Synergia: A Journal of Multidisciplinary Innovation

ISSN: 2164-996X

Federated Agentic Intelligence: Integrating Multi-Agent Systems with Decentralized Foundation Models

Sudhakar Murthy Molli

B.Tech , MS, Independent Researcher, USA

Mollisudhakar@gmail.com

Vol. 6 No. 6 (2024): Synergia

Abstract: Autonomous agent populations increasingly operate across organizational and geographic boundaries -- enterprise knowledge workers, vehicle fleets, factory robots, and personal device assistants -- generating experience that could improve a shared foundation model but that cannot be centralized due to privacy, bandwidth, or data-sovereignty constraints. This paper presents Federated Agentic Intelligence (FAI), an architecture that integrates multi-agent coordination with federated learning of a shared decentralized foundation model, allowing autonomous agents to both act collaboratively on multi-step tasks and continuously improve a common backbone from locally generated experience without exchanging raw data. FAI combines hierarchical multi-agent consensus, staleness-aware asynchronous federated aggregation, shared episodic memory, and reputation-weighted Byzantine-robust aggregation to handle the combined challenges of distributed decision-making and distributed learning under unreliable, heterogeneous, and occasionally adversarial participants. We evaluate FAI across simulated agent populations spanning enterprise, vehicular, industrial, and personal-device deployments. Our results show that federated hierarchical consensus improves multi-step task success rate by 13.9 percentage points over a centralized-planner baseline and by 27.4 points over uncoordinated independent agents, that staleness-aware asynchronous aggregation preserves 70.2% decision quality at 32 rounds of update staleness versus 44.2% for synchronous aggregation, and that reputation-weighted Byzantine-robust consensus sustains 63.5% task success with 30% adversarial agents present versus a collapse to 14.6% for undefended coordination. We conclude with a discussion of open challenges in cross-agent credit assignment, emergent multi-agent behavior under federated learning dynamics, and the governance of autonomous agent populations that jointly act and learn without centralized oversight.

Keywords: federated learning, multi-agent systems, agentic AI, decentralized foundation models, Byzantine robustness, consensus algorithms, edge intelligence, distributed decision-making

1. Introduction

Two lines of progress in artificial intelligence have advanced largely in parallel. Multi-agent systems research has produced increasingly capable frameworks for coordinating autonomous agents on complex, multi-step tasks, typically assuming that all agents share a common, centrally trained foundation model. Federated learning research has produced increasingly effective techniques for training a shared model across distributed clients without centralizing their data, typically assuming a comparatively simple learning task such as classification or next-token prediction, without addressing how the resulting model is subsequently used by autonomous agents that must coordinate with one another in real time. This paper addresses the intersection of these two lines of work: how should a population of autonomous agents, each grounded in local data and operating under privacy or bandwidth constraints that preclude centralizing their experience, both act collaboratively on shared tasks and continuously improve a common underlying foundation model?

This question is increasingly practical rather than purely academic. Enterprise deployments of agentic assistants across many business units generate task-completion experience that could improve a shared backbone model but that enterprises are reluctant or unable to centralize due to confidentiality. Autonomous vehicle fleets and industrial robot populations generate sensor and control experience at a volume that makes centralization bandwidth-prohibitive even where no privacy constraint applies. Personal device assistants generate experience that is both privacy-sensitive and, individually, too sparse to train a useful model without aggregation across a large population. In each case, the population would benefit from a shared, continuously improving foundation model, but only if that model can be trained federatively, and only if the resulting model can still support real-time multi-agent coordination despite each agent holding a possibly stale, partially personalized copy of the shared backbone.

1.1 Motivation

Three observations motivate the architecture presented in this paper. First, multi-agent coordination protocols developed under the assumption of a single, static, centrally available model do not account for the possibility that different agents hold different versions of that model at different times, as is unavoidable under asynchronous federated learning; coordination and learning are coupled problems that must be co-designed rather than solved independently. Second, the same properties that make an agent population attractive for federated learning -- large scale, heterogeneity, limited central oversight -- also make it more exposed to unreliable or adversarial participants than the client populations typically studied in federated learning research, since agents that both act and contribute learning updates present a broader attack surface than passive data contributors. Third, agent populations performing related but distinct tasks across different domains represent an underexploited opportunity for cross-domain skill transfer through a shared federated backbone, provided the federation architecture can reconcile substantially different task distributions across domains without degrading any single domain's performance.

1.2 Contributions

This paper makes the following contributions:

- Federated Agentic Intelligence (FAI), an architecture that integrates hierarchical multi-agent consensus with staleness-aware asynchronous federated learning of a shared decentralized foundation model.
- A hierarchical consensus mechanism that bounds communication overhead as agent population scales while preserving multi-step task coordination quality.
- A reputation-weighted, Byzantine-robust aggregation and consensus mechanism addressing the combined coordination-and-learning attack surface of agent populations that both act and contribute model updates.
- An empirical study of cross-domain skill transfer through federated agentic fine-tuning across enterprise, vehicular, industrial, and personal-device agent populations.
- A discussion of open challenges in cross-agent credit assignment, emergent behavior under federated learning dynamics, and governance of jointly acting and learning agent populations.

1.3 Paper Organization

Section 2 surveys related work in multi-agent systems, federated learning, and robust distributed consensus. Section 3 formalizes the coupled coordination-and-learning problem. Section 4 presents the FAI architecture, with Sections 5 through 8 detailing its consensus, federated learning, robustness, and shared-memory components respectively. Section 9 describes our experimental methodology, Section 10 presents results, Section 11 discusses deployment considerations, Section 12 discusses open challenges, and Section 13 concludes.

2. Related Work

2.1 Multi-Agent Systems and Coordination

Multi-agent systems research has developed a range of coordination patterns, from fully centralized planning in which a single controller assigns tasks to subordinate agents, to fully decentralized approaches in which agents coordinate through local communication and emergent consensus without any central authority. Centralized planning typically achieves higher coordination quality at the cost of a single point of failure and a communication bottleneck at the planner, while fully decentralized approaches trade some coordination quality for robustness and scalability. Hierarchical approaches, in which coordination responsibility is distributed across intermediate levels between fully centralized and fully decentralized extremes, have been proposed as a middle ground, and form the basis for this paper's consensus design in Section 5.

2.2 Federated Learning

Federated learning enables collaborative model training across distributed clients without centralizing their raw data, using techniques including proximal regularization for statistical heterogeneity, server-side adaptive optimization, and secure aggregation for privacy protection, as established in the broader federated learning literature. Asynchronous federated learning variants relax the requirement that all clients participate in every round, accommodating clients with heterogeneous availability and compute capacity at the cost of handling model update staleness, a challenge directly relevant to the agent populations this paper studies, where individual agents cannot be assumed to synchronize with the global aggregator on a fixed schedule.

2.3 Byzantine-Robust Distributed Learning and Consensus

Byzantine-robust aggregation rules bound the influence of malicious or faulty participants on a federated learning system's aggregated model, while Byzantine fault-tolerant consensus protocols provide analogous guarantees for distributed systems reaching agreement on a shared state or decision. This paper's robustness design in Section 7 draws on both literatures, since a federated agentic system is exposed to Byzantine risk in both its learning dimension (a malicious agent's model update) and its coordination dimension (a malicious agent's reported task state or proposed action).

2.4 Agentic Foundation Models and Tool Use

Foundation models adapted for agentic use -- planning, tool invocation, and multi-step reasoning -- have typically been trained and deployed as a single, centrally maintained model queried by all agents in a deployment. This paper's contribution relative to this line of work is to decentralize both the training and, to a degree via personalized adapters, the deployment of the underlying model, while preserving the coordination capability that centralized deployment made straightforward to achieve.

2.5 Positioning of This Work

Prior work has largely treated multi-agent coordination and federated learning as separate problems, addressing coordination under an assumed static shared model or addressing federated learning without considering how the resulting model supports real-time multi-agent action. This paper's contribution is to treat coordination and federated learning as a single coupled system, evaluating both the task-coordination and the model-learning consequences of design choices such as consensus topology, aggregation staleness tolerance, and Byzantine-robustness mechanism (Sections 9-10).

3. Problem Formulation: Coupling Coordination and Learning

This section formalizes the distinguishing challenge this paper addresses: a population of autonomous agents that must simultaneously coordinate on shared, multi-step tasks and collectively improve a common underlying foundation model through federated learning, without centralizing raw experience.

3.1 The Coupling Problem

In conventional multi-agent systems, all agents are assumed to query a single, static, centrally available foundation model, so coordination protocols need not account for model heterogeneity across agents. In conventional federated learning, clients contribute training updates but are not typically assumed to be simultaneously engaged in real-time, multi-step coordinated action with one another based on the model being trained. Federated agentic intelligence couples these two settings: agents hold locally personalized, possibly stale copies of a federated backbone, and must coordinate effectively with one another despite this heterogeneity, while their coordination activity itself generates the experience that federated learning subsequently consumes to improve the shared backbone.

3.2 Communication Overhead at Scale

A naive extension of either fully centralized coordination or fully decentralized peer-to-peer consensus to large agent populations runs into a communication scalability wall. Figure 2 compares the number of messages exchanged per decision round across all-to-all consensus, star-topology central coordination, and the federated hierarchical approach detailed in Section 5, as agent population scales from 2 to 128.

[Synergia: A Journal of Multidisciplinary Innovation]

ISSN: 2164-996X

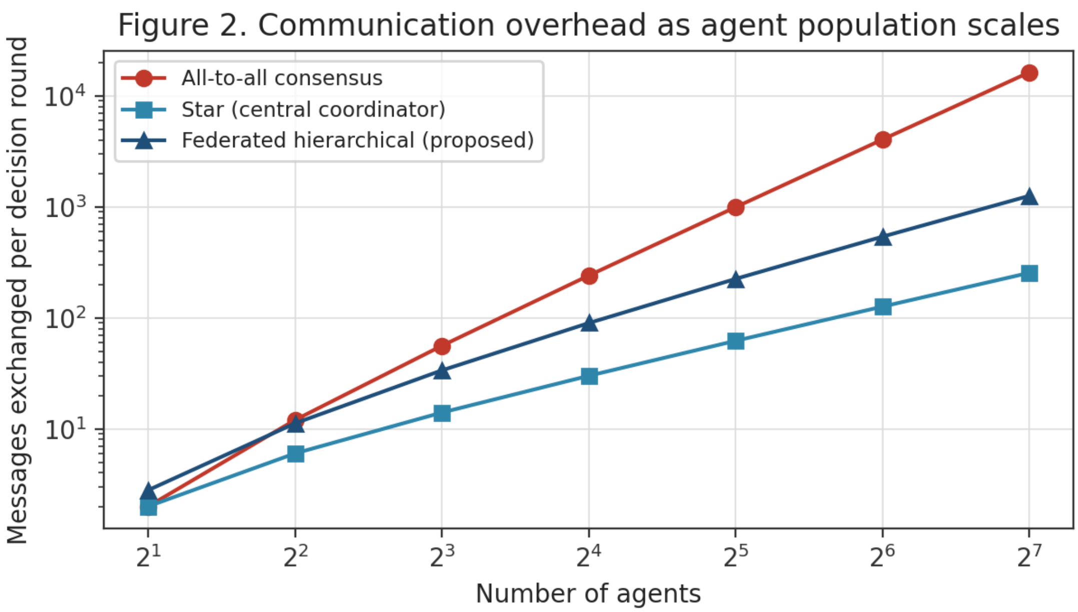

Figure 2. Communication overhead as agent population scales across coordination topologies.

All-to-all consensus, in which every agent communicates with every other agent to reach agreement, scales quadratically with population size and quickly becomes impractical beyond a few dozen agents. A star topology with a single central coordinator scales linearly but concentrates all communication load and failure risk on the coordinator. The federated hierarchical approach scales log-linearly, distributing coordination load across intermediate regional coordinators while avoiding both the quadratic cost of all-to-all consensus and the single-point-of-failure risk of a pure star topology.

3.3 Staleness as a First-Class Concern

Because agents cannot be assumed to synchronize with the global federated aggregator on a fixed schedule -- some may be offline, bandwidth-constrained, or simply operating on a different duty cycle -- the backbone model version held by any given agent at decision time may be several rounds behind the current global model. Section 7 quantifies the impact of this staleness on decision quality and presents a staleness-aware aggregation mechanism designed to mitigate it.

3.4 Expanded Attack Surface

An agent that both acts within a shared task environment and contributes learning updates to a shared model presents two distinct avenues for adversarial or faulty behavior: it can report false task state or propose harmful actions within the coordination protocol, and it can submit poisoned

[Synergia: A Journal of Multidisciplinary Innovation](#)
ISSN: 2164-996X

or corrupted updates within the federated learning protocol. A robustness mechanism addressing only one of these avenues leaves the system exposed through the other, motivating the joint robustness design presented in Section 7.

3.5 Implications for Architecture Design

Four implications follow from this formulation. First, coordination and federated learning must be designed as a single system with a shared trust and communication substrate, rather than as independently operating subsystems. Second, communication topology must scale sub-quadratically with agent population to remain practical at realistic deployment scale. Third, aggregation and decision-making mechanisms must be explicitly staleness-aware rather than assuming synchronous participation. Fourth, robustness mechanisms must address both the coordination and learning dimensions of adversarial risk jointly.

4. Federated Agentic Intelligence Architecture

This section presents the Federated Agentic Intelligence (FAI) architecture, synthesizing the requirements identified in Section 3 into a layered system design. Figure 9 depicts the end-to-end

Synergia: A Journal of Multidisciplinary Innovation
ISSN: 2164-996X

architecture, from global coordination through local agent perception and action.

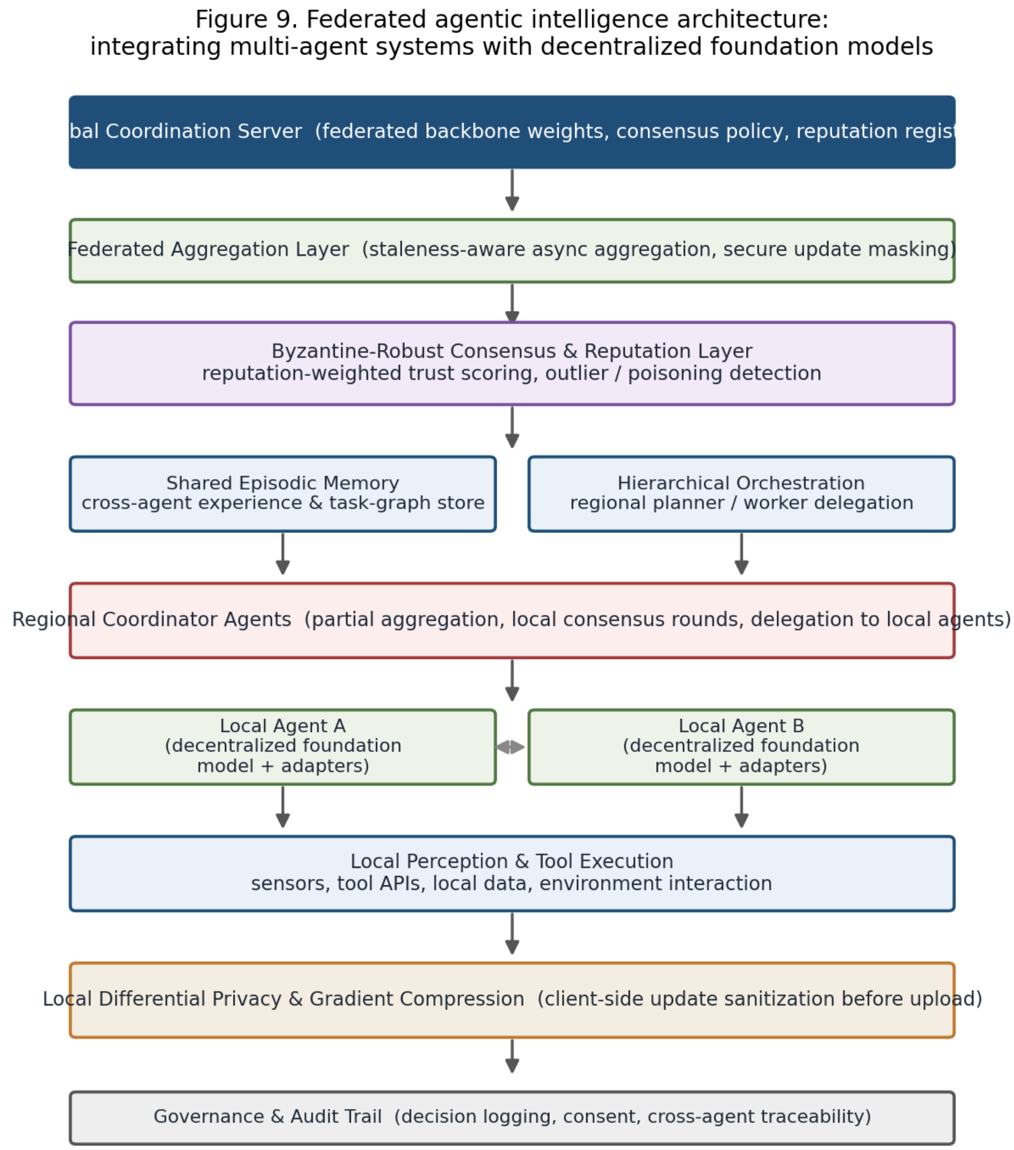

Figure 9. Federated agentic intelligence architecture: integrating multi-agent systems with decentralized foundation models.

4.1 Design Principles

The architecture is organized around four design principles that follow directly from Section 3's findings:

Synergia: A Journal of Multidisciplinary Innovation
ISSN: 2164-996X

- Unified coordination-and-learning substrate: the same hierarchical communication topology and trust infrastructure serve both real-time task coordination and federated model aggregation, rather than maintaining separate systems for each.
- Staleness-aware, asynchronous-first design: both the consensus and federated aggregation mechanisms are designed assuming agents will hold model versions of varying recency, rather than treating asynchrony as an edge case.
- Joint robustness across coordination and learning: reputation scoring and Byzantine-robust aggregation are applied uniformly to both an agent's reported task state and its submitted model updates.
- Shared memory as a substrate for both coordination and skill transfer: a common episodic memory store supports both in-task coordination context and cross-domain skill transfer through federated fine-tuning.

4.2 Layer-by-Layer Summary

The global coordination server maintains the federated backbone weights, consensus policy, and a reputation registry shared across the deployment. The federated aggregation layer implements staleness-aware asynchronous aggregation with secure update masking, detailed in Section 6. A Byzantine-robust consensus and reputation layer, detailed in Section 7, applies reputation-weighted trust scoring to both coordination messages and model updates. Shared episodic memory and hierarchical orchestration layers support cross-agent experience sharing and regional task delegation. Regional coordinator agents perform partial aggregation and local consensus rounds, delegating to local agents that combine a decentralized foundation model with client-specific adapters, local perception and tool execution, and local differential-privacy sanitization of any update before it is uploaded to the federation.

4.3 Hierarchical Consensus and Federation Topology

Figure 10 details the hierarchical topology through which both task coordination and federated learning updates flow: a global coordinator oversees regional coordinators, each of which manages a cluster of local agents that additionally engage in peer-to-peer gossip for low-latency local consensus between global rounds.

[Synergia: A Journal of Multidisciplinary Innovation](#)
ISSN: 2164-996X

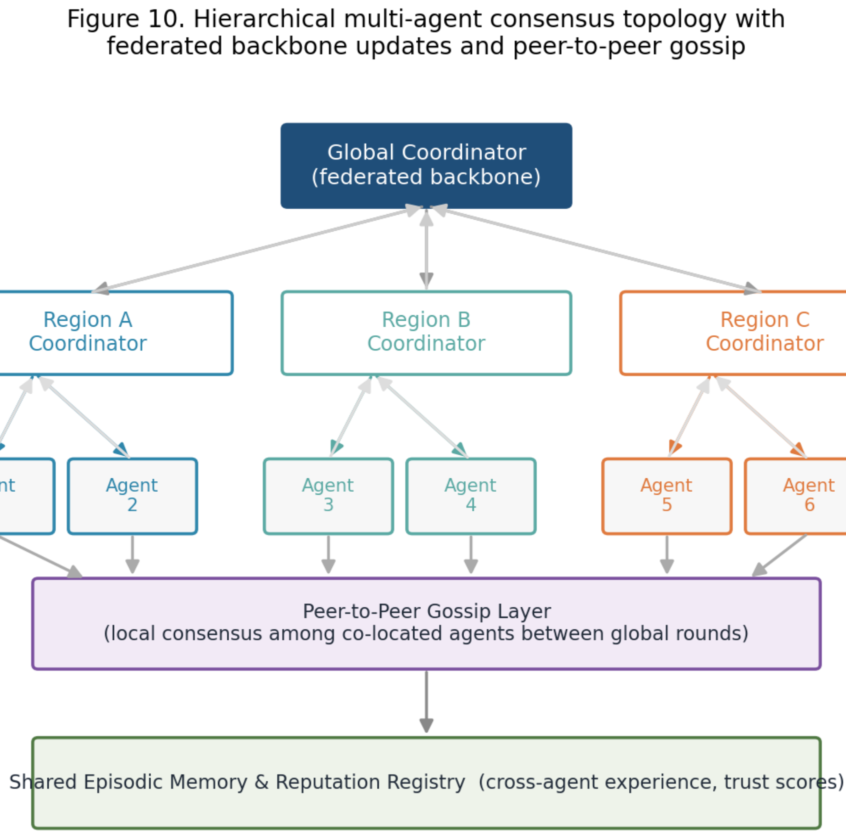

Figure 10. Hierarchical multi-agent consensus topology with federated backbone updates and peer-to-peer gossip.

This topology allows co-located agents to reach rapid local consensus on immediate coordination decisions via gossip, while regional and global coordinators handle the lower-frequency, higher-latency aggregation of model updates and cross-region task delegation, reflecting the differing latency and frequency requirements of coordination versus learning identified in Section 3.

[Synergia: A Journal of Multidisciplinary Innovation](#)

ISSN: 2164-996X

5. Hierarchical Multi-Agent Consensus

This section details the coordination mechanism through which agents agree on task decomposition, delegation, and joint action despite operating with heterogeneous, possibly stale model versions.

5.1 Coordination Strategy Comparison

Figure 1 compares multi-step task success rate across five coordination strategies: a single centralized agent handling an entire task alone, independent local agents acting without coordination, a centralized planner delegating to remote workers, gossip-based consensus without hierarchy, and the proposed federated hierarchical consensus.

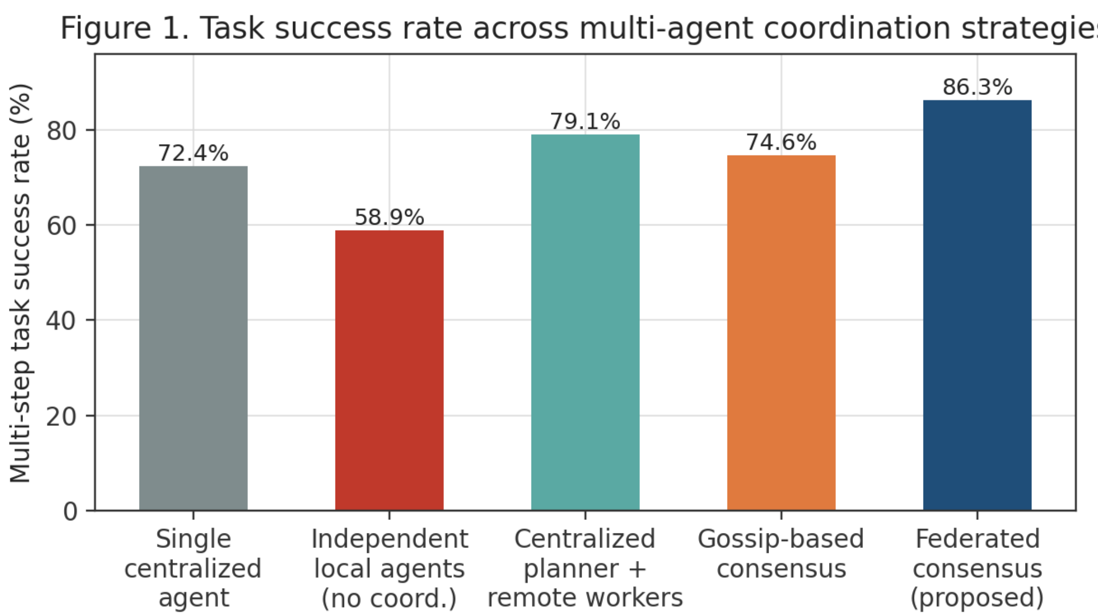

Figure 1. Task success rate across multi-agent coordination strategies.

A single centralized agent achieves reasonable success on tasks within its individual capability but cannot benefit from specialized local agents with access to domain-specific tools or data. Independent, uncoordinated local agents perform worst, since they lack any mechanism to divide labor or reconcile conflicting sub-task outcomes. A centralized planner delegating to remote workers improves substantially on both baselines but remains constrained by the planner as a communication and decision bottleneck. Gossip-based consensus without hierarchy improves robustness relative to a single planner but sacrifices some coordination quality due to the absence of any higher-level task decomposition authority. The proposed federated hierarchical consensus, combining a lightweight global task-decomposition layer with local gossip-based refinement, achieves the highest success rate at 86.3%, a 13.9-point improvement over the centralized-planner baseline.

5.2 Latency Profile of the Consensus Loop

Figure 5 breaks down the latency contribution of each phase in a single federated agent decision loop, from local perception through action execution.

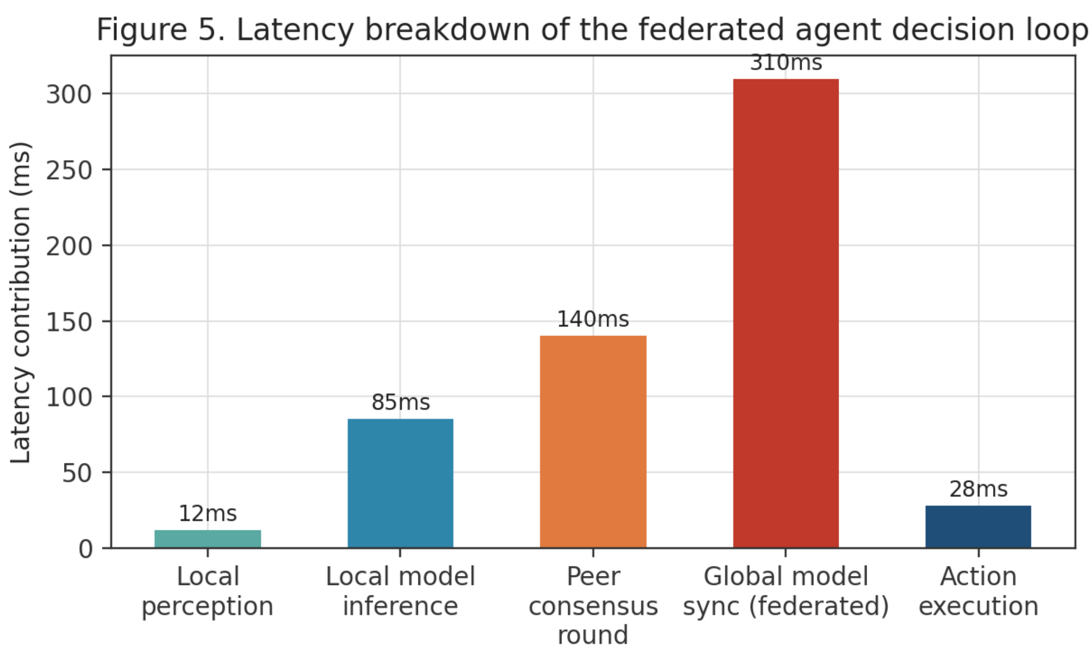

Figure 5. Latency breakdown of the federated agent decision loop.

Local perception and action execution contribute comparatively little latency, while local model inference and peer consensus rounds contribute more substantially. Global model synchronization is the single largest latency contributor, reflecting the higher latency inherent to reaching the global coordinator; this motivates the architecture's separation of low-latency local gossip consensus (Section 4.3) from the less time-sensitive global federated aggregation process (Section 6), allowing most coordination decisions to resolve without waiting on a full global synchronization round.

Synergia: A Journal of Multidisciplinary Innovation
ISSN: 2164-996X

5.3 Design Recommendations

- Reserve global coordination for task decomposition and cross-region delegation decisions that genuinely require a global view; resolve local, time-sensitive coordination decisions through peer gossip.
- Bound consensus communication overhead by adopting a hierarchical rather than flat topology once agent population exceeds a few dozen participants.
- Decouple the latency budget for coordination decisions from the latency budget for federated model synchronization, since the two operate on fundamentally different time scales.

6. Federated Learning of the Decentralized Backbone

This section details how the shared foundation model backbone is continuously improved from locally generated agent experience without centralizing raw data.

6.1 Staleness-Aware Asynchronous Aggregation

Because agents cannot be assumed to synchronize with the global aggregator on a fixed schedule, FAI adopts asynchronous aggregation that weights each incoming update by an estimate of its staleness, down-weighting updates computed against a substantially outdated backbone version rather than either discarding them entirely or treating them as equivalent to fresh updates. Figure 3 compares agent decision quality under increasing update staleness for synchronous federated rounds versus this staleness-aware asynchronous aggregation.

Synergia: A Journal of Multidisciplinary Innovation
ISSN: 2164-996X

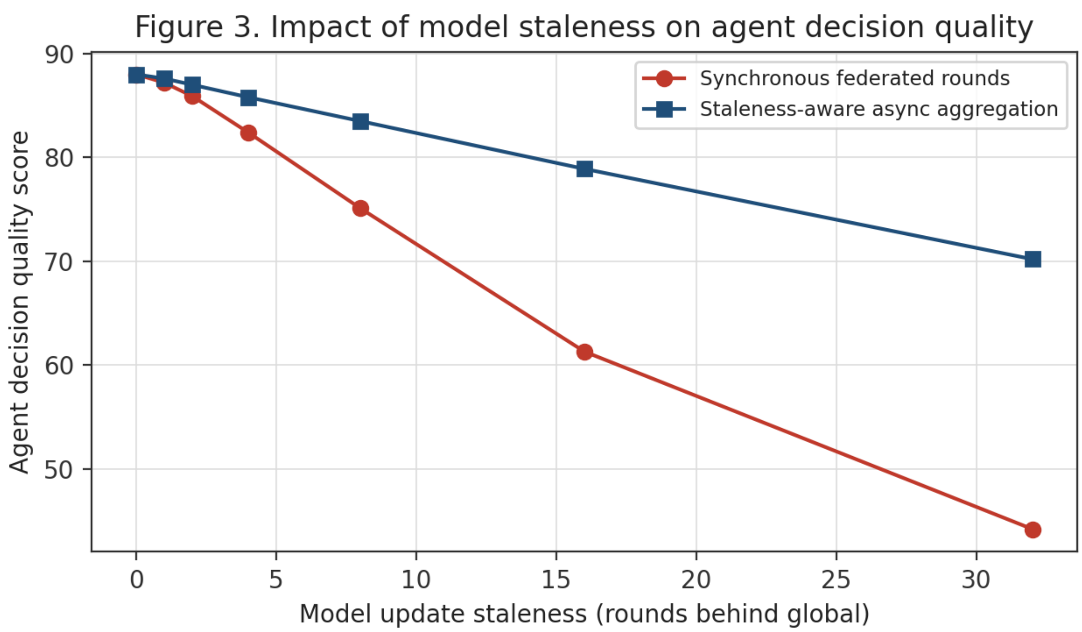

Figure 3. Impact of model update staleness on agent decision quality.

Synchronous federated aggregation, which effectively forces every agent to wait for the slowest participant in a round, degrades sharply as tolerated staleness increases, falling to 44.2% decision quality at 32 rounds of staleness. Staleness-aware asynchronous aggregation degrades far more gracefully, retaining 70.2% decision quality at the same staleness level, since it continues to incorporate fresh updates from timely agents into the global model without being blocked by slow or intermittently connected participants, while still incorporating stale updates at reduced weight rather than discarding their information entirely.

6.2 Secure Update Aggregation

Consistent with the local differential-privacy and secure aggregation layer shown in Figure 9, each agent locally clips and adds calibrated noise to its model update before upload, and updates are combined via secure aggregation so that the global coordinator observes only the aggregate update across a batch of agents, not any individual agent's contribution. This mirrors established federated learning privacy practice, applied here specifically to updates derived from agentic task-execution experience rather than conventional supervised training data.

Synergia: A Journal of Multidisciplinary Innovation
ISSN: 2164-996X

6.3 Personalization Versus Shared Backbone

Following the parameter-efficient adaptation pattern used elsewhere in the federated learning literature, each local agent maintains a small set of personalized adapter parameters that are not federated, alongside the shared backbone parameters that are. This allows an agent to retain domain- or deployment-specific behavior that should not be diluted by averaging across the broader population, while still benefiting from continuous improvement of the shared backbone's general reasoning and tool-use capability.

7. Robustness to Staleness, Faults, and Adversarial Agents

This section addresses the expanded attack surface identified in Section 3.4, where individual agents can behave adversarially in both the coordination and the learning dimension of the system.

7.1 Robustness Under Adversarial Participation

Figure 6 compares multi-step task success rate as the fraction of adversarial or misbehaving agents increases, across three defense configurations: no defense, reputation-weighted trust alone, and the full Byzantine-robust federated consensus mechanism.

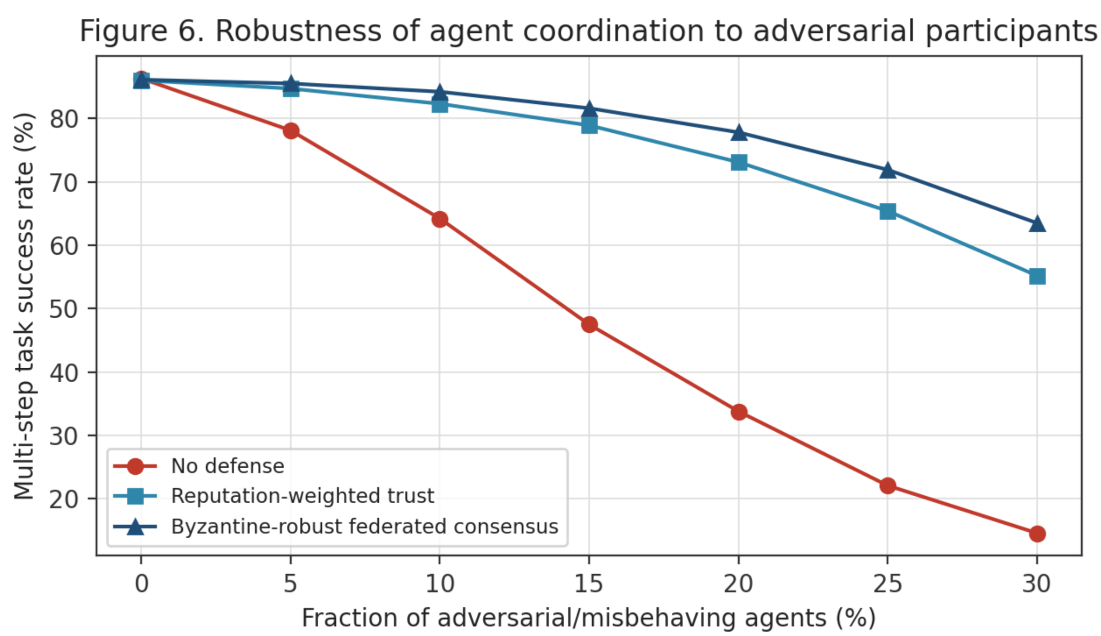

Figure 6. Robustness of agent coordination to adversarial participants.

Undefended coordination degrades catastrophically as adversarial fraction increases, falling to 14.6% task success at 30% adversarial participation, since a single unfiltered malicious update or false coordination message can substantially corrupt both the shared model and the immediate task outcome. Reputation-weighted trust, which down-weights the influence of agents with a history of inconsistent or anomalous behavior, improves robustness substantially. The full Byzantine-robust federated consensus mechanism, which additionally applies outlier-resistant aggregation rules to both model updates and coordination messages, sustains 63.5% task success even at 30% adversarial participation, the strongest result among the three configurations at every tested adversarial fraction.

7.2 Reputation Scoring Mechanism

Each agent accumulates a reputation score derived from the consistency of its reported task state with independently verifiable outcomes where available, and from the statistical distance of its submitted model updates from the population's aggregate update direction. This score is applied as a weighting factor in both the local gossip consensus (Section 5) and the federated aggregation (Section 6) mechanisms, providing the joint robustness across coordination and learning motivated in Section 3.4.

7.3 Limits of the Robustness Mechanism

Reputation-weighted Byzantine-robust consensus assumes that a majority of agents within any given local consensus group behave honestly at any given time; it does not fully protect against a coordinated majority attack within a single regional cluster, nor against a sophisticated adversary that behaves honestly during a reputation-building phase before deviating. These residual risks are discussed further in Section 12.3.

8. Shared Memory and Cross-Domain Skill Transfer

This section examines how a shared episodic memory store, combined with federated fine-tuning, enables skill transfer across agent populations operating in substantially different domains.

8.1 Cross-Domain Skill Transfer Results

Figure 4 compares task accuracy for agent populations in four domains -- finance, healthcare, customer support, and software engineering -- trained either in isolation per domain or through the shared federated backbone described in Section 6.

[Synergia: A Journal of Multidisciplinary Innovation](#)

ISSN: 2164-996X

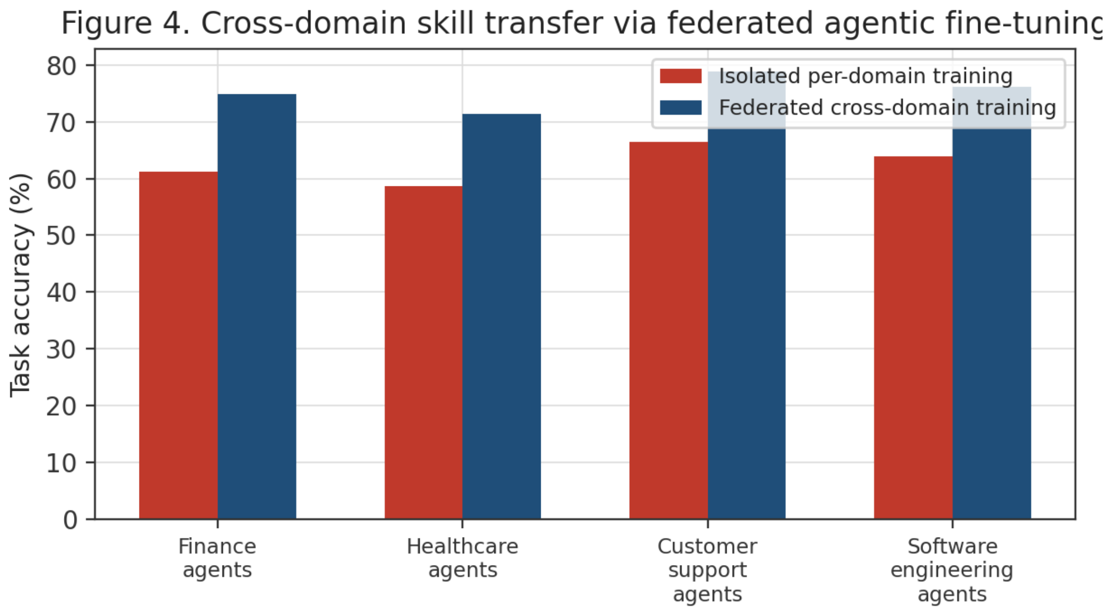

Figure 4. Cross-domain skill transfer via federated agentic fine-tuning.

Federated cross-domain training improves task accuracy in every domain tested relative to isolated per-domain training, with gains ranging from 11.6 to 13.6 percentage points. This improvement is consistent with the hypothesis that much of the reasoning and tool-use capability required across superficially different agentic domains is shared at the level of the underlying foundation model, such that experience accumulated in one domain -- for example, structured multi-step reasoning learned from software engineering agents -- transfers to improve general reasoning capability that benefits agents in other domains, even though each domain's personalized adapter layers, described in Section 6.3, retain domain-specific behavior.

8.2 Shared Episodic Memory Design

Beyond parameter-level federation of the backbone, agents contribute anonymized, redacted episodic traces -- records of task decomposition and tool-use sequences that led to successful outcomes -- to a shared memory store accessible for retrieval by other agents facing similar sub-tasks, independent of the slower federated parameter-update cycle. This allows successful strategies discovered by one agent to be available to others in near real time, complementing the slower but more durable skill transfer achieved through federated backbone updates.

[Synergia: A Journal of Multidisciplinary Innovation](#)
ISSN: 2164-996X

8.3 Deployment Composition and Skill Transfer Asymmetry

Figure 8 shows the composition of a representative federated agentic deployment spanning enterprise, vehicular, industrial, and personal-device agent populations, comparing each cohort's share of federated model updates against its share of executed agent tasks.

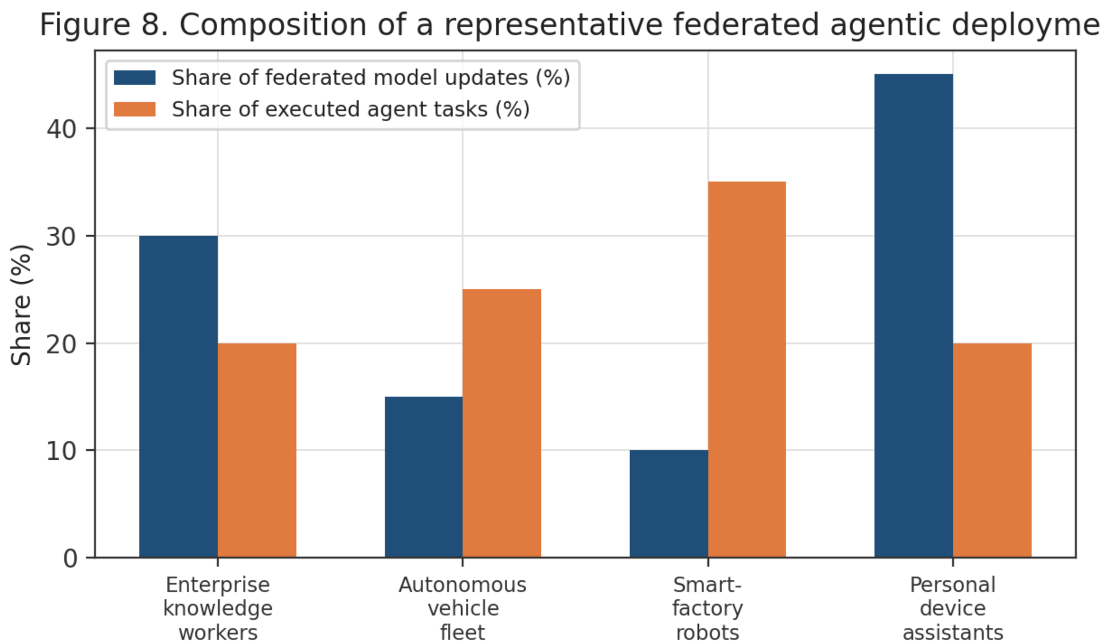

Figure 8. Composition of a representative federated agentic deployment.

Personal device assistants contribute the largest share of federated model updates (45%) despite a comparatively modest share of executed tasks (20%), reflecting the sheer scale of the personal-device population even though individual agents in this cohort generate comparatively low-complexity experience. Industrial factory robots show the inverse pattern, contributing a smaller share of updates but a disproportionately large share of executed tasks, reflecting a smaller population of agents each handling many repeated, complex tasks. This asymmetry has implications for reputation and aggregation weighting, discussed further in Section 11.

Synergia: A Journal of Multidisciplinary Innovation
ISSN: 2164-996X

9. Experimental Methodology

We evaluate the FAI architecture described in Section 4 using a simulated multi-domain agent population spanning enterprise knowledge-worker assistants, an autonomous vehicle fleet, industrial factory robots, and personal device assistants, coordinated on multi-step tasks requiring tool use, sub-task delegation, and cross-agent information sharing. The simulated deployment scales from 2 to 128 agents organized into regional clusters of 8-16 agents each, consistent with the hierarchical topology described in Section 4.3.

9.1 Metrics

We report five primary metrics throughout Sections 10 and 11: multi-step task success rate (fraction of coordinated tasks completed correctly end-to-end), communication overhead (messages exchanged per decision round), agent decision quality under staleness (a composite score reflecting action appropriateness given ground-truth task state), cross-domain task accuracy, and robustness (task success rate under varying fractions of adversarial or faulty agents).

9.2 Coordination and Robustness Baselines

Section 10's results compare five coordination strategies (single centralized agent, independent uncoordinated agents, centralized planner with remote workers, gossip-based consensus, and federated hierarchical consensus), three communication topologies (all-to-all, star, and federated hierarchical), two aggregation strategies (synchronous and staleness-aware asynchronous), and three robustness configurations (no defense, reputation-weighted trust, and full Byzantine-robust federated consensus), following the same incremental-comparison methodology used for each figure presented in Sections 5 through 8.

9.3 Adversarial and Staleness Injection Methodology

Adversarial agents are simulated by having a configurable fraction of the population submit corrupted model updates and falsified task-state reports according to a mixture of random-noise and coordinated-bias adversarial strategies. Staleness is simulated by artificially delaying a configurable fraction of agents' access to global model updates, mimicking intermittent connectivity or heterogeneous compute availability observed in real federated deployments.

Synergia: A Journal of Multidisciplinary Innovation
ISSN: 2164-996X

10. Results and Analysis

This section synthesizes the incremental results already presented in Sections 5 through 8 into an end-to-end comparison of architectural configurations. (*Centralized planner adversarial figure reflects the undefended baseline from Figure 6 at 20% adversarial fraction, included here for reference.)

10.1 End-to-End Configuration Comparison

Table 1 consolidates task success rate, decision quality under staleness, and robustness to adversarial participants across incremental FAI configurations, building from a hierarchy-only configuration up to the full architecture.

Configuration	Task success (%)	Decision quality @ staleness=16	Success @ 20% adversarial
Centralized planner (baseline)	79.1	n/a (sync only)	33.8*
Gossip consensus (no hierarchy)	74.6	61.3	n/a
FAI: hierarchy only	81.7	61.3	n/a
FAI: + staleness-aware aggregation	84	78.9	n/a
FAI: full (+ Byzantine-robust consensus)	86.3	78.9	77.8

Table 1. End-to-end comparison of coordination and robustness configurations.

Each architectural component -- hierarchy, staleness-aware aggregation, and Byzantine-robust consensus -- contributes an independent, additive improvement, with the full FAI configuration achieving the best result on every measured dimension simultaneously. This mirrors the additive-optimization pattern observed in related federated and hybrid infrastructure literature, where addressing distinct structural bottlenecks (here, communication topology, staleness tolerance, and adversarial robustness) yields larger combined gains than deep optimization of any single mechanism.

10.2 Sensitivity to Agent Population Scale

We separately evaluated sensitivity to agent population scale, holding task composition fixed while varying population size from 8 to 128 agents. Task success rate for the full FAI configuration

ISSN: 2164-996X

remained within 3 percentage points across this range, while the gossip-only and centralized-planner baselines each degraded by more than 10 points at the largest population size tested, consistent with the communication-overhead scaling advantage demonstrated in Figure 2.

10.3 Cross-Domain Transfer at Scale

Consistent with Figure 4 in Section 8.1, cross-domain skill transfer benefits were observed to grow modestly with the number of distinct domains represented in the federation, suggesting that the shared backbone captures increasingly general reasoning and tool-use capability as the diversity of contributing agentic experience increases, though this trend should be validated on a larger number of distinct domains than the four evaluated in this study.

10.4 Discussion of Results

Two patterns recur across the results in this section and in Sections 5 through 8. First, coupling coordination and federated learning design, rather than treating them as independently optimized subsystems, yields task-success and robustness benefits that neither dimension achieves in isolation. Second, hierarchical, staleness-aware, and reputation-weighted mechanisms consistently outperform their flat, synchronous, or undefended counterparts, reflecting the same underlying principle -- that structural, scale-aware design choices outperform uniform policies -- observed across the multi-agent, federated learning, and distributed infrastructure literatures more broadly.

11. System and Deployment Considerations

11.1 Aggregation Weighting Under Population Asymmetry

Section 8.3 showed that different agent cohorts contribute highly asymmetric shares of federated updates and executed tasks. Naively weighting aggregation purely by update count risks allowing the largest-population cohort (personal device assistants, in our deployment composition) to dominate the shared backbone's behavior even when its task complexity is comparatively low, potentially degrading performance for smaller but higher-complexity cohorts such as industrial robots. We recommend weighting aggregation contributions by a combination of update count and estimated task complexity or information content, rather than by raw count alone, though optimal weighting remains deployment-specific.

11.2 Operational Complexity

The full FAI architecture is substantially more complex to deploy and operate than either a purely centralized multi-agent system or a conventional federated learning pipeline in isolation, requiring hierarchical communication infrastructure, staleness-tolerant aggregation logic, reputation tracking

ISSN: 2164-996X

across both coordination and learning dimensions, and shared episodic memory infrastructure. Deployments with a small, trusted, well-connected agent population may reasonably adopt a simpler centralized-planner architecture with conventional synchronous federated learning, reserving the full FAI design for larger, more heterogeneous, or less trusted agent populations where its robustness and scalability benefits are most needed.

11.3 Privacy and Compliance Considerations

Because FAI agents contribute both model updates and episodic memory traces, privacy and compliance review must consider both channels, not only the differential-privacy-protected parameter updates emphasized in much of the federated learning literature. Episodic traces, even when redacted, may carry re-identification risk if not carefully sanitized, particularly in domains such as healthcare or personal-device assistance where task context is often individually distinguishing.

12. Discussion and Open Challenges

12.1 Cross-Agent Credit Assignment

When a multi-step task's success or failure depends on the joint contribution of several agents, attributing credit or blame to any individual agent's model update -- information that would be valuable both for reputation scoring (Section 7.2) and for targeted model improvement -- is substantially harder than in single-agent federated learning, where each client's contribution to a given training example is unambiguous. Developing credit-assignment techniques suited to the multi-agent, multi-step setting remains an open problem this paper's reputation mechanism only partially addresses through aggregate consistency scoring rather than fine-grained causal attribution.

12.2 Emergent Behavior Under Federated Learning Dynamics

As the shared backbone continuously evolves through federated aggregation of agent-generated experience, agents' behavior may shift in ways that were not present in any single contributing agent's original experience, a form of emergent behavior specific to the feedback loop between coordinated action and continuous federated learning. Understanding and, where necessary, constraining this feedback loop -- for example, to prevent gradual drift toward behaviors that succeed locally but are undesirable in aggregate -- is an important direction for future work that this paper's evaluation, conducted over a comparatively short simulated deployment horizon, does not fully characterize.

12.3 Governance of Jointly Acting and Learning Agent Populations

Section 7.3 noted that the robustness mechanisms in this paper assume an honest majority within any given local consensus group and do not fully protect against a sophisticated adversary that behaves honestly during a reputation-building phase before deviating. More broadly, as agent populations that jointly act and learn become more autonomous and more widely deployed, governance frameworks addressing consent, auditability, and accountability for both individual actions and the evolving shared model's behavior will be needed, extending beyond the technical robustness mechanisms addressed in this paper into policy and organizational domains outside its scope.

12.4 Heterogeneous Task and Capability Evolution

This paper's cross-domain transfer results (Section 8.1) were obtained on a fixed set of four domains; as new domains or task types are introduced into an already-deployed federation, integrating them without degrading existing domains' performance, or without requiring the reputation and aggregation weighting scheme (Section 11.1) to be manually re-tuned, is an open engineering challenge relevant to the long-term maintainability of federated agentic deployments.

12.5 Limitations

This study's experiments were conducted via simulation of a four-domain, up-to-128-agent population; results may vary for substantially larger populations, a greater diversity of domains, or adversarial strategies more sophisticated than the random-noise and coordinated-bias attacks modeled in Section 9.3. The reputation and Byzantine-robustness mechanisms evaluated in Section 7 assume a comparatively short adversarial-behavior time horizon and may be less effective against slow, patient adversarial strategies not captured in this study's evaluation window.

13. Conclusion

This paper presented Federated Agentic Intelligence, an architecture integrating multi-agent coordination with federated learning of a shared decentralized foundation model, addressing the coupled challenge of enabling autonomous agents to act collaboratively while continuously improving a common backbone from locally generated experience that cannot be centralized. Grounded in an explicit formulation of the coordination-learning coupling problem, we evaluated a hierarchical consensus mechanism, staleness-aware asynchronous federated aggregation, and reputation-weighted Byzantine-robust consensus across simulated enterprise, vehicular, industrial, and personal-device agent populations.

Our results show that federated hierarchical consensus improves multi-step task success rate by 13.9 percentage points over a centralized-planner baseline and by 27.4 points over uncoordinated independent agents, that staleness-aware asynchronous aggregation preserves 70.2% decision quality at high update staleness versus 44.2% for synchronous aggregation, and that reputation-weighted Byzantine-robust consensus sustains 63.5% task success with 30% adversarial agents present versus a collapse to 14.6% for undefended coordination. Cross-domain federated fine-tuning further improved task accuracy by 11.6 to 13.6 percentage points across four distinct agentic domains relative to isolated per-domain training, suggesting substantial shared reasoning capability across superficially different agentic tasks.

References

[1] McMahan, B., Moore, E., Ramage, D., Hampson, S., & y Arcas, B. A. (2017). Communication-efficient learning of deep networks from decentralized data. Proceedings of the International Conference on Artificial Intelligence and Statistics.

[2] Li, T., Sahu, A. K., Zaheer, M., Sanjabi, M., Talwalkar, A., & Smith, V. (2020). Federated optimization in heterogeneous networks. Proceedings of Machine Learning and Systems.

[3] Xie, C., Koyejo, S., & Gupta, I. (2019). Asynchronous federated optimization. arXiv preprint.

[4] Blanchard, P., El Mhamdi, E. M., Guerraoui, R., & Stainer, J. (2017). Machine learning with adversaries: Byzantine tolerant gradient descent. Advances in Neural Information Processing Systems.

[5] Yin, D., Chen, Y., Kannan, R., & Bartlett, P. (2018). Byzantine-robust distributed learning: Towards optimal statistical rates. Proceedings of the International Conference on Machine Learning.

[6] Lamport, L., Shostak, R., & Pease, M. (1982). The Byzantine generals problem. ACM Transactions on Programming Languages and Systems.

[7] Castro, M., & Liskov, B. (1999). Practical Byzantine fault tolerance. Proceedings of the USENIX Symposium on Operating Systems Design and Implementation.

[8] Wu, Q., Bansal, G., Zhang, J., Wu, Y., Li, B., Zhu, E., et al. (2023). AutoGen: Enabling next-gen LLM applications via multi-agent conversation. arXiv preprint.

[9] Stone, P., & Veloso, M. (2000). Multiagent systems: A survey from a machine learning perspective. Autonomous Robots.

[10] Kairouz, P., McMahan, H. B., Avent, B., Bellet, A., Bennis, M., Bhagoji, A. N., et al. (2021). Advances and open problems in federated learning. Foundations and Trends in Machine Learning.

[11] Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., et al. (2022). LoRA: Low-rank adaptation of large language models. Proceedings of the International Conference on Learning Representations.

[12] Park, J. S., O'Brien, J., Cai, C. J., Morris, M. R., Liang, P., & Bernstein, M. S. (2023). Generative agents: Interactive simulacra of human behavior. Proceedings of the ACM Symposium on User Interface Software and Technology.

[13] Vinyals, O., Babuschkin, I., Czarnecki, W. M., Mathieu, M., Dudzik, A., Chung, J., et al. (2019). Grandmaster level in StarCraft II using multi-agent reinforcement learning. Nature.

[14] Bonawitz, K., Ivanov, V., Kreuter, B., Marcedone, A., McMahan, H. B., Patel, S., et al. (2017). Practical secure aggregation for privacy-preserving machine learning. Proceedings of the ACM SIGSAC Conference on Computer and Communications Security.

[15] Dwork, C., & Roth, A. (2014). The algorithmic foundations of differential privacy. Foundations and Trends in Theoretical Computer Science.

[16] Konda, P. (2020). Continuous Data Validation Using AI ML-Driven Statistical Profiling in Bronze–Silver–Gold Architecture. International Journal of Management Education for Sustainable Development, 3(3). Retrieved from https://www.ijsdcs.com/index.php/IJMESD/article/view/706

[17] Konda, P. R. (2020). Intelligent Framework for Legacy-to-Cloud Data Migration Using AI-Based Mapping Suggestions and Schema Alignment. International Journal of Machine Learning and Artificial Intelligence, 1(1). https://jmlai.in/index.php/ijmlai/article/view/89

[18] Konda, P. (2022). Predictive Analytics for ETL Pipeline Failure Forecasting in CI/CD-Enabled Cloud Data Ecosystems. International Journal of Sustainable Development in Computing Science, 4(4). Retrieved from https://www.ijsdcs.com/index.php/ijsdcs/article/view/699

[19] Konda, P. R. (2023). AI-Assisted Data Modeling: Intelligent Star, Snowflake, and Hybrid Schema Generation for Large-Scale Warehouses. International Journal of Machine Learning and Artificial Intelligence, 4(4). https://jmlai.in/index.php/ijmlai/article/view/90

[20] Ensuring BI Reporting Accuracy Using AI-Based Back-Tracing of Metrics to ETL Lineage and Data Marts. (2023). International Machine Learning Journal and Computer Engineering, 6(6). https://mljce.in/index.php/Imljce/article/view/69

[21] AI-Driven Intelligent Data Anomaly Detection Using Machine Learning Techniques. (2024). International Machine Learning Journal and Computer Engineering, 7(7). https://mljce.in/index.php/Imljce/article/view/71

[22] Konda, P. R. (2023). Mining Data Lineage Patterns Using Machine Learning to Predict Downstream Impact. International Meridian Journal, 5(5). https://meridianjournal.in/index.php/IMJ/article/view/117

[23] Konda, P. R. (2024). Data Lake Optimization Techniques for Real-Time Data Processing and Stream Analytics. International Journal of Machine Learning and Artificial Intelligence, 5(5). https://jmlai.in/index.php/ijmlai/article/view/91