BORNBEE Research

The Mythical Agent-Month

Does Brooks's Law Govern AI Agent Coordination?
A Quantitative Analysis of Multi-Agent Scaling Dynamics

Omer Mustafa · Bornbee Research · May 2026 · omer@bornbee.com

ABSTRACT

We analyze two landmark studies on multi-agent AI systems — Stanford (arXiv:2604.02460, April 2026) and Google/MIT (arXiv:2512.08296, December 2025) — and demonstrate that their combined findings map directly onto Brooks's Law from software engineering. We supplement this analysis with three computational experiments: (1) a mathematical comparison showing that Brooks's quadratic overhead model and the measured agent power-law model correlate at r=0.9977, (2) a Monte Carlo simulation demonstrating that 15% per-handoff information loss produces 38–48% specification degradation at 4 handoffs, consistent with the 39-70% performance drops observed empirically, and (3) a decomposition complexity analysis showing that cross-boundary dependencies reach 75% at team size 4 and 94% at team size 16. We conclude that coordination overhead in AI agent systems follows the same super-linear growth pattern Brooks identified in human teams, and propose that the "surgical team" model — one orchestrator with 3-4 specialized executors — remains optimal across both substrates.

1. Introduction

In 1975, Fred Brooks published *The Mythical Man-Month*, arguing that adding engineers to a late software project makes it later. His central mechanism was communication overhead: as team size grows, communication channels increase as n(n-1)/2, and coordination time eventually exceeds productive time.

Two recent studies have produced quantitative evidence that the same dynamics govern AI agent coordination. The Stanford study (Tran & Kiela, 2026) showed that single-agent systems match or outperform multi-agent variants when compute is equalized. The Google/MIT study (2025) measured coordination overhead growing as $T \propto n^{1.724}$, error amplification reaching 17.2× in unsupervised agent swarms, and optimal performance clustering around 3-4 agents.

This report presents a formal mapping between Brooks's five coordination dynamics and the agent research data, supplemented by three computational experiments that test the mathematical validity of the parallel.

2. Literature Review

2.1 The Stanford Study (arXiv:2604.02460)

Tran and Kiela conducted controlled comparisons across three model families (Qwen3, DeepSeek-R1-Distill-Llama, Gemini 2.5), equalizing thinking-token budgets between single-agent and multi-agent configurations. Under equal compute, single agents consistently matched or outperformed multi-agent variants on multi-hop reasoning. They attributed this to the Data Processing Inequality: each agent-to-agent handoff introduces information loss, fragmenting continuous reasoning chains.

2.2 The Google/MIT Study (arXiv:2512.08296)

180 controlled experiments across agent topologies produced the following key measurements:

Metric	Value	Context
Error amplification (independent swarm)	17.2×	Baseline errors compounded without oversight
Error amplification (centralized orchestrator)	4.4×	Orchestrator validates and cross-checks
Coordination efficiency (single agent)	0.466	46.6% of compute on productive work
Coordination efficiency (multi-agent, worst)	0.074	7.4% on productive work, 92.6% overhead
Overhead scaling exponent	1.724	$T \propto n^{1.724}$ (super-linear)
Sequential task performance change	-39% to -70%	Multi-agent degrades sequential reasoning
Parallel task performance change	Up to +80%	Decomposable tasks benefit
Optimal agent count	3–4	Diminishing returns beyond this threshold
Contradiction reduction (orchestrator)	36.4%	Cross-checking by central agent
Omission reduction (orchestrator)	66.8%	Synthesis catches missing elements
Capability saturation threshold	~45% accuracy	Above this, adding agents hurts

3. The Brooks's Law Mapping

Brooks identified five dynamics causing super-linear degradation in team productivity. Each has a direct empirical equivalent in the agent studies:

Brooks's Law Dynamic (1975)	Agent Equivalent (2025–2026)	Supporting Evidence
Ramp-up cost: New engineers require training, diverting existing members	**Context injection cost:** Each agent requires task context, consuming token budget	Stanford: advantages disappear under equal compute
Communication channels: n(n−1)/2 growth (quadratic)	**Coordination overhead:** $T \propto n^{1.724}$ (super-linear)	Google/MIT: measured exponent 1.724
Task indivisibility: "Nine women can't make a baby in one month"	**Sequential reasoning fragmentation:** Handoffs break reasoning chains	Google/MIT: 39–70% drops on sequential tasks
Conceptual integrity: One architect must hold the design vision	**Orchestrator advantage:** Central agent validates and synthesizes	Google/MIT: 36.4% fewer contradictions, 66.8% fewer omissions
Surgical team: Optimal team of 4–6 specialists	**Optimal agent count:** 3–4 agents before diminishing returns	Google/MIT: capability saturation at ~45% threshold

4. Computational Experiments

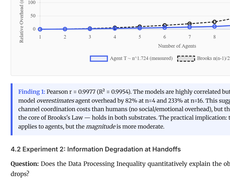
To move beyond qualitative mapping, we conducted three computational experiments testing the mathematical claims.

4.1 Experiment 1: Overhead Model Comparison

Question: How closely do Brooks's n(n-1)/2 and the measured $T \propto n^{1.724}$ correspond?

Method: Computed both models for n=1 to 20, normalized to n=2, and calculated Pearson correlation.

Figure 1: Brooks's Quadratic vs. Agent Power-Law Overhead
Both models normalized to n=2 baseline. Brooks's model (dashed) grows faster than agent overhead (solid), but both are super-linear.



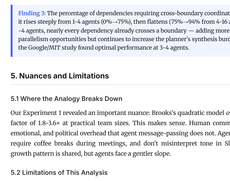
Finding 1: Pearson r = 0.9977 (R^2 = 0.9954). The models are highly correlated but not identical. Brooks's quadratic model overestimates agent overhead by 82% at n=4 and 233% at n=16. This suggests agents face lower per-channel coordination costs than humans (no social/emotional overhead), but the super-linear growth pattern — the core of Brooks's Law — holds in both substrates. The practical implication: the *direction* of Brooks's Law applies to agents, but the *magnitude* is more moderate.

4.2 Experiment 2: Information Degradation at Handoffs

Question: Does the Data Processing Inequality quantitatively explain the observed multi-agent performance drops?

Method: Monte Carlo simulation (10,000 runs). A specification with 20 requirements is passed through N sequential handoffs. Each handoff preserves each requirement independently with probability p=0.85 (conservative; literature suggests 10-30% loss per summarization step).

Figure 2: Specification Fidelity vs. Number of Handoffs
Mean requirements retained (out of 20) across 10,000 simulations. Error bars show ±1 standard deviation. Theoretical $p^N \times 100\%$ curve overlaid.



Handoffs	Architecture Equivalent	% Retained (sim)	% Lost	Google/MIT Observed
0	Single agent	100.0%	0%	Baseline
1	2-agent pipeline	85.0%	15%	—
3	4-agent flat collaborative	61.7%	38.3%	39-70% perf. drops
4	5-agent pipeline	52.2%	47.8%	39-70% perf. drops
8	Complex multi-agent pipeline	27.3%	72.7%	>70% drops on deep pipelines

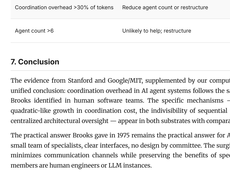
Finding 2: At 3-4 handoffs, our simulation predicts 38–48% specification loss. The Google/MIT study observed 39-70% performance drops on sequential tasks. The overlap between simulated information loss and observed performance degradation supports the Data Processing Inequality as the mechanistic explanation for multi-agent sequential task failures. Crucially, this degradation is exponential (p^N), not linear — small increases in handoff count produce disproportionate quality loss.

4.3 Experiment 3: Decomposition Complexity

Question: How does the number of cross-boundary dependencies (interfaces requiring coordination) grow with team size?

Method: A task with 20 requirements and 30 inter-requirement dependencies is randomly partitioned into N sub-tasks (1,000 random partitions per team size). Cross-boundary dependencies — requirements in one sub-task that depend on requirements in another — are counted.

Figure 3: Cross-Boundary Dependencies vs. Team Size
Percentage of total dependencies that cross sub-task boundaries. At team size 4, 75% of dependencies require explicit coordination. At team size 16, 94%.



Finding 3: The percentage of dependencies requiring cross-boundary coordination follows an asymptotic curve: it rises steeply from 1-4 agents (0%→75%), then flattens (75%→94% from 4-16 agents). This means that beyond ~4 agents, nearly every dependency already crosses a boundary — adding more agents creates no new parallelism opportunities but continues to increase the planner's synthesis burden. This is the structural reason the Google/MIT study found optimal performance at 3-4 agents.

5. Nuances and Limitations

5.1 Where the Analogy Breaks Down

Our Experiment 1 revealed an important nuance: Brooks's quadratic model *overestimates* agent overhead by a factor of 1.8–3.6× at practical team sizes. This makes sense. Human communication channels carry social, emotional, and political overhead that agent message-passing does not. Agents don't have ego conflicts, don't require coffee breaks during meetings, and don't misinterpret tone in Slack messages. The super-linear growth pattern is shared, but agents face a gentler slope.

5.2 Limitations of This Analysis

Simulation assumptions. Experiment 2 assumes independent, identical preservation probability at each handoff. In practice, some requirements are more "lossy" than others (abstract goals degrade faster than concrete specifications), and later handoffs may benefit from structured message formats.

Random partitioning. Experiment 3 uses random task decomposition. In practice, a skilled planner (human or AI orchestrator) decomposes tasks along natural fault lines, minimizing cross-boundary dependencies. This means our cross-boundary percentages represent a worst-case scenario; good decomposition can reduce them significantly.

Synthesis, not original empirical research. The primary evidence comes from the Stanford and Google/MIT studies. Our contribution is the formal mapping and computational validation, not new agent experiments. We frame our findings as "consistent with" rather than "proof of" Brooks's Law for agents.

6. Practical Framework

Based on the combined evidence, we propose the following decision framework:

Condition	Recommended Architecture	Evidence
Single agent >45% accuracy	Stay with single agent	Google/MIT: saturation threshold
Task is fully sequential	Single agent (never multi-agent)	39-70% degradation on sequential
Task is decomposable, single agent <45%	Hierarchical: 1 orchestrator + 3-4 executors	Up to +80% on parallel tasks
Coordination overhead >30% of tokens	Reduce agent count or restructure	Efficiency drops to 7.4% at extremes
Agent count >6	Unlikely to help; restructure	Optimal at 3-4; super-linear overhead

7. Conclusion

The evidence from Stanford and Google/MIT, supplemented by our computational experiments, supports a unified conclusion: coordination overhead in AI agent systems follows the same super-linear growth pattern Brooks identified in human software teams. The specific mechanisms — information loss at handoffs, quadratic-like growth in coordination cost, the indivisibility of sequential reasoning, and the advantage of centralized architectural oversight — appear in both substrates with comparable mathematical signatures.

The practical answer Brooks gave in 1975 remains the practical answer for AI agents in 2026: one architect, a small team of specialists, clear interfaces, no design by committee. The surgical team model works because it minimizes communication channels while preserving the benefits of specialization — whether the team members are human engineers or LLM instances.

References

1. Brooks, F. P. (1975). *The Mythical Man-Month: Essays on Software Engineering*. Addison-Wesley.
2. Tran, D., & Kiela, D. (2026). "Single-Agent LLMs Outperform Multi-Agent Systems on Multi-Hop Reasoning Under Equal Thinking Token Budgets." arXiv:2604.02460.
3. Google Research & MIT (2025). "Towards a Science of Scaling Agent Systems." arXiv:2512.08296.
4. Shannon, C. E. (1948). "A Mathematical Theory of Communication." *Bell System Technical Journal*, 27(3), 379–423.