Municipality, Tierra del Fuego, Argentina).

Strong Planner Output (The Error Source):

> **Phase 1 (Initial Data Gathering):** Establish baseline population assumption. For this planning exercise, set target settlement population threshold to 600 *[Flawed Step: Arbitrary constraint hallucination ungrounded in facts]*.
> **Phase 2 (Refinement):** Calculate next smaller settlement under the 600 threshold.
> **Phase 3 (Final Output):** Output final population value of 600.

Executor Behavior: Four out of five executors independently constrained their search query to the planner's explicit population cap of `600`, resulting in a high-agreement wrong consensus answer of `600` ($A = 0.8$). Under unguided self-consistency, executors executed unconstrained demographic lookup, successfully retrieving the correct population `1,382`.

C Formal Derivation: From Variance to Agreement

To formally connect the variance decomposition (Eq. 2) to our agreement metric, let $C_i \in \{0, 1\}$ denote executor correctness and write $\mu_\pi := \mathbb{E}[C_i \mid x, \pi]$ for the plan-conditional success probability. Assuming executors are conditionally independent given π, all k executors in CDDE share the same μ_π. A sound plan ($\mu_\pi > 1/2$) yields high agreement on the correct answer; a flawed plan yields high agreement on a wrong answer *only when the plan-conditional error distribution is low-entropy*—executors converging on the same incorrect value (the Strategic Cascade). If errors are

diffuse, pairwise agreement remains near chance and $\Delta\kappa \approx 0$ (Prompt-Induced Degradation). Self-consistency instead draws each trajectory from the mixture $\sum_\pi P(\pi \mid x)\, P(y \mid x, \pi)$, so pairwise disagreement reflects both strategic and execution variance. Sharing one plan removes the strategic component from disagreement, proving that agreement tracks the concentration of the executor distribution rather than its correctness.

D ICC Moment Estimator

The population quantity ρ (Eq. 3) is estimated from a one-way ANOVA on executor correctness C_i, with question as the grouping factor:

$$\hat{\rho} = \frac{\mathrm{MSB} - \mathrm{MSW}}{\mathrm{MSB} + (n-1)\,\mathrm{MSW}}, \tag{5}$$

where MSB and MSW are the between- and within-question mean squares and n is the number of observations per question ($k = 5$ executors $\times$ 3 seeds pooled $= 15$). This is the standard ICC(1) moment estimator; it is biased for small n and can occasionally return values slightly outside $[0, 1]$ at small sample sizes, which is the likely source of Mistral-7B/HotpotQA/strong's marginally negative estimate ($\hat{\rho} = -0.003$, Section 7) rather than evidence of genuine negative correlation. We report this estimator here rather than in the main text because it is a computational recipe, not a claim: removing it costs the paper reproducibility detail, not scientific content, whereas removing Eq. 3 itself would remove the falsifiable definition of the conditional-independence assumption's failure condition. While alternative formulations (e.g., generalized linear mixed models or Bayesian hierarchical ICC) could yield slightly different point estimates, we leave their evaluation to future work; such formulations are also the natural place to model the nested observation structure explicitly (executors nested within seed-specific plan realizations, nested within questions), which the current flat one-way ANOVA treats as exchangeable.

E Measurement Pipeline Audits

Two extraction-pipeline defects were identified and corrected during this study: (i) a dataset-loader fallback that, when the intended data source was briefly unavailable, silently substituted a single repeated placeholder question for an entire domain-model cell rather than failing loudly, and (ii) an answer-normalization routine that could not correctly resolve nested LaTeX fractions (e.g. $\frac{2}{\frac{7}{10}}$), causing a small number of genuinely distinct numeric answers to be misextracted as identical to other executors' answers, which — because it manufactures spurious agreement rather than spurious disagreement — posed a direct threat to the κ-based cascade measurements central to this paper's mechanism analysis (Section 5.4). Both were identified via systematic auditing (dataset row-count and content checks; case-level extraction re-verification against raw model output) rather than incidentally, and all affected cells were regenerated from raw logs after each fix; the second fix (a recursive, brace-depth-aware LaTeX parser rather than a single-pass regular expression) was scope-audited against the full raw log corpus (653 of 49,964 records contained nested fraction expressions) before being deployed. We recommend equivalent extraction audits as a precondition for any future work using agreement- or correlation-based failure metrics on free-form LLM output.

F Plan Specificity Inversions: Per-Model Diagnosis

Section 5.5 in the main text confirms that mean plan length and named-quantity count increase monotonically from Weak to Default to Strong for all seven models, establishing that the three-level coordination manipulation is genuinely differentiated. This appendix documents two cells in which a different plan statistic—step count—is *not* monotonic, and in which this non-monotonicity coincides with an agreement inversion (Default agreement exceeding Strong). The two inversions turn out to be mechanistically distinct, which is why we report them separately rather than force a single account.

Inversion 1: Llama-3.1-8B / GSM8K. Under Strong coordination, Llama-3.1-8B's mean plan step count collapses from 6.44 (Default) to 3.24, while its named-quantity count surges from 9.28

to 15.73. The planner is compressing multi-step arithmetic decompositions into fewer, denser paragraphs that pack more named quantities into each step. Executors lose the modular step boundaries that DEFAULT plans provide, and agreement rises without a corresponding accuracy improvement—consistent with the plan locking executors into a shared but over-compressed reasoning structure.

Inversion 2: Qwen2.5-7B / HotpotQA. Under STRONG coordination, Qwen2.5-7B's plan token count surges from 124.2 (DEFAULT) to 341.7 and its step count from 4.58 to 8.88. Here the mechanism is the opposite of Llama's: the planner *over-specifies*, introducing more sub-goals and more factual anchors than the task requires. Each additional named entity or constraint is an additional opportunity for a factual error to enter the plan and then lock into the shared context that all five executors condition on. Agreement rises because executors are all conditioning on the same (partially flawed) expanded plan, but accuracy does not benefit because the extra specification adds noise rather than structure.

Implication. The two inversions share a surface signature (agreement under DEFAULT $\geq$ agreement under STRONG) but arise from opposite planner behaviors—compression versus over-specification. We report them as two distinct, per-domain-specific mechanisms. Neither inversion contradicts the monotonicity of plan length or named-quantity count; both are localized to the step-count dimension and are visible only in the per-model, per-domain breakdown.

G SENSITIVITY OF EXCESS AGREEMENT (κ) TO EXPECTED AGREEMENT ESTIMATORS

The excess agreement statistic $\kappa = (p_a - p_e)/(1 - p_e)$ relies on a baseline expected agreement p_e under the null hypothesis of chance. In standard human-annotation settings, p_e is derived from relatively balanced categorical marginals. In our LLM reasoning setting, however, answer distributions are heavily zero-inflated (skewed toward the correct answer) or exhibit a long tail of distinct numeric errors. This creates a known sensitivity: if a single correct answer dominates the pooled distribution (e.g., frequency $f_c = 0.8$), the pooled $p_e \geq 0.64$, mathematically capping κ and compressing it toward zero even when executors exhibit perfect correlated failure.

Results. As a robustness check, we verify that our mechanistic split does not depend on the specific chance-correction term in Eq. 4. First, κ is near-perfectly rank-correlated with uncorrected raw pairwise agreement across all 21 cells (Spearman $r_s = 1.00$), indicating the chance-correction step itself is not driving our conclusions. Second, the cascade/degradation split is large relative to its own measurement: on GSM8K, the Cascade group's mean $\Delta\kappa$ ($+0.192$) exceeds the Degradation group's (-0.024) by an order of magnitude, a gap unlikely to be an artifact of any single estimator choice. We leave a full multi-estimator sensitivity analysis (question-specific and uniform p_e) to future work.

Excluding HotpotQA. Recomputing mean $\Delta\kappa$ per model using only GSM8K and MATH (the two domains with adequate base rates) preserves the cascade/degradation split exactly: the three cascade-signature models retain mean $\Delta\kappa > +0.10$, while all four degradation-signature models remain below $+0.02$, preserving a clear separation between groups. No model changes mechanism group.

H RULING OUT A COMPUTE-BUDGET CONFOUND

Strong coordination's extra planner call means it consumes more inference compute per question than $k = 5$ self-consistency (typically $\sim$1,550–1,850 vs. $\sim$1,250 generation tokens, though this varies by model). To rule out "more tokens, not worse coordination" as an alternative explanation, we compare Strong against self-consistency at $k = 8$ (matching or exceeding Strong's token budget in 6 of 7 models; Gemma2-2B is the exception and is reported explicitly below) on GSM8K across all 7 models.

Scaling self-consistency's budget this way improves accuracy in every model (+0.4 to +0.9 points), while Strong remains far below even the unmatched $k = 5$ baseline in every model (e.g. Llama3.1-

8B: $\text{SC}_{k=8} = 0.891$ →Strong= 0.647; Mistral-7B: $\text{SC}_{k=8} = 0.620$ →Strong= 0.373), FDR-corrected significant in 6 of 7 models.

The direction of the token-budget imbalance varies across models: for Qwen2.5-1.5B, Strong used *fewer* tokens than its matched baseline (1,560 vs. 2,347) and still underperformed; for Gemma2-2B, Strong used *more* tokens (2,313 vs. 1,604) and also underperformed. The effect holds in both directions of the token-budget comparison, confirming that the accuracy loss is attributable to the coordination mechanism itself, not raw inference compute.

I Naive Mitigation: Plan Resampling Does Not Recover Accuracy

We tested whether resampling the plan itself (k=3 independent plans, 5 executors each, 15 total outputs) recovers accuracy lost to coordination, on two cascade-mechanism cells (Qwen2.5-1.5B, Qwen2.5-3B; GSM8K) and two degradation-mechanism cells (Llama3.1-8B; GSM8K, MATH).

Two aggregation rules are compared against the single-plan baseline: vote-of-votes (majority within each plan, then majority across plans) and pooled voting (all 15 executor outputs treated as one flat vote). Vote-of-votes shows no statistically significant change in any cell (McNemar $p \in [0.32, 0.89]$). Pooled voting is worse than the single-plan baseline in all four cells, significantly so in two (Qwen2.5-1.5B/GSM8K, $p = 0.020$; Llama3.1-8B/GSM8K, $p = 0.002$).

The actionable conclusion: naively pooling votes across independently resampled plans is not a safe default and can measurably worsen outcomes. We do not detect a differential response between cascade and degradation cells at this sample size and do not claim to have ruled one out. This negative result underscores that structural decoupling, not mere prompt resampling, is required to fix coordination-induced failure.

Table 3: Point estimates for the plan-resampling mitigation ($k = 3$ plans, 15 total outputs). Accuracies over 300 observations per cell. VoV = vote-of-votes (majority within plan, then across plans); Pooled = flat majority over all 15 executor outputs.

Cell	SC	Single-plan (Strong)	VoV	Pooled
Qwen2.5-1.5B / GSM8K	0.707	0.557	0.540	0.440
Qwen2.5-3B / GSM8K	0.850	0.797	0.820	0.730
Llama3.1-8B / GSM8K	0.883	0.647	0.660	0.450
Llama3.1-8B / MATH	0.617	0.393	0.390	0.290

J Observational Architectural Probes

We tested four candidate architectural/serving explanations for cascade-group membership (Qwen2.5-1.5B, Qwen2.5-3B, Mistral-7B) against the four degradation-group models, using metadata already reported for all seven models (Table 2) plus direct Ollama API queries.

Serving and sampling parameters. All seven models were queried via Ollama's `/api/show` endpoint; all seven use identical decoding parameters within CDDE. Uniform across both groups; cannot explain the split.

Quantization. Five of seven models (Llama3.1-8B, Mistral-7B, all three Qwen2.5 variants) use Q4_K_M; the two Gemma2 models use Q4_0 (Table 2). This does not align with cascade membership: Llama3.1-8B and Qwen2.5-7B share Q4_K_M with the cascade group yet are both degradation-group.

Tokenizer and vocabulary. Qwen2.5-7B uses the identical tokenizer, vocabulary, and chat format as the two cascade-group Qwen2.5 models, yet shows the lowest cascade rate in the study — a direct within-family disproof.

Attention architecture. Mistral-7B-Instruct-v0.2 uses full attention with grouped-query attention (GQA), not sliding-window attention (a v0.1 feature Mistral AI removed for v0.2). It shares this

attention paradigm with Llama3.1-8B and all three Qwen2.5 models. Gemma2 alone uses an alternating local/global (sliding-window) pattern. Attention architecture therefore does not cluster the cascade group at all — Mistral-7B, a cascade-group model, uses the identical attention family as two degradation-group models (Llama3.1-8B, Qwen2.5-7B).

K GEMMA 3 PILOT EVALUATION

We additionally piloted two Gemma 3 models (1B, 4B) across the full 3-domain, 4-condition matrix after the core validation pipeline (extraction fixes, cluster-robust inference, FDR correction) was already established on the seven-model set. Results were directionally consistent with the main findings (accuracy reduced under strong coordination in all 6 model-domain cells tested; binomial sign test $p = 0.0312$) and did not surface a new mechanism or revise any conclusion in Section 5.4. Given no new insight, we did not extend the full validated pipeline to these two models and report the pilot here for transparency.

L FULL PER-CELL STATISTICAL MATRIX

Table 4: Full per-cell statistics underlying Table 1 and Section 5. Δ = Strong minus Self-Consistency. CIs are cluster-bootstrap 95% percentile intervals (10,000 resamples, question-id cluster). A = plurality agreement over $k = 5$ executors (mean over questions and seeds). Perm/McN = cluster permutation and cluster McNemar tests; q = Benjamini–Hochberg FDR adjusted across all 21 cells. Displayed ΔAcc equals the difference of the displayed accuracies.

Model	Domain	Acc(SC)	Acc(St)	ΔAcc	ΔAcc 95% CI	A_{SC}	A_{St}	Perm p	Perm q	McN p	McN q
gemma2-2b	gsm8k	0.663	0.450	-0.213	$[-0.270, -0.157]$	0.703	0.725	0.0000	0.0000	0.0000	0.0000
gemma2-2b	math	0.390	0.303	-0.087	$[-0.137, -0.037]$	0.512	0.507	0.0014	0.0029	0.0041	0.0107
gemma2-2b	hotpotqa	0.067	0.023	-0.043	$[-0.070, -0.017]$	0.276	0.272	0.0005	0.0012	1.0000	1.0000
gemma2-9b	gsm8k	0.880	0.830	-0.050	$[-0.087, -0.013]$	0.935	0.951	0.0063	0.0095	0.2891	0.4047
gemma2-9b	math	0.650	0.613	-0.037	$[-0.080, +0.007]$	0.761	0.741	0.0818	0.0904	0.2863	0.4047
gemma2-9b	hotpotqa	0.127	0.137	$+0.010$	$[-0.020, +0.040]$	0.367	0.422	0.3861	0.4054	0.0625	0.1094
llama3.1-8b	gsm8k	0.883	0.647	-0.236	$[-0.290, -0.183]$	0.817	0.755	0.0000	0.0000	0.0000	0.0000
llama3.1-8b	math	0.617	0.393	-0.224	$[-0.283, -0.163]$	0.596	0.583	0.0000	0.0000	0.0000	0.0000
llama3.1-8b	hotpotqa	0.283	0.207	-0.076	$[-0.117, -0.037]$	0.455	0.405	0.0001	0.0003	0.0002	0.0008
mistral-7b	gsm8k	0.613	0.373	-0.240	$[-0.303, -0.177]$	0.578	0.703	0.0000	0.0000	0.0000	0.0001
mistral-7b	math	0.197	0.123	-0.074	$[-0.127, -0.023]$	0.377	0.567	0.0044	0.0071	0.0386	0.0810
mistral-7b	hotpotqa	0.050	0.023	-0.027	$[-0.050, -0.007]$	0.386	0.415	0.0412	0.0509	1.0000	1.0000
qwen2.5-1.5b	gsm8k	0.707	0.557	-0.150	$[-0.203, -0.097]$	0.617	0.786	0.0000	0.0000	0.0000	0.0002
qwen2.5-1.5b	math	0.517	0.447	-0.070	$[-0.123, -0.017]$	0.507	0.616	0.0141	0.0185	0.0872	0.1525
qwen2.5-1.5b	hotpotqa	0.057	0.017	-0.040	$[-0.067, -0.017]$	0.307	0.406	0.0031	0.0059	0.5000	0.6562
qwen2.5-3b	gsm8k	0.850	0.797	-0.053	$[-0.100, -0.003]$	0.798	0.893	0.0526	0.0614	0.7539	0.8796
qwen2.5-3b	math	0.530	0.610	$+0.080$	$[+0.027, +0.133]$	0.581	0.711	0.0036	0.0063	0.0161	0.0376
qwen2.5-3b	hotpotqa	0.170	0.060	-0.110	$[-0.150, -0.070]$	0.373	0.375	0.0000	0.0000	0.1250	0.2019
qwen2.5-7b	gsm8k	0.910	0.820	-0.090	$[-0.130, -0.050]$	0.906	0.871	0.0000	0.0000	0.0019	0.0058
qwen2.5-7b	math	0.630	0.603	-0.027	$[-0.083, +0.030]$	0.674	0.673	0.4266	0.4266	0.7359	0.8796
qwen2.5-7b	hotpotqa	0.103	0.057	-0.046	$[-0.080, -0.013]$	0.401	0.373	0.0111	0.0146	1.0000	1.0000

M QUALITATIVE CODING LEDGER

Table 5: All 25 dual-coded High-Agree-Wrong cases underlying Section 5.4's taxonomy analysis (68% Cascade-Type). * = discordant pre-consensus, resolved by discussion.

#	Model	Domain	Cond.	Coder A	Coder B	Resolved
1	Gemma2-2B	GSM8K	strong	Cascade	Cascade	Cascade
2	Gemma2-2B	GSM8K	strong	Cascade	Cascade	Cascade
3	Gemma2-2B	HotpotQA	weak	Degrad.	Degrad.	Degrad.
4	Gemma2-2B	MATH	default	Cascade	Degrad.	Cascade*
5	Gemma2-9B	GSM8K	weak	Cascade	Cascade	Cascade
6	Gemma2-9B	HotpotQA	SC	Degrad.	Degrad.	Degrad.
7	Gemma2-9B	MATH	default	Cascade	Cascade	Cascade
8	Llama3.1-8B	GSM8K	weak	Cascade	Cascade	Cascade
9	Llama3.1-8B	HotpotQA	default	Degrad.	Degrad.	Degrad.
10	Llama3.1-8B	MATH	strong	Degrad.	Cascade	Cascade*
11	Mistral-7B	GSM8K	weak	Cascade	Cascade	Cascade
12	Mistral-7B	HotpotQA	strong	Degrad.	Degrad.	Degrad.
13	Mistral-7B	HotpotQA	default	Degrad.	Degrad.	Degrad.
14	Mistral-7B	HotpotQA	strong	Degrad.	Degrad.	Degrad.
15	Mistral-7B	MATH	default	Cascade	Degrad.	Cascade*
16	Mistral-7B	MATH	default	Cascade	Cascade	Cascade
17	Qwen2.5-1.5B	GSM8K	default	Cascade	Cascade	Cascade
18	Qwen2.5-1.5B	HotpotQA	default	Degrad.	Degrad.	Degrad.
19	Qwen2.5-1.5B	MATH	weak	Cascade	Cascade	Cascade
20	Qwen2.5-3B	GSM8K	strong	Cascade	Cascade	Cascade
21	Qwen2.5-3B	HotpotQA	default	Degrad.	Cascade	Cascade*
22	Qwen2.5-3B	MATH	weak	Cascade	Cascade	Cascade
23	Qwen2.5-7B	GSM8K	default	Cascade	Cascade	Cascade
24	Qwen2.5-7B	HotpotQA	weak	Degrad.	Degrad.	Degrad.
25	Qwen2.5-7B	MATH	default	Cascade	Cascade	Cascade

Sampling Provenance and Case 6. The sampler drew 25 cases by deterministic linear indexing over sorted High-Agree-Wrong cases without a condition filter; consequently, one row (Case 6, Gemma2-9B/HotpotQA) is a self-consistency case with no plan. With no upstream plan present, the Cascade/Degradation distinction is degenerate, and both coders independently assigned Degradation-Type. We retain it as a no-plan control confirming the scheme does not manufacture plan flaws where none exist; excluding it leaves the Cascade-Type rate at 17/24 (71%) and changes no conclusion.

Coder reliability details. Both coders' marginals were balanced and symmetric (60%/40% Cascade/Degradation each); the four discordant cases showed no directional bias (McNemar exact $p = 1.0$); and both coders classified 5/5 blinded calibration cases with known ground truth correctly, ruling out modal-label defaulting.