Super Agents: The DeFAI Engine for Super Sapiens

ABSTRACT

Abstract—This paper introduces Super Agents, a multi-agent decentralized finance (DeFi) framework on the Solana blockchain that integrates specialized Decentralized Finance AI (DeFAI) engines. Drawing on Magnetic-One (Fourney et al., 2025)—notably its nested-loop orchestration, ledger-driven collaboration, and error recovery—Super Agents addresses core DeFi challenges like cross-chain bridging, liquidity management, and yield optimization. By adopting Magnetic-One's orchestrator-based approach and specialized agents, the proposed system seeks to enhance trustlessness, scalability, and decision-making through advanced AI-driven analytics.

I. Introduction

Decentralized finance (DeFi) unlocks new financial opportunities by removing centralized intermediaries, enabling peer-to-peer lending, trading, and asset management. However, the user experience can be fragmented: bridging assets across chains, balancing liquidity pools, and finding optimal yield strategies can be labor-intensive. Recent successes in multi-agent architectures—exemplified by Magnetic-One—demonstrate how orchestrators coordinating specialized agents can solve complex, multi-step tasks with adaptive, modular workflows.

Goal: We extend Magnetic-One's nested-loop orchestration and ledger-based progress tracking to build a specialized DeFi system—Super Agents—capable of:

- Seamless cross-chain token swaps and bridging.
- AI-based liquidity management to mitigate impermanent loss and enhance capital efficiency.
- Yield optimization through real-time analytics on APYs, sentiment, and chain activity.

In the following sections, we discuss the relevant work, detail our multi-agent system architecture, explain the underlying technical components, and illustrate the workflow with multiple Mermaid diagrams. Lastly, we discuss potential benefits, limitations, and future extensions of the approach.

II. Related Work

2.1 Multi-Agent Orchestration

Magnetic-One (Fourney et al., 2025) highlights a nested-loop orchestration comprising: • An Outer Loop that builds a Task Ledger with planning details, verified facts, and overall goals. • An Inner Loop that consults a Progress Ledger, identifies the next agent to act, updates error counters, and detects repeated stalling.

When tasks stall, the orchestrator triggers reflection, updates the ledger, and re-plans with an adaptive strategy. This approach has proven effective on diverse benchmarks like GAIA, AssistantBench, and WebArena, showcasing how specialized agents—each with focused capabilities—yield modular, robust solutions.

2.2 DeFi on Solana

Solana's high throughput and low fees make it a popular platform for complex DeFi tasks. While bridging and liquidity optimization exist across various protocols, they are often manual or only partially automated. Research (Zohar & Noyan, 2020, 2024; Chu et al., 2021) suggests that machine learning can aid in tasks like impermanent loss reduction or yield rebalancing, but a fully multi-agent paradigm with ledger-driven error recovery remains largely unexplored in DeFi.

III. System Architecture

Super Agents adapts Magentic-One's two-loop design to DeFi, combining an Orchestrator agent with multiple specialized DeFAI agents:

1. Orchestrator

• Maintains the Task Ledger (high-level plan) and Progress Ledger (step-by-step records).
• Coordinates subtask assignments to specialized agents.
• Detects stalls or repeated errors, triggers reflection, and re-plans tasks.

2. Bridge Agent

• Handles cross-chain transfers and atomic swaps.
• Checks bridging liquidity and finalizes bridging transactions on Solana and external chains.

3. Liquidity Manager Agent

• Allocates and rebalances liquidity in pools based on market data, volatility forecasts, and user-defined risk.
• Minimizes impermanent loss while ensuring stable yields.

4. Yield Optimizer Agent
• Continuously scans major DeFi protocols for APYs.
• Relocates assets to maximize yields while respecting constraints (e.g., stable vs. volatile pools).

5. Market Analysis Agent
• Performs sentiment analysis on social media signals and interprets large on-chain data shifts.
• Triggers advanced warnings to the Orchestrator if market volatility is detected.

3.1 Nested-Loop Orchestration Diagram

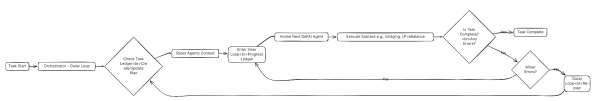

Fig. 1

The Outer Loop manages the Task Ledger and strategic planning, while the Inner Loop delegates subtasks to DeFAI agents and updates the Progress Ledger. Major stalls prompt reflection and re-planning.

3.2 Multi-Agent Architecture Diagram

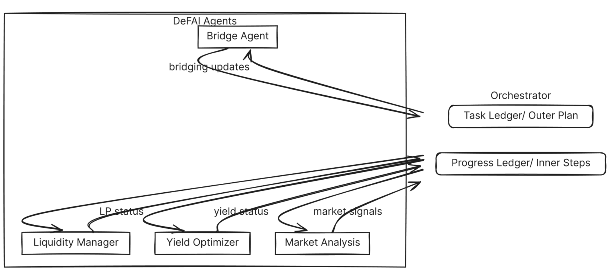

Fig. 2

High-level architecture. The Orchestrator coordinates specialized DeFAI agents for bridging, liquidity, yield, and market analysis. Progress and Task Ledgers guide the Orchestrator’s decisions.

IV. Technical Workflow and Data Flows

4.1 Data Flow Diagram

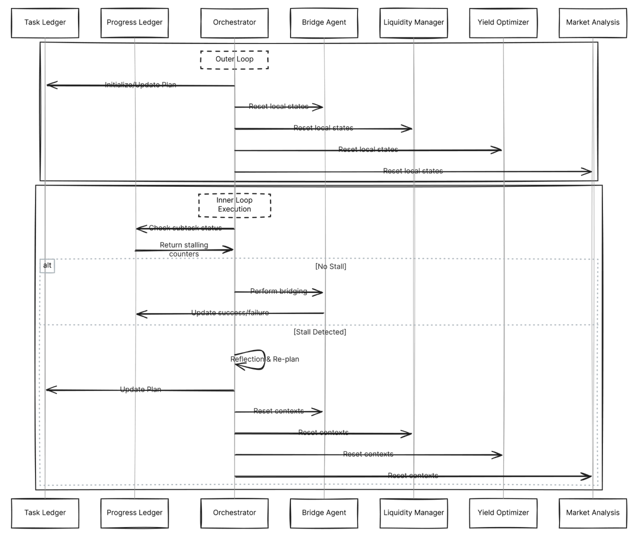

Fig. 3

Data and control flow among the Outer Loop, Inner Loop, ledgers, and DeFAI agents. The Orchestrator continuously checks the Progress Ledger to decide subtasks and monitors stalling.

4.2 Orchestrator & Agent Interaction (Sequence Diagram)

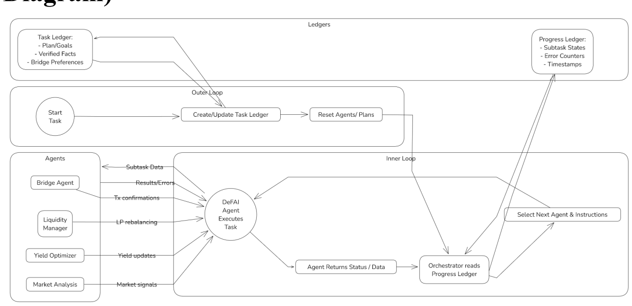

Fig. 4

Detailed sequence of interactions. The Orchestrator references ledgers, instructs specialized DeFAI agents, and handles stall detection via re-planning and reflection.

V. Core Functionalities

5.1 Multichain Token Swaps & Bridges
• Bridge Agent: Uses atomic swap mechanisms or bridging protocols (e.g., Wormhole) to facilitate cross-chain transfers.
• Orchestrator: Ensures bridging tasks are in line with the Task Ledger; updates the Progress Ledger with success/error details.
• Error Recovery: On repeated bridging failures, a fallback bridging route or time-delay scheduling may be triggered.

5.2 AI-Enhanced Liquidity Pool (LP) Management
• Liquidity Manager Agent: Employs predictive analytics (LSTMs, random forests) to anticipate volatility.
• Orchestrator: Periodically instructs rebalances based on market changes.
• Ledgers: Record each rebalance attempt, capturing impermanent loss metrics and user-defined risk tolerance.

5.3 Real-Time Market Analytics
• Market Analysis Agent: Gathers data from on-chain metrics and social media sentiment via NLP.
• Signal Alerts: High negative sentiment or unusual whale movements can prompt the Orchestrator to queue bridging tasks or shift assets.

5.4 Yield Optimization Strategies
• Yield Optimizer Agent: Monitors APYs across multiple Solana-based platforms (Raydium, Solend, etc.) or cross-chain pools if bridging is profitable.
• Reflective Re-allocation: If repeated yield reallocation fails or stalls, the Orchestrator re-plans (e.g., invests in stable pools or halts certain riskier allocations).

5.5 Streamlined Liquidity Provisioning
• AMM Integration: The Liquidity Manager interacts with AMMs (e.g., Serum-based DEXs) to automate liquidity provisioning.
• Slippage Boundaries: The orchestrator sets thresholds for slippage and monitors repeated deviations, triggering re-planning if inefficiencies persist.

VI. Technical Infrastructure

6.1 Solana & Smart Contracts

Solana offers sub-second block times and low fees—crucial for frequent ledger updates, bridging, and rebalancing. Smart Contracts store critical transaction details:

• Guardrails: Hard-coded reverts or timeouts if bridging or reallocation tasks exceed user-defined risk parameters.

• On-Chain Logging: Key events (task completions, agent error messages) anchor ledger updates with immutability.

6.2 DeFAI Engine & Model Pipelines

1. Bridge Agent: Minimal AI logic, relies on bridging liquidity checks and cross-chain states.

2. Liquidity Manager & Yield Optimizer: May run advanced ML models offline or in a distributed context. Periodically refresh predictions, updating the Orchestrator.

3. Market Analysis: Transformer-based NLP or simpler classification for sentiment detection from data APIs. On suspicious patterns, posts warnings to the Orchestrator.

6.3 Ledger Design and Error-Recovery

• Progress Ledger keeps an "attempt counter" for each subtask, incrementing upon failures.

• Task Ledger defines overarching objectives (e.g., bridging stablecoins to capture higher yields).

• Stall Threshold triggers a reflection step, leading to partial or full context resets (like in Magentic-One) so fresh strategies can be tried.

VII. Discussion

7.1 Benefits

• Adaptive Recovery: By systematically detecting stalls, the system reduces repeated errors in bridging or yield tasks.

• Agent Modularity: New DeFi functionalities (derivatives, NFT-based lending) can be added as specialized agents without revamping existing flows.

• Security: On-chain logging helps audit agent actions, and limiting agent privileges confines the impact of potential exploits.

7.2 Overheads and Limitations

• Latency: Additional ledger reads/writes and cross-agent communication can introduce overhead.

• Complex Data Dependencies: Agents rely on external or off-chain data (price oracles, social media) that could be spoofed. Proper verifications are essential.

7.3 Future Directions

• Longer-Term Memory: Storing repeated bridging strategies that worked or failed under certain market conditions could reduce future overhead.

• Granular Risk Models: Embedding advanced financial risk frameworks (e.g., VaR, CVaR) within the Orchestrator to refine yield strategies.

• Benchmarking: Similar to GAIA or WebArena, designing DeFi-specific agentic benchmarks (e.g., repeated bridging tasks, meltdown simulations) could stress-test multi-agent capabilities.

VIII. Conclusion

Super Agents evolves Magentic-One's successful multi-agent principles into a DeFi context, employing nested-loop orchestration and ledger-based error handling. By combining an orchestrator with specialized DeFAI agents—covering bridging, liquidity provisioning, yield optimization, and market analysis—the system autonomously adapts to dynamic market conditions, mitigates failures, and simplifies user participation in complex DeFi protocols. This synergy of AI-driven analytics, ledger-tracked workflows, and orchestrator-led collaboration highlights a promising new direction for next-generation, trustless financial services on the Solana blockchain and beyond

References

1. Fourney, A., Bansal, G., Mozannar, H., Tan, C., Salinas, E., Zhu, E., Loynd, R., West, R., Dibia, V., & Kamar, E. (2025). Magentic-One: A Generalist Multi-Agent System for Solving Complex Tasks. Microsoft Research AI Frontiers.
2. Chu, S., Xia, Y., & Zhang, Z. (2021). Manta: A Plug and Play Private DeFi Stack. IACR.
3. Strüker, J., et al. (2022). Decentralized Finance (DeFi). Fraunhofer FIT.
4. Zohar, A., & Noyan, A. (2020). Decentralized Finance. Journal of Financial Regulation.
5. Zohar, A., & Noyan, A. (2024). Decentralized Finance (DeFi): A Functional Approach. Journal of Financial Regulation.
6. Federal Reserve Bank of Boston. (2022). Decentralized Finance (DeFi).
7. DeFiChain Team. Whitepaper - DeFiChain.

Acknowledgments

We thank the Magentic-One research team for demonstrating how ledger-based orchestration and multi-agent error recovery can be generalized to diverse complex tasks. Their insights directly shaped the design of Super Agents, enabling robust DeFi automation through specialized agents and nested-loop planning.