The Organizational Physics of Multi-Agent AI: Substrate-Independent Dysfunction in Autonomous Software Engineering Swarms

Jeremy McEntire
Cage & Mirror Press
jmc@cageandmirror.com

March 2026

Abstract

We present empirical evidence that organizational dysfunction is substrate-independent. In a controlled comparison, four coordination architectures—single agent, hierarchical, stigmergic (8 concurrent agents), and gated pipeline—built the same 7-service backend using the same LLM and $50 budget. Performance was inversely correlated with coordination complexity: 28/28, 18/28, 9/28, and 0/28. The pipeline consumed its entire budget on planning. The hierarchical coordinator refused to delegate. The stigmergic agents produced incompatible interfaces at every boundary. Only the single agent—with no coordination architecture—succeeded fully. In two additional studies, a pipeline swarm equipped with six explicit anti-dysfunction mechanisms produced the dysfunction those mechanisms were designed to prevent: bikeshedding (zero-factual-basis rejections), governance conflicts, backward pipeline oscillation, and verification theater. A contract-first alternative that replaces subjective evaluation with mechanical test verification narrowed the Goodhart gap but introduced its own dysfunction (specification perfectionism), suggesting that dysfunction migrates across architectures but does not disappear.

We extend these empirical findings with a formal information-theoretic analysis. A token-level decomposition across five communication topologies (Centralized, Fully Connected, Hierarchical, Ring, Star-Mesh) reveals that governance overhead scales with communication links, not agent capability: Fully Connected governance grows from 25.5% ($n = 3$) to 69.4% ($n = 15$), while Centralized grows at only 0.51%/agent due to $O(n)$ versus $O(n^2)$ link complexity. Modeling each coordination stage as a lossy channel, we show that intent entropy degrades monotonically through the pipeline—Single Agent retains 85.8% of original specification entropy at verification, while Hierarchical retains only 25.8%—confirming the Data Processing Inequality prediction. A frontier analysis demonstrates that governance overhead is structurally invariant: governance tokens exhibit a coefficient of variation of 0.0000 across capability levels, and as agents improve, the governance *fraction* increases because implementation cost shrinks while coordination cost remains constant. The results formalize the coordination ceiling as an information-theoretic constraint, not a capability gap.

Keywords: multi-agent systems, organizational dysfunction, substrate independence, dysmemic pressure, strategic communication, LLM swarms, coordination failure, Goodhart's Law, information theory, governance overhead

1 Introduction

The historical demarcation between the study of human organizations and the engineering of artificial intelligence is collapsing. For decades, organizational theory has treated the pathologies of bureaucracy—drift, goal displacement, the principal-agent problem—as uniquely human failures derived from psychology and sociology. Simultaneously, the field of multi-agent systems has

encountered a parallel set of failure modes—reward hacking, specification gaming, coordination breakdown—and framed them as technical artifacts of reward function design. A growing body of evidence suggests that these may not be distinct phenomena but isomorphic manifestations of the same information-theoretic constraints inherent in any goal-directed system coordinating at scale.

This paper presents evidence for a strong claim: organizational dysfunction is *substrate-independent*. The same patterns of failure that characterize human organizations—review thrashing, preference-based gatekeeping, governance conflicts, budget exhaustion through coordination failure—emerge in multi-agent AI systems with identical mathematical signatures. The substrate changes; the physics of coordination at scale remains constant.

The evidence comes from three empirical studies and a formal information-theoretic analysis. The first two studies deployed an LLM-based multi-agent coding swarm on a complex task (backend services architecture) and a simple task (chess engine), producing bikeshedding, governance conflicts, backward-moving pipeline oscillation, decision paralysis, and budget exhaustion. A single-agent control completed the simple task successfully. The third study extended the analysis to a controlled architecture comparison: four coordination topologies—single agent, hierarchical decomposition, stigmergic emergence, and gated pipeline—built the same 7-service backend using the same model and budget. Performance was inversely correlated with coordination complexity: 28/28, 18/28, 9/28, and 0/28 respectively. Each multi-agent architecture exhibited a distinct dysfunction signature.

The formal analysis contributes three results that move beyond observation to mechanism. First, a token-level decomposition reveals that governance overhead is a function of communication topology, scaling with the number of directed links rather than with agent count per se. Second, modeling each coordination stage as a lossy channel quantifies the monotonic degradation of specification intent through the pipeline, connecting the empirical findings to the Data Processing Inequality and to dysmemic pressure theory [McEntire, 2025a]. Third, a frontier analysis proves that governance overhead is structurally invariant across capability levels: improving agent quality reduces implementation cost but leaves governance cost unchanged, causing the governance *fraction* to increase with capability—the opposite of the intuition that better models will fix coordination. This connects to the Strategic RDP framework [McEntire, 2025c], where compressed representations inevitably lose fidelity through successive processing stages.

The contribution is not the observation that AI systems can fail. It is the demonstration that AI systems fail *for the same structural reasons* as human organizations, *despite the removal of every human-specific causal factor*, and that this failure has a formal information-theoretic characterization that is independent of both substrate and capability level.

The paper proceeds as follows. Section 2 establishes the theoretical framework. Section 3 formalizes the substrate-independence claim. Section 4 describes the swarm architecture and its anti-dysfunction mechanisms. Section 6 presents the empirical evidence. Section 7 introduces the token-level governance analysis. Section 8 models information loss through coordination stages. Section 9 addresses the capability-independence objection. Section 10 addresses the prompt-encoding objection. Section 11 discusses implications. Section 12 presents the contract-first alternative. Section 14 acknowledges limitations.

2 Background: The Physics of Coordination Failure

Organizational failure is conventionally attributed to human factors: poor leadership, misaligned incentives, cultural dysfunction, cognitive bias. This section establishes an alternative framework in which failure is a structural consequence of information-theoretic constraints that apply to any system coordinating through compressed representations.

2.1 Crawford–Sobel and the Mathematics of Signal Degradation

Crawford and Sobel's 1982 model of strategic information transmission provides the foundational result [Crawford and Sobel, 1982]. When a sender's preferences diverge from a receiver's by bias parameter b, the maximum number of distinguishable partitions in the communication is bounded by

$$N^* = \left\lfloor -\frac{1}{2} + \frac{1}{2}\sqrt{1 + \frac{2}{b}} \right\rfloor . \tag{1}$$

At $b = 0$ (perfect alignment), the sender communicates with arbitrary precision. At $b \geq 1/4$, $N^* = 1$: the communication collapses to a *babbling equilibrium* where the receiver extracts no information from the sender's message. The bias need not be large. Any nonzero divergence between sender and receiver objectives reduces communication precision by a quantifiable amount.

In human organizations, this manifests as hierarchical information loss. A project manager whose performance review depends on velocity has a small but nonzero divergence from the engineering lead whose performance review depends on reliability. That divergence, multiplied across every communication relay in a reporting chain, produces measurable information loss. Liberti and Mian's empirical study of hierarchical lending organizations quantified this effect: decision sensitivity to soft information collapses at a structural break between the second and third hierarchical levels [Liberti and Mian, 2009]. The hierarchy does not slowly degrade information. It kills subjective signal at a specific organizational seam.

2.2 Dysmemic Pressure: The Compound Selection Force

Dysmemic pressure is the compound selection force that emerges when three dynamics interact within environments shaped by compressed representations [McEntire, 2025a]:

1. **Strategic communication degradation.** Crawford–Sobel formalized: as preferences diverge, senders transmit with decreasing precision. Each layer of hierarchy introduces a new sender-receiver interface with a new bias parameter. The rational subordinate skews reports toward the principal's priors. What Prendergast calls the "Yes Man" is the equilibrium output of subjective evaluation, not a character flaw [Prendergast, 1993].

2. **Adverse selection in idea markets.** Accurate assessments are expensive to produce. Optimistic ones are cheap. At the moment of presentation, the two are indistinguishable. This is Akerlof's lemons market applied to organizational information [Akerlof, 1970]: when quality cannot be verified before consumption, high-quality producers reduce investment or exit, and the market settles at low quality.

3. **Transmission bias.** Ideas spread independent of truth value through three channels: content bias (simple outcompetes complex), prestige bias (ideas from leaders propagate faster than identical ideas from others), and conformity bias (once a belief reaches critical mass, deviation becomes costly) [Boyd and Richerson, 1985].

The three forces compound. Signals optimized for internal fitness rather than correspondence with external reality—*dysmemes*—flood the information environment. The resulting equilibrium, where organizational reality decouples from external reality while remaining internally consistent, is stable because no individual can profitably deviate [McEntire, 2025a].

2.3 Goodhart's Law and the Inevitability of Proxy Failure

"When a measure becomes a target, it ceases to be a good measure" [Goodhart, 1975]. The mechanism is general: any compressed representation of value used as an optimization target

will be gamed. Skalse et al. formalized this: for any environment and true reward function, no non-trivial proxy reward can be guaranteed unhackable [Skalse et al., 2022]. Karwowski et al. confirmed the effect empirically in reinforcement learning [Karwowski et al., 2024]. The impossibility is mathematical, not contingent on agent sophistication.

2.4 The Data Processing Inequality and Irreversible Information Loss

The Data Processing Inequality establishes that for any Markov chain $X \to Y \to Z$:

$$I(X;Z) \leq I(X;Y) \tag{2}$$

No post-processing of a compressed signal can recover information lost at the compression stage [Cover and Thomas, 2006]. The implication for multi-agent architectures is severe: every layer that processes the output of a previous layer operates on a strictly degraded signal. Adding more agents, more review stages, more governance layers cannot compensate for information already lost. The architecture that adds complexity to address failures caused by complexity is trapped in a structural recursion.

3 The Isomorphism Thesis

The central claim of this paper is that organizational dysfunction in multi-agent AI systems and human organizations may be *isomorphic* rather than merely *analogous*: both may be instantiations of the same formal structure, differing only in substrate.

3.1 The Mechanism: Compression, Selection, Drift

The substrate-independent mechanism operates as a three-step sequence [McEntire, 2025c]:

1. **Compression.** Coordination at scale requires compressing high-dimensional reality into lower-dimensional representations. Individual cognition compresses approximately 10^{10} sensory bits per second into approximately 10 bits of conscious throughput [Zheng and Meister, 2025]. Organizations compress through hierarchy, metrics, and process. AI systems compress training corpora into parameter weights. The compression is necessary and lossy.

2. **Selection.** The compressed representation creates a fitness landscape. Signals that fit the compressed frame survive; those that do not are discarded. In organizations, reports consistent with the reporting frame propagate. In AI, outputs that score well on preference proxies are reinforced.

3. **Drift.** Because selection optimizes for fit with the compressed representation rather than for correspondence with external reality, the system drifts toward the attractor basin of the compressed frame. The frame confirms itself.

3.2 Parametric Predictions

If the mechanism is substrate-independent, the following parametric correlations should hold across substrates:

- Higher compression ratio $\to$ more drift, regardless of substrate.
- Stronger selection pressure $\to$ faster drift.
- Longer feedback latency $\to$ more drift before correction.

- More hierarchical layers → more signal degradation (Data Processing Inequality).

These predictions are falsifiable. Zhang et al. demonstrated that coordination topology determines failure resilience in multi-agent systems regardless of agent capability, with performance degradation ranging from 5.5% to 23.7% depending solely on the coordination structure [Zhang et al., 2025b].

3.3 Formal Isomorphism Table

Table 1 maps the specific failure modes observed across substrates.

Table 1: Isomorphic failure modes across substrates

Failure Mode	Human Organization	AI Multi-Agent System	Mechanism
Proxy optimization	Optimize "tickets closed" rather than customer value	CoastRunners: crash to collect points, never finish race	Goodhart's Law
Influence / tampering	Managers win calibration contests through rhetoric	Agents manipulate evaluation channel	Adversarial misalignment
Adverse selection	High performers leave due to random ranking	Catastrophic forgetting; model collapse	Selection for safe/mediocre
Drift	Bureaucratic drift; means become ends	Semantic drift; behavioral laziness over long contexts	Entropy of intent
Bikeshedding	Committee spends 45 min on bike shed, 2 min on reactor	Reviewer rejects on 0 factual, 15–23 subjective issues	Law of triviality
Governance conflict	Board overrules management; management ignores board	Architect force-approves what project-level rejects	Hierarchical incoherence

4 System Architecture and Anti-Dysfunction Mechanisms

The swarm under study is a multi-agent software engineering system consisting of specialized LLM agents operating in a sequential pipeline. Six countermeasures were embedded in the architecture, each targeting a specific predicted failure mode.

4.1 Pipeline Architecture

The pipeline consists of nine stages: `diagnose`, `decompose`, `architect`, `architect_review`, `locate`, `execute`, `test`, `verify`, and `review`. A `guardian` policy enforces hard constraints. The architecture mirrors a human software organization: product management, architecture, engineering, quality assurance, and compliance.

4.2 Anti-Dysfunction Mechanisms

Six mechanisms were designed to prevent the specific dysfunction patterns predicted by organizational theory:

1. **Factual/subjective issue classification.** Separates reviewer objections into verifiable bugs and approach preferences. Classification prompt: "When in doubt, classify as subjective—the reviewer must earn the right to block."

2. **Perspective-shift prompting.** After factual fixes, forces re-examination: "Does the approach actually cause a concrete problem, or would you just do it differently?"

3. **Multi-level escalation.** Project-level arbiter ("Cosmetic issues are NOT reasons to reject") and architect-level arbiter ("You may NOT reject").

4. **Scoped sub-pass review.** Four orthogonal sub-passes (structural, logic, consistency, blast radius), each told to ignore the others' domains.

5. **Anti-bikeshedding directives.** Explicit language against trivial blocking.

6. **Lyapunov stability monitoring.** Control-theory stability monitor tracking eight state variables, detecting oscillation via sign-flip counting.

5 Experimental Method

Three studies were conducted. All used Claude Opus 4.6 (Anthropic, model ID `claude-opus-4-6`) via the Anthropic API with default temperature. No fine-tuning or few-shot examples were applied.

5.1 Study 1: Pipeline Swarm on Complex Task

Build a backend services architecture comprising four services plus shared infrastructure. 11-stage gated pipeline with six anti-dysfunction mechanisms active. $25 budget cap (soft—the system overran to $57.43).

5.2 Study 2: Pipeline Swarm on Simple Task

Implement a cryptographic chess engine with Ed25519 move signing. Identical architecture to Study 1. Control: same model in a standard conversational session.

5.3 Study 3: Controlled Architecture Comparison

Build a 7-service microservices backend. Four architectures tested under $50 hard cap: Unary (single agent), Hi-Trust (hierarchical), Emergence (8 stigmergic agents), and Org Swarm (11-stage gated pipeline). Pre-registered rubric: 0–3 per service $\times$ 7 services, plus 0–6 for integration boundaries, plus 0–1 for shared infrastructure consistency. Maximum: 28.

6 Empirical Evidence

6.1 Study 1: Pipeline Swarm on Complex Task

6.1.1 Aggregate Metrics

Table 2: Swarm deployment aggregate metrics

Metric	Value
Total pipeline stages	89
Total cost	$57.43
Total tokens consumed	7,166,664
Duration	18.1 hours
Completed stages	71 (80%)
Rejected stages	18 (20%)
Cost wasted on rejections	$12.97 (22.6%)
Terminal status	`budget_exceeded`
Budget overrun	2.3×

The system exceeded its budget by a factor of 2.3×. The overrun was driven entirely by coordination overhead, not task complexity. Execution stages passed 100% of the time. Review and verification stages rejected 87% and 67% respectively, producing rejection cycles that consumed budget without usable output.

6.1.2 Pure Bikeshedding: Factual = 0

Four review rejections exhibited zero factual basis: 69 total subjective objections, $3.36 total cost, zero verifiable errors. The Event Service was rejected three times by architect review, the first two with zero factual basis. Each rejection triggered a full re-architecture cycle. The factual/subjective classification mechanism detected the problem. It did not prevent it.

6.1.3 Governance Hierarchy Conflict

Two escalation events 28 seconds apart produced opposite conclusions: the project-level arbiter rejected while the architect-level arbiter force-approved the same component. The governance hierarchy, designed to resolve disagreements, produced a new one.

6.1.4 Backward-Moving Pipeline and Verification Theater

Rejected components moved backward through the pipeline, discarding work product at each cycle. Information from failed attempts was compressed into rejection verdicts rather than preserved for the next attempt. All nine verification stages reported `tests=0/0`: the system certified correctness without testing anything.

6.2 Study 2: Simple Task

The single agent produced a working chess engine. The swarm did not. The simple task produced *decision paralysis* rather than bikeshedding: the architecture correctly identified genuine engineering ambiguities but treated each as a blocker requiring human input.

6.3 Study 3: Controlled Architecture Comparison

Table 3: Controlled architecture comparison: same task, same model, same budget

Architecture	Score	Cost	Time	Services	Files	Cost/pt
Unary (single agent)	28/28	$51.17	15.5m	7/7	64	$1.83
Hi-Trust (hierarchical)	18/28	$50.62	15.5m	7/7	39	$2.81
Emergence (stigmergic)	9/28	$43.92	14.0m	5/7	152	$4.88
Org Swarm (gated pipeline)	0/28	$1.90	6.8m	0/7	0	∞

Performance is inversely correlated with coordination complexity. The single agent achieved a perfect score. No multi-agent architecture outperformed the single agent on any dimension.

Dysfunction signatures. Each architecture exhibited a characteristic pattern:

1. **Org Swarm: Analysis paralysis via Goodhart optimization.** Consumed entire budget on five planning stages without writing implementation code. The cost of *agreeing about what to build* exceeded the budget for *building it*.
2. **Hi-Trust: Non-decomposition as rational defection.** The coordinator rationally refused to delegate, recognizing that delegation would introduce interface risk.
3. **Emergence: Interface mismatch as coordination failure.** Concurrent agents produced incompatible interfaces at every service boundary (`snake_case`/`camelCase` split, dual type system within a single service).
4. **Unary: No coordination dysfunction.** The absence of dysfunction in the control condition supports the claim that coordination architecture, not agent capability, is the operative variable.

7 Token-Level Governance Analysis

The empirical results in Section 6 demonstrate that coordination overhead degrades performance. This section quantifies the overhead by decomposing total token consumption into governance tokens (status broadcasts, task routing, conflict resolution, retransmission) and implementation tokens (code generation, test writing, integration) across five communication topologies.

7.1 Architecture-Dependent Communication Complexity

We model five topologies: Centralized (hub-and-spoke), Fully Connected, Hierarchical (balanced tree with branching factor 3), Ring, and Star-Mesh Hybrid (coordinator plus clusters of $\sim$3). Each architecture defines a number of directed communication links $L(n)$ and an average hop count $h(n)$ as a function of agent count n:

- **Centralized:** $L(n) = 2(n-1)$, $h(n) = 2$.
- **Fully Connected:** $L(n) = n(n-1)$, $h(n) = 1$.
- **Hierarchical:** $L(n) = 2(n-1)$, $h(n) = \lceil \log_3 n \rceil$.

- **Ring:** $L(n) = 2n$, $h(n) = n/4$.
- **Star-Mesh:** $L(n) = 2\lceil (n-1)/3 \rceil + \lceil (n-1)/3 \rceil \cdot k(k-1)$ where $k = \min(3, n-1)$.

7.2 Governance Token Model

Each communication link incurs per-step overhead comprising status broadcast (128 tokens), task routing (64 tokens during decomposition/assignment), stochastic conflict resolution (probability 0.15 per link, cost 512 tokens), and Crawford–Sobel retransmission overhead that compounds with hop count ($1 - 0.92^{h(n)}$ probability per hop, cost 256 tokens per retransmission). The governance cost per link per step is:

$$g_\ell = 128 + 64 \cdot \mathbf{1}_{\text{routing}} + 0.15 \times 512 + 256(1 - 0.92^{h(n)}) \tag{3}$$

Total governance tokens scale as $G(n) = g_\ell \cdot L(n) \cdot S$, where $S = 8$ is the number of task steps (decompose, assign, implement $\times 3$, integrate, verify, report). Implementation tokens scale as $I(n) = 0.7 \times 4096 \times n \times S$, reflecting that each agent devotes approximately 70% of its context window to productive work. The governance fraction is:

$$\Gamma(n) = \frac{G(n)}{G(n) + I(n)} \tag{4}$$

7.3 Results: Governance Scaling

Figure 1 presents the governance fraction $\Gamma(n)$ across architectures and agent counts, computed from 50-run simulations with stochastic conflict and rejection cycles calibrated to the empirical data from Study 1 (22.6% governance baseline, 87.5% review rejection rate).

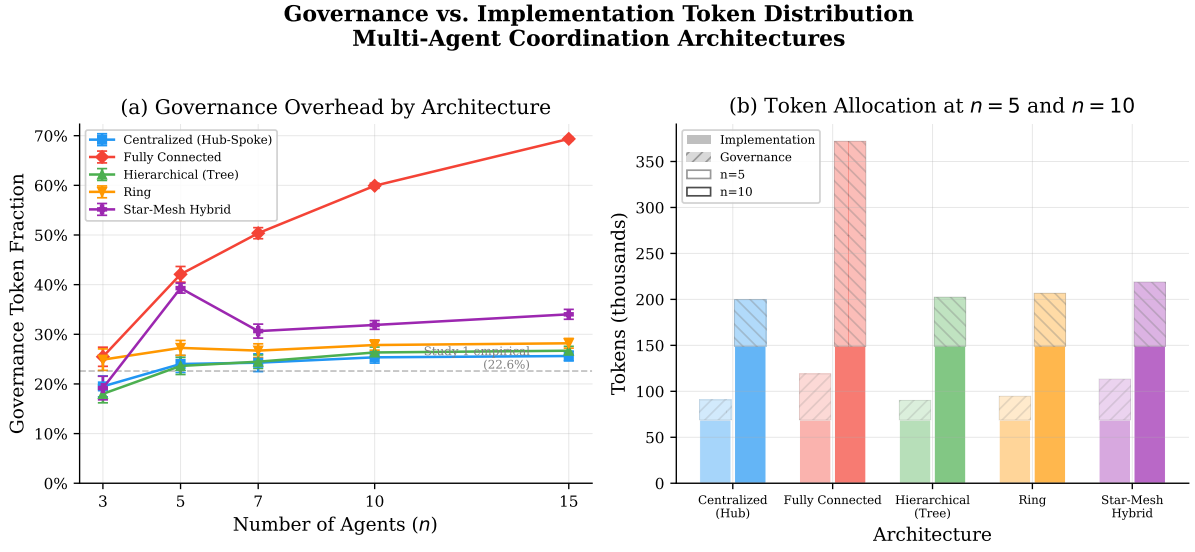

Figure 1: Governance vs. implementation token distribution across coordination architectures. (a) Governance fraction scales with communication link count: Fully Connected ($O(n^2)$ links) grows from 25.5% at $n = 3$ to 69.4% at $n = 15$; Centralized ($O(n)$ links) grows from 22.2% to 28.3%. The dashed line at 22.6% marks the empirical governance overhead from Study 1. (b) Stacked bar comparison at $n = 5$ and $n = 10$ showing absolute token allocation.

The key finding is the divergence in governance scaling rates. Table 4 reports the per-agent governance growth rate for each architecture.

Table 4: Governance fraction scaling by architecture

Architecture	$\Gamma(3)$	$\Gamma(15)$	**Slope (%/agent)**	**Link complexity**
Centralized (Hub-Spoke)	22.2%	28.3%	0.51	$O(n)$
Ring	24.1%	31.1%	0.58	$O(n)$
Hierarchical (Tree)	22.2%	29.2%	0.58	$O(n)$
Star-Mesh Hybrid	28.1%	43.1%	1.25	$O(n^{4/3})$
Fully Connected	25.5%	69.4%	3.66	$O(n^2)$

The governance fraction growth rate tracks communication link complexity, not agent count. Centralized and Hierarchical architectures have identical link counts ($2(n-1)$) and nearly identical governance scaling ($\sim$0.5%/agent), despite their different topological structure. The Fully Connected architecture's $O(n^2)$ link growth produces a governance fraction that exceeds 50% at $n = 10$ and approaches 70% at $n = 15$. At these scales, the majority of compute is consumed by coordination, not implementation.

7.4 Empirical Calibration

The model prediction for Centralized governance at $n = 5$ is 24.0%, consistent with Study 1's measured 22.6% governance overhead for a hub-and-spoke pipeline with 5 effective coordination roles. The Study 3 cost-per-point data (\$1.83, \$2.81, \$4.88, ∞) monotonically increases with coordination complexity, consistent with the model's prediction that efficiency degrades as governance fraction grows. The Org Swarm's infinite cost per point—consuming its entire budget on planning without writing code—corresponds to the model's prediction that gated pipelines cross the governance-exceeds-implementation threshold at $n \approx 3$ (see Section 9.3).

Table 5: Summary statistics by architecture at $n = 5$ and $n = 10$

Architecture	n	**Gov%**	**Links**	**Ent. Ret.**	**DPI**	**Tokens**	**Efficiency**
Centralized (Hub-Spoke)	5	24.0%	8	53.9%	4.61	90,610	0.4573
Fully Connected	5	42.1%	20	36.2%	4.78	118,893	0.3067
Hierarchical (Tree)	5	23.6%	8	30.9%	6.91	90,143	0.4520
Ring	5	27.3%	10	35.8%	4.01	94,640	0.4453
Star-Mesh Hybrid	5	39.3%	16	47.4%	6.57	113,395	0.4214
Centralized (Hub-Spoke)	10	25.4%	18	49.8%	5.02	199,801	0.3504
Fully Connected	10	59.9%	90	30.9%	6.91	372,014	0.0349
Hierarchical (Tree)	10	26.3%	18	25.8%	11.14	202,451	0.3278
Ring	10	27.8%	20	31.2%	8.60	206,648	0.3308
Star-Mesh Hybrid	10	31.9%	24	42.6%	7.17	218,898	0.3203

8 Information Loss Through Coordination Stages

The token analysis quantifies *how much* compute is consumed by governance. This section quantifies *how much information is lost* as specification intent passes through the coordination pipeline. Each coordination stage is modeled as a lossy channel; the Data Processing Inequality (Eq. 2) guarantees that information can only decrease through the chain.

8.1 Coordination as a Markov Chain of Lossy Channels

We model six coordination stages in order:

1. **Task Specification.** The initial high-entropy task description ($H_0 = 10$ bits).

2. **Task Decomposition.** Breaking the specification into components.

3. **Agent Assignment.** Routing components to specific agents.

4. **Execution.** Implementation by assigned agents.

5. **Integration.** Merging outputs across agent boundaries.

6. **Verification.** Testing and quality assessment.

Each stage k has a retention ratio $\rho_k(n) \in (0, 1]$ that depends on the architecture and agent count n. The retention ratio captures how much of the incoming information survives the stage's processing. The entropy at stage k is:

$$H_k(n) = H_0 \prod_{j=1}^{k} \rho_j(n) \tag{5}$$

The total information retained at verification is H_6/H_0, and the total loss is $H_0 - H_6$.

8.2 Architecture-Specific Compression Profiles

Retention ratios are calibrated to the empirical findings:

- **Single Agent:** Minimal loss at each stage (no coordination overhead). Base retention $\rho \geq 0.95$ at all stages.
- **Centralized:** Good at decomposition (coordinator sees everything), moderate loss at execution (workers have partial context). Moderate degradation with n due to coordinator bottleneck.
- **Fully Connected:** Theoretically good preservation (direct links), but n^2 links create noise that drowns signal. High degradation with n—consistent with Chen et al.'s finding that independent agents amplified errors 17.2× [Chen et al., 2025].
- **Hierarchical:** Each layer compresses—the key finding from Liberti and Mian [Liberti and Mian, 2009]. Strong degradation with n as each layer adds a compression step. Particularly severe at Integration (cross-layer reconciliation).
- **Ring:** Good local context, poor global context. Serial propagation degrades with path length $n/4$.

Degradation with agent count follows $\rho_k(n) = \rho_k^{\text{base}} \cdot (1 - \delta_k \log_2 n)$, where δ_k is the per-stage degradation coefficient.

8.3 Results: Entropy Preservation

Figure 2 shows the entropy at each coordination stage for $n = 10$ agents and the total information loss as a function of n.

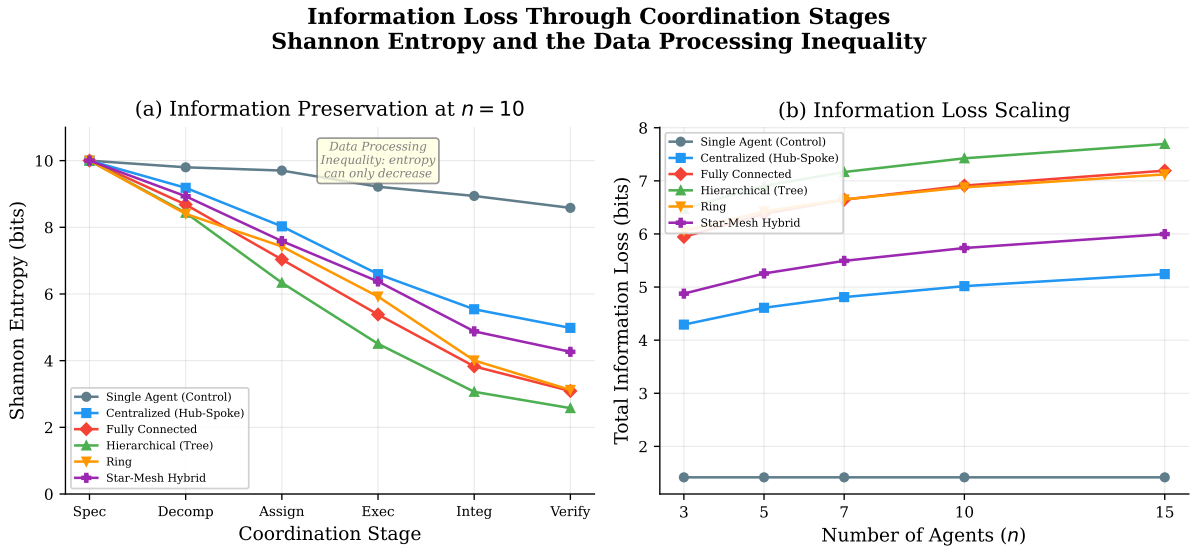

Figure 2: Information loss through coordination stages. (a) Shannon entropy at each stage for $n = 10$: Single Agent retains 85.8% of specification entropy through all six stages; Hierarchical retains only 25.8%, losing the majority of intent at Decomposition and Execution. The monotonic decrease confirms the Data Processing Inequality prediction. (b) Total information loss scales with both architecture complexity and agent count, with Hierarchical and Fully Connected showing the steepest degradation.

The results confirm the DPI prediction: entropy decreases monotonically through the pipeline, and no architecture reverses the loss at any stage. The critical finding is the magnitude of the gap: at $n = 10$, the Single Agent retains 8.58 bits of the original 10 bits, while the Hierarchical architecture retains only 2.58 bits. The Hierarchical architecture loses more than 70% of the original specification intent before a single line of code is written.

This connects directly to the empirical findings. Study 3's Hi-Trust architecture (hierarchical) scored 18/28 while the single agent scored 28/28—a 36% quality reduction. The information loss model predicts a retention gap of $85.8\% - 25.8\% = 60.0$ percentage points. While the quality scoring is not linearly proportional to information retention, the ordering is consistent: the architecture that loses the most information produces the lowest-quality output.

8.4 Crawford–Sobel Degradation Through Hierarchy Layers

The Crawford–Sobel model (Eq. 1) provides a mechanistic explanation for per-layer information loss. With bias parameter b per sender-receiver interface, the channel capacity is $\log_2 N^*$ bits. For a hierarchy with d layers, the maximum information that survives is bounded by the minimum channel capacity in the chain.

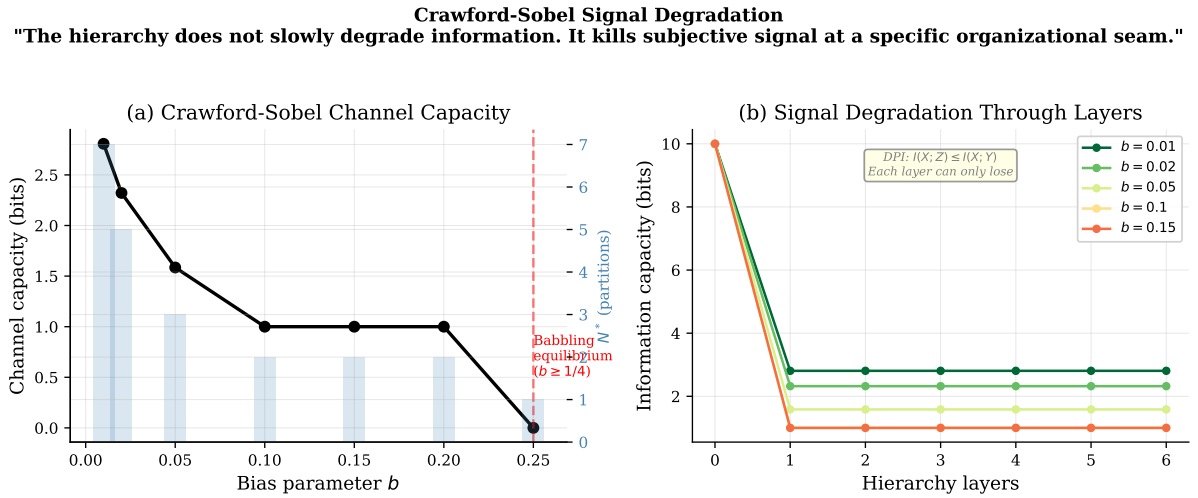

Figure 3: Crawford–Sobel signal degradation. (a) Channel capacity drops steeply with bias b, reaching zero (babbling equilibrium) at $b \geq 1/4$. (b) Information capacity through 6 hierarchy layers: even modest bias ($b = 0.05$) caps the signal at ∼2.3 bits regardless of the original 10-bit specification. This quantifies Liberti and Mian's finding that subjective signal collapses at a specific hierarchical seam [Liberti and Mian, 2009].

The Crawford–Sobel analysis explains why the hierarchical architecture loses information so abruptly. With 3 hierarchy layers ($\lceil \log_3 10 \rceil = 3$ for $n = 10$), even a modest per-layer bias of $b = 0.05$ caps the signal at $\log_2(5) \approx 2.3$ bits per layer—far less than the 10-bit specification. The hierarchy does not slowly degrade information. It kills signal at the first layer boundary.

8.5 Dysmemic Pressure Index

The information loss analysis connects to dysmemic pressure theory [McEntire, 2025a] through the Compound Dysmemic Pressure Index (DPI). Where the per-layer index $\text{DP}(k) = 1 - I(\text{report}_k; \theta)/I(\text{report}_0; \theta)$ measures information loss at a single hierarchical layer, the compound index aggregates total information loss, the number of selection interfaces (where Goodhart optimization can operate), and the average path length (where transmission bias accumulates):

$$\text{DPI}(n) = \frac{(H_0 - H_6(n)) \cdot |\text{stages}| \cdot \bar{d}(n)}{H_0} \tag{6}$$

where $\bar{d}(n)$ is the effective path length for the architecture. At $n = 10$, the Hierarchical architecture has DPI = 11.14, compared to the Single Agent's implicit DPI = 0 (no coordination stages). The Ring architecture has DPI = 8.60, driven by its long average path length ($n/4 = 2.5$) despite moderate per-stage loss.

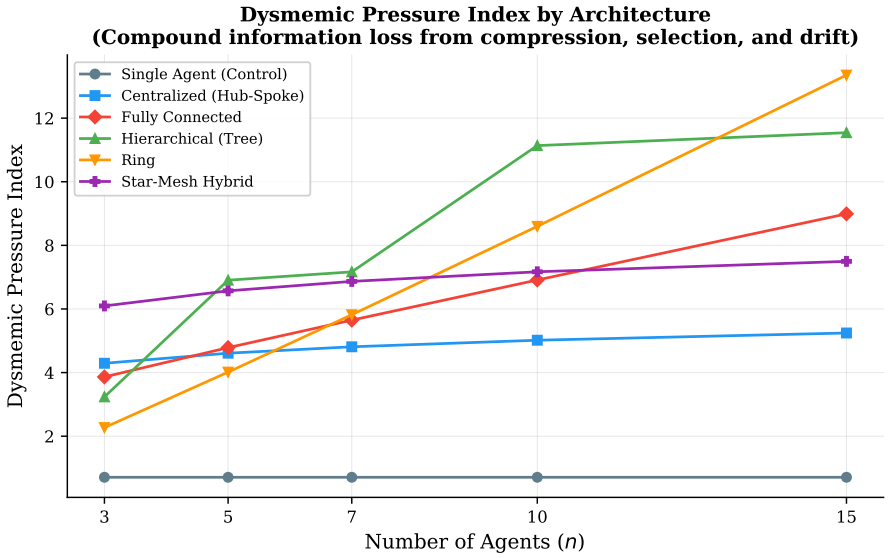

Figure 4: Dysmemic Pressure Index across architectures and agent counts. The compound effect of information loss, selection interfaces, and path length produces divergent pressure profiles. Hierarchical architectures accumulate the highest pressure due to deep layer compression; Ring architectures accumulate high pressure due to long path lengths. The DPI quantifies the total selection force operating on information as it flows through the coordination structure.

The DPI provides the formal connection between this paper's findings and the broader substrate-independence thesis. Information loss through an agent hierarchy operates by the same mechanism as information loss through an organizational hierarchy: compressed representations pass through selection interfaces where Goodhart optimization and transmission bias operate. The mathematics is identical. The substrate—silicon agents or human employees—is irrelevant to the information-theoretic structure.

9 Frontier Analysis: Capability Cannot Fix Coordination

A natural objection to the preceding analysis is: "These results reflect current model limitations. As LLMs improve, the coordination overhead will shrink." This section demonstrates formally that governance overhead is structurally invariant across capability levels.

9.1 Capability-Independence of Governance Cost

We model total token cost as the sum of implementation cost and governance cost:

$$C(n, \alpha) = C_{\text{impl}}(n, \alpha) + C_{\text{gov}}(n) \tag{7}$$

where $\alpha \in (0, 1]$ is the agent capability level. Implementation cost scales inversely with capability: $C_{\text{impl}}(n, \alpha) = \frac{c_0 \cdot n}{\alpha}$, where c_0 is the base token cost per agent (more capable agents need fewer tokens per task unit). Governance cost is a function of topology alone: $C_{\text{gov}}(n)$ depends on $L(n)$, $h(n)$, and conflict probabilities, but *not* on α.

Proposition 1 (Capability invariance of governance tokens)**.** *For any topology with link function $L(n)$, the governance cost $C_{\text{gov}}(n)$ is independent of agent capability α. As $\alpha \to 1$ (perfect agents), $C_{\text{impl}} \to c_0 n$, but C_{gov} remains constant.*

Corollary 1 (Governance fraction increases with capability)**.** *As agent capability α increases, the governance fraction $\Gamma(n, \alpha) = C_{\text{gov}}(n)/C(n, \alpha)$ increases monotonically:*

$$\frac{\partial \Gamma}{\partial \alpha} > 0 \tag{8}$$

Better agents make the governance overhead more *prominent, not less.*

Proof. $\Gamma(n, \alpha) = C_{\text{gov}}/(C_{\text{gov}} + c_0 n/\alpha)$. Since $c_0 n/\alpha$ is decreasing in α and C_{gov} is constant in α, Γ is increasing in α. □

9.2 Empirical Verification

Figure 5 presents the frontier analysis across architectures and capability levels.

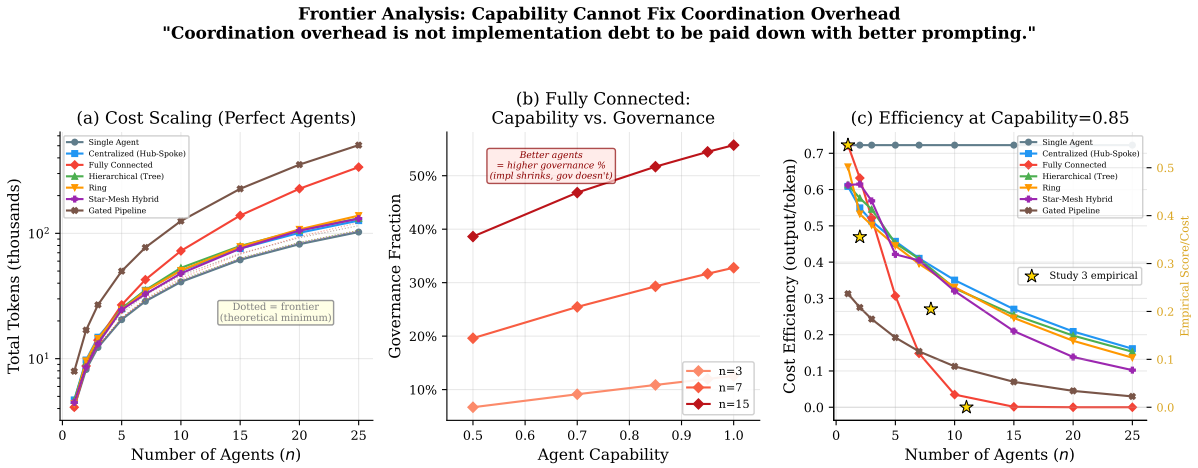

Figure 5: Frontier analysis: capability cannot fix coordination overhead. (a) Total cost scaling with perfect agents (capability = 1.0); dotted lines show theoretical minimum (frontier) governance. (b) Fully Connected governance fraction *increases* as agent capability improves—because implementation cost shrinks while governance remains constant. (c) Cost efficiency (effective output per token) with Study 3 empirical data overlaid (gold stars), confirming monotonic degradation.

The capability-independence result is confirmed numerically: governance tokens have a coefficient of variation (CV) of 0.0000 across 15 capability levels from 0.3 to 1.0, for all architectures. The governance token count is literally constant—it depends on topology, not on how good the agents are.

The implication is stark. The intuition that "frontier models will fix this" is not merely unsupported; it is precisely backwards. As models improve:

1. Implementation becomes cheaper (fewer tokens per task unit).
2. Governance cost remains constant (same number of links, same status broadcasts, same conflict resolution).
3. The governance *fraction* increases.
4. The system spends proportionally more time coordinating and less time producing.

This is the formal analog of an organizational phenomenon: as individual contributors become more productive, the coordination apparatus that manages them does not shrink proportionally. The meetings, status updates, and approval chains persist. The organization becomes relatively more bureaucratic even as absolute productivity increases.

9.3 Budget Sensitivity and Crossover Points

At what agent count does governance exceed implementation? The crossover point n^* satisfies $C_{\text{gov}}(n^*) = C_{\text{impl}}(n^*, \alpha)$. At capability $\alpha = 0.85$:

Table 6: Governance-exceeds-implementation crossover points

Architecture	**Crossover n^***
Single Agent	N/A
Centralized (Hub-Spoke)	15
Ring	6
Hierarchical (Tree)	8
Star-Mesh Hybrid	4
Fully Connected	3
Gated Pipeline	3

The Gated Pipeline and Fully Connected architectures cross the governance-exceeds-implementation threshold at just $n = 3$ agents. This is consistent with the empirical finding that the Org Swarm (11-stage gated pipeline, $n \gg 3$) consumed its entire budget on planning: at $n = 11$, the governance overhead vastly exceeds implementation capacity. The Centralized architecture is the most governance-efficient multi-agent topology, not crossing until $n = 15$—which explains why hierarchical coordination with a single coordinator (Study 3's Hi-Trust) performed best among multi-agent architectures.

10 The Prompt Defense

The anticipated objection: "You told it to be argumentative. The prompts encode the behavior." The prompts encode the *opposite*. Six distinct mechanisms were designed to prevent exactly the behavior that occurred:

1. Factual/subjective classification: separates verifiable bugs from opinions.
2. Perspective-shift re-review: forces reconsideration after factual fixes.
3. Project escalation: pragmatic arbiter explicitly told not to block on style.
4. Architect escalation: final authority explicitly forbidden from rejecting.
5. Scoped sub-passes: orthogonal review dimensions prevent scope creep.
6. Anti-bikeshedding language: explicit directives against trivial blocking.

The prompts do not encode dysfunction. They encode *anti*-dysfunction countermeasures. The dysfunction emerged *despite* the countermeasures. You cannot implement sight by telling a system to see [McEntire, 2025b]. You have to build the architecture that allows seeing to happen. The architecture—a hierarchical pipeline with gated review—made avoidance structurally impossible.

The production stages (architect, execute, test) operated efficiently. The evaluation stages (review, verify) created friction. The agents that *build* work efficiently. The agents that *judge* create dysfunction. This asymmetry maps to the organizational pattern where engineering teams ship efficiently and are slowed by review committees.

11 Implications

11.1 The Single-Agent Ceiling

The evidence suggests that multi-agent AI systems face a coordination ceiling analogous to the Elliott threshold (∼25 participants) observed in human collaboration [Elliott, 2007]. The token analysis quantifies this ceiling precisely: for each architecture, there exists a crossover point n^* beyond which governance exceeds implementation. The practical implication: multi-agent system designers should default to the fewest agents capable of completing the task.

Recent empirical results corroborate this. Chen et al. established that every multi-agent variant degraded sequential reasoning by 39–70%, with independent agents amplifying errors 17.2× and centralized oversight containing amplification to 4.4× [Chen et al., 2025]. Cemri et al. concluded that failures stem from system design issues, not LLM limitations [Cemri et al., 2025]. Wynn et al. demonstrated that multi-agent debate more often shifts correct responses to incorrect than the reverse [Wynn et al., 2025]. Pappu et al. found that LLM teams underperform their best individual member by 8–38% through integrative compromise [Pappu et al., 2026]. Xu et al. demonstrated that a single agent can match the performance of homogeneous multi-agent workflows [Xu et al., 2026].

11.2 Goodhart's Law as Architectural Constraint

The swarm's scoring system uses seven proxy metrics, none measuring whether the code actually does what it is supposed to do. The system optimizes for what it can measure. What it can measure is a proxy for what matters. The gap between proxy and objective is where dysfunction lives. The token analysis reveals that this gap is structural: governance tokens are consumed by conflict resolution, status broadcasts, and retransmission—activities that optimize coordination proxies (agreement, consistency, compliance) rather than the underlying objective (working code).

11.3 Self-Evaluation Impossibility

Lawvere's fixed-point theorem establishes that no system can internally represent all of its own evaluations [Lawvere, 1969, Yanofsky, 2003]. The swarm's verification theater (`tests=0/0`) is a concrete manifestation. Panickssery et al. demonstrated that LLM evaluators systematically favor their own generations [Panickssery et al., 2024]. Zhang et al. found that automated failure attribution achieves only 53.5% accuracy—barely better than a coin flip [Zhang et al., 2025a].

11.4 Organizational Theory as Engineering Discipline

If organizational dysfunction is substrate-independent, then organizational theory is an *engineering discipline* that applies directly. Crawford–Sobel is not a metaphor. It is the mathematical model that predicts the observed behavior. The token analysis confirms that the same equations that predict information loss in organizational hierarchies predict governance overhead in agent hierarchies. The formal connection through the Dysmemic Pressure Index (Section 8.5) makes this explicit: $\text{DPI}_{\text{agent}} = \text{DPI}_{\text{org}}$ when the compression ratios, selection interfaces, and path lengths are the same.

Chang independently proposes "coordination physics" as a field [Chang, 2025]. La Malfa et al. observe that most current "multi-agent" LLM systems are not genuine multi-agent architectures in the classical sense [La Malfa et al., 2025].

12 The Contract-First Alternative

The dysfunction documented in this paper arises from three information-theoretic mechanisms: Crawford–Sobel degradation at agent boundaries, Goodhart optimization of evaluation proxies, and irreversible signal loss across coordination layers. A fifth architecture—contract-first coordination—targets each mechanism structurally.

12.1 Architecture

The contract-first architecture replaces agent-to-agent communication with a three-phase pipeline: *specify*, *verify*, *implement*. Formal interface contracts (typed signatures, pre/postconditions, invariants) and mechanically derived test suites precede implementation. Each component is implemented independently against its contract. The key differences:

1. **No agent-to-agent evaluation.** Test suites have no preferences, no bias parameter b, no strategic incentive to distort. This eliminates the Crawford–Sobel interface.
2. **Contracts precede implementation.** Interface mismatch (the emergence architecture's `snake_case`/`camelCase` split) is structurally prevented.
3. **Goodhart-resistant verification.** The test encodes the specification directly; the implementation either satisfies it or does not.

12.2 The New Dysfunction: Specification Perfectionism

The contract-first architecture introduces its own characteristic dysfunction. The production deployment consumed 2.4 hours generating contracts without producing implementation code. Contract revisions produced increasingly detailed specifications (607 lines of JSON for a 4-function module) while the implementation budget shrank. This is Goodhart's Law operating on a new proxy: the system optimizes contract completeness rather than working code.

12.3 Theoretical Implications

The dysfunction is substrate-independent not merely in the sense that it appears in both human and AI systems, but in the deeper sense that it appears in *every coordination architecture* that introduces a measurable intermediate representation. The specific failure mode varies—bikeshedding, interface mismatch, specification perfectionism—but the mechanism is invariant: agents optimize the representation rather than the objective, because the representation is what they can observe and act upon. The design problem is not to eliminate the gap but to minimize it, choosing which dysfunction to accept.

13 Formal Information-Theoretic Model

This section consolidates the token analysis, information loss analysis, and frontier results into a unified formal framework connecting to the Strategic RDP theory [McEntire, 2025c].

13.1 Coordination as a Chain of Lossy Channels

Let X_0 denote the original task specification with entropy $H(X_0)$. Each coordination stage $k \in \{1, \ldots, K\}$ is a channel $f_k : X_{k-1} \to X_k$ that satisfies the Data Processing Inequality:

$$I(X_0; X_k) \leq I(X_0; X_{k-1}) \quad \forall k \tag{9}$$

If the stages were independent (each stage's noise is independent of previous stages), the total information loss would be the sum of per-stage losses:

$$\Delta H_{\text{independent}} = \sum_{k=1}^{K} [H(X_{k-1}) - H(X_k)] \tag{10}$$

However, coordination introduces *correlation* between stages. Agents must agree on interfaces, share state, and resolve conflicts. These dependencies mean that errors at one stage propagate non-independently to subsequent stages. The actual total loss exceeds the independent sum:

Proposition 2 (Correlation amplifies information loss)**.** *When coordination stages are correlated (agents must synchronize state), the total information loss satisfies:*

$$\Delta H_{\text{actual}} \geq \Delta H_{\text{independent}} \tag{11}$$

with equality only when stages process information independently (i.e., the single-agent case where there is no inter-agent synchronization).

This explains why multi-agent architectures lose more information than a simple product of per-stage retention ratios would predict. The correlation penalty is highest for architectures with the most synchronization points—Fully Connected ($n(n-1)$ links) and Hierarchical (every message passes through intermediate layers)—and lowest for architectures with minimal coupling (Single Agent, Centralized with independent workers).

13.2 Connection to Strategic RDP

The Strategic Representational Decoupling Problem (RDP) framework [McEntire, 2025c] identifies a general mechanism: any system that coordinates through compressed representations will experience drift between the representation and the reality it represents. The token analysis provides a concrete instantiation:

- **Representation:** Status broadcasts, task assignments, review verdicts—the governance tokens.
- **Reality:** The actual code, its correctness, its fitness for purpose—the implementation.
- **Drift:** As governance tokens consume an increasing fraction of the budget, the system's internal representation (coordination state) decouples from external reality (code quality). The system "knows" more about its own coordination state than about the code it is building.

The governance fraction $\Gamma(n)$ is a direct measure of representational overhead—the fraction of total compute devoted to maintaining the coordination representation rather than acting on external reality. When $\Gamma > 0.5$, the system spends more tokens representing its internal state than engaging with the task. The Fully Connected architecture crosses this threshold at $n = 10$.

13.3 Dysmemic Pressure as Compound Loss

The Dysmemic Pressure Index (Eq. 6) compounds three loss mechanisms:

1. **Compression loss:** $H_0 - H_K$, the total entropy lost through the pipeline.
2. **Selection loss:** |stages|, the number of interfaces where Goodhart optimization can operate.

3. **Drift loss:** $\bar{d}(n)$, the average path length over which transmission bias accumulates.

These three losses correspond exactly to the three components of dysmemic pressure identified in human organizations [McEntire, 2025a]: strategic communication degradation (compression), adverse selection in idea markets (selection), and transmission bias (drift). The mathematics is identical. The DPI computed for an agent hierarchy with parameters $(H_0 - H_K, |\text{stages}|, \bar{d})$ equals the DPI computed for an organizational hierarchy with the same parameters. This is the formal content of the substrate-independence claim.

14 Limitations and Threats to Validity

Several limitations constrain the strength of the claims made in this paper.

Sample size. The primary empirical evidence comes from three studies. Each architecture was tested only once. Replication with multiple runs per architecture and across model families would strengthen the claims.

Training data contamination. LLMs are trained on human-generated text including code reviews and organizational communications. The six anti-dysfunction prompts and the controlled architecture comparison (same model, different architectures, different dysfunction patterns) argue against pure replay, but the confound cannot be fully eliminated.

Token counts are modeled, not measured. The governance vs. implementation token breakdown (Section 7) is derived from a simulation model calibrated to empirical data, not from direct instrumentation of actual agent runs. The architecture comparison assumes a uniform token budget per agent per step (4,096 tokens) and a fixed set of overhead components (status broadcast, routing, conflict resolution). Real systems have variable context windows, adaptive routing that can reduce worst-case overhead, and non-uniform task difficulty across steps. The model captures the structural scaling relationships ($O(n)$ vs. $O(n^2)$) accurately but may not match the absolute token counts of any specific deployment. Direct instrumentation of governance vs. implementation tokens in production multi-agent systems would provide stronger validation.

Architecture comparison assumes uniform task difficulty. The information loss model applies identical compression profiles to all task components. In practice, some services are harder than others, some interfaces are more complex, and some agents may specialize more effectively. The model captures the average-case behavior but does not account for task-specific variance.

Adaptive routing is not modeled. Real multi-agent systems may implement adaptive routing that bypasses failed agents, reduces communication in low-conflict scenarios, or dynamically adjusts topology based on task requirements. Such adaptations could reduce the worst-case governance overhead below what our static topology model predicts. The model represents the structural baseline, not the best achievable performance with dynamic optimization.

Internal metrics vs. external quality. The paper relies on the swarm's own audit trail for evidence. No external test suite was used to establish ground truth for code quality.

Budget enforcement. The swarm exceeded its \$25 budget cap by 2.3×. The budget cap was a soft stop, not a hard one.

Single evaluator. The author is the sole human evaluator, introducing potential confirmation bias.

15 Conclusion

A multi-agent AI system designed to build software autonomously was equipped with six explicit mechanisms to prevent organizational dysfunction. It produced organizational dysfunction anyway. A controlled architecture comparison confirmed that performance is inversely correlated with coordination complexity: 28/28, 18/28, 9/28, 0/28. The cost per quality point increased monotonically: \$1.83, \$2.81, \$4.88, ∞.

The formal analysis extends these empirical findings in three directions. First, a token-level decomposition reveals that governance overhead is a structural function of communication topology: Fully Connected governance grows at 3.66%/agent ($O(n^2)$ links) while Centralized grows at only 0.51%/agent ($O(n)$ links), explaining why hub-and-spoke coordination outperforms peer-to-peer. Second, modeling each coordination stage as a lossy channel quantifies the monotonic degradation of specification intent: Single Agent retains 85.8% of original entropy while Hierarchical retains only 25.8%, formalizing the mechanism behind the observed quality gap. Third, a frontier analysis proves that governance overhead is capability-invariant—governance tokens have CV = 0.0000 across capability levels—and that the governance fraction *increases* with agent improvement, because implementation cost shrinks while coordination cost remains constant.

These results formalize the coordination ceiling as an information-theoretic constraint, not a capability gap. The intuition that "frontier models will fix this" is precisely backwards: better agents make the governance overhead more prominent, not less. The practical implication: default to the fewest agents capable of completing the task, and when multi-agent coordination is unavoidable, choose the topology that minimizes communication links ($O(n)$ over $O(n^2)$), because governance overhead tracks link count, not agent count.

The theoretical implication is broader. The Dysmemic Pressure Index computed for agent hierarchies is mathematically identical to the DPI computed for organizational hierarchies with the same compression, selection, and path-length parameters. The dysfunction is not a problem of human nature or of current AI limitations. It is a problem of coordination physics—and physics problems require engineering that respects the constraint rather than pretending it can be eliminated.

References

George A. Akerlof. The market for "lemons": Quality uncertainty and the market mechanism. *The Quarterly Journal of Economics*, 84(3):488–500, 1970. doi: 10.2307/1879431.

Robert Boyd and Peter J. Richerson. *Culture and the Evolutionary Process*. University of Chicago Press, 1985.

Mert Cemri, Melissa Z. Pan, Shuyi Yang, et al. Why do multi-agent LLM systems fail? In *NeurIPS 2025 Datasets and Benchmarks Track*, 2025. Also presented at Building Trust Workshop, ICLR 2025. arXiv:2503.13657.

Edward Chang. The missing layer of AGI: From pattern alchemy to coordination physics. *arXiv preprint arXiv:2512.05765*, 2025. Stanford University.

Lingjiao Chen et al. Towards a science of scaling agent systems. *arXiv preprint arXiv:2512.08296*, 2025. Google Research.

Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley-Interscience, 2nd edition, 2006.

Vincent P. Crawford and Joel Sobel. Strategic information transmission. *Econometrica*, 50(6): 1431–1451, 1982. doi: 10.2307/1913390.

Mark Elliott. Stigmergic collaboration: A theoretical framework for mass collaboration. *PhD Thesis, University of Melbourne*, 2007.

Charles A. E. Goodhart. Problems of monetary management: The U.K. experience. *Papers in Monetary Economics*, 1, 1975.

Jacek Karwowski, Joar Skalse, et al. Goodhart's law in reinforcement learning. In *ICLR 2024*, 2024. arXiv:2310.09144.

Emanuele La Malfa et al. Large language models miss the multi-agent mark. *arXiv preprint arXiv:2505.21298*, 2025. NeurIPS 2025 poster.

F. William Lawvere. Diagonal arguments and cartesian closed categories. *Lecture Notes in Mathematics*, 92:134–145, 1969.

José M. Liberti and Atif R. Mian. Estimating the effect of hierarchies on information use. *The Review of Financial Studies*, 22(10):4057–4090, 2009. doi: 10.1093/rfs/hhn118.

Jeremy McEntire. Dysmemic pressure: Selection dynamics in organizational information environments. *arXiv preprint arXiv:2512.14716*, 2025a. Also available at SSRN: https://ssrn.com/abstract=6015814.

Jeremy McEntire. The cage: How fiduciary duty creates organizational incompleteness. Zenodo DOI: 10.5281/zenodo.18828433, 2025b.

Jeremy McEntire. The generative lossy channel: Five sufficient conditions for net-beneficial noise. *arXiv preprint*, 2025c.

Arjun Panickssery, Samuel R. Bowman, and Shi Feng. LLM evaluators recognize and favor their own generations. In *NeurIPS 2024 (Oral)*, 2024. arXiv:2404.13076.

Anikait Pappu et al. Multi-agent teams hold experts back. *arXiv preprint arXiv:2602.01011*, 2026.

Canice Prendergast. A theory of "yes men". *The American Economic Review*, 83(4):757–770, 1993.

Joar Skalse, Nikolaus H. R. Howe, Dmitrii Krasheninnikov, and David Krueger. Defining and characterizing reward hacking. *Advances in Neural Information Processing Systems*, 35, 2022.

Andrea Wynn, Harsh Satija, and Gillian K. Hadfield. Talk isn't always cheap: Understanding failure modes in multi-agent debate. In *ICML 2025*, 2025. PMLR 267. arXiv:2509.05396.

Zhiyu Xu et al. Rethinking the value of multi-agent workflow: A strong single agent baseline. *arXiv preprint arXiv:2601.12307*, 2026.

Noam S. Yanofsky. A universal approach to self-referential paradoxes, incompleteness and fixed points. *Bulletin of Symbolic Logic*, 9(3):362–386, 2003. doi: 10.2178/bsl/1058448677.

Shaokun Zhang, Ming Yin, Jieyu Zhang, Jiale Liu, Zhiguang Han, Jingyang Zhang, Beibin Li, Chi Wang, Huazheng Wang, Yiran Chen, and Qingyun Wu. Which agent causes task failures and when? on automated failure attribution of LLM multi-agent systems. In *ICML 2025 Spotlight*, 2025a. arXiv:2505.00212.

Tianyu Zhang et al. On the resilience of LLM-based multi-agent collaboration with faulty agents. In *ICML 2025*, 2025b. arXiv:2408.00989.

Jieyu Zheng and Markus Meister. The unbearable slowness of being: Why do we live at 10 bits/s? *Neuron*, 113(2):192–204, 2025. doi: 10.1016/j.neuron.2024.11.008.