Svedberg Open
DISSEMINATIONOFKNOWLEDGE

International Journal of Artificial Intelligence and Machine Learning

Publisher's Home Page: https://www.svedbergopen.com/

Research Paper Open Access

A Scalable Multi-Agent Architecture For Agentic AI-Based Autonomous Task Coordination and Resource Optimization

Mahendrakumar Kalal[1] , Janardhana Naidu Kola[2]

[1]Senior SAP Analyst O.C.Tanner Houston, Texas, USA, Email: mahendrakalal14051989@gmail.com, ORCID ID: 0009-0006-4150-8937
[2]Director of Business Intelligence ADP Miami, United States, Email: Reachjanardhanakola@gmail.com ORCID ID: 0009-0003-1845-2216

Abstract
The study establishes a scalable multi-agent architecture for agentic artificial intelligence (Agentic AI) systems that can support autonomous task coordination, distributed execution, and efficient computational resource utilization under changing workload conditions. The objective is to integrate agent-level reasoning, task planning, autonomous task allocation, execution monitoring, resource management, and load balancing within a unified modular framework. The study also examines task-allocation behaviour under different problem scales using established allocation approaches, including Hungarian, Auction, Consensus-Based Bundle Algorithm (CBBA), and Auction 2-opt. The evaluated benchmark scenarios indicate that task scale influences computational requirements and communication overhead. At small scale, the Hungarian method achieved the lowest total cost, while Auction 2-opt required the highest computational effort and communication. At medium scale, the Auction method produced the lowest total cost, whereas Auction 2-opt again showed greater computational requirements and communication overhead. At large scale, differences in total cost remained relatively small, but computational requirements increased considerably with problem size. These findings demonstrate that allocation performance varies according to workload scale and that resource-aware coordination is important for scalable multi-agent execution. The proposed architecture provides a structured foundation for integrating autonomous reasoning, multi-agent coordination, task allocation, and resource optimization. The benchmark evidence identifies important considerations for future implementation and supports further validation through original simulations and real-world experiments involving dynamic workloads, heterogeneous resources, failures, and distributed edge-cloud environments.

Keywords: Agentic artificial intelligence, Autonomous task coordination, Multi-agent systems, Resource optimization, Scalable architecture.

1. Introduction

Artificial intelligence (AI) is evolving from predominantly task-specific computational systems toward autonomous frameworks capable of interpreting objectives, reasoning over contextual information, planning actions, interacting with external tools, and adapting their behaviour during execution. This transition has contributed to the emergence of agentic artificial intelligence (Agentic AI), in which AI systems are designed to pursue specified objectives with a greater degree of autonomy. Agentic architectures integrate capabilities including reasoning, planning, memory, action generation, environmental interaction, and feedback-based adaptation to support complex decision-making processes (Abou et al., 2025). The emergence of generative AI and large language models (LLMs) has further expanded this paradigm by enabling AI systems to progress beyond response generation toward multi-step reasoning, task execution, and autonomous orchestration (Haque et al., 2026). These capabilities are particularly important for dynamic intelligent environments in which task requirements, computational workloads, and operational conditions may change continuously.

The growing application of Agentic AI across diverse domains demonstrates the increasing demand for autonomous and adaptive computational systems. Agentic architectures have been investigated in healthcare, where autonomous agents can support complex decision processes and workflow management, as well as in autonomous aerial intelligence and other intelligent environments (Banerjie et al., 2025). Agentic AI has also emerged as an important paradigm for autonomous decision-making and intelligent-system development, indicating a broader transition from static AI models toward systems capable of planning, execution, monitoring, and adaptation (Chahar et al., 2025). In software engineering, agent-based approaches have been explored for autonomous deployment and scalable operational processes (Felipe, 2026). These developments indicate that effective Agentic AI requires not only capable underlying models but also architectural mechanisms for coordinating actions, managing computational resources, and maintaining system-level efficiency.

Agentic AI extends conventional AI by emphasizing goal-directed behaviour and autonomous interaction with computational or physical environments. Rather than producing an isolated prediction or response, an agentic system can interpret a goal, formulate an action strategy, execute individual steps, evaluate intermediate outcomes, and adjust subsequent actions. This capability creates opportunities for AI systems to manage complex workflows with reduced dependence on predefined procedural instructions (Ajmera, 2026). Recent perspectives on Agentic AI further emphasize architectures that combine generative models with planning, tool use, memory, and autonomous execution (Raghavendra and Saikia, 2026). The resulting systems are particularly relevant to intelligent environments that require continuous adaptation and decision-making.

As AI-enabled tasks become more complex, single-agent architectures may face limitations when one agent is responsible for reasoning, planning, execution, monitoring, and resource management simultaneously (Allmendinger et al., 2026). Multi-agent artificial intelligence addresses this challenge by distributing responsibilities among multiple interacting agents. Agents can perform specialized functions while collaborating to achieve shared objectives, thereby supporting task decomposition, parallel processing, and distributed decision-making (Ali and Grudén, 2025). Hierarchical multi-agent frameworks further facilitate structured execution by allowing higher-level agents to formulate objectives while subordinate agents perform specialized subtasks (Reda et al., 2026). Similar approaches have been investigated for autonomous enterprise operations, where multiple AI agents collaborate across interconnected workflows (Srivastav and Rohit, 2025). Increasing the number of interacting agents may also introduce communication overhead, synchronisation requirements, conflicting decisions, and additional computational demands.

Effective coordination is fundamental to the operation of multi-agent systems. Autonomous task coordination involves translating high-level objectives into executable subtasks, selecting appropriate agents, allocating responsibilities, exchanging information, monitoring progress, and adapting execution strategies when workload or environmental conditions change. Agentic AI-driven orchestration has demonstrated potential for real-time project coordination and intelligent workflow management (Harini et al., 2025). Distributed agent coordination has also gained importance in next-generation communication systems, where AI agents can interpret objectives, coordinate strategies, and support autonomous network control (Dev et al., 2025). Research on future intelligent communications similarly highlights the transition from large AI models toward autonomous systems capable of interaction, reasoning, and coordinated decision-making (Jiang et al., 2026). These developments establish autonomous coordination as a central requirement for scalable distributed Agentic AI.

The benefits of multi-agent architectures depend strongly on efficient resource management. Each additional agent may introduce computational, memory, communication, and processing requirements, while heterogeneous tasks can demand different levels of computational capacity (Elkael et al., 2026). A scalable agentic system must therefore allocate resources according to task requirements, workload intensity, execution priorities, and agent availability. Cloud-oriented Agentic AI architectures emphasise scalability as an important requirement for supporting increasing numbers of autonomous tasks and users (Alva and Pandey, 2026). Similarly, scalable autonomous decision-making frameworks highlight the need for infrastructure capable of sustaining continuous agent operation without excessive computational overhead (Molli, 2023). Secure collaboration among autonomous agents can further increase communication and computational requirements, making resource-aware coordination essential for practical deployment.

Despite rapid advances, existing Agentic AI architectures remain fragmented across reasoning, workflow automation, autonomous decision-making, resource management, and domain-specific applications. Contemporary architectures differ in reasoning mechanisms, orchestration strategies, memory structures, communication models, and degrees of autonomy (Rekha et al., 2026). Existing approaches have demonstrated autonomous task execution and multi-agent collaboration, but many focus on individual domains or specific architectural components rather than jointly addressing coordination, resource optimisation, scalability, and distributed execution (Sapkota et al., 2025). Moreover, increasing agent populations can amplify communication overhead, redundant operations, synchronisation requirements, and computational consumption. Trust, reliability, security, governance, and human oversight therefore remain important considerations for practical autonomous AI deployment (Karim et al., 2025).

The existing literature provides strong foundations for autonomous AI, multi-agent collaboration, workflow automation, and distributed intelligent systems; an important gap remains in their integration within a unified, scalable, and resource-aware architecture. Current studies frequently investigate individual dimensions, including agent reasoning, data analytics, enterprise automation, healthcare, autonomous aerial systems, or communication networks. Other research demonstrates autonomous orchestration, scalable deployment, and distributed control but leaves opportunities for deeper integration of dynamic resource optimisation with multi-agent task coordination. The central gap is therefore the limited availability of an integrated architecture that treats autonomous task coordination, dynamic resource optimisation, distributed execution, and scalability as interconnected objectives. Addressing this gap is essential for developing Agentic AI systems capable of maintaining efficient performance under changing workloads while controlling computational and coordination overhead.

This study proposes a scalable multi-agent architecture for Agentic AI-based autonomous task coordination and resource optimisation. The primary objective is to develop an integrated framework in which autonomous agents can interpret high-level objectives, decompose complex tasks, assign subtasks according to agent capabilities, coordinate execution, and dynamically utilise available computational resources. The proposed architecture is designed to support efficient operation under changing workloads and varying numbers of interacting agents.

The objectives of this study are to develop a scalable multi-agent architecture that integrates autonomous reasoning, task planning, task coordination, execution, and resource management for efficient Agentic AI-based task processing, and to assess task-allocation performance under different workload scales using task execution efficiency, computational requirements, communication overhead, scalability, and related performance measures.

2. Materials and Methods

2.1 Proposed Multi-Agent Architecture

The proposed framework was designed as a scalable multi-agent architecture for autonomous task coordination and resource optimisation. The system consists of multiple specialised agents connected through a coordination layer responsible for task distribution, communication, execution monitoring, and resource management. Each agent performs defined functions according to its capabilities and current operational state. A high-level objective is initially interpreted and divided into manageable subtasks, which are subsequently assigned to suitable agents. The architecture separates reasoning, planning, communication, execution, and resource-management functions to support modular operation. This organization enables multiple agents to operate concurrently while maintaining coordination and allowing the system to adapt to changing workload conditions.

2.2 Agentic AI-Based Task Reasoning and Planning

The reasoning mechanism enables agents to interpret assigned objectives, evaluate available information, and determine appropriate actions. Complex objectives are decomposed into smaller subtasks according to their requirements, dependencies, and execution conditions. Each agent maintains a state representation containing its assigned task, workload, execution status, and available computational resources. Based on this information, agents generate or select suitable execution plans and continuously update their decisions using feedback from completed or ongoing operations. The framework supports AI-based reasoning for task interpretation and autonomous decision-making, while separating reasoning from execution control. This design allows the agents to modify their plans when task requirements, workload conditions, or resource availability change.

2.3 Autonomous Task Coordination and Allocation

Task coordination determines how subtasks are distributed among participating agents. The coordination mechanism evaluates task requirements, agent capabilities, current workload, task priority, and resource availability before assigning responsibilities. Independent tasks can be executed concurrently, whereas dependent tasks are coordinated according to their required execution sequence. During operation, agents communicate task status, intermediate results, and resource conditions through the coordination layer. If an assigned agent becomes overloaded or unavailable, the affected task can be reassigned to another eligible agent. Dynamic prioritization is applied when workload conditions change, allowing urgent or computationally important tasks to receive appropriate attention. This mechanism supports autonomous, distributed, and adaptive task execution.

2.4 Resource Optimization and Load Balancing

Resource optimization is integrated with task coordination to regulate the computational resources used by participating agents. Resource allocation considers task requirements, agent workload, processing capacity, and current resource availability. The framework dynamically adjusts resource assignments as tasks arrive, progress, or reach completion. Load balancing is applied to prevent excessive concentration of computational demand on individual agents. When an agent becomes heavily loaded, suitable pending tasks can be directed toward agents with greater available capacity. Conversely, underutilized resources can be assigned additional eligible workloads. This integrated strategy aims to improve resource utilization, reduce computational bottlenecks, and maintain efficient task execution as workload intensity changes.

2.5 Scalability and Distributed Execution

The scalability of the proposed architecture is evaluated by varying the number of participating agents and the intensity of computational workloads. Additional agents can be incorporated through the modular architecture without fundamentally modifying existing agent functions. Distributed execution allows independent subtasks to be processed concurrently, while dependent operations follow their specified relationships. System behaviour is examined under increasing workload conditions to determine changes in task completion, processing time, resource utilization, and coordination overhead. Robustness is also considered by examining the system's response to agent unavailability or incomplete task execution. These procedures provide a basis for determining whether the architecture can maintain efficient autonomous operation as system size and workload complexity increase.

2.6 Experimental Evaluation and Performance Metrics

The proposed framework is evaluated against suitable baseline configurations under comparable workload conditions. The experimental design considers different agent populations and workload levels to examine coordination efficiency, resource management, and scalability. Performance is assessed using task completion rate, execution time, computational resource utilization, throughput, task allocation efficiency, and communication overhead. Load distribution and system behaviour under increasing workloads are additionally examined to identify potential bottlenecks. Where applicable, repeated experimental runs are used to obtain stable performance estimates, with appropriate measures of variation reported. Comparative and ablation evaluations can further determine the contribution of multi-agent coordination, dynamic resource allocation, and autonomous planning to overall system performance.

3. Results

3.1 Small-Scale Task Allocation Performance

The performance of four task-allocation approaches under a small-scale static multi-agent setting. The Hungarian method achieved the lowest total cost at 92.2 ± 26.0, while Auction and CBBA produced slightly higher costs of 103.5 ± 32.1 and 103.6 ± 32.5, respectively, as shown in Table 1. Auction 2-opt recorded a cost of 98.1 ± 27.7. All methods achieved the same completion rate of 0.50, with no response delay reported. The Hungarian method required the lowest computational time, whereas Auction 2-opt required the highest. Communication requirements were absent for Hungarian but increased across the distributed approaches, with Auction 2-opt generating the highest communication overhead.

Table 1. Small-Scale Static Multi-Agent Task Allocation

Method	Total Cost	Completion Rate	Response Delay	Computational Time ($\times 10^{-6}$ s)	Communication Messages
Hungarian	92.2 ± 26.0	0.50 ± 0	0 ± 0	212.5 ± 32.7	0 ± 0
Auction	103.5 ± 32.1	0.50 ± 0	0 ± 0	456.1 ± 84.9	6.17 ± 2.04
CBBA	103.6 ± 32.5	0.50 ± 0	0 ± 0	214.2 ± 19.3	7.20 ± 1.73
Auction 2-opt	98.1 ± 27.7	0.50 ± 0	0 ± 0	628.5 ± 92.3	10.0 ± 0

Source: Song et al. (2025)

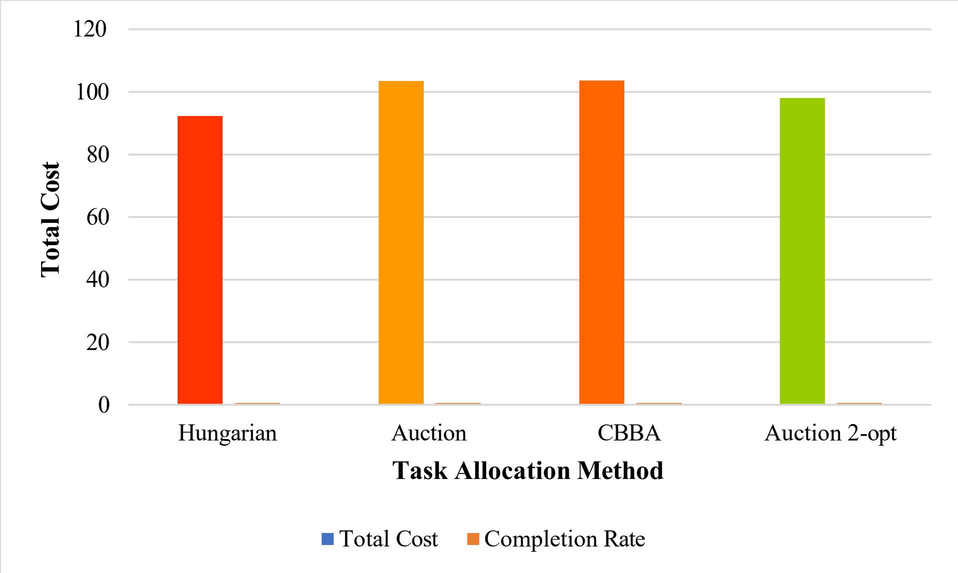

Figure 1. Total Cost Comparison Across Task Allocation Methods

The total cost associated with four multi-agent task allocation methods: Hungarian, Auction, CBBA, and Auction 2-opt. The vertical axis represents total cost, scaled from 0 to 120, while the horizontal axis lists the four methods under comparison as shown in Figure 1. Each bar carries a unique colour, with Hungarian in red, Auction in amber, CBBA in orange, and Auction 2-opt in light green. The chart shows that Hungarian achieves the lowest total cost among all methods, whereas Auction and CBBA record nearly identical and the highest costs, with Auction 2-opt positioned slightly below them.

3.2 Medium-Scale Task Allocation Performance

The four allocation methods under a medium-scale static multi-agent scenario. The Auction method produced the lowest total cost of 115.9 ± 13.5, followed by CBBA at 120.7 ± 18.5 and the Hungarian method at 121.7 ± 21.6, as shown in Table 2. Auction 2-opt showed the highest total cost of 122.8 ± 20.9. The completion rate remained 0.30 for all approaches, while response delay was reported as zero. Computational requirements increased considerably compared with the small-scale scenario. Hungarian and CBBA required approximately $2.8 \times 10^3 \times 10^{-6}$ s, whereas Auction 2-opt required $4.972 \times 10^3 \times 10^{-6}$ s. Communication messages were highest for Auction 2-opt.

Table 2. Medium-Scale Static Multi-Agent Task Allocation

Method	Total Cost	Completion Rate	Response Delay	Computational Time ($\times 10^{-6}$ s)	Communication Messages
Hungarian	121.7 ± 21.6	0.30 ± 0	0 ± 0	2783 ± 250	0 ± 0
Auction	115.9 ± 13.5	0.30 ± 0	0 ± 0	3199 ± 221	17.4 ± 3.14
CBBA	120.7 ± 18.5	0.30 ± 0	0 ± 0	2804.4 ± 37.2	19.0 ± 2.77
Auction 2-opt	122.8 ± 20.9	0.30 ± 0	0 ± 0	4972 ± 334	105 ± 0

Source: Song et al. (2025)

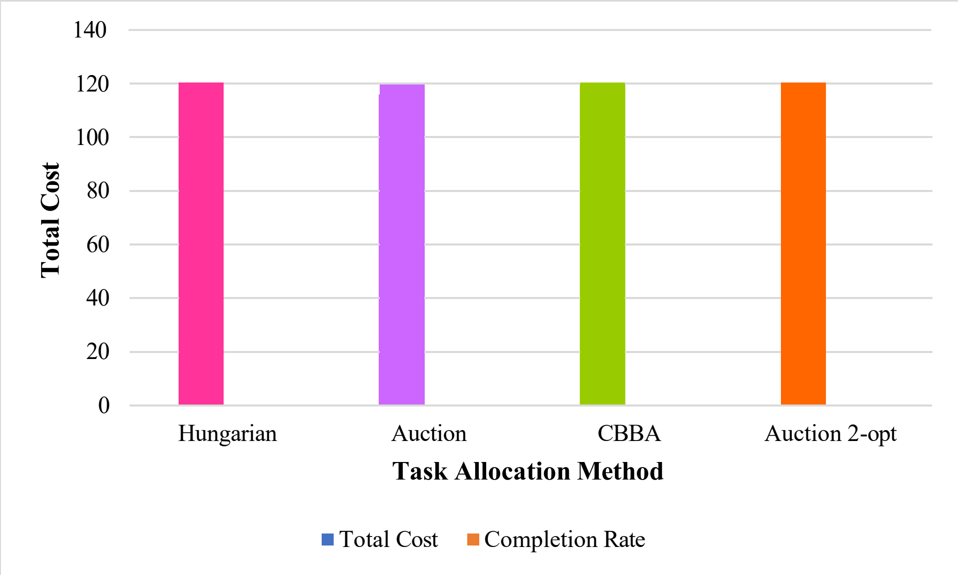

Figure 2. Total Cost Comparison Across Task Allocation Methods

The total cost incurred by four multi-agent task allocation methods: Hungarian, Auction, CBBA, and Auction 2-opt. The vertical axis denotes total cost, ranging from 0 to 140, while the horizontal axis lists the four allocation methods evaluated, as shown in Figure 2. Each bar is rendered in a distinct colour for visual clarity, with Hungarian shown in pink, Auction in purple, CBBA in green, and Auction 2-opt in orange. The results indicate that Auction 2-opt records the highest total cost, closely followed by Hungarian, while Auction yields the lowest cost among the four methods, with CBBA falling in between these values.

3.3 Large-Scale Task Allocation Performance

The performance of the allocation methods when the task-allocation problem is increased to a large scale. Total cost values were relatively close, ranging from 150.4 ± 13.5 for Auction to 152.7 ± 11.7 for CBBA, as shown in Table 3. All four approaches reported a completion rate of 0.20 and zero response delay in the evaluated setting. Computational requirements increased substantially with problem size. Hungarian required 28,574 ± 511 × 10^{-6} s, while Auction 2-opt required 42,380 ± 5,570 × 10^{-6} s. Communication overhead also increased markedly, reaching 780 messages for Auction 2-opt, compared with no messages for the Hungarian approach.

Table 3. Large-Scale Static Multi-Agent Task Allocation

Method	Total Cost	Completion Rate	Response Delay	Computational Time ($\times 10^{-6}$ s)	Communication Messages
Hungarian	150.8 ± 12.4	0.20 ± 0	0 ± 0	28,574 ± 511	0 ± 0
Auction	150.4 ± 13.5	0.20 ± 0	0 ± 0	28,870 ± 1,020	45.2 ± 6.51
CBBA	152.7 ± 11.7	0.20 ± 0	0 ± 0	29,505 ± 424	45.8 ± 2.54
Auction 2-opt	152.1 ± 10.8	0.20 ± 0	0 ± 0	42,380 ± 5,570	780 ± 0

Source: Song et al. (2025)

4. Discussion

The findings demonstrate that multi-agent task-allocation performance is strongly influenced by problem scale, computational requirements, and communication overhead. In the small-scale scenario, the Hungarian method produced the lowest total cost, whereas the distributed approaches required additional communication between participating agents. As the number of tasks and agents increased, differences in computational requirements became more evident. In the medium-scale scenario, Auction achieved the lowest total cost, while Auction 2-opt required the highest computational time and generated substantially greater communication overhead. The large-scale results further demonstrated that computational and communication requirements increase considerably as task-allocation complexity increases. These observations are relevant to Agentic AI because autonomous systems must coordinate multiple agents while simultaneously managing computational resources and communication demands.

The results also indicate that a single allocation strategy may not remain optimal under all workload conditions. The change in relative performance between small-, medium-, and large-scale scenarios suggests that task-allocation mechanisms should be selected according to system size and operational requirements. This supports the architectural principle of dynamically coordinating autonomous agents rather than relying on a fixed allocation mechanism. Agentic AI systems are increasingly characterised by autonomous reasoning, planning, cooperation, and adaptation, making coordination an essential component of their architecture. Therefore, the observed increase in computational time and communication overhead with system scale provides an important justification for integrating resource-aware coordination into scalable multi-agent architectures.

The findings are consistent with recent research emphasising coordination and cooperation as fundamental requirements of Agentic AI systems. Talasila (2026) highlighted coordination, negotiation, and cooperation as central mechanisms for autonomous agents operating in collaborative and competitive environments. The present analysis similarly demonstrates that increasing agent participation introduces additional coordination requirements and communication overhead. This indicates that scalability cannot be considered solely in terms of adding more agents; the associated computational and communication costs must also be controlled.

The findings also correspond with the architectural challenges discussed by Thakur (2026), who described Agentic AI systems as requiring combinations of reasoning, planning, autonomous action, environmental interaction, and adaptive behaviour. The increasing computational requirements observed across the benchmark scenarios reinforce the need for architectures that separate task reasoning from execution and resource management. In this context, the proposed architecture's emphasis on task decomposition, autonomous allocation, load balancing, and distributed execution is conceptually aligned with current Agentic AI development.

Healthcare-oriented research provides further support for this architectural direction. Subir et al. (2026) emphasized multi-agent collaboration, architectural organisation, safety, and reliability when applying Agentic AI to complex healthcare workflows. Similarly, Warrier and Abhilash (2025) investigated autonomous agentic AI for clinical workflow orchestration, demonstrating the importance of coordinating multiple operational activities within dynamic environments. Although the present study is not restricted to healthcare, the same architectural requirement applies: autonomous agents must coordinate tasks efficiently while maintaining reliable execution.

The findings are also relevant to edge-oriented Agentic AI. Zhang et al. (2026) discussed agentic AI and agentification in the context of edge general intelligence, where distributed computation, resource constraints, and autonomous coordination become particularly important. The substantial increase in computational time and communication messages observed in the larger scenarios indicates why resource-aware coordination is important for distributed and edge-based intelligent systems.

The study has implications for the design of scalable AI systems in which multiple autonomous agents operate concurrently. First, task allocation should be integrated with resource management rather than treated as an independent process. Second, communication overhead should be considered when increasing the number of participating agents. Third, modular agent architectures can provide greater flexibility by allowing specialised agents to perform reasoning, planning, execution, monitoring, and resource-management functions independently. Such an architecture can support intelligent applications requiring autonomous workflow orchestration, distributed decision-making, and adaptive execution.

Computer Science and Artificial Intelligence perspective, these findings support the development of Agentic AI systems that combine autonomous reasoning with multi-agent coordination and computational resource optimisation. From a distributed-systems perspective, the results highlight the importance of scalability, communication efficiency, and workload distribution. The study therefore aligns particularly strongly with the Artificial Intelligence subject area while also maintaining relevance to Computer Networks and Communications through its focus on distributed agents and communication overhead.

Several limitations should be considered. First, the numerical benchmark results presented in Tables 1–3 originate from Song et al. (2025); therefore, they should be interpreted as comparative benchmark evidence rather than direct experimental validation of the proposed Agentic AI architecture. Second, the evaluated scenarios primarily represent static task-allocation conditions and therefore do not fully capture continuous workload changes, dynamic task arrivals, or complex environmental uncertainty. Third, the reported metrics emphasise cost, completion rate, response delay, computational time, and communication messages, while broader Agentic AI characteristics such as reasoning quality, autonomy, reliability, explainability, security, and human oversight require additional evaluation. Finally, real-world deployment may introduce hardware limitations, network variability, heterogeneous agent capabilities, and failure conditions that are not completely represented by the benchmark scenarios.

Future research should validate the proposed architecture through original simulation and experimental implementation rather than relying only on published benchmark data. Experiments can evaluate different numbers of agents, task intensities, heterogeneous computational resources, dynamic task arrivals, and agent failures. Future versions can incorporate reinforcement learning or LLM-based planning for adaptive task decomposition and autonomous decision-making. Resource optimisation can also be extended to edge-cloud environments, where communication latency and limited computational capacity are important considerations. Further work should investigate security, trust, explainability, safety, and human oversight, particularly for high-impact applications. Such developments would strengthen the architecture's relevance to scalable Artificial Intelligence, distributed intelligent systems, and autonomous multi-agent computing.

5. Conclusion

The study highlights the importance of scalable and coordinated architectures for the development of autonomous Agentic AI systems. The evaluated task-allocation scenarios demonstrate that increasing problem scale can substantially influence computational requirements, communication overhead, and overall allocation behaviour. The comparison of Hungarian, Auction, CBBA, and Auction 2-opt approaches further indicates that no single allocation strategy consistently provides the most favourable performance across different workload conditions. These observations reinforce the need for adaptive mechanisms capable of considering task requirements, agent capabilities, workload distribution, and available computational resources during autonomous execution. The proposed multi-agent architecture addresses these requirements by integrating task reasoning, planning, autonomous task allocation, communication, execution monitoring, resource management, and load balancing within a modular framework. Such integration provides a structured basis for coordinating specialised agents while supporting distributed execution and scalable operation. The architecture is particularly relevant to intelligent environments in which workloads and computational demands may vary over time. Overall, the study establishes a foundation for combining autonomous decision-making with multi-agent coordination and resource-aware execution. Although the presented benchmark evidence does not constitute direct experimental validation of the proposed architecture, it identifies important performance considerations that should guide its future implementation. Original simulation and real-world evaluation can further establish its effectiveness under dynamic workloads, heterogeneous resources, agent failures, and edge-cloud environments. The framework therefore provides a useful direction toward efficient, scalable, and autonomous multi-agent AI systems.

References

1. Abou Ali, M., Dornaika, F., & Charafeddine, J. (2025). Agentic AI: a comprehensive survey of architectures, applications, and future directions. Artificial Intelligence Review, 59(1), 11.
2. Ajmera, V. (2026). Agentic AI for Autonomous Decision-Making. SAMRIDDHI: A Journal of Physical Sciences, Engineering and Technology, 18(03), 73-81.
3. Ali, Z., & Grudén, K. (2025). Intelligent multi-agent framework for automated data analytics and machine learning tasks (AI agent).
4. Allmendinger, S., Bonenberger, L., Endres, K., Fetzer, D., Gimpel, H., & Kühl, N. (2026). Multi-agent AI. Electronic Markets, 36(1), 18.
5. Alva, L. R., & Pandey, B. (2026). Agentic AI systems in the age of generative models: architectures, cloud scalability, and real-world applications. Artificial Intelligence Review, 59(3), 88.
6. Banerjie, S., Zhu, Y., Freeman, I., Machado, J. V., Ahmed, A., Sarker, A., & Al-Garadi, M. (2025). Agentic AI in healthcare: a comprehensive survey of foundations, taxonomy, and applications.
7. Chahar, S., Gaba, P., Kaswan, K. S., Dhatterwal, J. S., & Baliyan, A. (2025, December). Agentic AI: A Paradigm Shift in Autonomous Decision-Making and Intelligent Systems. In 2025 IEEE 1st International Conference on Recent Trends in Computing and Smart Mobility (RCSM) (pp. 1-6). IEEE.
8. Dev, K., Khowaja, S. A., Zeydan, E., Singh, K., & Debbah, M. (2025). Advanced architectures integrated with agentic AI for next-generation wireless networks. IEEE Communications Standards Magazine.
9. Elkael, M., D'Oro, S., Bonati, L., Polese, M., Lee, Y., Furueda, K., & Melodia, T. (2026). AgentRAN: An agentic AI architecture for autonomous control of open 6G networks. IEEE Communications Magazine.
10. Felipe, L. G. (2026). Agentic AI Software Deployment: Enhancing Software Deployment with Agentic AI for Autonomous and Scalable Operations.
11. Haque, A. K. M., Ridoy, A. A. I., Rayhan, M., & Porres, I. (2026). Emergence of Agentic AI: A Review on Evolution, Background, Working Principles, Applications, Adoption Factors, and Future Research Directions. arXiv preprint arXiv:2608.18110.

12. Harini, P., Thota, B., Mallik, B., Kandi, V. P., & Chakravarthi, M. K. (2025, October). Agentic AI-Driven Decision Orchestration System for Real-Time Project Coordination. In 2025 2nd International Conference on Artificial Intelligence and Knowledge Discovery in Concurrent Engineering (ICECONF) (pp. 1-7). IEEE.
13. Jiang, F., Pan, C., Wang, K., Michiardi, P., Dobre, O. A., & Debbah, M. (2026). From large ai models to agentic ai: A tutorial on future intelligent communications. IEEE Journal on Selected Areas in Communications.
14. Karim, M. M., Van, D. H., Khan, S., Qu, Q., & Kholodov, Y. (2025). Ai agents meet blockchain: A survey on secure and scalable collaboration for multi-agents. Future Internet, 17(2), 57.
15. Molli, S. M. (2023). Trustworthy Agentic AI Systems: A Hybrid Cloud Framework for Scalable Autonomous Decision-Making. International Journal of Science, Technology and Convergence, 5(5).
16. Raghavendra, P., & Saikia, M. J. (2026). Agentic AI: A Perspective on Architecture, Frameworks and Applications. AI, 7(6), 219.
17. Reda, E., Yasser, M., & El-Metwally, S. (2026). Autonoma: A Hierarchical Multi-Agent Framework for End-to-End Workflow Automation. arXiv preprint arXiv:2603.19270.
18. Rekha, M., Ravi, C., & AlZadjali, N. S. H. (2026). Architecture of Agentic AI Depicting the Core Components Involved. Agentic Artificial Intelligence: Engineering AI Agents for Autonomous AI Systems, 31.
19. Sapkota, R., Roumeliotis, K. I., & Karkee, M. (2025). UAVs meet agentic AI: A multidomain survey of autonomous aerial intelligence and agentic UAVs. arXiv preprint arXiv:2506.08045.
20. Song, Y., Ma, Z., Chen, N., Zhou, S., & Srigrarom, S. (2025). Comparative Analysis of Centralized and Distributed Multi-UAV Task Allocation Algorithms: A Unified Evaluation Framework. Drones, 9(8), 530. https://doi.org/10.3390/drones9080530
21. Srivastav, P., & Rohit, P. (2025). Multi-Agent Artificial Intelligence for Autonomous Enterprise Automation.
22. Subir, B., Rajib, M., & Saikia, M. J. (2026). Agentic AI and Multi-Agent Collaboration in Healthcare: A Comprehensive Survey of Architectures, Clinical Safety, and Future Directions. Future Internet, 18(8), 391.
23. Talasila, D. (2026). Agentic AI in multi-agent systems: exploring the coordination, negotiation, and cooperation of autonomous artificial agents in competitive and collaborative digital ecosystems. Journal of Integrative Science and Societal Impact, 2(1), 64-71.
24. Thakur, L. (2026). Artificial intelligence agents and agentic AI systems: Architectures, capabilities, applications, challenges, and future directions. International Journal of Applied Resilience and Sustainability, 2(2), 721-744.
25. Warrier, A., & Abhilash, K. S. (2025, December). Autonomous agentic AI for clinical workflow orchestration: Self-managing healthcare operations. In 2025 6th International Conference on IoT Based Control Networks and Intelligent Systems (ICICNIS) (pp. 1450-1457). IEEE.
26. Zhang, R., Liu, G., Liu, Y., Zhao, C., Wang, J., Xu, Y., ... & Kim, D. I. (2026). Toward edge general intelligence with agentic AI and agentification: Concepts, technologies, and future directions. IEEE Communications Surveys & Tutorials, 28, 4285-4318.