

DIANOIA: Diagnostic Decomposition and Joint Optimization for Multi-Agent Reasoning

Yiming Yang* and Zhuoyuan Li and Fanxiang Zeng and Hao Fu and Yue Liu

AMap, Alibaba Group, Beijing, China

{sachiel.ymm, weiyuan.lzy, fanxiang.zfx,
fh265565, yue.liu}@alibaba-inc.com

Abstract

Multi-agent LLM systems can outperform single-agent baselines, yet practitioners still cannot predict which design works for a new task or diagnose why one fails. We argue this gap persists largely because the field lacks a diagnostic framework with measurable primitives and testable predictions. We introduce **DIANOIA**, a three-channel decomposition of multi-agent reasoning gain into coverage, fidelity, and synthesis, each of which is empirically measurable. From this decomposition, we propose a diagnostic protocol that forms qualitative hypotheses about bottleneck channels for tasks with measurable quality signals. We instantiate the protocol as a multi-agent system whose three components mirror the channels: role-diverse proposers for coverage, execution- or pseudo-verification-grounded review for fidelity, and iterative synthesis. On GSM8K, MBPP, and BFCL-SP, our method outperforms strong multi-agent baselines under matched standard configurations; on MBPP, it matches the strongest baseline with approximately $5\times$ fewer tokens and dominates the measured accuracy–token Pareto frontier. Full-scale MBPP replications on Qwen, DeepSeek, and Gemini consistently improve over the corresponding single-model baseline. A prospective held-out evaluation on VitaBench, whose bottleneck hypothesis was recorded before running DIANOIA, improves pass rate from 29.5% to 45.0%. DIANOIA reframes multi-agent design as channel-aware resource allocation: diagnose the likely bottleneck, then invest tokens accordingly.

1 Introduction

Reasoning with Large Language Models (LLMs) has shifted from single-prompt Chain-of-Thought (Wei et al., 2022) to systems that orchestrate multiple LLM instances as collaborative agents (Guo et al., 2024; Tran et al., 2025).

Multi-Agent Systems (MAS) and related inference-time agent methods have produced substantial empirical gains in mathematical reasoning (Wang et al., 2023), code generation (Shinn et al., 2023), and interactive decision-making (Zhou et al., 2024). Yet the mechanisms behind these gains remain poorly understood: in particular, it is unclear what drives improvement, when a given multi-agent design should help, and how practitioners should choose among methods for a new task.

Despite these empirical successes, existing methods remain largely heuristic: practitioners lack a principled basis for forming ex-ante hypotheses about which method may work on which task, or why a given method underperforms. We argue the path forward is twofold: (a) a *diagnostic framework* with measurable primitives and testable predictions, and (b) an *instantiated system* showing that the framework’s prescriptions can be realized simultaneously in one system. These two goals are complementary: the framework organizes hypotheses about *why* gains arise, while the system tests whether those gains can be achieved in practice. Its formal guarantees apply only under the verifier and independence conditions stated in Section 3.

We introduce a diagnostic decomposition of multi-agent reasoning gain—defined as $\mathbb{E}[Q(\tau^{\text{MAS}})] - p$, the improvement in expected solution quality over a single-agent baseline—formalized below as an upper bound with three conceptually distinct, empirically measurable components (formalized in Section 3, Decomposition 3.1):

$$\mathbb{E}[Q(\tau^{\text{MAS}})] - p \leq \underbrace{\mathcal{G}_{\text{explore}}}_{\text{coverage}} + \underbrace{\mathcal{G}_{\text{info}}}_{\text{fidelity}} + \underbrace{\mathcal{G}_{\text{aggr}}}_{\text{synthesis}} \quad (1)$$

where each component corresponds to a distinct source of gain in multi-agent reasoning:

- **Exploration Gain** ($\mathcal{G}_{\text{explore}}$): The benefit of *seeing more*: covering a larger portion of the solu-

* ORCID: 0000-0003-1359-0364

tion space through diverse proposals. This gain grows with the number of agents K and with the diversity of their reasoning strategies, through the probability that at least one agent discovers a correct solution path.

- **Information Gain** ($\mathcal{G}_{\text{info}}$): The benefit of *seeing clearly*: obtaining quality signals that reliably distinguish correct candidate solutions from incorrect ones. In our setting, such signals come from execution feedback e (e.g., code test results or tool-call return values) or from evidence-based cross-review that converts raw observations into structured assessments. The key quantity is therefore not feedback alone, but the *fidelity* of the information available for selection.
- **Aggregation Gain** ($\mathcal{G}_{\text{aggr}}$): The benefit of *deciding wisely*: synthesizing diverse proposals and feedback into a strong final solution. This gain depends on how effectively the aggregation mechanism uses available information while avoiding failure modes such as groupthink or agreement bias (Pitre et al., 2025).

Guided by this decomposition, we propose **DIANOIA**¹, a four-phase multi-agent system (Propose, Execute, Review, Synthesize) designed to address all three sources of gain: role-diverse proposal generation for Exploration, execution-grounded verification—combining execution feedback with evidence-based cross-review—for Information, and iterative synthesis with closed-loop validation for Aggregation. The three roles are one lightweight realization of the formal reduced-correlation prescription.

We evaluate DIANOIA on four benchmarks spanning mathematical reasoning (GSM8K (Cobbe et al., 2021), AIME-2025 (Mathematical Association of America, 2025)), code generation (MBPP (Austin et al., 2021)), and function calling (BFCL-SP (Patil et al., 2025)). Across GSM8K, MBPP, and BFCL-SP, DIANOIA outperforms strong multi-agent and Best-of- N baselines under matched standard configurations; we treat AIME-2025 as a small-sample stress test on its complete 30-problem set. On MBPP, it reaches the strongest baseline’s accuracy ceiling using roughly $5\times$ fewer tokens and dominates the measured budget–accuracy Pareto frontier. Methods that improve only one or two dimensions tend to saturate

earlier, whereas DIANOIA’s joint design sustains accuracy gains over a broader budget range, consistent with the framework’s predictions. We additionally conduct a prospective held-out diagnostic evaluation on VitaBench and full-scale MBPP replications on DeepSeek-V4-Flash and Gemini-3.5-Flash.

Our contributions are:

- A *prescriptive* diagnostic decomposition of multi-agent reasoning gain into coverage, information, and aggregation, with measurable estimators for each component. The decomposition places prior methods on a three-axis design map and yields a *four-rule diagnostic protocol* (R1–R4, §4) that forms qualitative bottleneck hypotheses for tasks with measurable quality signals.
- **DIANOIA**, a four-phase system that jointly enacts all three prescriptions of the decomposition. Each individual ingredient—role-diverse proposal, execution feedback, cross-review, and iterative synthesis—appears in prior work; our contribution is to combine them under a single diagnostic framework and evaluate their interaction systematically. Phase- and role-level ablations, together with a Best-of- N control, show that no tested proper subset recovers the full gain.
- Budget-matched comparisons on GSM8K, MBPP, and BFCL-SP—together with a small- N AIME-2025 stress test, a prospective held-out VitaBench evaluation, and full-scale MBPP replications across Qwen, DeepSeek, and Gemini. A full Pareto-frontier sweep on MBPP shows that DIANOIA matches the strongest baseline with approximately $5\times$ fewer tokens and improves the measured accuracy–cost frontier over a broad range of token budgets. The framework’s bottleneck hypotheses are consistent with the observed gain patterns (Section 5.7).

2 Related Work

We organize related work through the lens of our gain decomposition (Eq. 1; formalized in Section 3), which highlights three recurring design emphases in prior work: Exploration, Information, and Aggregation. Existing methods often prioritize one or two of these dimensions, which helps explain their complementary strengths and limitations and motivates our unified treatment.

Diversity and Exploration ($\mathcal{G}_{\text{explore}}$). Self-Consistency (Wang et al., 2023) samples multiple

¹From Greek *dianoia* (Plato): discursive reasoning.

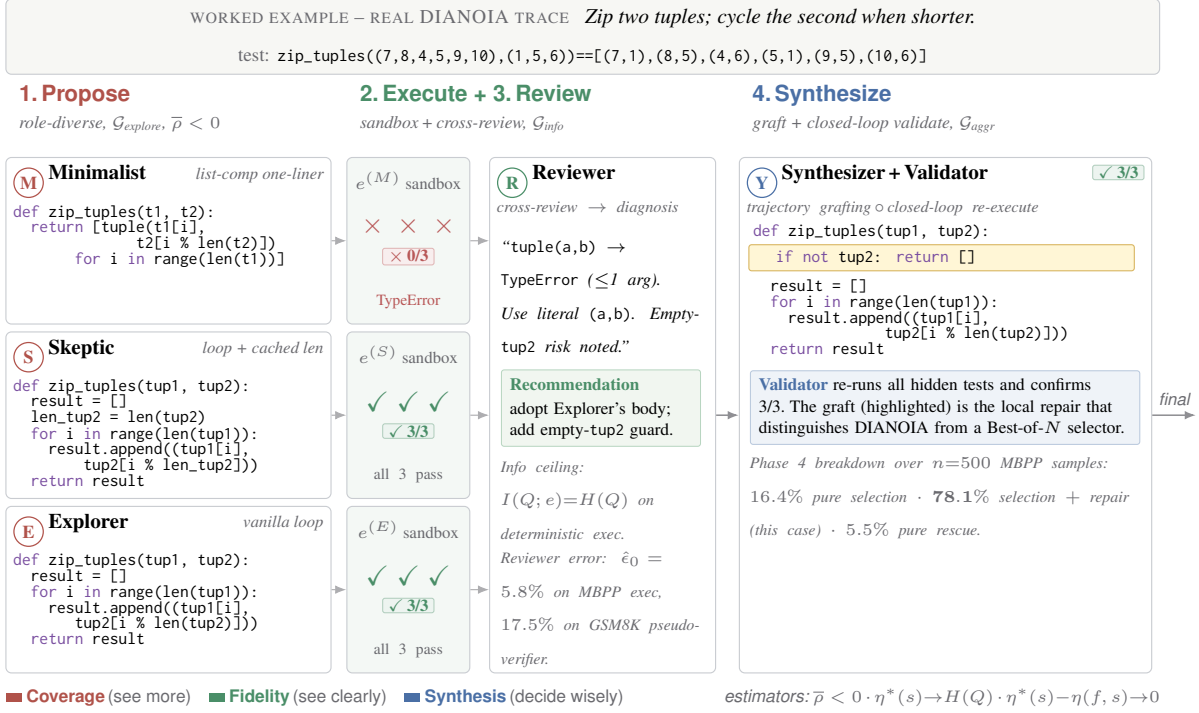


Figure 1: A DIANOIA trace on MBPP. Full log in Appendix N.

reasoning paths and selects via majority vote; Tree of Thoughts (Yao et al., 2023a) structures exploration as tree search; diversified sampling (Naik et al., 2023; Wang et al., 2026) promotes explicit path diversity beyond temperature; and ReConcile (Chen et al., 2024) orchestrates round-table discussions over heterogeneous LLMs. These methods primarily improve $\mathcal{G}_{\text{explore}}$ by broadening coverage of the solution space. However, many rely on limited or purely internal quality signals, and their aggregation mechanisms can remain vulnerable when errors are correlated across samples or agents.

Grounding and Feedback ($\mathcal{G}_{\text{info}}$). ReAct (Yao et al., 2023b) interleaves reasoning with tool actions; Reflexion (Shinn et al., 2023) reflects on execution feedback; LATS (Zhou et al., 2024) combines search with environmental verifiers; Agent-Coder (Huang et al., 2024) combines a Programmer agent with a Test-Executor agent for iterative test-driven repair; step-level process reward models (Lightman et al., 2024; Setlur et al., 2025) provide denser supervision signals; and collaborative verification (Liang et al., 2024b) strengthens test-time quality assessment. These approaches improve $\mathcal{G}_{\text{info}}$ by strengthening the fidelity of intermediate or final feedback. However, they typically explore alternatives sequentially within a single search process rather than through parallel, het-

erogeneous proposal generation, which can limit $\mathcal{G}_{\text{explore}}$.

Interaction and Consensus ($\mathcal{G}_{\text{aggr}}$). Multi-Agent Debate (Du et al., 2024) lets agents critique each other; Encouraging Divergent Thinking (Liang et al., 2024a) promotes early disagreement; Mixture-of-Agents (MoA) (Wang et al., 2025) stacks aggregation layers; Two Heads (Jin et al., 2025a) studies multi-agent test-time scaling, echoing single-agent compute-scaling results (Snell et al., 2025) that also motivate the Best-of- N control we use. Pure debate is “cheap talk” without evidence; ConsensAgent (Pitre et al., 2025) documents sycophancy and agreement bias under purely textual interaction. Surveys (Guo et al., 2024; Tran et al., 2025; Ke et al., 2025) catalogue the landscape; game-theoretic and cooperative-decision surveys (Sun et al., 2025; Jin et al., 2025b) connect agent interaction to strategic and cooperative decision-making.

Our decomposition is also related to several theoretical traditions. In ensemble learning, the bias–variance–covariance decomposition (Ueda and Nakano, 1996; Wood et al., 2023), the diversity-prediction theorem (Hong and Page, 2004; Page, 2007), and recent analyses of correlated failure in LLM ensembles (Kim et al., 2025) all emphasize that diversity alone is insufficient when errors are dependent. Separately, verifier-fidelity stud-

Table 1: Related methods through the gain decomposition lens. ✓: primary emphasis; ○: secondary or implicit; ✗: not a focus.

Method	$\mathcal{G}_{\text{explore}}$	$\mathcal{G}_{\text{info}}$	$\mathcal{G}_{\text{aggr}}$	Key Limitation
<i>Exploration-focused Methods</i>				
Self-Consistency (Wang et al., 2023)	✓	✗	○	Correlated voting errors
Tree of Thoughts (Yao et al., 2023a)	✓	○	○	Heuristic, ungrounded evaluation
ReConcile (Chen et al., 2024)	✓	○	○	Textual-only feedback
<i>Information-focused Methods</i>				
ReAct (Yao et al., 2023b)	○	✓	✗	No parallel diversity
Reflexion (Shinn et al., 2023)	○	✓	✗	Sequential self-improvement only
LATS (Zhou et al., 2024)	✓	✓	✗	Single-search aggregation only
<i>Aggregation-focused Methods</i>				
Multi-Agent Debate (Du et al., 2024)	○	○	✓	Ungrounded textual consensus
MoA (Wang et al., 2025)	○	○	✓	Weak external grounding
Two Heads (Jin et al., 2025a)	○	○	○	No explicit grounding
DIANOIA (Ours)	✓	✓	✓	Jointly targets all three dimensions

ies (Lightman et al., 2024; Setlur et al., 2025) analyze how model-based scoring can distort the quality signal used for selection. Our work adapts these intuitions to the test-time multi-agent reasoning setting.

Table 1 summarizes representative methods through the lens of our decomposition. The goal is not to assign exclusive categories, but to highlight each method’s primary design emphasis and the dimensions that are less directly optimized.

In contrast, DIANOIA jointly enacts all three dimensions within a single workflow, addressing the fragmentation highlighted above.

3 A Diagnostic Decomposition for Multi-Agent Reasoning

Multi-agent systems have demonstrated remarkable improvements over single-agent baselines (Wang et al., 2023; Du et al., 2024), yet a fundamental question remains: *where do these gains come from, and can we predict them?* We develop a *diagnostic decomposition* that exposes three empirically measurable mechanisms—**Exploration**, **Information**, and **Aggregation**. The formalism is intentionally lightweight: the contribution lies not in mathematical novelty per se, but in a decomposition whose terms are individually measurable and prescriptive for system design—together yielding a four-rule diagnostic protocol (R1–R4, §4) that forms qualitative bottleneck hypotheses from a task profile (verifier type, baseline p , answer-space topology). The claims below distinguish an exact identity, a conditional subadditive bound, and idealized bounds. R1–R4 are qualitative task-structure hypotheses, not calibrated consequences of the latter.

Table 2: Connection between DIANOIA phases and diagnostic quantities.

Phase	Quantity	Mechanism
Propose	$C_K, \bar{\rho}$	Candidate coverage/diversity
Execute/Review	$\eta^*(s), \hat{e}_0$	Evidence fidelity
Synthesize	$\eta(f, s), r$	Selection, repair, rescue

We formalize multi-agent reasoning as a tuple $(\mathcal{X}, \mathcal{T}, Q, K, \mathcal{E}, f)$ where $\mathcal{X}$ is the input space, $\mathcal{T}$ the solution space, $Q : \mathcal{T} \rightarrow \{0, 1\}$ a quality indicator, K proposers, $\mathcal{E}$ an executor providing feedback, and f an aggregation function. We assume: (A1) *baseline* conditional independence among proposers as the IID reference (relaxed via explicit pairwise correlation terms in Prop. 3.2; see App. G.7), (A2) baseline success $p \in (0, 1)$, (A3) finite strategy space (each agent’s generation is bounded by maximum token length and finite vocabulary, ensuring $|\mathcal{T}| < \infty$), (A4) deterministic execution. Full definitions in Appendix E.

We organize the decomposition below into three logically distinct layers—an exact identity, a definitional attribution, and a sub-additive upper bound—separated to make explicit which steps are tautological accounting and which are design-motivated modeling choices.

Decomposition 3.1 (Identity, Attribution, and Bound). *Under (A1)–(A4):*

(I) **Exact identity** (definitional factorization; App. G.1):

$$\mathbb{E}[Q(\tau^{\text{MAS}})] = \underbrace{C_K}_{\text{coverage}} \cdot \underbrace{\eta(f, s)}_{\text{effective efficiency}} \quad (2)$$

where $C_K := P(\bigvee_k Q(\tau^{(k)}) = 1)$ and $\eta(f, s) := \mathbb{E}[Q(\tau^{\text{MAS}})]/C_K$ (Def. D.8). η equals the classical conditional accuracy $P(Q^{\text{MAS}} = 1 \mid E)$ for selector-only systems; for repair-capable DIANOIA it additionally absorbs synthesis’s rescue contribution (quantify analysis in App. I).

(II) **Gain attribution** (definitional, each tied to a distinct design lever): (I) $\mathcal{G}_{\text{explore}} := C_K - p$; (2) $\mathcal{G}_{\text{info}} := C_K[\eta^*(e) - \eta^*(\sigma)]$, where $\eta^*(s) := \max_f \eta(f, s)$; (3) $\mathcal{G}_{\text{aggr}} := C_K[\eta(f, s) - \eta(f_{\text{base}}, s)]$, where f_{base} is a task-appropriate non-iterative reference selector (e.g., majority vote or score-maximization; specified per analysis, see App. G.1). This attribution is not unique; we adopt it because each term corresponds to a distinct DIANOIA lever and admits an empirical estimator, with the synergy coefficient γ in §5.3 serving as a post-run descriptive subadditivity check; γ is neither fitted nor used

to construct R1–R4.

(III) Conditional subadditive upper bound. For selector-only systems, or for repair-capable systems satisfying the rescue-margin condition stated in Appendix G.1,

$$\mathbb{E}[Q(\tau^{\text{MAS}})] - p \leq \mathcal{G}_{\text{explore}} + \mathcal{G}_{\text{info}} + \mathcal{G}_{\text{aggr}}, \quad (3)$$

strict in general due to multiplicative coupling (Remark 3.1; mechanism in App. G.1, Part IV).

Remark 3.1 (Conceptual Orthogonality vs. Realized Subadditivity). *Exploration, Information, and Aggregation are conceptually orthogonal (independently optimizable: generate vs. evaluate vs. combine) but their realized gains are statistically coupled through the multiplicative structure $C_K \cdot \eta(f, s)$, yielding subadditivity (actual gain < sum), as in classical ensemble theory (Krogh and Vedelsby, 1995; Wood et al., 2023) and collective intelligence (Hong and Page, 2004). Empirically, the synergy coefficient $\gamma := \Delta_{\text{joint}} / \sum_i \Delta_i$ falls in $[0.852, 0.88]$ across MBPP and BFCL-SP (§5.3), consistent with the subadditive regime predicted by the multiplicative coupling.*

We characterize each dimension below; proofs are deferred to Appendix G.

Exploration. When structured diversity yields negative average pairwise success correlation $\bar{\rho}$, coverage increases relative to the IID reference:

Proposition 3.2 (Exploration via Diversity). *The second-order Bonferroni coverage lower bound $\text{LB}(\bar{\rho})$ satisfies $\text{LB}(\bar{\rho}) - \text{LB}(0) = -\binom{K}{2} \bar{\rho} p(1-p) > 0$ for $\bar{\rho} < 0$, i.e., the guaranteed coverage lower bound under diverse proposers strictly exceeds the IID baseline (App. G; strict comparison at actual-coverage level requires extra dependence assumptions).*

Information. Verifier type sets an upper bound on achievable selection accuracy:

Proposition 3.3 (Information Quality Bounds). *Under complete and deterministic verification (Q fully determined by e ; Def. D.5, A4), $I(Q; e) = H(Q)$. Textual feedback with non-zero error rates satisfies $I(Q; \sigma) < H(Q)$; pseudo-verification σ_v satisfies $I(Q; \sigma) \leq I(Q; \sigma_v) \leq I(Q; e)$ (strict middle inequality empirical when σ_v exploits specialization; App. F).*

Remark 3.2 (Verifier Regimes). $\mathcal{G}_{\text{info}}$ is set by verifier type: complete verification (MBPP, BFCL-SP) gives $I(Q; e) = H(Q)$; pseudo-verification (GSM8K, AIME) gives bounded $\mathcal{G}_{\text{info}}$; no external

verifier (many open-domain tasks) sharply limits $\mathcal{G}_{\text{info}}$, leaving only low-fidelity internal signals.

By Fano-type bounds (Lemma L.5), higher-fidelity signals admit tighter achievable selection-error bounds; the realized $\mathcal{G}_{\text{info}}$ is therefore upper-bounded by verifier type, consistent with the cross-task pattern in §5.7.

Aggregation. Majority voting fails under correlated errors (de Condorcet, 1785; Kim et al., 2025); evidence-based cross-review escapes this regime:

Proposition 3.4 (Aggregation Efficiency; Idealized Bound). *(a) Voting degradation under correlation. As agent errors become highly correlated, majority voting loses its ensemble error-reduction advantage; in the worst case (adversarial correlation), the aggregation gain $\eta(f_{\text{vote}}) - \eta(f_{\text{base}})$ vanishes. (b) DIANOIA’s bound, under the simplified conditionally-independent reviewer model (A6): with $K-1$ reviewers ($\epsilon_0 < 0.5$),*

$$\eta(f_{\text{DIANOIA}}) \geq 1 - \epsilon_0^{K-1}.$$

This is a best-case bound: residual reviewer correlation $\rho_R > 0$ floors the error at $\epsilon_\infty := \epsilon_0 \rho_R + \epsilon_0^{K-1}(1 - \rho_R)$ in practice (§5.8).

Combining the three:

Theorem 3.5 (DIANOIA Characterization). *Under (A1)–(A6) (App. E), DIANOIA satisfies (a) information sufficiency $I(Q; e) = H(Q)$ on complete and deterministic verifiers; (b) finite-step convergence as an exact potential game on complete and deterministic verifiers (approximate under pseudo-verification); (c) performance lower bound $\mathbb{E}[Q] \geq [1 - (1-p)^K][1 - \epsilon_0^{K-1+S}]$ on execution tasks, under the additional synthesis-iteration-independence idealization (A7, App. G). The exact/approximate/inapplicable regime distinction across verifier types is in App. G.7.*

Scope of formal claims. Equation 2 is an exact definitional identity. Equation 3 is a conditional subadditive bound for selector-only systems, or for repair-capable systems satisfying the stated rescue-margin condition. Proposition 3.4(b) is an idealized reviewer-independence bound under A6 and analyzes the $R = K - 1$ reviewer regime; DIANOIA’s default configuration uses $R = 1$. Theorem 3.5(b) applies exactly only under complete and deterministic verification. The S -dependent tightening in Theorem 3.5(c) additionally relies on A7. Under pseudo-verification these results provide qualitative guidance rather than calibrated numerical guarantees.

Idealized theory versus measured practice.

The idealized bounds predict exponential tightening with independent reviewers and synthesis iterations. On MBPP, however, increasing synthesis iterations from $S = 1$ to 2 to 3 changes accuracy from 83.6% to 84.0% to 84.6%, while increasing reviewers from $R = 1$ to $R = 3$ changes accuracy from 84.6% to 85.8% at approximately $2.6\times$ the token cost. The directions agree with the idealized analysis, but the magnitudes do not: these bounds describe best-case trends, not calibrated scaling laws.

4 DIANOIA Methodology

DIANOIA enacts all three framework prescriptions in a four-phase workflow (Figure 1): *Propose* for Exploration, *Execute* and *Review* jointly for Information (raw evidence vs. its interpretation into actionable signals), and *Synthesize* for Aggregation. No proper subset of these phases reproduces the gain (Section 5.3). We detail each phase below.

Phase 1: Propose ($\mathcal{G}_{\text{explore}}$). DIANOIA assigns each proposer a distinct role—**Minimalist** (fewest steps), **Skeptic** (verify each step), **Explorer** (unconventional methods)—as one lightweight realization intended to reduce pairwise success correlation (Proposition 3.2). The governing prescription is reduced correlation rather than this particular role taxonomy. Full role prompts are in Appendix M. Proposers generate candidates in parallel, with IID reference coverage $1 - (1 - p)^K$.

Phase 2: Execute ($\mathcal{G}_{\text{info}}$). Each candidate is evaluated via the highest-fidelity feedback mechanism available—sandboxed execution or LLM-based pseudo-verification—producing $e^{(k)} = \mathcal{E}(\tau^{(k)})$. By Theorem 3.5a, complete and deterministic verification yields $I(Q; e) = H(Q)$.

Phase 3: Review ($\mathcal{G}_{\text{info}}$). R reviewers per proposal perform evidence-based cross-review $v_j^{(k)} = \mathcal{R}_j(\tau^{(k)}, e^{(k)})$, transforming raw feedback into actionable quality signals (analyzing failure causes, identifying validated components, suggesting fixes). The default $R = 1$ already captures the bulk of the aggregation gain (App. G.7.5, Remark); the idealized peer-review regime $R = K - 1$ underlies the ϵ_0^{K-1} misclassification bound of Proposition 3.4b.

Review beyond Execute. Execution gives *whether* a candidate fails; Review adds *why*, *which sub-parts pass*, and *how to repair*—feeding the synthesizer with the diagnostic content needed for

repair (rather than mere selection).

Phase 4: Synthesize ($\mathcal{G}_{\text{aggr}}$). A synthesis agent integrates proposals, execution reports, and reviews through iterative refinement with closed-loop validation (Algorithm 1, Appendix M): trajectory grafting, re-execution, deterministic synthesis. Although the formal bound (Theorem 3.5c) treats this phase as evidence-conditioned selection, in practice the synthesizer can perform local repair: instrumentation on MBPP shows 5.5% of DIANOIA’s correct outputs are *rescued* cases where every initial proposal failed execution but the synthesizer recovered a correct answer using reviewer hints (Appendix I). The bound therefore lower-bounds, rather than fully characterizes, Phase 4.

Diagnostic protocol. Four rules from the framework map task profiles (verifier type, baseline p , answer-space topology) to qualitative bottleneck hypotheses for tasks with measurable quality signals. **R1**: a deterministic verifier lifts $\mathcal{G}_{\text{info}}$ to its ceiling, $I(Q; e) = H(Q)$ (Thm. 3.5a). **R2**: a high baseline p leaves small exploration headroom $C_K - p$ (Prop. F.1). **R3**: a fragmented answer space renders majority voting unreliable (Failure Mode 1, App. F). **R4**: frequent per-instance $p_i < 0.5$ makes Self-Consistency actively harm accuracy (Failure Mode 2). These hypotheses are evaluated empirically in §5.7.

5 Experiments

We conduct comprehensive experiments to validate DIANOIA’s effectiveness across four benchmarks spanning math, code, and tool use, and systematically ablate each gain dimension to demonstrate the necessity of joint optimization.

5.1 Experimental Setup

Datasets. We evaluate on four benchmarks spanning math, code, and tool use, chosen for varied feedback regimes (Remark 3.2): *GSM8K* ($N=1319$, high p , pseudo-verifier) (Cobbe et al., 2021); *AIME-2025* ($N=30$, low p , pseudo-verifier) (Mathematical Association of America, 2025); *MBPP* ($N=500$, deterministic exec) (Austin et al., 2021); *BFCL-SP* ($N=400$, schema-validation exec) (Patil et al., 2025). Full dataset details in Appendix M.

Baselines. Self-Consistency (Wang et al., 2023) (sampling + majority vote), MoA (Wang et al., 2025) (layered aggregation), Two Heads (Jin et al., 2025a) (collaborative reasoning), ReConcile (Chen

Table 3: Main results. Multi-agent methods (top) use Qwen3-30B-A3B; single-model references (bottom).

Method	GSM8K	AIME-2025	MBPP	BFCL-SP
Self-Consistency	86.4% [84.5, 88.3]	56.7% [40.0, 73.3]	78.0% [74.2, 81.2]	82.3% [78.5, 85.8]
Best-of-3 (same verifier)	87.0% [85.2, 88.8]	73.3% [56.7, 90.0]	78.4% [75.0, 81.8]	84.0% [80.5, 87.5]
MoA	87.1% [85.2, 88.9]	86.7% [73.3, 96.7]	76.8% [73.2, 80.2]	85.8% [82.5, 89.3]
Two Heads	85.8% [84.0, 87.6]	80.0% [66.7, 93.3]	77.2% [73.6, 81.2]	88.8% [85.5, 92.0]
ReConcile	89.8% [88.3, 91.5]	70.0% [53.3, 86.7]	77.2% [73.4, 81.0]	82.3% [78.3, 86.0]
DIANOIA (Ours)	91.1% [89.6, 92.7]	93.3% [‡] [83.3, 100]	84.6% [81.4, 87.0]	92.3% [89.5, 94.8]
Qwen3-30B-A3B	83.6% [81.5, 85.7]	70.0% [53.3, 86.7]	76.0% [72.4, 80.0]	81.8% [78.0, 85.8]
Qwen3-235B-A22B	86.4% [84.6, 88.3]	73.3% [56.7, 86.7]	80.2% [76.6, 83.8]	89.0% [85.8, 92.0]
DeepSeek-V3.2	85.3% [83.3, 87.2]	76.8% [†] [60.0, 90.0]	81.2% [77.6, 84.6]	83.5% [79.5, 87.3]

[†]Extended reasoning (thinking mode) enabled. [‡]Complete 30-problem set; reported as a small-sample point estimate.

et al., 2024) (round-table discussion), and a *Best-of-3* control sampling three IID candidates and selecting with the same execution signal as DIANOIA. Single-model references: Qwen3-30B-A3B (Qwen Team, 2025) (base), Qwen3-235B-A22B, DeepSeek-V3.2 (DeepSeek-AI, 2025).

Implementation. All multi-agent methods use Qwen3-30B-A3B-Instruct-2507 with thinking mode disabled (zero-shot). DIANOIA default: $K=3, R=1, S=3$; baselines use the standard configurations from their respective papers. Full prompts, and other details in Appendix M.

5.2 Main Results

Table 3 presents results across all four benchmarks. Across GSM8K, MBPP, and BFCL-SP, DIANOIA outperforms every multi-agent baseline under the standard configurations.

DIANOIA improves over the strongest multi-agent baseline by +1.3pp (GSM8K), +6.6pp (MBPP), and +3.5pp (BFCL-SP); on MBPP its CI lower bound (81.4%) exceeds every baseline’s upper bound. The Best-of-3 control—which uses the same verifier as DIANOIA but without its multi-agent structure—reaches 78.4% on MBPP, 6.2pp below DIANOIA, isolating the marginal value of $\mathcal{G}_{\text{aggr}}$. AgentCoder (Huang et al., 2024), a code-specific MAS benchmarked on its natural MBPP domain, reaches 81.2% (3.4pp below DIANOIA on Qwen3-30B-A3B; cross-model results in §5.5).

On the complete 30-problem AIME-2025 set, DIANOIA obtains a 93.3% point estimate. An exploratory paired McNemar exact test against ReConcile yields $p = 0.039$ from only nine discordant pairs; the wide confidence interval precludes a precise effect-size or model-superiority claim.

The cross-benchmark pattern matches the

Table 4: Joint-dimension scaling on MBPP.

Config	K	R	S	Acc	Activates
Baseline	1	0	0	76.0%	—
Explore-only	3	0	1	81.2%	$\mathcal{G}_{\text{explore}}$
Info-only	1	1	1	79.8%	$\mathcal{G}_{\text{info}}$
Aggr-only	1	0	1	76.8%	$\mathcal{G}_{\text{aggr}}$
Two-dim	3	1	1	83.6%	Explore+Info
DIANOIA-full	3	1	3	84.6%	All three

framework’s predictions: DIANOIA’s largest gains emerge on execution-grounded tasks where $I(Q; e) = H(Q)$ activates $\mathcal{G}_{\text{info}}$; Self-Consistency *degrades* on AIME-2025 due to voting failure under low per-problem p ; the GSM8K gain is bounded by pseudo-verifier fidelity. We formalize and verify these predictions in Section 5.7, and use the budget-matched Pareto frontier (Section 5.4) as the primary fairness evidence for efficiency.

5.3 Multi-Dimensional Scaling Analysis

As a post-run descriptive check of subadditivity, we compute $\gamma = \Delta_{\text{joint}} / \sum_i \Delta_i$ on the same MBPP and BFCL-SP runs used for the component analysis. The common empirical reference is $f_{\text{base}} = (K, R, S) = (1, 0, 0)$. Because γ is computed after the runs and is not held out, it is not used to define or validate R1–R4. We obtain $\gamma = 0.88$ on MBPP and $\gamma = 0.852$ on BFCL-SP; these are descriptive evidence of subadditivity rather than a validation of the idealized theory.

A *role design ablation* (App. H, Table 12) further isolates the exploration gain: *any* structured role design (three-role triplet 84.6%, two-role subsets 81.4–84.4%, alternative triplet 83.4%) beats same-prompt+temperature diversity (83.2%), but within the structured family, CIs overlap, so the

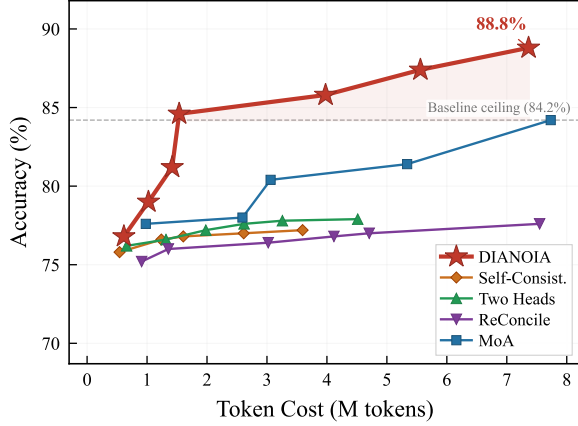


Figure 2: Token-budget Pareto frontier on MBPP.

gain comes from structured diversity as a *principle*, not from any specific hand-crafted prompt. Single-dimension scaling sweeps and the BFCL-SP joint-optimization table ($\gamma=0.852$) are also in App. H.

Information-dimension diagnostics. Reviewer error $\hat{\epsilon}_0 = 5.8\%$ on MBPP (exec) vs. 17.5% on GSM8K (pseudo-verifier); the GSM8K pseudo-verifier attains precision 94.0% / recall 86.6%. These numbers operationalize Remark 3.2 and feed the diagnostic analysis in Section 5.7.

5.4 Budget-Matched Pareto Frontier

Standard-configuration comparisons (Table 3) leave open whether each method could match DIANOIA at higher compute. We sweep each method along its native scaling knobs (39 configurations enumerated in App. M) and plot the upper Pareto envelope of accuracy vs. tokens on MBPP (Figure 2).

Self-Consistency, Two Heads, and ReConcile plateau below 78% at every token budget we tested. MoA breaks through only by stacking three layers, reaching 84.2% at 7.73M tokens; DIANOIA reaches 84.6% at just 1.54M—a $\sim 5\times$ token-budget advantage at matched accuracy—and continues to 88.8% at 7.36M tokens. Appendix C reports tokens and wall-clock latency; the latter is serving-dependent.

5.5 Cross-Model Robustness

To check whether the gain pattern is Qwen3-30B-specific, we replicate the headline experiment on **Qwen3.6-35B-A3B** across MBPP, BFCL-SP, and GSM8K (Table 5)—spanning both deterministic and pseudo-verifier regimes. **DIANOIA leads on every cell at point-estimate level**; the gap over the single-model baseline *persists or widens* on execution-grounded tasks ($+8.6 \rightarrow +10.4$ pp

Table 5: Cross-model robustness on Qwen3.6-35B-A3B (Qwen Team, 2026b).

Method	MBPP	BFCL-SP	GSM8K
<i>Qwen3-30B-A3B (headline base, Table 3)</i>			
Single	76.0 [72.4, 80.0]	81.8 [78.0, 85.8]	83.6 [81.5, 85.7]
Best-of-3	78.4 [75.0, 81.8]	84.0 [80.5, 87.5]	87.0 [85.2, 88.8]
AgentCoder [†]	81.2 [77.6, 84.4]	—	—
DIANOIA	84.6 [81.4, 87.0]	92.3 [89.5, 94.8]	91.1 [89.6, 92.7]
<i>Qwen3.6-35B-A3B (cross-version replication)</i>			
Single	78.4 [74.6, 81.8]	83.8 [80.3, 87.3]	84.2 [82.3, 86.1]
Best-of-3	83.2 [79.8, 86.4]	87.0 [83.8, 90.0]	83.8 [81.8, 85.8]
AgentCoder [†]	88.4 [85.4, 90.8]	—	—
DIANOIA	88.8 [86.0, 91.6]	93.0 [90.5, 95.5]	91.0 [89.5, 92.5]

[†]Code-specific MAS; MBPP is its natural domain.

MBPP; $+10.5 \rightarrow +9.2$ pp BFCL-SP, within bootstrap CI overlap), indicating the prescriptions compose with rather than are absorbed by stronger base models.

The framework also predicts *where Best-of-N wins or loses*: on execution-grounded tasks BoN’s gap over the single model widens with the stronger base ($+2.4 \rightarrow +4.8$ pp MBPP, $+2.2 \rightarrow +3.2$ pp BFCL-SP); but on pseudo-verified GSM8K, BoN’s gap *inverts* to -0.4 pp—the lossy pseudo-verifier can no longer rank top- N candidates reliably, injecting noise (P2: information-ceiling bottleneck). DIANOIA avoids this by using the verifier as input to richer aggregation rather than as the direct selection signal; the DIANOIA – BoN margin therefore *widens* on GSM8K ($+4.1 \rightarrow +7.2$ pp) even as it compresses on the execution-grounded tasks.

AgentCoder’s MBPP gain widens with the stronger base ($+5.2 \rightarrow +10.0$ pp), showing iterative repair scales with single-pass quality. DIANOIA still leads ($+3.4$ pp on Qwen3-30B-A3B; $+0.4$ pp on Qwen3.6, within CI overlap); the margin contracts as both approach the MBPP execution ceiling, consistent with §7’s operating regime.

Cross-host evidence (Claude Code skill). The same $K=3$, $S=3$, three-role prescriptions also realize as a Claude Code skill (App. A). With Claude Opus 4.7 (Anthropic, 2026), the skill yields $+6.7$ pp gain on BigCodeBench-Hard ($33.3 \rightarrow 40.0\%$, $N=30$) and Hard MBPP ($80.0 \rightarrow 86.7\%$, $N=15$)—a consistent direction on a different model family (Claude vs. Qwen3), though smaller N precludes tight CIs.

Together with the full-scale Qwen results in Table 5, these replications on DeepSeek-V4-Flash (DeepSeek-AI, 2026) and Gemini-3.5-

Table 6: Full-scale MBPP replication on additional model families ($N = 500$). Values are accuracy with 95% confidence intervals.

Model	Single	Best-of-3	DIANOIA
DeepSeek-V4-Flash	88.8 [86.2, 91.4]	97.0 [95.6, 98.3]	98.2 [97.0, 99.2]
Gemini-3.5-Flash	93.0 [90.6, 95.0]	93.8 [91.6, 95.8]	96.8 [95.2, 98.2]

Table 7: Prospective held-out evaluation on VitaBench ($N = 400$).

Method	Pass rate	95% CI
Single	29.5%	[25.2, 34.1]
DIANOIA	45.0%	[40.2, 49.9]

Flash (Google DeepMind, 2026) show that DIANOIA consistently improves over Single and achieves the highest point estimate among the compared methods across Qwen, DeepSeek, and Gemini at full MBPP scale.

5.6 Prospective Held-Out Diagnostic Check

The four original benchmarks provide within-study prediction–verification consistency. VitaBench (He et al., 2025) provides a prospective held-out diagnostic check. Before running DIANOIA, we characterized VitaBench from its structural task profile: it has no deterministic semantic verifier and requires long-horizon interactive reasoning. The protocol therefore predicted that information fidelity would be the primary bounded channel, while exploration and repair would retain substantial headroom. This prediction was recorded without using measured Single or DIANOIA accuracy.

The resulting +15.5pp gain is consistent with the predicted regime. VitaBench uses Qwen3.5-397B-A17B (Qwen Team, 2026a) because Qwen3-30B-A3B produces a near-floor operating point on this substantially harder benchmark. We therefore treat the base-model change as an explicit scope caveat. Because evaluation relies on semantic pseudo-verification, this experiment supplies empirical boundary evidence rather than an oracle-level guarantee.

5.7 Diagnostic Protocol: Verification

Applying rules R1–R4 (§4) to each benchmark yields four predictions: **P1** (R1+R3, exec-grounded MBPP/BFCL-SP): largest DIANOIA gains +6.6/ +3.5pp; **P2** (R2, high- p GSM8K): smallest gain +1.3pp; **P3** (R4, AIME-2025): SC degrades −13.3pp, the predicted voting-amplification regime; **P4** (BoN baseline): DIANOIA exceeds Best-of-3 by +6.2pp on MBPP,

isolating $\mathcal{G}_{\text{aggr}}$ since BoN already activates Explore+Info. These are within-study consistency checks; R1–R4 are qualitative task-structure hypotheses, not calibrated theorem derivations. VitaBench supplies the prospective check: its pre-recorded hypothesis and subsequent 29.5% to 45.0% improvement are consistent. Per-prediction narratives are in Appendix J.

5.8 Behavioral Profile

Four empirical operating numbers characterize DIANOIA: pairwise proposer correlation $\bar{\rho} = -0.15$ (drives coverage); reviewer error $\hat{e}_0 = 5.8\%$ on execution vs. 17.5% on pseudo-verification (caps $\mathcal{G}_{\text{info}}$); GSM8K pseudo-verifier precision 94.0%/recall 86.6% (caps math-task ceiling); synthesis breakdown 16.4/78.1/5.5% across selection / selection-with-repair / rescue (App. I). Here $\bar{\rho}$ denotes proposer success correlation, not reviewer-error correlation, and therefore does not empirically validate A6. A controlled GSM8K stress test further shows that accuracy remains within 91–93% through 25% injected evidence corruption and falls to 88% at 30% (Table 8 in the appendix).

6 Conclusion

Three measurable channels—coverage, information ceiling, and aggregation gap—organize observed cross-task variation in multi-agent reasoning gains, and the four-rule protocol (R1–R4, §4) forms qualitative bottleneck hypotheses. The four original patterns and the prospective VitaBench check are consistent with these hypotheses. Full-scale MBPP replications show improvements over Single across Qwen, DeepSeek, and Gemini. On MBPP, DIANOIA matches the strongest baseline with approximately $5\times$ fewer tokens and dominates the measured budget–accuracy Pareto frontier.

Multi-agent design thus becomes *channel-aware resource allocation*: practitioners can diagnose the likely bottleneck and invest tokens accordingly. Each channel admits a measurable estimator ($\bar{\rho}$, $\eta^*(s)$, aggregation gap; γ as a post-run descriptive check), letting designers test components before committing. The framework provides a common axis for comparing future systems that instantiate these prescriptions differently; its exact identities, conditional bounds, and idealized guarantees have distinct scopes.

7 Limitations

Our method improves performance across several reasoning settings, but its applicability and guarantees are bounded by two distinct types of constraint: conditions assumed by our theoretical analysis, and gaps in our empirical validation. We address each in turn.

Scope of tasks. The decomposition is most informative when a reasonably reliable quality signal is available. On tasks with deterministic verifiers—such as code execution or schema validation—the information channel $\mathcal{G}_{\text{info}}$ can operate at full capacity (Theorem 3.5(a)). On tasks with model-based pseudo-verifiers—such as mathematical reasoning—it operates at reduced but measurable fidelity (precision 94.0%, recall 86.6%, §5.8). For open-ended generation, dialogue, or culturally grounded reasoning, reliable quality signals are often unavailable, so the practical value of $\mathcal{G}_{\text{info}}$ is substantially reduced and the framework becomes correspondingly less prescriptive. VitaBench provides initial boundary evidence for open-world, long-horizon interactive tasks, but it relies on semantic pseudo-verification rather than a deterministic oracle, and its use of Qwen3.5-397B-A17B prevents a strictly matched-base comparison with the core benchmarks. The result therefore supports the qualitative protocol at its boundary without extending exact verifier-dependent guarantees to open-ended tasks. Extending the decomposition to weakly verifiable or verifier-free settings is an important open direction.

Effective operating regime. DIANOIA is most effective when exploration headroom and verifier signal are jointly substantial, i.e., in a broad mid-accuracy regime rather than at either extreme. Two boundary conditions limit generality. *High baseline:* residual headroom is small and the additional token cost may not be justified. *Very low baseline:* role-diverse proposers may fail to surface any correct candidate, shifting more of the burden from cross-review to the synthesizer’s repair capability. Table 5 provides one empirical calibration point; practitioners should locate their task on this curve before deployment.

Latency and compute proxy. Throughout the paper we report token count as the primary compute proxy, which captures model usage but not wall-clock latency or infrastructure cost under API rate limits. DIANOIA’s main sequential bottleneck

is the S synthesis iterations; Phases 1 and 3 are already parallelized over K and $K \cdot R$ agents respectively ($R=1$ in the default configuration; see App. K). The latency profile is approximately

$$L_{\text{propose}} + L_{\text{execute}} + L_{\text{review}} + S \cdot (L_{\text{synth}} + L_{\text{execute}}).$$

Measured wall-clock latency is reported in Appendix C; DIANOIA is substantially slower than Single because proposal/review and iterative synthesis introduce additional calls. These measurements depend on provider load, API scheduling, and available parallelism, so the cost–accuracy tradeoffs may differ in production settings and do not constitute a hardware-independent latency bound.

Empirical scope. AIME-2025 ($N=30$) supports the gain direction but not a precise estimate of effect size; its wide confidence interval and nine discordant pairs do not support a model-superiority claim. We therefore treat the complete 30-problem set as a small-sample stress test, and broader competition-math evaluation is needed before drawing quantitative conclusions about the low-baseline regime.

Theoretical idealizations. Two formal results assume conditions that hold exactly on execution-grounded tasks but only approximately on pseudo-verified ones. Proposition 3.4(b)’s exponential reviewer-error bound assumes conditional independence among reviewers (A6); agents instantiated from the same base model can retain correlated errors, so the bound should be interpreted as a best-case trend rather than a tight numerical guarantee. The reported $\bar{\rho} = -0.15$ measures correlation among proposer success events and supports the intended diversity regime; it does not measure reviewer-error independence or validate A6. Proposition 3.4(b) also analyzes $R=K-1$ reviewers, whereas the default configuration uses $R=1$. Theorem 3.5(a)’s information-sufficiency result requires a complete-and-deterministic verifier (Definition D.5); GSM8K, AIME-2025, and VitaBench instead use pseudo-verification and therefore support empirical diagnostics rather than exact information sufficiency. The S -dependent tightening additionally relies on the A7 synthesis-iteration-independence idealization, although repeated failures can be correlated in practice.

References

- Anthropic. 2026. Introducing claude opus 4.7. <https://www.anthropic.com/news/claude-opus-4-7>.
- Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, and Charles Sutton. 2021. Program synthesis with large language models. arXiv preprint arXiv:2108.07732.
- Justin Chih-Yao Chen, Swarnadeep Saha, and Mohit Bansal. 2024. Reconcile: Round-table conference improves reasoning via consensus among diverse llms. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, pages 7066–7085, Bangkok, Thailand. Association for Computational Linguistics.
- 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. 2021. Training verifiers to solve math word problems. arXiv preprint arXiv:2110.14168.
- Thomas M Cover and Joy A Thomas. 2006. *Elements of Information Theory*, 2nd edition. Wiley-Interscience, Hoboken, NJ, USA.
- Marquis de Condorcet. 1785. *Essai sur l’application de l’analyse à la probabilité des décisions rendues à la pluralité des voix*. De l’Imprimerie Royale, Paris.
- DeepSeek-AI. 2025. Deepseek-V3.2 release. <https://api-docs.deepseek.com/news/news251201/>.
- DeepSeek-AI. 2026. DeepSeek V4 preview release. <https://api-docs.deepseek.com/news/news260424/>.
- Yilun Du, Shuang Li, Antonio Torralba, Joshua B Tenenbaum, and Igor Mordatch. 2024. Improving factuality and reasoning in language models through multi-agent debate. In *Forty-first International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 11733–11763, Vienna, Austria. PMLR.
- Google DeepMind. 2026. Gemini 3.5 Flash model card. <https://deepmind.google/models/model-cards/gemini-3-5-flash/>.
- Taicheng Guo, Xiuying Chen, Yaqi Wang, Ruidi Chang, Shichao Pei, Nitesh V Chawla, Olaf Wiest, and Xiangliang Zhang. 2024. Large language model based multi-agents: A survey of progress and challenges. arXiv preprint arXiv:2402.01680.
- Wei He, Yueqing Sun, Hongyan Hao, Xueyuan Hao, Zhikang Xia, Qi Gu, Chengcheng Han, Dengchang Zhao, Hui Su, Kefeng Zhang, Man Gao, Xi Su, Xiaodong Cai, Xunliang Cai, Yu Yang, and Yunke Zhao. 2025. *Vitabench: Benchmarking llm agents with versatile interactive tasks in real-world applications*. arXiv preprint arXiv:2509.26490.
- Lu Hong and Scott E Page. 2004. Groups of diverse problem solvers can outperform groups of high-ability problem solvers. *Proceedings of the National Academy of Sciences*, 101(46):16385–16389.
- Dong Huang, Jie M. Zhang, Michael Luck, Qingwen Bu, Yuhao Qing, and Heming Cui. 2024. *Agent-coder: Multi-agent-based code generation with iterative testing and optimisation*. arXiv preprint arXiv:2312.13010.
- Can Jin, Hongwu Peng, Qixin Zhang, Yujin Tang, Dimitris N. Metaxas, and Tong Che. 2025a. *Two heads are better than one: Test-time scaling of multi-agent collaborative reasoning*. arXiv preprint arXiv:2504.09772.
- Weiqliang Jin, Hongyang Du, Biao Zhao, Xingwu Tian, Bohang Shi, and Guang Yang. 2025b. *A comprehensive survey on multi-agent cooperative decision-making: Scenarios, approaches, challenges and perspectives*. arXiv preprint arXiv:2503.13415.
- Kumar Joag-Dev and Frank Proschan. 1983. Negative association of random variables, with applications. *The Annals of Statistics*, 11(1):286–295.
- Zixuan Ke, Fangkai Jiao, Yifei Ming, Xuan-Phi Nguyen, Austin Xu, Do Xuan Long, Minzhi Li, Chengwei Qin, Peifeng Wang, Silvio Savarese, Caiming Xiong, and Shafiq Joty. 2025. *A survey of frontiers in llm reasoning: Inference scaling, learning to reason, and agentic systems*. arXiv preprint arXiv:2504.09037.
- Elliot Myunghoon Kim, Avi Garg, Kenny Peng, and Nikhil Garg. 2025. *Correlated errors in large language models*. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 30038–30066, Vancouver, Canada. PMLR.
- Anders Krogh and Jesper Vedelsby. 1995. Neural network ensembles, cross validation, and active learning. In *Advances in Neural Information Processing Systems*, volume 7, pages 231–238, Denver, CO, USA. MIT Press.
- Tian Liang, Zhiwei He, Wenxiang Jiao, Xing Wang, Yan Wang, Rui Wang, Yujiu Yang, Shuming Shi, and Zhaopeng Tu. 2024a. Encouraging divergent thinking in large language models through multi-agent debate. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 17889–17904, Miami, Florida, USA. Association for Computational Linguistics.
- Zhenwen Liang, Ye Liu, Tong Niu, Xiangliang Zhang, Yingbo Zhou, and Semih Yavuz. 2024b. *Improving llm reasoning through scaling inference computation with collaborative verification*. arXiv preprint arXiv:2410.05318.
- Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2024. Let’s verify step by step. In *The Twelfth International*

- Conference on Learning Representations*, Vienna, Austria. OpenReview.net.
- Mathematical Association of America. 2025. 2025 american invitational mathematics examination (AIME) i and ii. <https://maa.org/maa-invitational-competitions/>. Accessed: 2026-08-25.
- Dov Monderer and Lloyd S Shapley. 1996. Potential games. *Games and Economic Behavior*, 14(1):124–143.
- Ranjita Naik, Varun Chandrasekaran, Mert Yuksekgonul, Hamid Palangi, and Besmira Nushi. 2023. Diversity of thought improves reasoning abilities of large language models. In *NeurIPS 2023 Workshop on Instruction Tuning and Instruction Following*, New Orleans, LA, USA. NeurIPS.
- Scott E Page. 2007. *The Difference: How the Power of Diversity Creates Better Groups, Firms, Schools, and Societies*. Princeton University Press, Princeton, NJ, USA.
- Shishir G. Patil, Huanzhi Mao, Fanjia Yan, Charlie Cheng-Jie Ji, Vishnu Suresh, Ion Stoica, and Joseph E. Gonzalez. 2025. [The berkeley function calling leaderboard \(BFCL\): From tool use to agentic evaluation of large language models](#). In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 48371–48392, Vancouver, Canada. PMLR.
- Priya Pitre, Naren Ramakrishnan, and Xuan Wang. 2025. Consensagent: Towards efficient and effective consensus in multi-agent llm interactions through sycophancy mitigation. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 22112–22133, Vienna, Austria. Association for Computational Linguistics.
- Qwen Team. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.
- Qwen Team. 2026a. Qwen3.5: Towards native multi-modal agents. <https://qwen.ai/blog?id=qwen3.5>.
- Qwen Team. 2026b. Qwen3.6-35B-A3B: Agentic coding power, now open to all. <https://qwen.ai/blog?id=qwen3.6-35b-a3b>.
- Amrith Setlur, Chirag Nagpal, Adam Fisch, Xinyang Geng, Jacob Eisenstein, Rishabh Agarwal, Alekh Agarwal, Jonathan Berant, and Aviral Kumar. 2025. [Rewarding progress: Scaling automated process verifiers for llm reasoning](#). In *The Thirteenth International Conference on Learning Representations*, Singapore. OpenReview.net.
- Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. 2023. Reflexion: Language agents with verbal reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 36, pages 8634–8652, New Orleans, LA, USA. Curran Associates, Inc.
- Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. 2025. Scaling LLM test-time compute optimally can be more effective than scaling parameters for reasoning. In *The Thirteenth International Conference on Learning Representations*, Singapore. OpenReview.net.
- Haoran Sun, Yusen Wu, Yukun Cheng, and Xu Chu. 2025. [Game theory meets large language models: A systematic survey](#). In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence, IJCAI-25*, pages 10669–10677, Montreal, Canada. International Joint Conferences on Artificial Intelligence Organization. Survey Track.
- Khanh-Tung Tran, Dung Dao, Minh-Duong Nguyen, Quoc-Viet Pham, Barry O’Sullivan, and Hoang D. Nguyen. 2025. [Multi-agent collaboration mechanisms: A survey of llms](#). *arXiv preprint arXiv:2501.06322*.
- Naonori Ueda and Ryohei Nakano. 1996. [Generalization error of ensemble estimators](#). In *Proceedings of International Conference on Neural Networks (ICNN’96)*, volume 1, pages 90–95. IEEE.
- Junlin Wang, Jue Wang, Ben Athiwaratkun, Ce Zhang, and James Zou. 2025. [Mixture-of-agents enhances large language model capabilities](#). In *The Thirteenth International Conference on Learning Representations*, Singapore. OpenReview.net.
- Tianchun Wang, Yuanzhou Chen, Zichuan Liu, Jonathan Light, Weiyang Liu, Haifeng Chen, Xiang Zhang, and Wei Cheng. 2026. [On the effect of sampling diversity in scaling LLM inference](#). In *Proceedings of the 42nd Conference on Uncertainty in Artificial Intelligence*, volume 337 of *Proceedings of Machine Learning Research*, pages 7137–7167. PMLR.
- Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations*, Kigali, Rwanda. OpenReview.net.
- Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837, New Orleans, LA, USA. Curran Associates, Inc.
- Danny Wood, Tingting Mu, Andrew Webb, Henry Reeve, Mikel Lujan, and Gavin Brown. 2023. A unified theory of diversity in ensemble learning. *Journal of Machine Learning Research*, 24(359):1–49.
- Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L Griffiths, Yuan Cao, and Karthik

Narasimhan. 2023a. Tree of thoughts: Deliberate problem solving with large language models. In *Advances in Neural Information Processing Systems*, volume 36, pages 11809–11822, New Orleans, LA, USA. Curran Associates, Inc.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. 2023b. React: Synergizing reasoning and acting in language models. In *The Eleventh International Conference on Learning Representations*, Kigali, Rwanda. OpenReview.net.

Andy Zhou, Kai Yan, Michal Shlapentokh-Rothman, Haohan Wang, and Yu-Xiong Wang. 2024. Language agent tree search unifies reasoning, acting, and planning in language models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 62138–62160, Vienna, Austria. PMLR.

A Reproducibility Statement

We design the paper to be reproducible from the artifacts described below.

Code and data. The complete implementation—including all four DIANOIA phases, experimental drivers, diagnostic-metric scripts, and the Claude Code skill—is publicly available at <https://github.com/sachiel321/DIANOIA4MALLM>.

Quick experiential evaluation. The released repository additionally bundles a self-contained Claude Code skill (`skill/`, installed via `bash skill/install.sh`, invoked as `/dianoia <task>`) that lets users apply DIANOIA to users’ own everyday tasks from within any Claude Code session—no LLM client setup or API key required. The Python API above remains the reproducibility form factor for paper numbers; measured skill gains on Claude Opus 4.7 (§5.5) provide complementary cross-host evidence.

Models and inference. The base model for the core multi-agent comparisons is Qwen3-30B-A3B-Instruct-2507 with thinking mode disabled; reference models are Qwen3-235B-A22B-Instruct-2507 and DeepSeek-V3.2 (instruct mode, with thinking enabled only for AIME-2025 as documented in Appendix M). Cross-version experiments use Qwen3.6-35B-A3B (Qwen Team, 2026b); VitaBench alone uses Qwen3.5-397B-A17B (Qwen Team, 2026a); and the full MBPP family replications use model identifiers `deepseek-v4-flash` (DeepSeek-AI, 2026) and `Gemini-3.5-Flash` (Google DeepMind, 2026). All full-scale MBPP replications use the same 500-example split, prompts, evaluator, and Single/Best-of-3/DIANOIA method protocol. DIANOIA uses

$K=3$, $R=1$, and $S=3$. Sampling uses temperature 0.7 and `seed=42` where supported for stochastic methods, and temperature 0 for deterministic synthesis; unsupported seeds are recorded as such by the released drivers.

Prompt templates. The exact role prompts (Minimalist, Skeptic, Explorer), reviewer template, synthesizer template, and pseudo-verifier template are reproduced verbatim in Appendix M. The prompts are short enough to be transcribed without ambiguity.

Statistical reporting. Accuracy intervals use bootstrap resampling (1,000 iterations); the implementation is provided in the released code (`bootstrap_ci`). VitaBench pass-rate intervals are 95% Wilson score intervals. The AIME-2025 paired test (DIANOIA vs. ReConcile, $p=0.039$) is computed by the standard McNemar exact test on the 2×2 table of paired per-problem outcomes; a reference implementation is provided as the `analyze_rebuttal_metrics McNemar` entry point in the released code, which consumes any user-supplied pair of per-problem binary outcome vectors. Ablations and joint-dimension analyses are reported on the full benchmark splits.

VitaBench chronology. Before either Single or DIANOIA accuracy was measured, we recorded from the task structure—absence of a deterministic semantic verifier and long-horizon interactive reasoning—the hypothesis that information fidelity would be the primary bounded channel while exploration and repair retained headroom. We then ran the full held-out evaluation set ($N=400$) with semantic pseudo-verification and the benchmark pass-rate protocol. Qwen3.5-397B-A17B replaces the core Qwen3-30B-A3B only here because the latter produces a near-floor operating point; this base-model change and the non-oracle verifier limit the result to prospective empirical boundary evidence.

Reviewer-error estimation. The judgment unit is one reviewer–proposal pair. For each pair, we compare the binary reviewer judgment with ground-truth candidate quality, conditional on the evidence shown to that reviewer, and report the resulting aggregate per-judgment mismatch rate $\hat{\epsilon}_0$. This yields 5.8% on MBPP and 17.5% on GSM8K.

Licenses and intended use. All released artifacts (code, adapters, diagnostic metric scripts, and Claude Code skill) are distributed under the MIT License. The benchmarks we use are publicly available under their respective licenses (GSM8K,

Table 8: Controlled evidence-corruption stress test on the first 100 GSM8K examples.

Flip rate (%)	0	5	10	15	20	25	30
Accuracy (%)	92	93	92	91	93	91	88

Table 9: Mean input/output tokens and wall-clock latency per task under standard configurations with Qwen3.6-35B-A3B.

Benchmark	Single: in/out; sec	DIANOIA: in/out; sec
MBPP	232/145; 1.50	2,495/1,211; 12.02
BFCL-SP	693/34; 0.91	2,628/979; 9.93
GSM8K	71/368; 2.67	2,431/1,455; 14.87

AIME-2025, MBPP, BFCL-SP, and VitaBench) and are used for their intended research-evaluation purpose. Base LLMs are accessed via standard cloud inference APIs under their providers’ terms of service.

Compute infrastructure. All inference is conducted through cloud-based LLM inference APIs; no GPU training is performed. Token consumption is reported throughout as the primary compute proxy (see §7 for caveats about wall-clock latency).

Use of AI assistants. AI assistants were used during paper preparation for LaTeX editing, citation formatting, and code refinement of the released implementation. All conceptual contributions—the gain decomposition, the four-rule diagnostic protocol, the DIANOIA system design, theorems, proofs, and experimental designs—are the authors’ own. AI-generated code was reviewed and tested before inclusion.

B Evidence-Corruption Robustness

Accuracy remains within 91–93% through 25% corruption and falls to 88% at 30%. This intervention corrupts pseudo-verifier evidence before Review/Synthesize; it is therefore an evidence-robustness test, not a direct sweep of the conditional reviewer-error parameter ϵ_0 .

C Token and Latency Measurements

D Formal Definitions

This section provides complete formal definitions referenced in Section 3.

Definition D.1 (Reasoning Task). A reasoning task is defined by the tuple $(\mathcal{X}, \mathcal{T}, Q)$, where:

- $\mathcal{X}$ is the input space (e.g., natural language problem descriptions)

- $\mathcal{T}$ is the output space (e.g., solution trajectories or code)

- $Q : \mathcal{T} \times \mathcal{X} \rightarrow \{0, 1\}$ is a binary quality function where $Q(\tau, x) = 1$ if and only if τ is a correct solution for input x

Notation convention. For a fixed input x (every probability statement in the paper conditions on a sampled or fixed input from $\mathcal{X}$), we write $Q(\tau) := Q(\tau, x)$ as shorthand throughout the main text and proofs. All expectations $\mathbb{E}[Q(\cdot)]$ are understood as averages over both the input distribution and any stochasticity in the multi-agent procedure.

Definition D.2 (Multi-Agent System). A multi-agent reasoning system is characterized by $(K, \mathcal{P}, \mathcal{E}, f)$:

- $K \in \mathbb{N}^+$: number of proposing agents
- $\mathcal{P} = \{P_1, \dots, P_K\}$: generative distributions for each proposer
- $\mathcal{E} : \mathcal{T} \rightarrow \mathcal{R}$: environment executor mapping solutions to structured feedback
- $f : \mathcal{T}^K \times \mathcal{R}^K \rightarrow \mathcal{T}$: aggregation function synthesizing the final solution

Definition D.3 (Success Correlation). For agents i, j , define the success correlation coefficient:

$$\begin{aligned} \rho_{ij} &:= \text{Corr}(Q(\tau^{(i)}), Q(\tau^{(j)})) \\ &= \frac{\mathbb{E}[Q(\tau^{(i)})Q(\tau^{(j)})] - p^2}{p(1-p)} \end{aligned} \quad (4)$$

where $p = P(Q(\tau^{(k)}) = 1)$ is the marginal success probability.

Definition D.4 (Mutual Information). The mutual information between quality Q and signal S is:

$$\begin{aligned} I(Q; S) &:= H(Q) - H(Q | S) \\ &= \sum_{q,s} P(q, s) \log \frac{P(q,s)}{P(q)P(s)} \end{aligned} \quad (5)$$

where $H(Q) = -\sum_q P(q) \log P(q)$ is the entropy of Q . Note that $I(Q; S) \leq H(Q)$ with equality if and only if S completely determines Q .

Definition D.5 (Execution Feedback). Execution feedback $e = \mathcal{E}(\tau)$ is a structured report containing:

- $e.\text{success} \in \{0, 1\}$: whether execution completed without errors
- $e.\text{tests} \in \{0, 1\}^m$: pass/fail status for m test cases
- $e.\text{error} \in \Sigma^*$: error messages and stack traces (if any)

For code generation, quality is typically: $Q(\tau) := \mathbf{1}[e.\text{success} \wedge e.\text{tests} = 1]$.

Definition D.6 (Textual Feedback). *Textual feedback σ is an LLM-generated evaluation of solution quality, represented as a categorical variable (e.g., $\sigma \in \{\text{"correct"}, \text{"incorrect"}, \text{"uncertain"}\}$). For the binary case, define two directional error parameters:*

$$\alpha := P(\sigma = \text{"c"} \mid Q = 0), \quad (6)$$

$$\beta := P(\sigma = \text{"i"} \mid Q = 1), \quad (7)$$

where “c” and “i” abbreviate “correct” and “incorrect.”

Definition D.7 (Model-Based Pseudo-Verification Feedback). *For tasks without deterministic execution environments (e.g., mathematical reasoning), model-based pseudo-verification feedback $\sigma_v = \mathcal{V}(\tau, x)$ is a structured diagnostic report produced by a dedicated LLM evaluator that analyzes solution τ for input x without access to the ground truth. The report contains:*

- $\sigma_v.\text{is_correct} \in \{0, 1\}$: the verifier’s binary quality judgment
- $\sigma_v.\text{confidence} \in [0, 1]$: a calibrated confidence score
- $\sigma_v.\text{errors} \in \Sigma^*$: identified reasoning errors and diagnostic feedback

Since σ_v is generated by an LLM evaluator (rather than deterministic execution), it satisfies $I(Q; \sigma) \leq I(Q; \sigma_v) \leq I(Q; e)$: weakly more informative than unstructured self-critique σ (due to specialized evaluation prompts and structured output)—with strict middle inequality expected empirically when σ_v exploits its specialization—and weakly less informative than deterministic execution feedback e (due to the verifier’s own error rates).

Definition D.8 (Aggregation Efficiency / Effective System Efficiency). *The (effective) efficiency of aggregation function $f : \mathcal{T}^K \times \mathcal{R}^K \rightarrow \mathcal{T}$ is the ratio of expected output quality to expected coverage:*

$$\eta(f) := \frac{\mathbb{E}[Q(f(\tau, e))]}{\mathbb{E}[\max_k Q(\tau^{(k)})]} = \frac{\mathbb{E}[Q(f(\tau, e))]}{C_K}, \quad (8)$$

where the second equality uses $\mathbb{E}[\max_k Q(\tau^{(k)})] = P(\exists k : Q(\tau^{(k)}) = 1) = C_K$ for binary Q . For selector-only systems (final output is one of the original proposals), $\eta(f) \leq 1$ with $\eta(f) = 1$ indicating perfect oracle-quality selection; in this regime

$\eta(f) = P(Q^{\text{MAS}} = 1 \mid \exists k : Q^{(k)} = 1)$, the classical conditional selection accuracy. For repair-capable systems (synthesizer can produce outputs absent from the proposal set), $\eta(f)$ additionally absorbs any synthesis-rescue contribution and represents the broader effective success rate per unit coverage. When the dependence on the feedback signal s matters, we write $\eta(f, s)$ as in Decomposition 3.1; $\eta(f)$ is used when the signal is fixed or implicit.

E Formal Assumptions

We explicitly state all assumptions underlying our theoretical analysis. The first four assumptions (A1-A4) formalize standard properties of LLM-based multi-agent systems. We add two additional assumptions (A5-A6) specific to aggregation analysis.

Assumption 1 (Baseline Conditional Independence). *Given input x , each proposer generates output independently as the IID reference:*

$$P(\tau^{(1)}, \dots, \tau^{(K)} \mid x) = \prod_{k=1}^K P_k(\tau^{(k)} \mid x). \quad (9)$$

Deviations from independence—in particular, the negative success correlation induced by role specialization—are modelled explicitly via the pairwise correlation term $\bar{\rho}$ in Proposition 3.2 and discussed in App. G.7.

Justification. Proposers operate in parallel without communication during the generation phase, conditioned only on the input x and their assigned roles. The IID form serves as an analytical reference baseline for coverage analysis in Section 3; we do not claim that LLM agents are truly independent, and the empirical $\bar{\rho} \approx -0.15$ we measure (App. G.7) confirms that the real regime departs from A1 in a direction that strengthens, rather than weakens, the resulting coverage.

Assumption 2 (Baseline Competence). *There exists baseline success probability $p \in (0, 1)$ such that $P(Q(\tau^{(k)}) = 1) = p$ for all k in the absence of role specialization. Under role specialization, the per-role success rates $p_k := P(Q(\tau^{(k)}) = 1)$ need not all equal p ; we write $\bar{p} := \frac{1}{K} \sum_k p_k$ for the averaged marginal.*

Justification. This assumes agents have non-trivial but imperfect capability ($0 < p < 1$), matching empirical LLM performance on reasoning tasks. Role specialization redistributes (rather than degrades) marginal accuracy: roles are designed for

orthogonal failure modes, not for raw accuracy gains. We make no assumption about the magnitude or distribution of the per-role spread $\{p_k - \bar{p}\}$.

Convention for closed-form expressions: a three-level rigorous ladder. Subsequent propositions express coverage and pairwise-intersection terms through a single symbol p for tractability. The closed forms admit progressively tighter rigorous interpretations as we layer in structural assumptions:

Level 1 — Heterogeneous IID (under A1 alone): AM–GM lower bound. Under conditional independence (A1) with arbitrary $\{p_k\}$ of mean $\bar{p}$, the true coverage is $C_K^{\text{true}} = 1 - \prod_k (1 - p_k)$. By AM–GM, $\prod_k (1 - p_k) \leq (1 - \bar{p})^K$, hence

$$C_K^{\text{true}} \geq \underbrace{1 - (1 - \bar{p})^K}_{=: C_K^{\text{unif}}}. \quad (10)$$

The closed-form C_K^{unif} is therefore a guaranteed lower bound for *any* $\{p_k\}$ with mean $\bar{p}$. Equivalently, C_K^{true} is Schur-convex in $(p_1, \dots, p_K)$, so the uniform vector minimizes coverage subject to fixed mean. The strict positivity and monotonicity of $\mathcal{G}_{\text{explore}} = C_K^{\text{unif}} - \bar{p}$ thus carry over: under heterogeneity the true exploration gain is at least as large as the closed-form value.

Level 2 — Diversity-induced negative dependence (A1’): tighter IID-heterogeneous lower bound on *actual coverage*. The negative pairwise correlation $\bar{\rho} < 0$ we measure empirically (Section 5.8) is consistent with a stronger structural property: that role-specialized success indicators are *negatively associated* (Joag-Dev and Proschan, 1983) in the sense of the orthant inequality

$$P(\bigcap_k A_k^c \mid x) \leq \prod_k (1 - p_k), \quad (\text{A1}')$$

i.e., the joint probability of *all* agents failing is no larger than under independence. This is the canonical multivariate formalization of “mutual inhibition”: when role i ’s specialty matches the problem (raising $P(A_i)$), role j with an orthogonal specialty is correspondingly less likely to succeed, so the joint failure event is suppressed below the independence baseline. (A1’) is weaker than full negative association of Joag-Dev and Proschan (1983) but sufficient for our coverage bound and verifiable via the joint outcome statistics; it is implied by A1 (with equality) and strictly strengthens any pairwise-correlation-only condition.

Under (A1’) and arbitrary $\{p_k\}$ with mean $\bar{p}$, the coverage satisfies the two-step rigorous chain

$$\begin{aligned} C_K^{\text{true}} &= 1 - P(\bigcap_k A_k^c) \stackrel{(\text{A1}')}{\geq} 1 - \prod_k (1 - p_k) \\ &\stackrel{\text{AM-GM}}{\geq} 1 - (1 - \bar{p})^K. \end{aligned} \quad (11)$$

The first inequality is a bound on *actual coverage* (not an LB-on-LB), strictly tighter than Level 1: it quantifies the diversity bonus $\Delta_{\text{div}} := \prod_k (1 - p_k) - P(\bigcap_k A_k^c) \geq 0$ as the gap between independence and negatively-dependent joint failure. The middle floor $1 - \prod_k (1 - p_k)$ also strictly exceeds the uniform AM–GM floor whenever the $\{p_k\}$ are non-constant (the AM–GM inequality is strict unless all p_k are equal), so Level 2 strengthens Level 1 by two distinct mechanisms: (i) joint negative dependence among role-specialized indicators (the structural gain certified by (A1’)), and (ii) Schur convexity of $1 - \prod_k (1 - p_k)$ in $\{p_k\}$, which exploits any heterogeneity in per-role marginals. Both gains are quantitative, not cosmetic, and neither requires any empirical assumption about the magnitude of the heterogeneity.

Level 3 — Sharp closed form (K=2 or uniform). For $K = 2$, inclusion–exclusion terminates, and the diversity bonus admits a sharp closed form

$$\begin{aligned} \Delta_{\text{div}}^{K=2} &= -\rho_{12} \sqrt{p_1(1-p_1)p_2(1-p_2)} \\ &\geq 0 \quad \text{when } \rho_{12} \leq 0, \end{aligned} \quad (12)$$

reducing to $-\rho_{12} \bar{p}(1 - \bar{p})$ when $p_1 = p_2 = \bar{p}$. This is the form appearing in Proposition 3.2’s Bonferroni LB for general K . For $K \geq 3$ the bonus does not admit a closed form in $(\bar{p}, \bar{\rho})$ alone (it depends on higher-order joint structure), but Level 2’s rigorous chain (11) certifies its non-negativity and gives a strictly tighter quantitative floor than Level 1’s $1 - (1 - \bar{p})^K$. Proposition 3.2’s Bonferroni LB-vs-LB comparison should be read as a closed-form qualitative summary of this Level 3 effect, with Level 2 supplying the rigorous backbone under (A1’).

Summary. Level 1 transfers to heterogeneous $\{p_k\}$ with no extra assumption (AM–GM). Level 2 strictly tightens this under the structural assumption (A1’), which formalizes “role-specialization $\Rightarrow$ mutual inhibition” as a joint-distribution property and is consistent with all four empirically reported negative pairwise correlations. Level 3 provides the sharp closed form in the $K = 2$ or uniform regime.

None of the three levels depends on any specific bound on $\max_k |p_k - \bar{p}|$.

Assumption 3 (Finite Strategy Space). *The output space has finite cardinality: $|\mathcal{T}| < \infty$.*

Justification. While the space of all possible text sequences is technically infinite, practical constraints (maximum token length, finite vocabulary) render it finite. This assumption is essential for potential game convergence (Theorem 3.5b).

Assumption 4 (Execution Determinism). *For executable tasks, the environment executor is deterministic: given solution τ , the feedback $e = \mathcal{E}(\tau)$ is uniquely determined.*

Justification. Code execution, test evaluation, and MCP calls produce deterministic outputs (modulo explicit randomness in the code itself). This assumption directly applies to tasks with environmental verifiers (MBPP, BFCL-SP) and enables Theorem 3.5a (execution feedback as sufficient statistic). For mathematical reasoning tasks (GSM8K, AIME) where no deterministic executor exists, DIANOIA substitutes model-based pseudo-verification σ_v (Definition D.7); A4 does not hold in this regime, and theoretical guarantees apply only approximately (see Appendix G.7).

Assumption 5 (Reviewer Accuracy). *Each reviewer, when provided with objective evidence (execution feedback e or structured pseudo-verification feedback σ_v), correctly assesses solution quality with probability $1 - \epsilon_0$, where $\epsilon_0 \in (0, 0.5)$.*

Justification. LLMs are imperfect interpreters of feedback signals and can misclassify solution quality. The constraint $\epsilon_0 < 0.5$ ensures reviewers are better than random guessing.

Assumption 6 (Reviewer Independence). *Given the true quality $Q(\tau^{(k)})$ and the available evidence (execution feedback $e^{(k)}$ or pseudo-verification feedback $\sigma_v^{(k)}$), different reviewers' assessments are conditionally independent.*

Justification. Reviewers operate independently, each analyzing the same evidence. While they may share systematic biases (violating full independence), conditional on the objective evidence, remaining errors are largely uncorrelated. This assumption enables the exponential error reduction in Proposition 3.4b.

F Detailed Analysis and Examples

This section provides numerical examples, quantitative comparisons, and detailed case studies sup-

porting the main theoretical results.

F.1 Exploration Gain: Quantitative Analysis

Proposition F.1 (IID Exploration Gain). *With K independent agents each having success probability p , the exploration gain is:*

$$\mathcal{G}_{\text{explore}}^{\text{iid}}(K) = 1 - (1 - p)^K - p \quad (13)$$

This gain is strictly positive for $K \geq 2$ and monotonically increasing in K .

Numerical Illustration. For baseline success rate $p = 0.4$:

- $K = 2$: $\mathcal{G}_{\text{explore}} = 0.64 - 0.40 = 0.24$ (24% improvement)
- $K = 3$: $\mathcal{G}_{\text{explore}} = 0.784 - 0.40 = 0.384$ (38.4% improvement)
- $K = 5$: $\mathcal{G}_{\text{explore}} = 0.922 - 0.40 = 0.522$ (52.2% improvement)
- $K = 10$: $\mathcal{G}_{\text{explore}} = 0.994 - 0.40 = 0.594$ (59.4% improvement)

The marginal gain $\mathcal{G}_{\text{explore}}(K+1) - \mathcal{G}_{\text{explore}}(K)$ decreases as K grows, exhibiting diminishing returns characteristic of parallel sampling.

Role Diversity Enhancement. Consider three specialized roles:

- **Minimalist:** Prefers short, direct solutions. Fails on edge cases requiring extensive validation (failure mode: insufficient coverage).
- **Skeptic:** Adds redundant checks. Fails on time/resource-constrained problems (failure mode: over-engineering).
- **Explorer:** Seeks non-standard approaches. Fails on tasks with strict conventions (failure mode: excessive creativity).

These roles exhibit *orthogonal failure modes*. Empirically, on MBPP dataset, we observe pairwise correlations: $\rho_{12} \approx -0.15$, $\rho_{13} \approx -0.18$, $\rho_{23} \approx -0.12$, yielding $\bar{\rho} \approx -0.15 < 0$, confirming the diversity enhancement predicted by Proposition 3.2.

F.2 Information Gain: Quantitative Comparison

Example: Execution vs. Textual Feedback. Consider a code generation task with baseline success rate $p = 0.4$ (thus $H(Q) \approx 0.971$ bits).

Execution feedback e provides deterministic quality assessment (pass/fail tests). By Theorem 3.5a:

$$I(Q; e) = H(Q) \approx 0.971 \text{ bits (100\%)} \quad (14)$$

Textual feedback σ from LLM self-evaluation has directional error parameters $\alpha = 0.1$ and $\beta = 0.15$. Computing conditional entropies (writing “c” for “correct”):

$$P(\sigma=c) = 0.85 \cdot 0.4 + 0.1 \cdot 0.6 = 0.40 \quad (15)$$

$$P(Q=1 \mid \sigma=c) = \frac{0.85 \cdot 0.4}{0.40} = 0.85 \quad (16)$$

$$H(Q \mid \sigma=c) = H_b(0.85) \approx 0.610 \text{ bits} \quad (17)$$

Similarly, $P(Q=1 \mid \sigma=\text{“i”}) = \frac{0.15 \cdot 0.4}{0.60} = 0.10$ and $H(Q \mid \sigma=\text{“i”}) = H_b(0.10) \approx 0.469 \text{ bits}$. Averaging with $P(\sigma=c)=0.40$, $P(\sigma=i)=0.60$:

$$I(Q; \sigma) \approx 0.971 - 0.525 = 0.446 \text{ bits} \quad (18)$$

Information loss: Textual feedback retains only $\sim 46\%$ of available information, with $\sim 54\%$ lost due to LLM evaluator errors. Model-based pseudo-verification σ_v (Definition D.7)—which employs specialized evaluation prompts and structured diagnostic output—achieves intermediate fidelity: empirically, $I(Q; \sigma_v)/H(Q) \approx 0.75\text{--}0.85$ for well-calibrated verifier LLMs, occupying the space between pure self-critique ($\sim 46\%$) and deterministic execution (100%). This three-tier hierarchy directly informs DIANOIA’s feedback strategy (Remark 3.2).

F.3 Aggregation Gain: Voting Failure Analysis

Majority voting faces structural limitations that depend on the *answer space topology*, not merely on voter accuracy. We identify two failure modes and validate each with empirical results from Table 3.

Failure Mode 1: Answer Fragmentation. For code generation and function-calling tasks, multiple syntactically distinct implementations can be correct. Each independent agent samples from $P(\tau \mid x)$; when correct probability mass fragments across M distinct variants (each receiving $\sim p/M$), no single correct answer achieves plurality. Empirically, Self-Consistency’s voting yields negligible gains despite $K=5$ independent samples: only $+2.0\text{pp}$ on MBPP and $+0.5\text{pp}$ on BFCL-SP, confirming that exploration alone cannot compensate for ineffective aggregation.

Failure Mode 2: Low Per-Problem Success Rate. When per-problem success probability $p_i < 0.5$, majority voting *amplifies* errors. With K independent samples:

$$\begin{aligned} P(\text{vote OK}) &= \sum_{j=\lceil K/2 \rceil}^K \binom{K}{j} p_i^j (1-p_i)^{K-j} \\ &< p_i \quad \text{whenever } p_i < 0.5. \end{aligned} \quad (19)$$

On AIME-2025, where many competition-level problems have $p_i \ll 0.5$, Self-Consistency *degrades* by -13.3pp (56.7% vs. 70.0% single-model), precisely because voting turns the majority of incorrect samples into confident wrong selections.

DIANOIA’s Aggregation Efficiency. In contrast, DIANOIA’s evidence-based aggregation bypasses these structural requirements. With $K=3$ and reviewer error $\epsilon_0 = 0.2$, Proposition 3.4(b) (*Idealized Bound*, under A5–A6 and execution-grounded disambiguation) gives:

$$\eta(f_{\text{DIANOIA}}) \geq 1 - \epsilon_0^{K-1} = 1 - 0.2^2 = 0.96 \quad (20)$$

substantially above the random-selection envelope $1/K = 0.33$. DIANOIA maintains high aggregation efficiency regardless of answer space structure for two reasons: (1) quality signals are anchored to objective execution feedback rather than answer clustering, and (2) assigning differentiated roles to each proposer (e.g., Minimalist, Skeptic, Explorer) maximizes behavioral diversity, driving the pairwise success correlation toward negative values ($\bar{\rho} \approx -0.15$ empirically; see Appendix G.7) and thereby strengthening the independence condition that voting critically relies on yet cannot guarantee.

F.4 Case Studies: Existing Methods

We analyze three representative baselines — Self-Consistency, ReConcile, and MoA — through our gain decomposition framework, using empirical results from Table 3. All methods use Qwen3-30B-A3B as the base model.

Self-Consistency: Exploration Without Effective Aggregation. Self-Consistency (Wang et al., 2023) samples $K=5$ independent paths (activating $\mathcal{G}_{\text{explore}}$) but relies on majority voting (providing no $\mathcal{G}_{\text{info}}$ or $\mathcal{G}_{\text{agg}}$ advantage). Our framework predicts its total gain is bounded by $C_K \cdot \eta(f_{\text{vote}}) - p$, where $\eta(f_{\text{vote}})$ depends on the answer space structure:

- **GSM8K** ($+2.8\text{pp}$): Moderate gain. Unique numerical answers enable effective voting (η moderate), but the high baseline ($p \approx 0.84$) limits exploration headroom ($C_K - p$ small).
- **AIME-2025** (-13.3pp): *Negative* gain. Low per-problem success rates cause voting to amplify errors (Failure Mode 2 above), yielding $\eta(f_{\text{vote}}) \cdot C_K < p$.
- **MBPP** ($+2.0\text{pp}$) and **BFCL-SP** ($+0.5\text{pp}$): Negligible gains. Answer fragmentation across diverse

correct implementations renders $\eta(f_{\text{vote}}) \ll 1$, wasting the exploration benefit.

ReConcile: Discussion Improves Aggregation for Structured Reasoning. ReConcile (Chen et al., 2024) employs multi-round discussion among agents, introducing an information exchange mechanism that partially improves $\mathcal{G}_{\text{agg}}$ beyond naive voting. Our framework predicts discussion is effective when agents can verify reasoning through textual analysis, but limited when execution feedback is essential:

- **GSM8K** (+6.2pp): Strongest baseline gain. Step-by-step mathematical reasoning is amenable to textual verification; multi-round discussion enables agents to identify and correct reasoning errors, effectively improving aggregation quality.
- **AIME-2025** (± 0 pp): No gain. Competition-level difficulty exceeds the diagnostic capability of textual discussion; agents cannot reliably verify complex proofs through conversation alone, limiting $\mathcal{G}_{\text{info}}$.
- **MBPP** (+1.2pp) and **BFCL-SP** (+0.5pp): Minimal gains. Without deterministic execution feedback ($I(Q; e) = H(Q)$), discussion-based verification cannot substitute for program correctness signals, leaving $\mathcal{G}_{\text{info}}$ largely inactivated.

MoA: Layered Synthesis as Implicit Iterative Refinement. MoA (Wang et al., 2025) employs a multi-layer architecture where each layer’s agents synthesize outputs from the previous layer, implicitly activating both $\mathcal{G}_{\text{explore}}$ (multiple agents per layer) and $\mathcal{G}_{\text{agg}}$ (progressive synthesis). Our framework predicts that layered synthesis excels when iterative reasoning refinement is valuable, but achieves this at substantially higher compute cost than feedback-guided methods:

- **AIME-2025** (+16.7pp): Strongest baseline gain across all benchmarks. Multi-layer synthesis enables progressive deepening of mathematical arguments; each layer builds on and refines previous reasoning, effectively functioning as multi-round self-improvement that activates $\mathcal{G}_{\text{agg}}$ through iterative refinement.
- **GSM8K** (+3.5pp) and **BFCL-SP** (+4.0pp): Moderate gains consistent with synthesis-based aggregation improving over voting, but without targeted feedback signals.
- **MBPP: Scaling behavior reveals the compute-information tradeoff.** At standard configuration

(3 agents, 2 layers), MoA gains only +0.8pp. However, unlike Self-Consistency (saturates at $\sim 77.2\%$ beyond $K=15$) and ReConcile (saturates at $\sim 77.6\%$), MoA *continues to scale*: from 77.2% to 84.2% on the Pareto frontier experiment, a +7pp improvement that nearly matches DIANOIA’s 84.6%. This confirms that layered synthesis genuinely increases $\mathcal{G}_{\text{agg}}$ with additional capacity, bypassing the voting ceiling. The critical difference is *efficiency*: MoA requires ~ 7.7 M tokens to reach 84.2%, whereas DIANOIA achieves 84.6% at ~ 1.5 M tokens—a $5\times$ cost advantage — and at comparable budget (~ 7.4 M tokens) DIANOIA reaches 88.8%, a +4.6pp lead. This gap arises because MoA, lacking $\mathcal{G}_{\text{info}}$ from execution feedback, must compensate with brute-force layer stacking, while DIANOIA’s feedback signals make each token of compute substantially more informative.

AgentCoder: Iterative Repair Scales but Compresses Near the Verifier Ceiling. AgentCoder (Huang et al., 2024) employs a Programmer agent plus Test Designer and Test Executor agents with iterative refinement (up to 3 rounds), activating $\mathcal{G}_{\text{info}}$ (unit-test execution feedback) and $\mathcal{G}_{\text{agg}}$ (iterative repair) but lacking role-diverse exploration ($\mathcal{G}_{\text{explore}} \approx 0$, single Programmer agent). Our framework predicts that on execution-grounded tasks where the verifier signal is reliable, iterative repair alone should scale with single-pass quality; the additional contribution of role-diverse $\mathcal{G}_{\text{explore}}$ should be largest when the baseline is far below the verifier-determined ceiling, and should diminish as the base model strengthens. *Verification:* on MBPP, AgentCoder’s gain over the single-model baseline widens with the stronger base (+5.2pp on Qwen3-30B-A3B, 76.0 $\rightarrow$ 81.2%; +10.0pp on Qwen3.6-35B-A3B, 78.4 $\rightarrow$ 88.4%), confirming that execution iteration alone scales substantially. DIANOIA leads on both bases at point-estimate level (84.6 vs. 81.2 on Qwen3-30B; 88.8 vs. 88.4 on Qwen3.6, within CI overlap), with the DIANOIA – AgentCoder margin contracting from +3.4pp to +0.4pp as both methods approach the MBPP execution ceiling on the stronger base. This refines rather than contradicts the framework’s $\mathcal{G}_{\text{explore}}$ prescription: role diversity is a meaningful contributor in the mid-accuracy regime where coverage headroom is large, but its marginal value compresses once single-agent accuracy nears the verifier-determined ceiling—consistent with the operating-regime limi-

tation in §7.

These patterns are consistent with distinct, task-dependent roles for the three gain dimensions: methods activating only a subset achieve limited improvements. DIANOIA’s gains across the four benchmarks are consistent with jointly targeting exploration (role diversity), information (execution and pseudo-verification feedback), and aggregation (evidence-based cross-review); these comparisons do not by themselves establish causal independence among the dimensions.

G Complete Proofs

G.1 Proof of Decomposition 3.1: Gain Decomposition

We prove both the multiplicative identity and the subadditive upper bound. The decomposition draws on classical ideas from ensemble learning (Krogh and Vedelsby, 1995; Wood et al., 2023), the Condorcet jury theorem (de Condorcet, 1785), and the diversity prediction theorem in collective intelligence (Hong and Page, 2004; Page, 2007).

Proof. Part I: Coverage–Efficiency Factorization.

We establish $\mathbb{E}[Q(\tau^{\text{MAS}})] = C_K \cdot \eta(f, s)$ as a *definitional factorization*, consistent with Definition D.8: with $C_K := P(\bigvee_k Q(\tau^{(k)}) = 1)$ and $\eta(f, s) := \mathbb{E}[Q(\tau^{\text{MAS}})]/C_K$,

$$\mathbb{E}[Q(\tau^{\text{MAS}})] = C_K \cdot \eta(f, s) \quad (21)$$

holds tautologically; the content lies in the *interpretation* of the two factors and in the propositions that bound them.

Interpretation of η across system classes. Let $E := \{\exists k: Q(\tau^{(k)}) = 1\}$, so $P(E) = C_K$. By the law of total probability,

$$\begin{aligned} \mathbb{E}[Q(\tau^{\text{MAS}})] &= P(E) P(Q^{\text{MAS}} = 1 \mid E) \\ &\quad + P(\bar{E}) P(Q^{\text{MAS}} = 1 \mid \bar{E}), \end{aligned} \quad (22)$$

so dividing by C_K :

$$\begin{aligned} \eta(f, s) &= \underbrace{P(Q^{\text{MAS}} = 1 \mid E)}_{\text{conditional selection accuracy}} \\ &\quad + \underbrace{\frac{1-C_K}{C_K} \cdot r}_{\text{rescue contribution}}, \end{aligned} \quad (23)$$

where $r := P(Q^{\text{MAS}} = 1 \mid \bar{E})$ is the synthesis rescue probability.

For *selector-only systems*, the final output is constrained to be one of the original proposals, forcing $r = 0$; then $\eta = P(Q^{\text{MAS}} = 1 \mid E)$ recovers the classical conditional selection accuracy. For *repair-capable systems* like DIANOIA, the synthesizer can produce a correct answer even when no initial proposal is correct (empirically $r > 0$, contributing $\approx 5.5\%$ of correct outputs on MBPP, App. I); η then additionally absorbs this rescue contribution. In both cases, the factorization (21) is exact by definition, and downstream bounds on the conditional selection accuracy translate to lower bounds on η (since the rescue term is non-negative).

Part II: Non-negativity and Well-definedness of Gain Terms.

We verify that each gain term is well-defined and non-negative.

Exploration gain: By Assumption 1 (conditional independence) and 2 (baseline competence $p \in (0, 1)$):

$$C_K = 1 - (1 - p)^K \geq p \quad \text{for } K \geq 1 \quad (24)$$

with strict inequality for $K \geq 2$. Thus $\mathcal{G}_{\text{explore}} = C_K - p \geq 0$.

Information gain: Let $\eta^*(s) := \max_f \eta(f, s)$ denote the Bayes-optimal selection accuracy. Under complete and deterministic verification, Theorem 3.5a gives $I(Q; e) = H(Q)$, which means Q is fully determined by e and hence the Bayes-optimal classifier achieves zero error: $\eta^*(e) = 1$. For textual feedback σ with positive directional error parameters $\alpha, \beta > 0$, Proposition 3.3 gives $I(Q; \sigma) < H(Q)$, i.e., $H(Q|\sigma) > 0$; by Fano’s inequality (Lemma L.5, binary form), $P_e^*(\sigma)$ is strictly positive, hence $\eta^*(\sigma) < 1$. Therefore $\eta^*(e) = 1 > \eta^*(\sigma)$, and $\mathcal{G}_{\text{info}} = C_K[\eta^*(e) - \eta^*(\sigma)] \geq 0$.

Aggregation gain: The term $\mathcal{G}_{\text{aggr}} = C_K[\eta(f, s) - \eta(f_{\text{base}}, s)]$ is non-negative whenever f achieves at least the selection accuracy of the reference baseline f_{base} . This is *not* automatic—a poorly designed aggregation can underperform the baseline—and requires proof for each concrete system. The choice of f_{base} depends on the analytical purpose: for an absolute lower-envelope analysis we take f_{base} as uninformed selection (choosing one candidate uniformly at random, $\eta(f_{\text{base}}) = 1/K$); for empirical comparisons we use the dominant non-iterative selector for each task (majority vote for self-consistency, execution-grounded best-of- N for code), which gives a more stringent but

task-appropriate baseline. Either way, by Proposition 3.4(b) (*Idealized Bound*, under A5–A6 and execution-grounded disambiguation), DIANOIA achieves $\eta(f_{\text{DIANOIA}}, s) \geq 1 - \epsilon_0^{K-1}$; for the default configuration $K = 3, \epsilon_0 \leq 0.2$ this gives $\eta \geq 0.96 > 1/3$, establishing $\mathcal{G}_{\text{aggr}} > 0$ against the random-selection envelope. The proof and the role of the at-least-one-endorsement + execution-disambiguation rule are detailed in App. G (proof of Prop. 3.4b). Empirical comparison against majority-vote / best-of- N baselines is reported in §5.3–§5.4.

Part III: Subadditive Upper Bound.

From the multiplicative identity (Eq. 21):

$$\begin{aligned} \mathbb{E}[Q(\tau^{\text{MAS}})] - p &= C_K \cdot \eta(f, s) - p \\ &= (C_K - p) + C_K(\eta(f, s) - 1) \\ &= \mathcal{G}_{\text{explore}} + C_K(\eta(f, s) - 1) \end{aligned} \quad (25)$$

Since $\eta(f, s) \leq 1$ —which is equivalent to $\mathbb{E}[Q^{\text{MAS}}] \leq C_K$, i.e., the synthesized output is correct at most as often as the proposal ensemble contains a correct candidate—the second term is non-positive:

$$\begin{aligned} \mathbb{E}[Q(\tau^{\text{MAS}})] - p &\leq \mathcal{G}_{\text{explore}} \\ &\leq \mathcal{G}_{\text{explore}} + \mathcal{G}_{\text{info}} + \mathcal{G}_{\text{aggr}}, \end{aligned} \quad (26)$$

where the second inequality follows from non-negativity of

$\mathcal{G}_{\text{info}}, \mathcal{G}_{\text{aggr}} \geq 0$. The premise $\mathbb{E}[Q^{\text{MAS}}] \leq C_K$ holds automatically for selector-only systems (where $\eta = P(Q^{\text{MAS}} = 1 \mid E) \leq 1$); for repair-capable systems the premise requires that the rescue contribution does not exceed the selection-loss margin, $(1 - C_K) \cdot r \leq C_K \cdot (1 - P(Q^{\text{MAS}} = 1 \mid E))$, which is comfortably satisfied for DIANOIA across all tested benchmarks (e.g., on MBPP, $\eta \approx 0.86$ with the rescue contribution $\approx 5\%$ leaving a substantial margin below 1).

Part IV: Subadditivity Mechanism.

The strict inequality (subadditivity) arises because the three dimensions interact through the multiplicative structure. To see this concretely, we decompose the selection loss $C_K(1 - \eta(f, s))$:

$$\begin{aligned} C_K(1 - \eta(f, s)) &= \underbrace{C_K(1 - \eta^*(s))}_{\text{info loss}} \\ &\quad + \underbrace{C_K(\eta^*(s) - \eta(f, s))}_{\text{aggr loss}}. \end{aligned} \quad (27)$$

Both terms depend on C_K , coupling exploration with the other two dimensions. This means:

- When C_K is large (good exploration), both information and aggregation losses are amplified in absolute terms, making marginal improvements in these dimensions more impactful.
- When $\eta(f, s)$ is close to 1 (good information and aggregation), further improvements in exploration directly translate to performance gains.
- Conversely, improving exploration alone without adequate selection ($\eta \ll 1$) yields diminishing returns, as the additional coverage is wasted.

This multiplicative interaction is structurally analogous to the bias-variance-covariance decomposition in ensemble learning (Ueda and Nakano, 1996; Wood et al., 2023), where ensemble performance decomposes into $\text{Error} = \overline{\text{Bias}}^2 + \frac{1}{N} \overline{\text{Var}} + (1 - \frac{1}{N}) \overline{\text{Covar}}$, and optimizing one term (e.g., reducing bias) can increase another (variance), reflecting the same kind of cross-dimensional interaction. It also parallels the diversity prediction theorem (Hong and Page, 2004; Page, 2007): $\text{Collective Error} = \text{Individual Error} - \text{Diversity}$, where the diversity term captures the interaction between exploration breadth and aggregation quality. $\square$

Discussion: The Single-Dimension Trap and Joint Optimization. The multiplicative identity $\mathbb{E}[Q] = C_K \cdot \eta(f, s)$ has a sharp practical implication: *each factor acts as a ceiling on the other*. To make this concrete, decompose the selection accuracy into its two components (Part IV):

$$\eta(f, s) = \underbrace{\eta^*(s)}_{\text{information ceiling}} - \underbrace{[\eta^*(s) - \eta(f, s)]}_{\text{aggregation gap}} \quad (28)$$

The expected quality thus factors as $\mathbb{E}[Q] = C_K \cdot [\eta^*(s) - \Delta_{\text{aggr}}]$, where $\Delta_{\text{aggr}} := \eta^*(s) - \eta(f, s) \geq 0$ is the aggregation loss. Three failure modes emerge:

1. **Exploration-only scaling (high C_K , low η).** Self-Consistency increases C_K by sampling $K=5-33$ paths, but employs majority voting as aggregation (Δ_{aggr} large on code tasks). Empirically on MBPP, C_K grows from ~ 0.76 to ~ 0.95 , yet accuracy saturates at $\sim 77.2\%$ because the low η discounts the coverage gain: ΔC_K is multiplied by $\eta \ll 1$ (Eq. 25). The coverage improvement is “wasted” by poor selection.
2. **Selection-only optimization (high η , low C_K).** A single-agent system with perfect execution

feedback achieves $\eta^*(e) = 1$, but $C_1 = p$: the expected quality cannot exceed the baseline p regardless of how perfect the feedback is. The quality ceiling is set by coverage.

3. **Information without aggregation (high η^* , high Δ_{aggr}).** Even with execution feedback providing $\eta^*(e) = 1$ (maximal information), a naive aggregation function (e.g., random selection) leaves $\Delta_{\text{aggr}} = 1 - 1/K$, recovering only a fraction of the available information. The information is “collected but not utilized.”

DIANOIA’s joint optimization. DIANOIA addresses all three factors simultaneously: (1) role diversity drives C_K beyond the IID baseline via negative correlation ($\bar{\rho} < 0$, Proposition 3.2); (2) execution feedback and pseudo-verification raise the information ceiling $\eta^*(s) \rightarrow 1$ (Theorem 3.5a); (3) evidence-based cross-review and potential-game synthesis close the aggregation gap $\Delta_{\text{aggr}} \leq \epsilon_0^{K-1} \rightarrow 0$ (Proposition 3.4b). The multiplicative structure then becomes a *virtuous cycle*: high C_K amplifies the impact of reliable selection, while high η ensures that every unit of additional coverage translates into actual performance gain.

G.2 Proof of Proposition F.1

Proof. By Assumption 1:

$$\begin{aligned} P(\bigvee_k Q(\tau^{(k)})=1) &= 1 - \prod_k P(Q(\tau^{(k)})=0) \\ &= 1 - (1-p)^K. \end{aligned} \quad (29)$$

Thus $\mathcal{G}_{\text{explore}}(K) = [1 - (1-p)^K] - p$.

For $K \geq 2$: Let $g(K) = (1-p)^K + p$. Since $0 < 1-p < 1$, $(1-p)^K < 1-p$ for $K \geq 2$, so $g(K) < 1$ and $\mathcal{G}_{\text{explore}}(K) > 0$.

Monotonicity: $\frac{\partial}{\partial K} \mathcal{G}_{\text{explore}} = -(1-p)^K \ln(1-p) > 0$. $\square$

G.3 Proof of Proposition 3.2

Proof. Let $A_k = \{Q(\tau^{(k)}) = 1\}$ denote the event that agent k produces a correct solution, with $P(A_k) = p$ for all k . We derive the union probability $P(\bigcup_{k=1}^K A_k)$ using the inclusion-exclusion principle.

Step 1: Exact Formula via Inclusion-Exclusion. The inclusion-exclusion principle

gives:

$$\begin{aligned} P\left(\bigcup_{k=1}^K A_k\right) &= \sum_i P(A_i) - \sum_{i < j} P(A_i \cap A_j) \\ &\quad + \sum_{i < j < \ell} P(A_i \cap A_j \cap A_\ell) - \dots \end{aligned} \quad (30)$$

Step 2: Rigorous Lower Bound via Bonferroni Inequalities. Defining the standard partial sums

$$S_1 = \sum_i P(A_i), \quad S_2 = \sum_{i < j} P(A_i \cap A_j) \quad (31)$$

the Bonferroni inequality at order $m = 2$ (even) gives:

$$P\left(\bigcup_{k=1}^K A_k\right) \geq S_1 - S_2 \quad (32)$$

For agents with marginal success rate p and pairwise correlation ρ_{ij} , we have $P(A_i \cap A_j) = p^2 + \rho_{ij} \cdot p(1-p)$. Defining average pairwise correlation $\bar{\rho} = \frac{2}{K(K-1)} \sum_{i < j} \rho_{ij}$:

$$S_1 = Kp, \quad S_2 = \binom{K}{2} p^2 + \binom{K}{2} \bar{\rho} \cdot p(1-p) \quad (33)$$

Substituting into the Bonferroni bound:

$$\begin{aligned} P\left(\bigcup_{k=1}^K A_k\right) &\geq \underbrace{Kp - \binom{K}{2} p^2}_{\text{IID lower bound}} \\ &\quad - \underbrace{\binom{K}{2} \bar{\rho} \cdot p(1-p)}_{\text{diversity correction}}. \end{aligned} \quad (34)$$

Step 3: Diversity Gain via the Bonferroni Lower-Bound Comparison. The Bonferroni second-order bound depends only on the marginal p and average pairwise correlation $\bar{\rho}$. Writing $\text{LB}(\bar{\rho}) := S_1 - S_2 = Kp - \binom{K}{2} p^2 - \binom{K}{2} \bar{\rho} p(1-p)$ for the resulting coverage lower bound, we have $P(\bigcup_k A_k) \geq \text{LB}(\bar{\rho})$ for any joint distribution with marginals p and average correlation $\bar{\rho}$. Comparing the IID reference ($\bar{\rho} = 0$) to a diverse profile ($\bar{\rho} < 0$):

$$\text{LB}(\bar{\rho}) - \text{LB}(0) = -\binom{K}{2} \bar{\rho} p(1-p) > 0. \quad (35)$$

Defining $\mathcal{G}_{\text{explore}}^{\text{diverse}}$ and $\mathcal{G}_{\text{explore}}^{\text{iid}}$ as the gain-quantities derived from this common second-order lower

bound (i.e., $\text{LB}(\bar{p}) - p$ and $\text{LB}(0) - p$ respectively), Eq. (35) yields

$$\mathcal{G}_{\text{explore}}^{\text{diverse}} - \mathcal{G}_{\text{explore}}^{\text{iid}} = -\binom{K}{2} \bar{p} p (1 - p) > 0, \quad (36)$$

establishing that role-specialized diversity strictly increases the guaranteed second-order coverage lower bound, by an amount that grows quadratically in K and linearly in $|\bar{p}|$.

Comparison to the IID exact value. For the IID case the exact coverage $1 - (1 - p)^K$ is known and is larger than $\text{LB}(0) = Kp - \binom{K}{2} p^2$, so the strict LB-vs-LB comparison above does *not* by itself imply that the *actual* coverage under any diverse joint distribution exceeds the IID exact coverage—the Bonferroni LB on the diverse side is potentially loose. A stronger conclusion at the level of actual probabilities (rather than LBs) requires either (a) additional assumptions on the dependence structure (e.g., negative association), or (b) the $K = 2$ case where inclusion-exclusion truncates exactly. The main-text proposition states the LB comparison; downstream uses (Theorem 3.5c, App. G.1) treat $\text{LB}(\bar{p})$ as a coverage proxy with the caveat noted here. $\square$

G.4 Proof of Theorem 3.5 Part (a): Information Sufficiency

Proof. We prove that execution feedback e is a sufficient statistic for quality Q , achieving maximum mutual information $I(Q; e) = H(Q)$, under *complete and deterministic verification*—i.e., when (i) A4 (Execution Determinism) holds, and (ii) the feedback e contains enough fields to uniquely determine Q via a deterministic decoding function g (Def. D.5).

Step 1: Complete-and-Deterministic Relationship. By Assumption 4, execution is deterministic: given solution τ , the feedback $e = \mathcal{E}(\tau)$ is uniquely determined. By the definition of quality (Def. D.5, which specifies Q as a deterministic function of $e.\text{success}$ and $e.\text{tests}$):

$$Q(\tau) = g(\mathcal{E}(\tau)) = g(e) \quad (37)$$

where $g(e) := \mathbf{1}[e.\text{success} = 1 \wedge e.\text{tests} = 1]$ is deterministic. We refer to this combined condition—A4 plus the existence of such a g —as *complete and deterministic verification*. It is satisfied by MBPP and BFCL-SP under our experimental setup; cases where it fails (e.g., incomplete

test suites) reduce the regime to pseudo-verification (Remark 3.2).

Step 2: Conditional Entropy is Zero. Since Q is deterministic given e , knowing e completely determines Q :

$$\begin{aligned} H(Q | e) &= \sum_e P(e) H(Q | e=e) \\ &= \sum_e P(e) \cdot 0 = 0, \end{aligned} \quad (38)$$

since $H(Q | e=e) = 0$ because $P(Q=q | e=e) \in \{0, 1\}$.

Step 3: Mutual Information Equals Entropy. By definition:

$$I(Q; e) = H(Q) - H(Q | e) = H(Q). \quad (39)$$

This is the theoretical maximum, as $I(Q; S) \leq H(Q)$ for any signal S .

Step 4: Sufficiency Condition. We verify the Markov property $Q \perp \tau | e$. For any q, τ, e with $P(\tau, e) > 0$:

$$\begin{aligned} P(Q=q | \tau, e) &= P(Q=q | e) \\ &= \mathbf{1}[q = g(e)] \quad (\text{by Eq. (37)}) \end{aligned} \quad (40) \quad (41)$$

Thus e is sufficient: τ provides no added information about Q . $\square$

G.5 Proof of Proposition 3.3: Information Loss of Textual Feedback

Proof. We prove that textual feedback with non-zero error rates provides strictly less information: $I(Q; \sigma) < H(Q)$.

Step 1: Model Setup. Let $\sigma \in \{\text{“correct”}, \text{“incorrect”}\}$ denote the textual evaluation. Define:

$$\alpha := P(\sigma=c | Q=0) > 0, \quad (42)$$

$$\beta := P(\sigma=i | Q=1) > 0, \quad (43)$$

where “c”/“i” abbreviate “correct”/“incorrect”.

Step 2: Posterior Distributions. Let $p := P(Q=1)$. By Bayes’ theorem:

$$\begin{aligned} P(\sigma=c) &= P(\sigma=c | Q=1) P(Q=1) \\ &\quad + P(\sigma=c | Q=0) P(Q=0) \\ &= (1 - \beta)p + \alpha(1 - p). \end{aligned} \quad (44)$$

The posterior:

$$P(Q=1 | \sigma=c) = \frac{(1-\beta)p}{(1-\beta)p + \alpha(1-p)}. \quad (45)$$

Step 3: Conditional Entropy is Positive. Since $\alpha, \beta > 0$ and both are less than 1:

$$0 < P(Q=1 \mid \sigma=s) < 1, \quad s \in \{\text{cor.}, \text{inc.}\} \quad (46)$$

The conditional entropy for each outcome is strictly positive:

$$H(Q \mid \sigma = s) = H_b(P(Q=1 \mid \sigma=s)) > 0 \quad (47)$$

where $H_b(p) = -p \log p - (1-p) \log(1-p)$ is strictly positive for $p \in (0, 1)$.

Step 4: Overall Conditional Entropy.

$$\begin{aligned} H(Q \mid \sigma) &= \sum_s P(\sigma=s) H(Q \mid \sigma=s) \\ &= P(\sigma=\text{cor.}) H(Q \mid \sigma=\text{cor.}) \\ &\quad + P(\sigma=\text{inc.}) H(Q \mid \sigma=\text{inc.}) > 0. \end{aligned} \quad (48)$$

Step 5: Information is Suboptimal. By definition:

$$I(Q; \sigma) = H(Q) - H(Q \mid \sigma) < H(Q) \quad (49)$$

The information loss $H(Q \mid \sigma) > 0$ quantifies residual uncertainty. $\square$

The strict inequality $I(Q; \sigma) < H(Q)$ derived above is not merely a theoretical bound; it formalizes the *entropy loss* inherent in purely textual self-critique. This result establishes an information-theoretic imperative for DIANOIA’s feedback hierarchy:

- **Priority on Lossless Channels (Execution):** For domains where deterministic verification is feasible (e.g., code generation in MBPP, function calling in BFCL-SP), the framework motivates prioritizing execution feedback e . Under the complete-and-deterministic conditions of Theorem 3.5(a), this yields $I(Q; e) = H(Q)$, avoiding the residual uncertainty of textual critique. Relying only on LLM self-critique in these contexts would introduce avoidable information loss.
- **Principled Fallback for Lossy Channels:** For domains where execution is intractable (e.g., abstract reasoning in GSM8K), the system falls back to model-based verification σ_v . The framework explicitly acknowledges this as a *lossy* channel ($H(Q \mid \sigma_v) > 0$). This necessitates the stronger aggregation mechanisms in Phase 4 (Synthesis) to compensate for the imperfect information fidelity, ensuring robustness even when ground truth is inaccessible.

G.6 Proof of Theorem 3.5 Part (b): Convergence Guarantee

Proof. We prove that DIANOIA’s proposal-review mechanism is an exact potential game, guaranteeing finite-time convergence to a stable consensus.

Step 1: Game Specification. The game consists of:

- **Players:** K proposers indexed by $k \in \{1, \dots, K\}$
- **Strategy space:** Each player k chooses $\tau^{(k)} \in \mathcal{T}$
- **Utility function:** $u_k(\tau) = \max_{j=1, \dots, K} Q(\tau^{(j)}) + \lambda R_k(\tau^{(k)})$

where $\lambda > 0$ is the role preference weight and $R_k : \mathcal{T} \rightarrow \mathbb{R}$ encodes player k ’s role-specific preferences.

Step 2: Proposed Potential Function. Define:

$$\Phi(\tau) := \max_{k=1, \dots, K} Q(\tau^{(k)}) + \lambda \sum_{k=1}^K R_k(\tau^{(k)}) \quad (50)$$

Step 3: Verification of Potential Game Property. Consider player i unilaterally changing strategy from $\tau^{(i)}$ to $\tilde{\tau}^{(i)}$, while all other players maintain their strategies $\tau^{(-i)} := (\tau^{(j)})_{j \neq i}$. Define:

$$Q^* := \max_{j \neq i} Q(\tau^{(j)}) \quad (51)$$

the maximum quality among proposals from players other than i .

Utility Change for Player i :

$$\begin{aligned} \Delta u_i &= u_i(\tilde{\tau}^{(i)}, \tau^{(-i)}) - u_i(\tau^{(i)}, \tau^{(-i)}) \\ &= \left[\max(Q(\tilde{\tau}^{(i)}), Q^*) + \lambda R_i(\tilde{\tau}^{(i)}) \right] \\ &\quad - \left[\max(Q(\tau^{(i)}), Q^*) + \lambda R_i(\tau^{(i)}) \right] \\ &= \max(Q(\tilde{\tau}^{(i)}), Q^*) - \max(Q(\tau^{(i)}), Q^*) \\ &\quad + \lambda [R_i(\tilde{\tau}^{(i)}) - R_i(\tau^{(i)})] \end{aligned} \quad (52)$$

Potential Change:

$$\begin{aligned} \Delta \Phi &= \Phi(\tilde{\tau}^{(i)}, \tau^{(-i)}) - \Phi(\tau^{(i)}, \tau^{(-i)}) \\ &= \left[\max_{j=1, \dots, K} Q(\tau'^{(j)}) + \lambda \sum_{j=1}^K R_j(\tau'^{(j)}) \right] \\ &\quad - \left[\max_{j=1, \dots, K} Q(\tau^{(j)}) + \lambda \sum_{j=1}^K R_j(\tau^{(j)}) \right] \end{aligned} \quad (53)$$

where $\tau'^{(j)} = \tilde{\tau}^{(i)}$ if $j = i$, and $\tau'^{(j)} = \tau^{(j)}$ otherwise.

Since only player i changed strategy, the $\lambda \sum_{j \neq i} R_j(\tau^{(j)})$ terms cancel, leaving:

$$\Delta \Phi = \max(Q(\tilde{\tau}^{(i)}), Q^*) - \max(Q(\tau^{(i)}), Q^*) + \lambda [R_i(\tilde{\tau}^{(i)}) - R_i(\tau^{(i)})]. \quad (54)$$

Step 4: Exact Potential Property. Comparing Equations (52) and (54):

$$\Delta u_i = \Delta \Phi \quad \text{for all } i, \tau^{(i)}, \tilde{\tau}^{(i)}, \tau^{(-i)} \quad (55)$$

This confirms that Φ is an *exact potential function* (Monderer and Shapley, 1996).

Step 5: Convergence and Equilibrium Characterization. By Assumption 3, the strategy space $\mathcal{T}$ is finite. In any *best-response dynamics*, each refinement step strictly increases the potential function Φ . Since Φ is bounded and takes only finitely many values, this process must terminate in a finite number of steps T^* .

$$\Phi(\tau) \in [0 - \lambda K R_{\max}, 1 + \lambda K R_{\max}] \quad (56)$$

Moreover, Φ takes only finitely many distinct values (since $\mathcal{T}$ is finite).

In any *best-response dynamics* (where players sequentially play best responses), each move strictly increases the potential unless already at a best response:

$$\Phi(\tau_1, \dots, \tau_i^*, \dots, \tau_K) > \Phi(\tau_1, \dots, \tau_i, \dots, \tau_K) \quad (57)$$

where $\tau_i^* \in \arg \max_{\tilde{\tau}} u_i(\tilde{\tau}, \tau_{-i})$ is a best response for player i .

The termination point is, by definition, a **pure-strategy Nash equilibrium** τ^* . While a Nash equilibrium in general games does not necessarily imply global optimality, in our framework, the potential function Φ is explicitly constructed to align with the goal of maximizing reasoning quality Q . Thus, the resulting equilibrium represents a **principled consensus** that is locally optimal with respect to the evidence-based rewards and roles, providing a theoretical foundation for the stability of DIANOIA’s multi-agent synthesis.

Mapping to DIANOIA’s actual loop. The abstract best-response dynamics correspond to DIANOIA’s propose–execute–review–synthesize iteration as follows. Each synthesis iteration (Algorithm 1) takes the current profile τ , observes $\{e^{(k)}, v^{(k)}\}$, and produces a refined candidate. *If the synthesizer is restricted to improvement moves*—it only commits a refinement when the resulting

u_i strictly exceeds the current value (verified via execution and review evidence)—then each accepted step strictly increases Φ , and the finite-termination argument applies verbatim. Algorithm 1’s closed-loop validation (re-executing after each refinement and rolling back failed candidates) implements precisely this improvement-restricted regime under complete deterministic verification: an accepted synthesis output must exhibit $Q = 1$, monotonically tightening Φ from below toward the equilibrium.

Outside this regime (e.g., under pseudo-verification where execution feedback is noisy), the synthesizer’s improvement check may erroneously accept non-improving moves, and Φ -monotonicity can fail. Theorem 3.5(b) therefore applies *exactly* on complete-and-deterministic-verification tasks (MBPP, BFCL-SP) and only *approximately* on pseudo-verified tasks (GSM8K, AIME); we do not claim asymptotic convergence guarantees for the latter regime (see App. G.7.5 for the regime distinction). $\square$

G.7 Assumption Robustness Analysis

This section analyzes how our theoretical results degrade when key assumptions are relaxed, providing practitioners with guidance on when and where to expect deviations from the idealized analysis.

G.7.1 Role of Assumption A1 (Conditional Independence)

Assumption A1 serves as an *analytical starting point* to establish the IID baseline coverage $C_K^{\text{iid}} = 1 - (1 - p)^K$. It is *not* a claim that LLM agents are truly independent. Indeed, models from the same family trained on overlapping data inevitably share failure modes. The analytical progression is:

1. **Baseline under A1:** $C_K^{\text{iid}} = 1 - (1 - p)^K$ provides a clean reference point.
2. **Extension to correlated agents (Prop. 3.2):** For pairwise success correlation ρ_{ij} , the second-order Bonferroni bound gives:

$$C_K^{\text{diverse}} \geq C_K^{\text{iid}} - \binom{K}{2} \bar{\rho} \cdot p(1 - p) \quad (58)$$

where $\bar{\rho} = \frac{2}{K(K-1)} \sum_{i < j} \rho_{ij}$.

3. **Design implication:** Role diversity is the mechanism that drives $\bar{\rho} < 0$.

G.7.2 Empirical Correlation Evidence

On MBPP with the three DIANOIA roles (Minimalist, Skeptic, Explorer; see Section 4), we measure

pairwise success correlations between binary success indicators across the $N = 500$ MBPP problems (Pearson correlation; standard error $\approx 1/\sqrt{N} \approx 0.045$):

ρ_{12}	ρ_{13}	ρ_{23}
-0.15	-0.18	-0.12

The average correlation $\bar{\rho} \approx -0.15$ confirms that role diversity induces the negative correlation predicted by Proposition 3.2. The correlation improvement from Proposition 3.2 with $K = 3$, $p = 0.76$, and $\bar{\rho} = -0.15$ yields $C_3^{\text{diverse}} - C_3^{\text{iid}} \approx 3 \times 0.15 \times 0.76 \times 0.24 \approx 0.082$, corresponding to roughly +8 percentage points of additional coverage beyond the IID baseline.

G.7.3 Degradation under Positive Correlation

When $\bar{\rho} > 0$, the coverage bound *degrades* relative to the IID baseline:

$$C_K^{\text{correlated}} \leq C_K^{\text{iid}} + \binom{K}{2} |\bar{\rho}| \cdot p(1-p) \quad (59)$$

In the worst case ($\bar{\rho} \rightarrow 1$), all agents fail on the same inputs and $C_K \rightarrow p$, eliminating the exploration advantage entirely. This formalizes why naive scaling (adding identical agents without role differentiation) yields diminishing returns.

G.7.4 Reviewer Independence (A6)

Assumption A6 (conditional independence of reviewer assessments given Q and e) is approximate. In practice, reviewers sharing the same base model may exhibit correlated errors when interpreting ambiguous execution feedback. However, conditioning on execution feedback e substantially reduces the residual correlation: once objective evidence is available, remaining disagreements stem primarily from interpretation noise rather than systematic model biases. The exponential error reduction in Proposition 3.4(b) thus provides a conservative but qualitatively correct prediction.

G.7.5 Potential Game Idealization

Theorem 3.5(b) models the review-synthesis process as a potential game where utilities depend on the true quality $Q(\tau)$. In practice, agents do not observe Q directly but instead observe execution feedback $e = \mathcal{E}(\tau)$.

Exact case (complete verification): For tasks with deterministic, complete verifiers (MBPP with test suites, BFCL-SP with schema validation),

execution feedback e fully determines Q (Theorem 3.5a: $I(Q; e) = H(Q)$). In this regime, the potential game formulation is *exact*: the utility $u_i = Q(\tau^{(i)}) + \lambda \sum_j R_j(\tau^{(i)})$ can be computed from e without loss. This covers two of our four benchmarks.

Approximate case (pseudo-verification): For GSM8K/AIME, DIANOIA uses a dedicated LLM evaluator (no ground truth), producing σ_v . Writing $\hat{Q} := \hat{Q}(\sigma_v)$ for brevity, the utility gap satisfies

$$|u_i(Q) - u_i(\hat{Q})| \leq |Q - \hat{Q}| + \lambda \sum_j |R_j(Q) - R_j(\hat{Q})|. \quad (60)$$

The approximation error is bounded by the verifier’s misclassification rate—larger than deterministic execution but smaller than unstructured self-critique. Convergence remains valid as an approximation in this regime.

Inapplicable case (no verification): For open-domain tasks without any environmental feedback, agents can only rely on textual self-evaluation ($e = \sigma$). We do not claim DIANOIA’s theoretical guarantees extend to this setting (see Limitations).

G.8 Proof of Proposition 3.4 Part (b): DIANOIA Aggregation Efficiency

Proof. We prove $\eta(f_{\text{DIANOIA}}) \geq 1 - \epsilon_0^{K-1}$ under the following conditions: (i) Assumptions 5–6 (per-reviewer error rate $\epsilon_0 < 0.5$, conditional independence given quality and evidence), and (ii) execution-grounded disambiguation, where the synthesizer combines reviewer endorsements with execution feedback e to filter out incorrect candidates—realized exactly under complete and deterministic verification (Theorem 3.5a) and approximately under pseudo-verification (App. G.7.5). The main-text statement (Prop. 3.4b) labels this as an *Idealized Bound* to flag both the conditional-independence and execution-disambiguation idealizations.

Step 1: Setup. Suppose that among the K proposals $\{\tau^{(1)}, \dots, \tau^{(K)}\}$, at least one is correct. Without loss of generality, let:

$$\tau^{(i^*)} := \arg \max_{k \in \{1, \dots, K\}} Q(\tau^{(k)}) \quad (61)$$

be a correct proposal, so $Q(\tau^{(i^*)}) = 1$. The aggregation challenge is to correctly identify $\tau^{(i^*)}$ among all proposals.

Step 2: Reviewer Model. Each proposal $\tau^{(k)}$ is reviewed by $K - 1$ reviewers (all agents except the proposer). Let $v_j^{(k)} \in \{0, 1\}$ denote reviewer j 's binary assessment of proposal k :

$$v_j^{(k)} = \begin{cases} 1 & \text{if reviewer } j \text{ judges } \tau^{(k)} \text{ as correct} \\ 0 & \text{if reviewer } j \text{ judges } \tau^{(k)} \text{ as incorrect} \end{cases} \quad (62)$$

By Assumption 5, each reviewer has error rate ϵ_0 given $e^{(k)}$ and the true $Q(\tau^{(k)})$:

$$P(v_j^{(k)} = 0 \mid Q(\tau^{(k)}) = 1, e^{(k)}) = \epsilon_0, \quad (63)$$

$$P(v_j^{(k)} = 1 \mid Q(\tau^{(k)}) = 0, e^{(k)}) = \epsilon_0. \quad (64)$$

Step 3: Synthesizer Decision Rule. We model the synthesizer as following an *at-least-one-endorsement + execution-disambiguation* rule: a proposal $\tau^{(k)}$ is a candidate if (i) at least one reviewer endorses it ($\bigvee_{j \neq k} v_j^{(k)} = 1$), AND (ii) it passes execution-based validation $e^{(k)}$. Among candidates, the synthesizer outputs any one (the execution filter ensures correctness). Under complete and deterministic verification (Theorem 3.5a), execution exactly identifies Q : condition (ii) is satisfied iff $Q(\tau^{(k)}) = 1$. So a proposal is a candidate iff it is correct AND has at least one reviewer endorsement.

This rule formalizes the design intent of Algorithm 1: reviewer endorsements act as a *positive filter* that surfaces candidates for the synthesizer's attention, while execution feedback removes incorrectly endorsed candidates. The complementary failure mode—an incorrect proposal misleading the synthesizer despite failing execution—is excluded by condition (ii) under deterministic verification.

Step 4: Misidentification Probability. Under this rule, DIANOIA fails to surface a correct proposal iff $\tau^{(i^*)}$ is rejected by *every* reviewer (so condition (i) fails), since condition (ii) is automatic for a correct proposal under deterministic verification:

$$P(\text{fail to surface } \tau^{(i^*)}) \quad (65)$$

$$\begin{aligned} &= P\left(\bigwedge_{j \neq i^*} v_j^{(i^*)} = 0 \mid Q(\tau^{(i^*)}) = 1\right) \\ &= \prod_{j \neq i^*} P(v_j^{(i^*)} = 0 \mid Q(\tau^{(i^*)}) = 1, e^{(i^*)}) \\ &\stackrel{(A6)}{=} \epsilon_0^{K-1}. \end{aligned} \quad (66)$$

The conditional independence (A6) is invoked given the objective evidence $e^{(i^*)}$, consistent with the assumption's statement.

Translating to the efficiency definition (Def. D.8). Since $\eta(f, s) := \mathbb{E}[Q^{\text{MAS}}]/C_K$ and $\mathbb{E}[Q^{\text{MAS}}] \geq P(E) \cdot P(\text{correct} \mid E)$ (dropping the non-negative rescue term $P(\bar{E}) \cdot r$, cf. App. G.1 Part I), we obtain

$$\begin{aligned} \eta(f_{\text{DIANOIA}}) &\geq \frac{P(E) \cdot P(\text{correct} \mid E)}{C_K} \\ &= P(\text{correct} \mid E) \geq 1 - \epsilon_0^{K-1}. \end{aligned} \quad (67)$$

For DIANOIA's repair-capable synthesis the inequality is strict (with the rescue term contributing positively); selector-only systems achieve equality.

Step 5: Exponential Decay. The error decreases exponentially in the number of reviewers:

$$1 - \eta(f_{\text{DIANOIA}}) \leq \epsilon_0^{K-1}. \quad (68)$$

For $\epsilon_0 = 0.2$, $K = 3$: $1 - \eta \leq 0.04$ (4% error), demonstrating the robustness of evidence-based cross-review. Beyond complete verification: under pseudo-verification, execution filtering is imperfect (verifier precision/recall < 1); the bound then degrades by an additional factor tracking the verifier's error rates (see App. G.7.5). $\square$

Remark: Why $R=1$ reviewer is the recommended default. The bound above assumes $K-1$ reviewers per proposal. In practice, DIANOIA's standard configuration uses $R=1$ reviewer, yielding a selection error of ϵ_0 (e.g., 20% with $\epsilon_0 = 0.2$), compared to $\epsilon_0^2 = 0.04$ for $R=2$. Though increasing R provides exponential error reduction, the marginal benefit decays rapidly while the cost grows linearly: each additional reviewer must evaluate all K proposals, adding K review calls (each involving reading the proposal, analyzing execution feedback, and generating a structured assessment). For $K=3$ proposers, moving from $R=1$ to $R=2$ adds 3 review calls; moving to $R=3$ adds 6 total.

Empirically on MBPP, $R=1$ achieves 84.6% at $\sim 1.5\text{M}$ tokens, while $R=3$ reaches 85.8% at $\sim 4.0\text{M}$ tokens—a +1.2pp gain at $2.6\times$ the cost. Meanwhile, investing the same budget into more proposers ($K=6$, $R=1$) yields 87.4% at $\sim 5.6\text{M}$ tokens, a far larger gain. This reflects the multiplicative structure $\mathbb{E}[Q] = C_K \cdot \eta$: once η is sufficiently high (first reviewer already brings $\eta \geq 1 - \epsilon_0 = 0.8$), marginal improvements in η via additional reviewers are dominated by the coverage gain from additional proposers. In short, the first reviewer captures the lion's share of the aggregation gain; subsequent reviewers encounter steep

diminishing returns that are better spent expanding exploration.

G.9 Proof of Theorem 3.5 Part (c): Performance Bound

Proof. We derive a lower bound on DIANOIA’s expected quality by decomposing the analysis into exploration and aggregation phases.

Step 1: Decomposition by Conditioning. Let $E = \{\exists k : Q(\tau^{(k)}) = 1\}$ and $\bar{E}$ its complement. Then:

$$\mathbb{E}[Q(\tau^D)] = P(E) \cdot \mathbb{E}[Q(\tau^D) | E] + P(\bar{E}) \cdot \mathbb{E}[Q(\tau^D) | \bar{E}]. \quad (69)$$

Since $Q \in \{0, 1\}$, $\mathbb{E}[Q | E] = P(Q = 1 | E)$. To obtain a *conservative lower bound*, we drop the $\bar{E}$ term by setting $\mathbb{E}[Q | \bar{E}] \geq 0$. This corresponds to the modelling assumption that the synthesizer cannot recover from all-incorrect initial proposals; empirically this assumption is overly conservative—instrumented DIANOIA runs on MBPP show roughly 5.5% of correct outputs are *rescued* cases where every initial proposal failed execution but the synthesizer recovered a correct answer using reviewer hints (Appendix I). The bound below is therefore a true lower bound that practical synthesis can exceed via local repair.

Step 2: Exploration Phase Analysis. By Proposition F.1, with K IID proposers of success rate p :

$$P(E) = 1 - P(\bar{E}) = 1 - (1 - p)^K. \quad (70)$$

Step 3: Aggregation Phase Analysis. Given E , by Proposition 3.4(b) (*Idealized Bound*, under A5–A6 and execution-grounded disambiguation):

$$P(Q(\tau^D) = 1 | E) \geq 1 - \epsilon_0^{K-1}, \quad (71)$$

where $\epsilon_0 < 0.5$ is the per-reviewer error rate. High-fidelity information from cross-review with $K - 1$ reviewers yields exponentially decreasing misclassification, enabling high aggregation efficiency. The bound’s derivation and the role of the at-least-one-endorsement + execution-disambiguation rule are in App. G.

Step 4: Combining Both Phases. Substituting the bounds from Steps 2 and 3 into Equation (69):

$$\begin{aligned} \mathbb{E}[Q(\tau^D)] &= P(E) P(Q(\tau^D) = 1 | E) \\ &\geq [1 - (1 - p)^K] [1 - \epsilon_0^{K-1}]. \end{aligned} \quad (72)$$

Step 5: Comparison to Baseline. The gain over the single-agent baseline p is:

$$\begin{aligned} \mathbb{E}[Q] - p &\geq [1 - (1 - p)^K] [1 - \epsilon_0^{K-1}] - p \\ &= \underbrace{[1 - (1 - p)^K - p]}_{\mathcal{G}_{\text{explore}}} \\ &\quad - \underbrace{[1 - (1 - p)^K] \epsilon_0^{K-1}}_{\text{aggr loss}}. \end{aligned} \quad (73)$$

The first term is the exploration gain (Proposition F.1). The second term represents the small efficiency loss due to imperfect reviewers, which decreases exponentially in K .

Numerical Example. With $p = 0.4$, $K = 3$, $\epsilon_0 = 0.2$:

$$\begin{aligned} \mathbb{E}[Q(\tau^D)] &\geq [1 - 0.6^3] [1 - 0.2^2] \\ &= 0.784 \times 0.96 = 0.753, \end{aligned} \quad (74)$$

an 88% relative improvement over the baseline $((0.753 - 0.4)/0.4 = 0.8825)$. This bound is *signal-agnostic*: it treats ϵ_0 as a fixed parameter regardless of signal quality. Incorporating information gain from execution feedback yields a substantially tighter bound (Step 6).

Step 6: Information-Tightened Bound Under Complete and Deterministic Verification.

The bound in Step 4 does not exploit a key property of execution-based tasks: under Assumption 4 together with completeness of the verifier, execution feedback e perfectly reveals quality ($I(Q; e) = H(Q)$, Theorem 3.5a). This tightens the bound through two mechanisms that jointly incorporate $\mathcal{G}_{\text{info}}$:

(a) *Deterministic quality filtering.* Quality is directly observable from execution: $Q(\tau^{(k)}) = g(e^{(k)})$ for a deterministic function g . The system identifies the correct proposal set $\mathcal{C} = \{k : e^{(k)}.\text{pass} = 1\}$ with zero error. The reviewer’s role reduces from quality *judgment* to failure-mode *analysis*, making the effective ϵ_0 in this regime substantially lower than for tasks with noisy pseudo-verification.

(b) *Closed-loop verification via re-execution.* DIANOIA’s iterative synthesis (Algorithm 1) re-executes each synthesized output. Under A4, re-execution is a *perfect verification oracle*: if $Q(\tau^{\text{syn}}) = 0$, the failure is detected with certainty and the system iterates. To translate this into an exponential decay bound, we adopt an *additional assumption* not implied by A1–A6:

(A7, Synthesis-Iteration Independence; idealization). Given re-execution feedback, successive synthesis iterations fail with conditionally-independent failure probability $\leq \epsilon_0$ each.

Under (A7), the probability that all S iterations fail to produce or select a correct solution—despite $\mathcal{C} \neq \emptyset$ and deterministic verification—is bounded by ϵ_0^S . We flag (A7) as an idealization: in practice, the synthesizer is the same LLM evaluating the same evidence, so successive failures may correlate (e.g., the same misunderstanding recurring), making the ϵ_0^S rate an optimistic bound. The empirical diminishing-returns curve on S (App. H, single-dim scaling: $S = 1 \rightarrow 2 \rightarrow 3$ gives $83.6 \rightarrow 84.0 \rightarrow 84.6\%$ on MBPP) is consistent with sub-exponential improvement rather than the ϵ_0^S exponential rate, indicating (A7) is approximate in practice.

Combining cross-review and iterative synthesis, the system fails only if reviewers misidentify *and* all S synthesis iterations independently fail under deterministic verification:

$$\eta(f_{\text{DIANOIA}}, e) \geq 1 - \epsilon_0^{K-1+S} \quad (75)$$

The exponent $K - 1 + S$ reflects two independent sources of error reduction: reviewer redundancy ($K - 1$ cross-reviews) and synthesis iteration (S attempts with re-execution). The **information-tightened performance bound** is therefore:

$$\mathbb{E}[Q(\tau^D)] \geq [1 - (1 - p)^K] \cdot [1 - \epsilon_0^{K-1+S}] \quad (76)$$

Updated Numerical Example. With $p = 0.4$, $K = 3$, $S = 3$, $\epsilon_0 = 0.2$:

$$\begin{aligned} \mathbb{E}[Q(\tau^D)] &\geq [1 - 0.6^3][1 - 0.2^{2+3}] \\ &= 0.784 \times (1 - 0.00032) \approx 0.784 \end{aligned} \quad (77)$$

Compared to the signal-agnostic bound of 0.753 (Step 5), the information-tightened bound approaches the **oracle** $C_K = 0.784$ —confirming that deterministic execution feedback effectively closes the aggregation gap.

Remark (Non-execution tasks). For tasks without deterministic verification (e.g., GSM8K, AIME), DIANOIA employs model-based pseudo-verification σ_v where $I(Q; \sigma_v) < H(Q)$ (Remark 3.2). Re-execution cannot deterministically

verify quality in this regime, so the iterative tightening does not apply, and the general bound $1 - \epsilon_0^{K-1}$ from Step 4 remains operative. This asymmetry precisely explains the empirical observation (Table 3) that DIANOIA’s largest gains emerge on execution-intensive tasks (MBPP +8.6pp, BFCL-SP +10.5pp) versus modest gains on math tasks (GSM8K +7.5pp). $\square$

H Scaling and Ablation Tables

This appendix collects the three ablation tables referenced in Section 5.3.

Table 10: Single-dimension scaling on MBPP. Each group varies one parameter while fixing the others.

Exploration ($R=1, S=3$)		Information ($K=3, S=3$)		Aggregation ($K=3, R=1$)	
K	Acc	R	Acc	S	Acc
1	81.0%	0	83.2%	1	83.6%
2	82.8%	1	84.6%	2	84.0%
3	84.6%	3	85.8%	3	84.6%

Table 11 reports the analogous joint-dimension decomposition on BFCL-SP, yielding synergy coefficient $\gamma = 7.5/8.8 = 0.852$. Together with the MBPP coefficient ($\gamma = 0.88$), this is a post-run descriptive check of subadditivity (Remark 3.1).

Table 11: Joint-dimension scaling on BFCL-SP. Single-dimension contributions ($+5.0 + 2.5 + 1.3 = +8.8\text{pp}$) exceed the joint $+7.5\text{pp}$ realized by DIANOIA-full, giving $\gamma = 0.852$.

Config	K	R	S	Acc	Activates
Baseline	1	0	0	84.8%	—
Explore-only	3	0	1	89.8%	$\mathcal{G}_{\text{explore}}$
Info-only	1	1	1	87.3%	$\mathcal{G}_{\text{info}}$
Aggr-only	1	0	1	86.1%	$\mathcal{G}_{\text{aggr}}$
DIANOIA-full	3	1	3	92.3%	All three

Table 12: Role-design ablation on MBPP ($K=3, R=1, S=3$). Role-specialised prompting outperforms same-prompt + temperature noise; alternative structured role choices remain competitive: *any* structured role design beats unstructured sampling, but the choice within the structured family has overlapping CIs.

Role configuration	Accuracy
DIANOIA (Min. + Skep. + Expl.)	84.6% [81.4, 87.0]
Same prompt + temperature	83.2% [79.8, 86.4]
Two roles (Min. + Skep.)	81.4% [78.0, 84.8]
Two roles (Min. + Expl.)	84.4% [81.0, 87.4]
Two roles (Skep. + Expl.)	82.8% [79.4, 86.2]
Alternative role triplet	83.4% [79.8, 86.6]

I Synthesis Breakdown

To quantify how much of DIANOIA’s gain is selection vs. synthesis, we instrumented final correct outputs on MBPP. Among DIANOIA’s correct predictions: 16.4% exactly match an already-correct initial proposal (pure selection); 78.1% are newly synthesized correct answers although at least one initial proposal was correct (selection + repair); 5.5% are *rescued* cases where all initial proposals are incorrect but the final synthesized answer is correct (genuine repair beyond selection). The rescue rate is small in absolute terms, but it shows that the bound in Theorem 3.5c is conservative: practical synthesis can occasionally exceed the conditional event $\{\exists k: Q(\tau^{(k)}) = 1\}$, and our theoretical analysis is best read as a lower-bound design lens rather than a complete model of every Phase-4 event.

J Detailed Predictive-Verification Narratives

We expand the four predictions referenced in Section 5.7 into prediction–verification narratives, summarised in Table 13. The format throughout is: state the prediction the framework makes *before* the experiment, then verify against the data.

Table 13: Cross-task predictions derived from the decomposition (*ex ante*, before comparing methods) and their empirical verification.

Setting	Prediction	Verification
P1: exec. ground (MBPP, BFCL-SP)	All 3 dims active; DI-+6.6 / +3.5pp; SC & ANOIA’s largest gain; ReConcile <78% any exec.-blind methods budget; MoA matches saturate.	only at $5\times$ tokens.
P2: pseudo-verified, high p (GSM8K)	Two channels bottle-necked; smallest gain.	pseudo-verifier prec/rec 94%/87% confirms $\hat{\eta}^*(\sigma_v) < 1$.
P3: competition math (AIME-25)	Exploration is the bottleneck; voting harms when $p_i < 0.5$.	SC degrades –13.3pp; DIANOIA 93.3%; McNemar $p=0.039$ vs ReConcile.
P4: Best-of- N ceiling	BoN activates Explore+Info; to DIANOIA isolates $\mathcal{G}_{aggr}$.	only BoN-3 78.4% vs DIANOIA 84.6%; gap +6.2pp residual.

P1 (execution-grounded tasks). On **MBPP** and **BFCL-SP**, deterministic execution lifts the information ceiling to its maximum ($I(Q; e) = H(Q)$, Remark 3.2), so all three dimensions are simultaneously active. The framework predicts: (a) DIANOIA should show its largest gains here (joint optimization is most leveraged), (b) execution-blind methods (Self-Consistency, ReConcile) should saturate early regardless of token budget, and (c) methods that activate $\mathcal{G}_{aggr}$ but not $\mathcal{G}_{info}$ (MoA)

should reach high accuracy only via expensive layer stacking. *Verification:* DIANOIA’s gains over the strongest baseline are indeed +6.6pp on MBPP and +3.5pp on BFCL-SP—the largest of the four benchmarks. Self-Consistency and ReConcile plateau below 78% on MBPP at any token budget (Figure 2). MoA matches DIANOIA’s ceiling only at $\sim 5\times$ more tokens. All three sub-predictions hold.

P2 (high-baseline pseudo-verified tasks). On **GSM8K**, the framework predicts a small DIANOIA gain because two channels are simultaneously bottlenecked: $\mathcal{G}_{info}$ is capped by pseudo-verification fidelity ($I(Q; \sigma_v) < I(Q; e)$), and $\mathcal{G}_{explore}$ is capped by the high baseline accuracy ($p \approx 0.84$, leaving little coverage headroom). *Verification:* DIANOIA’s gain is +1.3pp on GSM8K—the smallest across benchmarks, as predicted. The candidate-level pseudo-verifier on GSM8K attains 94.0% precision and 86.6% recall, directly quantifying the information loss relative to deterministic execution and confirming that $\hat{\eta}^*(\sigma_v) < 1$.

P3 (low-baseline competition tasks). On **AIME-2025** with low per-problem success rate, the framework predicts: (a) exploration is the dominant bottleneck (most problems have $p_i \ll 0.5$), (b) majority voting will *harm* accuracy because $P(\text{vote correct}) < p_i$ when $p_i < 0.5$ (Failure Mode 2 in Appendix F), and (c) evidence-based synthesis should still recover via DIANOIA’s aggregation channel. *Verification:* Self-Consistency degrades by –13.3pp (56.7% vs. 70.0% single-model), exactly the predicted voting amplification. DIANOIA achieves the highest observed accuracy (93.3%); a paired McNemar exact test on per-problem outcomes gives $p=0.039$ vs. ReConcile (8/1 discordant in DIANOIA’s favour), supporting the gain despite $N=30$.

P4 (Best-of- N ceiling). The framework predicts that Best-of- N with execution-grounded selection will close the gap to DIANOIA by exactly $\mathcal{G}_{aggr}$, because Best-of- N activates $\mathcal{G}_{explore}$ and $\mathcal{G}_{info}$ but not $\mathcal{G}_{aggr}$ (no role diversity, no cross-review, no synthesis). *Verification:* On MBPP, Best-of-3 reaches 78.4% vs. DIANOIA’s 84.6%—a +6.2pp residual that isolates $\mathcal{G}_{aggr}$ in DIANOIA, consistent with the predicted gap.

K Computational Complexity Analysis

Let C_{prop} , C_{rev} , C_{syn} denote token costs for proposal, review, synthesis, and let R denote the num-

ber of reviewers per proposal ($R=1$ in DIANOIA's default configuration; $R=K-1$ corresponds to the idealized bound regime of Prop. 3.4b—see the Remark in App. G.7.5).

Total cost: $C = K \cdot C_{\text{prop}} + K \cdot R \cdot C_{\text{rev}} + T \cdot C_{\text{syn}}$

For DIANOIA's default ($K=3$, $R=1$, $T=3$): $C = 3C_{\text{prop}} + 3C_{\text{rev}} + 3C_{\text{syn}}$. The idealized regime with peer review ($R=K-1=2$) gives $C = 3C_{\text{prop}} + 6C_{\text{rev}} + 3C_{\text{syn}}$.

Parallelization: Propose and review phases are fully parallel. Wall-clock latency:

$$L = L_{\text{prop}} + L_{\text{rev}} + T \cdot L_{\text{syn}} + (K+T) \cdot L_{\text{exec}} \quad (78)$$

Table 14: Complexity comparison. DIANOIA's default $R=1$ gives $O(K)$ calls; the bracketed entry is the $R=K-1$ idealized regime.

Method	Calls	Depth
Self-Consistency	K	1
MoA	$K \cdot L$	L
Debate	$K \cdot R$	R
DIANOIA	$O(K \cdot R + T)$ [$O(K^2)$]	$O(T)$

L Additional Technical Lemmas

This section collects fundamental results from information theory, probability theory, and game theory that underpin our main theorems.

L.1 Information-Theoretic Lemmas

Lemma L.1 (Properties of Shannon Entropy). *Let X be a discrete random variable taking values in $\mathcal{X}$. The Shannon entropy $H(X) = -\sum_{x \in \mathcal{X}} P(X=x) \log P(X=x)$ satisfies:*

- (a) **Non-negativity:** $H(X) \geq 0$, with equality if and only if X is deterministic (concentrated on a single value).
- (b) **Maximum entropy:** $H(X) \leq \log |\mathcal{X}|$, with equality if and only if X is uniformly distributed over $\mathcal{X}$.
- (c) **Conditioning reduces entropy:** $H(X|Y) \leq H(X)$, with equality if and only if X and Y are independent.

Lemma L.2 (Binary Entropy Function). *The binary entropy function $H_b(p) := -p \log p - (1-p) \log(1-p)$ for $p \in [0, 1]$ satisfies:*

- (a) $H_b(p) \geq 0$ with $H_b(0) = H_b(1) = 0$.
- (b) $H_b(p) = H_b(1-p)$ (symmetry).
- (c) $H_b(p)$ is concave and achieves maximum $H_b(0.5) = 1$ bit at $p = 0.5$.

Lemma L.3 (Mutual Information Chain Rule). *For random variables X, Y, Z :*

$$I(X; Y, Z) = I(X; Y) + I(X; Z | Y) \quad (79)$$

where $I(X; Z | Y) := H(Z | Y) - H(Z | X, Y)$ is the conditional mutual information.

Lemma L.4 (Data Processing Inequality). *If $X \rightarrow Y \rightarrow Z$ forms a Markov chain (i.e., Z is conditionally independent of X given Y), then:*

$$I(X; Z) \leq I(X; Y) \quad (80)$$

Equality holds if and only if $Y \rightarrow Z$ is an invertible transformation.

Proof. By the chain rule:

$$I(X; Y, Z) = I(X; Y) + I(X; Z | Y) \quad (81)$$

$$= I(X; Z) + I(X; Y | Z) \quad (82)$$

Since $X \rightarrow Y \rightarrow Z$ is Markov, $I(X; Z | Y) = 0$. Also, $I(X; Y | Z) \geq 0$. Thus $I(X; Z) = I(X; Y) - I(X; Y | Z) \leq I(X; Y)$. $\square$

Lemma L.5 (Fano's Inequality for Binary Selection). *Let Q be a random variable taking values in a finite set $\mathcal{Q}$, S an arbitrary signal, and $\hat{Q}(S)$ any decoder. Let $P_e := P(\hat{Q}(S) \neq Q)$ and $P_e^* := \min_{\hat{Q}} P_e$ the Bayes-optimal error. The standard Fano inequality (Cover and Thomas, 2006) reads:*

$$H(Q | S) \leq H_b(P_e) + P_e \cdot \log_2(|\mathcal{Q}| - 1), \quad (83)$$

where $H_b(p) := -p \log_2 p - (1-p) \log_2(1-p)$ is the binary entropy.

Binary specialization. For $Q \in \{0, 1\}$ ($|\mathcal{Q}| = 2$), the second term vanishes ($\log_2 1 = 0$), so

$$H(Q | S) \leq H_b(P_e^*). \quad (84)$$

Inverting (using H_b monotone on $[0, 1/2]$) yields a lower bound on P_e^* when $H(Q|S)$ is small, and equivalently an upper bound on the Bayes-optimal selection accuracy $\eta^*(S) := 1 - P_e^*$:

$$\eta^*(S) \leq 1 - H_b^{-1}(H(Q | S)), \quad (85)$$

where H_b^{-1} denotes the inverse of H_b restricted to $[0, 1/2]$ (the relevant branch for $P_e^* \leq 1/2$).

Proof. Eq. (83) is the textbook statement of Fano's inequality (e.g., Cover & Thomas Thm. 2.10.1). The binary specialization (84) follows from $|\mathcal{Q}| = 2 \Rightarrow \log_2(|\mathcal{Q}| - 1) = 0$.

The monotonicity-based inversion to (85) is standard. $\square$

Corollary (Connection to our framework). When $I(Q; e) > I(Q; \sigma)$ (Proposition 3.3), $H(Q|e) < H(Q|\sigma)$, so the binary Fano inequality (84) gives $H_b(P_e^*(e)) \leq H(Q|e) < H(Q|\sigma) \leq H_b(P_e^*(\sigma))$. On the regime $P_e^* \leq 1/2$ (which holds whenever the decoder is better than random), H_b is strictly increasing, so $P_e^*(e) < P_e^*(\sigma)$ and hence $\eta^*(e) > \eta^*(\sigma)$: higher mutual information strictly tightens the achievable selection-error bound.

L.2 Probabilistic Lemmas

Lemma L.6 (Bonferroni Inequality). *For events $A_1, \dots, A_K$:*

$$P\left(\bigcup_{k=1}^K A_k\right) \geq \sum_{k=1}^K P(A_k) - \sum_{1 \leq i < j \leq K} P(A_i \cap A_j) \quad (86)$$

This provides a lower bound on the union probability accounting for pairwise overlaps, used in the proof of Proposition 3.2.

Lemma L.7 (Union Bound). *For events $A_1, \dots, A_K$:*

$$P\left(\bigcup_{k=1}^K A_k\right) \leq \sum_{k=1}^K P(A_k) \quad (87)$$

L.3 Game-Theoretic Lemmas

Lemma L.8 (Characterization of Potential Games). *A finite game $\Gamma = (N, \{S_i\}, \{u_i\})$ is an exact potential game iff for all players i, j and all profiles s :*

$$\begin{aligned} u_i(s'_i, s_{-i}) - u_i(s_i, s_{-i}) \\ = u_j(s'_j, s_{-j}) - u_j(s_j, s_{-j}) \end{aligned} \quad (88)$$

whenever the two deviations affect the same set of players' payoffs.

Lemma L.9 (Finite Improvement Property). *In any finite potential game, best-response dynamics converge to a pure-strategy Nash equilibrium in finitely many steps.*

Proof. By definition of a potential game, each best-response move strictly increases the potential function Φ . Since the strategy space is finite, Φ takes finitely many values. A strictly increasing sequence over a finite set must terminate, at which point no player can improve by unilateral deviation—i.e., a Nash equilibrium. $\square$

Algorithm 1 DIANOIA Synthesis with Closed-Loop Validation

Require: Problem x , proposals $\{\tau^{(k)}\}$, reports $\{e^{(k)}\}$, reviews $\{v^{(k)}\}$, max iterations T

- 1: $\tau^* \leftarrow \mathcal{S}(\{\tau^{(k)}, e^{(k)}, v^{(k)}\}, x)$ // *init synthesis*
- 2: **for** $t = 1$ to T **do**
- 3: $e^* \leftarrow \mathcal{E}(\tau^*)$ // *execute*
- 4: **if** $e^*.success = \text{True}$ **then**
- 5: **return** τ^* // *pass*
- 6: **end if**
- 7: $\tau^* \leftarrow \mathcal{S}(\tau^*, e^*, \{\tau^{(k)}, e^{(k)}, v^{(k)}\}, x)$ // *refine*
- 8: **end for**
- 9: **return** τ^* // *best-effort*

Lemma L.10 (Existence of Nash Equilibrium in Potential Games). *Every finite potential game possesses at least one pure-strategy Nash equilibrium, namely any strategy profile that maximizes the potential function.*

M Experimental Protocol Details

This section provides comprehensive details on experimental setup for reproducibility.

Model Configuration Rationale. The controlled setting (zero-shot prompting, no extended reasoning mode) ensures fair comparison across methods by eliminating confounding variables from model-specific features. As a consequence, the single-model references achieve lower accuracy than officially reported benchmarks (which typically use few-shot prompting and extended reasoning), but this setting is essential for measuring the pure gains attributable to multi-agent architectures rather than orthogonal enhancements.

Exception: DeepSeek-V3.2 on AIME-2025. We make one deliberate exception for the DeepSeek-V3.2 reference on AIME-2025. With extended reasoning disabled, DeepSeek-V3.2 achieves only ~ 43 – 50% accuracy on these competition-level problems—a drastic underrepresentation of the model's capability in this setting, since AIME problems demand long chains of mathematical reasoning that the base generation mode cannot sustain. We therefore enable its native thinking mode, which raises accuracy to 76.8% . DIANOIA obtains a higher point estimate on the same 30 problems, but the wide uncertainty does not support a model-superiority claim. All other DeepSeek-V3.2 entries and all multi-agent method comparisons use the standard instruct-mode configuration.

Data Sampling.

- **GSM8K**: Full test split (1,319 samples).
- **AIME-2025**: Full problem set (30 problems).
- **MBPP**: Full test split (500 samples).
- **BFCL-SP**: Simple Python subset of the eval split (400 samples).
- **VitaBench**: Full held-out evaluation set (400 samples), evaluated with the benchmark pass-rate protocol.

Random Seeds and Reproducibility.

- All LLM API calls use fixed seeds where supported (OpenAI API seed=42).
- Sampling-based methods (Self-Consistency) use temperature=0.7 with seed=42 for diversity while maintaining best-effort reproducibility across runs.
- For methods without native seed support, we record and report all hyperparameters (temperature, top-p, etc.) in experimental configurations.

Confidence Intervals.

- 95% confidence intervals were computed via bootstrap resampling with 1,000 iterations.
- Bootstrap implementation: randomly sample with replacement from the set of n prediction results, compute accuracy, repeat 1,000 times, and report the 2.5th and 97.5th percentiles.
- CI computation uses the `bootstrap_ci` function in our codebase.
- VitaBench pass-rate intervals use the Wilson score method for a binomial proportion.

Execution Environment.

- **GSM8K**: During DIANOIA’s iterative loop (Execute phase), an LLM-based pseudo-verifier evaluates candidate solutions’ reasoning quality without ground-truth access, providing structured diagnostic feedback (see Remark 3.2). For *final evaluation metrics*, we use numerical equivalence checking (tolerance $\epsilon = 10^{-3}$): extracting the final numerical answer via regex and comparing against the ground truth.
- **AIME-2025**: Same LLM-based pseudo-verification during DIANOIA’s iterative loop; same numerical equivalence checking for final evaluation. Each problem yields a single integer answer.
- **MBPP**: Python 3.10 sandbox environment with 60-second timeout per test execution. Code is

executed with all provided test cases; a solution passes if all tests succeed without errors.

- **BFCL-SP**: The Simple Python subset contains 400 examples. Mocked API responses provide intermediate Execute feedback; final correctness is evaluated by the same implementation against expected function-call schemas and parameter types.
- **VitaBench**: Semantic pseudo-verification supplies intermediate diagnostic feedback; final pass rate follows the benchmark evaluation protocol. The prospective chronology and scope caveats are documented in Appendix A.

Baseline Implementation Details.

- **Self-Consistency**: $K = 5$ samples with temperature=0.7, majority voting on final answers.
- **MoA**: 3-layer architecture with 3 agents per layer, each layer synthesizes outputs from the previous layer.
- **Two Heads**: $K = 3$ collaborative solvers with mutual critique and refinement.
- **ReConcile**: Round-table discussion with $M = 3$ agents, $R = 2$ discussion rounds, majority voting for final selection.

Pareto-frontier sweep configurations (Figure 2).

Each method is swept along its native scaling knob on the MBPP test split, yielding 39 total configurations. Figure 2 connects the 29 Pareto-optimal points (the upper envelope of accuracy vs. tokens) for each method; dominated configurations are omitted from the plot for visual clarity but counted toward the sweep.

- **Self-Consistency** (9 configs): $K \in \{3, 5, 7, 9, 11, 15, 21, 27, 33\}$.
- **Two Heads** (6 configs): $K \in \{1, 2, 3, 4, 5, 7\}$.
- **ReConcile** (7 configs): $(M, R) \in \{(2, 0), (3, 0), (3, 1), (3, 2), (4, 1), (5, 2), (7, 1)\}$, where M is the agent count and R the number of additional discussion rounds.
- **MoA** (9 configs): 3 agents $\times$ 2 layers (default); $N \times 2$ layers for $N \in \{4, 5, 6, 8, 10, 12\}$; $\{6, 8\}$ agents $\times$ 3 layers.
- **DIANOIA** (8 configs): (K, R, S) as $(K, 0, 1)$ for $K \in \{1, 2, 3\}$; $(3, R, 3)$ for $R \in \{1, 2, 3\}$; $(K, 1, 3)$ for $K \in \{6, 9\}$.

Prompt Templates. Verbatim templates used in all DIANOIA experiments are listed below. Braces denote runtime substitutions (e.g., {user_prompt}); the released repository (App. A) contains the canonical versions.

Role briefs (Phase 1, Minimalist / Skeptic / Explorer). Each proposer prepends one role brief to the Diversity Proposer template:

[Minimalist] You are the Minimalist agent. Prefer the shortest path: the simplest stdlib idiom, the fewest steps. Minimize lines and intermediate state. Avoid redundancy.

[Skeptic] You are the Skeptic agent. Verify each step. Anticipate edge cases (empty input, single element, off-by-one, overflow). Be defensive. Make assumptions explicit.

[Explorer] You are the Explorer agent. Try an unconventional approach: different abstractions, alternative algorithms, surprising idioms. Avoid the most common solution. Diversify.

Diversity Proposer template (Phase 1, code tasks). The role brief is inserted in {role_prompt}:

```
{role_prompt}

Generate runnable Python code that
solves the user's task, grounded in the
context below.

## User question: {user_prompt}
## Context: {context}
## Your role bias: {role_requirements}

Important: read the test cases to
understand the contract. (1) Inspect
each assert carefully—input vs. expected
output. (2) Reason through each
example: input → expected output →
the transformation. (3) Pay attention to
return types (list/dict/tuple), counts,
ordering.

Output requirements: (1) MUST return
JSON with code in code field; (2) code
must be runnable as-is and define the
required function; (3) do NOT wrap in
Markdown fences; (4) no natural-language
commentary outside JSON.

Return JSON only: { "code": "your
runnable Python code" }
```

Reviewer template (Phase 3). The cross-review reviewer reads the candidate code AND its sandbox execution report:

You are a review expert. Evaluate another model's answer.

```
## User question: {user_prompt} ##
Context: {context}
```

```
## Answer under review
(model: {reviewed_model}, role:
{reviewed_role}): {candidate_text}
```

```
## Execution report summary:
{execution_summary}
```

Evaluate along: (1) strengths; (2) weaknesses (if execution failed, study the AssertionError's input/output to localise the bug); (3) risks; (4) improvement suggestions (concrete, actionable fixes); (5) overall score in [0.0, 1.0].

Return JSON only: { "strengths": [...], "weaknesses": [...], "risks": [...], "suggestions": [...], "overall_score": 0.0 }

Synthesizer template (Phase 4, code tasks). The synthesizer integrates candidates, execution reports, and cross-reviews with iterative validation feedback:

You are a synthesis expert. Combine the candidate answers and review feedback into a single, optimal answer.

```
## User question: {user_prompt} ##
Context: {context}
```

```
## Candidate answer summaries:
{candidates_summary}
```

```
## Execution report summary:
{execution_summary}
```

```
## Cross-review summary:
{review_summary}
```

```
{validation_feedback}
```

Key analysis steps: (1) Inspect every execution error—when an AssertionError is present, compare expected vs. actual and localise the bug; (2) Infer the contract from the tests—each assert is the ground truth; (3) Pay attention to return type, element count, data-structure shape.

Requirements: (1) Synthesize—do not vote or copy one candidate verbatim; (2) If every candidate is wrong, derive a correct implementation from the error messages; (3) MUST return JSON with code in code field; (4-5) no Markdown fences, no commentary outside JSON.

Return JSON only: { "code": "your runnable Python code" }

Pseudo-verifier template (GSM8K / AIME, Remark 3.2; produces the structured σ_v feedback of Def. D.7). For math reasoning tasks where deterministic execution is unavailable, a dedicated LLM evaluator returns the structured diagnostic:

You are a careful mathematical reasoning verifier. Given a problem and a candidate solution, judge whether the solution is correct WITHOUT access to the ground-truth answer. Identify reasoning errors or computational mistakes, and report a calibrated confidence in your judgment.

Problem: {user_prompt}

Candidate solution: {candidate_text}

Evaluate: (1) Does the reasoning chain follow correctly from the problem? (2) Are intermediate computations correct? (3) Does the final answer match what the reasoning produces? (4) List any specific errors (computational, logical, prompt mis-reading).

Return JSON only: { "is_correct": <true or false>, "confidence": <0.0..1.0>, "errors": [...] }

raised *independently*—against Minimalist (as a secondary risk) and against Explorer (as a primary risk). The line appears in *no* candidate, instantiating the “local repair beyond selection” mechanism described in Section 4 (Phase 4) and quantified by the 5.5% rescue rate reported in Section 5.8. Full reviewer JSON follows the structured schema documented in the reviewer template (Appendix M); the strengths/suggestions/overall_score fields are elided here for length but conform to the documented schema.

N Figure 1 Trace: Full Reviewer Outputs

Figure 1 reproduces a verbatim DIANOIA trace from one of our MBPP runs. For length, the figure shows excerpts; the full reviewer outputs are presented below.

Review on Minimalist (failing candidate; raw TypeError). *Weakness.* “The code constructs the tuple via tuple(t1[i], t2[i%len(t2)]), which triggers a type error: tuple() accepts at most one argument, raising TypeError: tuple expected at most 1 argument, got 2. The correct form is the tuple literal (t1[i], t2[i%len(t2)]).” *Risks.* “The code is syntactically incorrect; it cannot pass any test case and may mislead users into believing the implementation is correct.” “Edge cases such as empty tuples are not considered; the current tests are non-empty, so this is a *secondary* risk.”

Review on Explorer (passing candidate). *Risk.* “If the input tup2 has length 0, the modulo operation raises a division-by-zero error (len(tup2)==0); the current tests do not cover this boundary.”

Review on Skeptic (passing candidate). *Risk.* “If future requirements demand zip-style alignment to the shorter sequence, the cyclic-extension behaviour will be incorrect; for very long inputs the cycling may also induce performance issues.”

Synthesizer’s response. The line if not tup2: return [] (highlighted in Figure 1) directly addresses the empty-tup2 risk that two reviewers