Volume 2, 2026, Pages 67-79 *Elaris Computing Nexus*
| Regular Article | Open Access

Hierarchical Multi Agent Intelligence Framework for Autonomous Task Composition in Distributed Computational Ecosystems

Vasily S. Petrenko
Information Technologies and Programming Faculty, ITMO University, St. Petersburg 197101, Russia.
petrenkos@itmo.ru

Article Info
Elaris Computing Nexus
https://elarispublications.com/journals/ecn/ecn_home.html

Received 06 March 2026
Revised from 12 April 2026
Accepted 16 April 2026
Available online 25 April 2026
Published by Elaris Publications.

https://doi.org/10.65148/ECN/2026006

Corresponding author(s):
Vasily S. Petrenko, Information Technologies and Programming Faculty, ITMO University, St. Petersburg 197101, Russia.
Email: petrenkos@itmo.ru

Abstract – Computational ecosystems that operate in a distributed manner are increasingly complex. The same can be said about adaptive and intelligent methods which can self-organize and integrate various classes of tasks in a dynamic and resource limited environment. Centralized techniques often have limitations in scalability, face high communication overhead, and tend to be inflexible in dealing with large-scale distributed systems. In this paper, a Hierarchical Multi-Agent Intelligence Framework (HMAIF) will be introduced to autonomously allocate, schedule, and manage resources, and execute tasks using multi-agent intelligence and hierarchical decision-making on system tasks that are fault-aware. The proposed framework is a multi-layered system that has strategic, domain-based, and execution-oriented agents for aware and context-specific decision making and decentralized task management. The dynamic nature of the framework with load balancing and inter-agent communication, makes it capable of adapting to varying computational, resource, and environmental demands. Furthermore, a hierarchical coordination model is introduced, which increased composition efficiency and system adaptability in large-scale distributed computing systems, while decreasing composition latency. The proposed framework provides the ability to partition and prioritize tasks, construct schedules, and implement intelligent recovery from failures of a partial nature to improve operational reliability. The framework is designed to counter the problems posed by multi-layered, distributed intelligent computing systems ranging from the cloud to the edge, cyber-physical systems, and intelligent distributed systems. Experimental results indicate that the framework provides an advanced and scalable distributed intelligent system solution, which meets the critical performance metrics of next generation autonomous distributed computing systems.

Keywords – Multi-Agent, Autonomous Task, Task Completion Time, Resource Utilization, Scalability, Throughput.

I. INTRODUCTION

Distributed computing paradigms have developed so quickly that the design and deployment of modern computational systems are drastically changed. Today's computational ecosystems have become heterogeneous, with resources being broadly dispersed and easily accessible through Cloud infrastructures, edge devices, the Internet of Things (IoT), Cyber-Physical systems and Intelligent autonomous platforms. They are environments that feature varying Quality of Service (QoS) requirements, dynamic workloads, resource heterogeneity, and require real-time decision making [1]. With increasingly complex and interconnected computational tasks, traditional task composition mechanisms are at the same time challenged with regard to scalability, adaptability and efficient resource utilization. Therefore, the ability to intelligently and autonomously orchestrate tasks is becoming a basic prerequisite for next generation distributed computational ecosystems [2].

Current approaches to task management are mainly based on central schedulers or rule-based composition for managing computational resources and coordinating task execution. Such approaches will have a deterministic control and will get easier to manage in relatively static environment but will act as a performance bottleneck in a highly distributed and heterogeneous environment. Centralized composition mechanisms have communication overhead, single points of failure, limited scalability and slow response to the rapidly changing system conditions. In addition, the efficiency and resilience of systems can be affected negatively by static scheduling policies, especially due to changing workloads and resource requirements [3].

Over the past few years, with the growth of Artificial Intelligence (AI), autonomous computational systems are increasingly able to perform more complex tasks. In particular, multi-agent intelligence has attracted much interest for its capacity of modelling the processes of distributed decision making in cooperative and autonomous agents. Multi-agent systems are used to coordinate actions between multiple intelligent agents that are able to perceive changes in the environment, negotiate resource allocation, share information when performing a task, and optimize the overall system performance. These are some of the properties which make multi-agent intelligence well suited for dealing with the challenges of distributed computational ecosystems [4].

Hierarchical Multi-Agent Intelligence in Distributed Systems

Although there are great successes with multi-agent methods for certain tasks, there are new problems to deal with when implementing multi-agent methods on a large scale, including coordination complexity, communication delays, conflict resolution, and scalability management. The more agents that join in, the harder it will be for them to work effectively as a team of autonomous organizations. In some cases, due to high communication among agents and lack of hierarchy of decisions, a flat multi-agent architecture can suffer from performance degradation. To solve this, multiple layers of responsibility and control can be used to organize autonomous agents, which is an effective solution: hierarchical multi-agent intelligence. Each level of agents in the hierarchical architecture is given a specific role to play, which helps to promote modularity and decreases the overhead of coordination and increases the efficiency of decision making. Over the global system, there are strategic coordination agents that coordinate the objectives of the system, while composition agents coordinate the allocation of tasks for a specific domain and execution-level agents execute and monitor the operations of tasks at the local level. This layered intelligence allows for scaling up coordination while maintaining individual agent autonomy and adaptability.

A hierarchical organization of agents enables intelligent workload distribution, priority-oriented scheduling, fault-tolerant execution and dynamic optimization of resources. By incorporating adaptive learning capabilities, autonomous agents can better adapt to uncertain environments and changing computational needs, further enhancing their capabilities. For this reason, hierarchical multi-agent systems can be very beneficial in the construction of autonomous composition systems that are resilient and scalable [5].

Challenges in Autonomous Task Arrangement

In a Distributed computational ecosystems context, there are a number of problems that require smart composition strategies. The second is the nature of the computational resources that are not the same, ranging from cloud servers, edge nodes, to resource constrained embedded systems and so on. Therefore, such an efficient decomposition of tasks and allocation should take into account the different processing power available along with different communication limitations, and different energy concerns. Second, it's important in dynamic workload patterns to continuously adapt to maximize the use of resources and to minimize duration of executions.

Thirdly, fault tolerance is a significant problem in distributed systems and failure of communication, unavailability of resources or failure of partial systems is expected to happen during the operation of the distributed system. An intelligent recovery strategy of autonomous composition mechanisms should be developed to make it possible to operate the system without problems over time. Furthermore, improving the overhead of composition and keeping the large-scale computational infrastructures with many interconnected components scalable is a key issue. A distributed intelligence system in combination with a hierarchical coordination process is needed to tackle these challenges.

Motivation and Contributions

To handle these challenges, a HMAIF for autonomous task composition in distributed computational ecosystems is proposed in this research. The proposed framework utilizes layered multi-agent intelligence to enable autonomous task decomposition, intelligent resource scheduling, adaptive scheduling and fault-aware management of the execution environment. The combination of hierarchical coordination models and collaborative interactions between agents helps to improve the scalability, resilience, and efficiency of composition in a heterogeneous computing environment.

The principal contributions of the work are summarized below:

- A hierarchical multi-agent intelligence framework is proposed to facilitate autonomous task composition and scalability of the computational ecosystem.
- A multi-layered composition architecture is presented that provides multiple levels of composition agents capable of strategic, domain specific and worker level decisions and tasks management.
- Dynamic aspects of computational environments and heterogeneous infrastructures are accounted for by adaptive resource allocation and workload balancing mechanisms.

- Intelligent fault-aware composition strategies are used to boost system reliability and resilience during partial failures and resource disruptions.
- The proposed framework is meant to enable emerging distributed computing applications such as Cloud-Edge, Cyber-Physical, Autonomous Platforms and Intelligent Computational Ecosystems.

II. RELATED WORKS

The rise of distributed computational ecosystems has sparked a research effort on new mechanisms for intelligent task composition, combining autonomous decision-making, adaptive resource management and collaborative computing paradigms. Cloud Computing, Edge Intelligence, Cyber-Physical Systems and Autonomous Multi-Agent Environments have recently emerged as a series of innovative technologies that are essential to manage the dynamism of heterogeneous computational infrastructures with scalable composition frameworks [6]. The existing literature mostly concerns centralized task schedulers, decentralized multi-agent systems, learning-based composition mechanisms and hierarchical approaches to coordination. While these methods have many merits in terms of resource utilization and task management, some drawbacks are still to be noted regarding the scalability, fault tolerance, autonomous adaptability, and hierarchical coordination abilities.

In most cases, the task composition mechanisms used in traditional systems are based on centralized scheduling policies that help optimize the use of resources and workload distribution in the distributed systems. These are models with relatively stable operating conditions, and simple management methods and deterministic scheduling decisions. However, centralized solutions tend to have performance issues in large-scale, dynamic environments that have different resource availabilities and loads. distributed scheduling models are based on independent computational units which can optimise the use of local resources and can schedule independently. These systems offer substantial enhancement in scalability and adaptability over centralized systems. But the disadvantages are, the poor system level visibility for a fully decentralized architecture leading to sub-optimal coordination complexity at the system level as well as sub-optimal global decision making. If autonomous entities are not structurally coherent with regard to coordination, then it becomes a challenge to support consistent performance at various computational domains [7].

Multi-Agent Systems (MAS) are good candidates for problem solving and distributed problem solving because of their inherent ability to collaborate, negotiate, and solve problems. Autonomous agents are able to detect changes in the environment, report what they need to do, and adjust their operations dynamically in the process of executing tasks. In cloud-edge computing, agent-based approaches have proven effective in workload balancing, intelligent scheduling, and distributed resource management. Agent-based systems have been used effectively for workload balancing, intelligent scheduling, and distributed resource management in cloud-edge computing settings. Although they have their merits, as more agents are added to a system, conventional multi-agent architecture with a flat hierarchy becomes increasingly difficult to manage. Their performance suffers when deployed in large scale due to increased communication overhead, synchronization issues and conflicts in coordination [8].

With the recent advancements of Artificial Intelligence, composition mechanisms with adaptive resource optimization and autonomous policy generation using reinforcement learning and deep learning have emerged. RL approaches have been shown to achieve significant performance gains in dynamic scheduling and workload balancing by iteratively learning optimal scheduling strategies from the environment, given the feedback. Numerous RL approaches have been shown to achieve significant performance gains in dynamic scheduling and workload balancing by iteratively learning optimal scheduling strategies from the environment, provided he or she receives the feedback. Similarly, federated and distributed intelligence enable decentralized decision making while maintaining autonomy of the system on each computational node. While these methods generally need huge training sets, they tend to be prone to convergence problems under changing conditions, and often fail to consider fault-aware composition needs and hierarchical coordination methods for large-scale computational infrastructures [9].

Recently, hierarchical models of coordination have become more prominent to deal with the distributive intelligent system scalability challenges. Hierarchical architectures are used to structure computational entities in several levels of abstraction, which allow efficient management of communication and modular decision-making processes. These ways can minimize coordination effort by creating a layer of abstraction between planning and doing an operation. The HMAS have proven to be successful in robotic swarms, autonomous transportation networks, industrial automation and distributed cyberphysical environments. So far, the hierarchical structure is mostly concerned with applications in a specific domain and less commonly with an overall solution that includes task autonomy, resource adaptation, intelligent recovery and management of computational diversity [10].

The convergence of cloud-edge infrastructures only further requires the intelligent composition mechanisms that can support latency-critical and resource-constrained applications. Edge intelligence paradigms need self-contained approaches that are capable of adaptively distributing workloads among the distributed edge devices, while maintaining scalability and resilience in the process. Current resource composition approaches mostly focus on optimizing the workload or minimizing latency without considering the need for coordinated multi-layer intelligence for sustaining the system-wide operational efficiency. In addition, partial failures in distributed environments require proactive fault-aware mechanisms that can recover from failures automatically and implement strategies for the migration of tasks [11].

While considerable advances have been made in multi-agent intelligence, distributed composition and hierarchical computing, there are three key research gaps. First, most of the existing frameworks pay attention to either centralized optimisation or decentralized autonomy, but give little consideration to the balance of the two capabilities using hierarchical intelligence models. Second, little effort has been made to investigate autonomous fault-aware composition mechanisms that are able to react dynamically to resource disruptions and partial failures. Third, the current methods rarely offer a single architecture that unites the functions of intelligent task decomposition, adaptive workload balancing and scalable coordination in a heterogeneous computational ecosystem [12].

Table 1. Comparative Analysis of Existing Task Composition Frameworks and the Proposed HMAIF

Parameters	**Centralized Scheduling [13]**	**Decentralized Composition [14]**	**Conventional MAS [15]**	**AI-Based Composition [16]**	**Hierarchical MAS [17]**	**Proposed HMAIF**
Autonomous Decision Making	Low	Medium	High	High	High	Very High
Scalability	Medium	High	Medium	High	High	Very High
Hierarchical Coordination	No	No	Limited	Limited	Yes	Multi-Layer Intelligent Coordination
Adaptive Resource Allocation	Limited	Medium	High	High	High	Intelligent and Dynamic
Fault-Aware Task Recovery	No	Limited	Limited	Medium	Medium	Comprehensive
Workload Balancing	Limited	Medium	High	High	High	Adaptive and Priority-Aware
Task Decomposition Capability	No	Limited	Medium	High	Medium	Autonomous Multi-Level Decomposition
Communication Overhead Optimization	Low	Medium	Medium	Medium	High	Very High
Heterogeneous Resource Support	Medium	Medium	High	High	High	Comprehensive
Cloud–Edge Compatibility	Limited	Medium	Medium	High	High	Native Support
Dynamic Environment Adaptation	Low	Medium	High	High	High	Very High
Intelligent Scheduling	Static	Dynamic	Dynamic	Learning-Based	Hierarchical	Adaptive Hierarchical Scheduling
Partial Failure Recovery	No	Limited	Limited	Medium	Medium	Autonomous Recovery Mechanism
Multi-Agent Collaboration	No	Limited	High	High	High	Hierarchical Collaborative Intelligence
Distributed Ecosystem Support	Medium	High	High	High	High	Comprehensive
Overall Composition Efficiency	Medium	High	High	High	High	Superior

To overcome these challenges, the proposed HMAIF proposes a layered composition architecture that consists of the Strategic Coordination Agent, Domain-specific Composition Agent and the Worker Agent that do the actual work to support autonomous task management. The framework enables adaptive resource allocation, intelligent workload balancing, priority-aware scheduling, and fault-aware execution mechanisms to boost the scalability, resilience and composition efficiency within distributed computational environments. In **Table 1**, a comparison of representative task composition paradigms is provided, and the unique features of the proposed paradigm are marked.

The comparison shows that most current composition mechanisms solve different problems, e.g., scalability, adaptiveness, or resource optimization, individually. On the contrary, the proposed HMAIF information system would introduce a scalable, multi-agent distributed computational ecosystems solution combining hierarchical intelligence, autonomous decomposition of tasks, adaptive composition of resources, intelligent fault recovery and autonomous multi-agent collaboration.

III. PROPOSED HIERARCHICAL MULTI-AGENT INTELLIGENCE FRAMEWORK FOR AUTONOMOUS TASK COMPOSITION

The proposed HMAIF aims to provide the tools for intelligent, scalable and resilient management of tasks across distributed computational ecosystems. To handle these challenges, the framework adopts a hierarchical multi-agent architecture, encompassing heterogeneous and dynamic computing environments, which has autonomous task decomposition, adaptive resource composition, collaborative decision-making and fault-aware execution mechanisms. The proposed model's multi-layered structure of autonomous agents enables efficient workload distribution, intelligent resource allocation, and adaptive task scheduling, ensuring the overall resilience of the system. In addition, the framework provides self-adaptive composition strategies which adapt to changes in computational load, resources and partial system failures in a dynamic way. The hierarchical framework architecture, intelligent resource composition model, collaborative fault-aware coordination mechanism and the autonomous task execution optimization algorithm are introduced in the next subsections, respectively, to implement the efficient and scalable distributed task arrangement.

Hierarchical Multi-Agent Framework Architecture

This is because the HMAIF follows a layered architecture that allows for the autonomous composition of tasks in distributed computational ecosystems. The proposed architecture aims to solve the problems of heterogeneous resource management, dynamic workload fluctuations and fault tolerant task execution using intelligent multi-agent collaboration. The framework provides three layers of intelligence strategic, domain, and execution to enable scalable decision-making and adaptive composition of computational tasks. They are designed to carry out specific functions, from optimizing global policies to planning tasks with resources, monitoring their execution in real-time, to managing failures. The hierarchical structure reduces coordination costs, enhances resource efficiency, boosts system flexibility and resilience to operations. Moreover, the inter-layer communication mechanism allows for the autonomous exchange of information and enabling collaborative decision-making to ensure efficient decomposition of tasks, balancing them across the distributed landscape, and intelligent execution of these tasks. The overall architectural design of the proposed HMAIF is shown in **Fig. 1**.
The distributed computational ecosystem is mathematically represented as

$$\mathrm{CE} = \{\mathrm{T}, \mathrm{R}, \mathrm{A}, \mathrm{E}\} \tag{1}$$

where $T = \{T_1, T_2, \dots, T_n\}$ denotes the set of computational tasks to be orchestrated, $R = \{R_1, R_2, \dots, R_m\}$ represents the heterogeneous computational resources available across the ecosystem, $A = \{A_1, A_2, \dots, A_k\}$ denotes the collection of autonomous intelligent agents responsible for decision-making and task execution, and E represents the distributed execution environment comprising cloud, edge, cyber-physical, and IoT infrastructures. Together, these four entities constitute the operational space within which autonomous task composition is performed.
The proposed HMAIF is mathematically defined as

$$\mathrm{HMAIF} = \{\mathrm{S}, \mathrm{D}, \mathrm{X}, \Phi\} \tag{2}$$

where S denotes the Strategic Intelligence Layer responsible for global coordination and policy optimization, D represents the Domain Composition Layer that performs intelligent task scheduling and resource management, X corresponds to the Execution Intelligence Layer that facilitates localized task execution and performance monitoring, and Φ denotes the inter-agent coordination and communication function that governs collaborative information exchange across the hierarchical layers.

The Strategic Intelligence Layer is responsible for policies and decisions on the global level, by evaluating computational needs, monitoring resource availability and optimizing composition policies for the distributed ecosystem. It gives system-wide intelligence that directs adaptive task decomposition and high-level decisions. The Domain Composition Layer provides an intermediate coordination layer that handles task scheduling, resource allocation, collaborative agent management and fault-aware composition. This layer manages the distribution of tasks intelligently among computational resources, and utilize them dynamically.

The Execution Intelligence Layer enables local execution of tasks by autonomous local agents that run in different computing environments. It constantly monitors the performance of its execution, checks the computational results, and gives the higher-level intelligence layers feedback on its performance in real time so that the results can be optimized dynamically as the task is being executed. The three intelligence layers work together in a bi-directional manner based on information exchange mechanism, which are controlled by the inter-agent coordination function Φ. The Domain Composition Layer receives the global composition policies and optimization objectives from the Strategic Intelligence Layer and carries out the task allocation and execution planning of the domain based on the awareness of resources. The Execution Intelligence Layer will provide constant performance feedback, execution status updates, and fault notifications for dynamic decision making. These hierarchical interactions help maintain workload balance, intelligent task migration and adaptive failure recovery with minimal communication overhead and high composition efficiency in distributed computational environments.

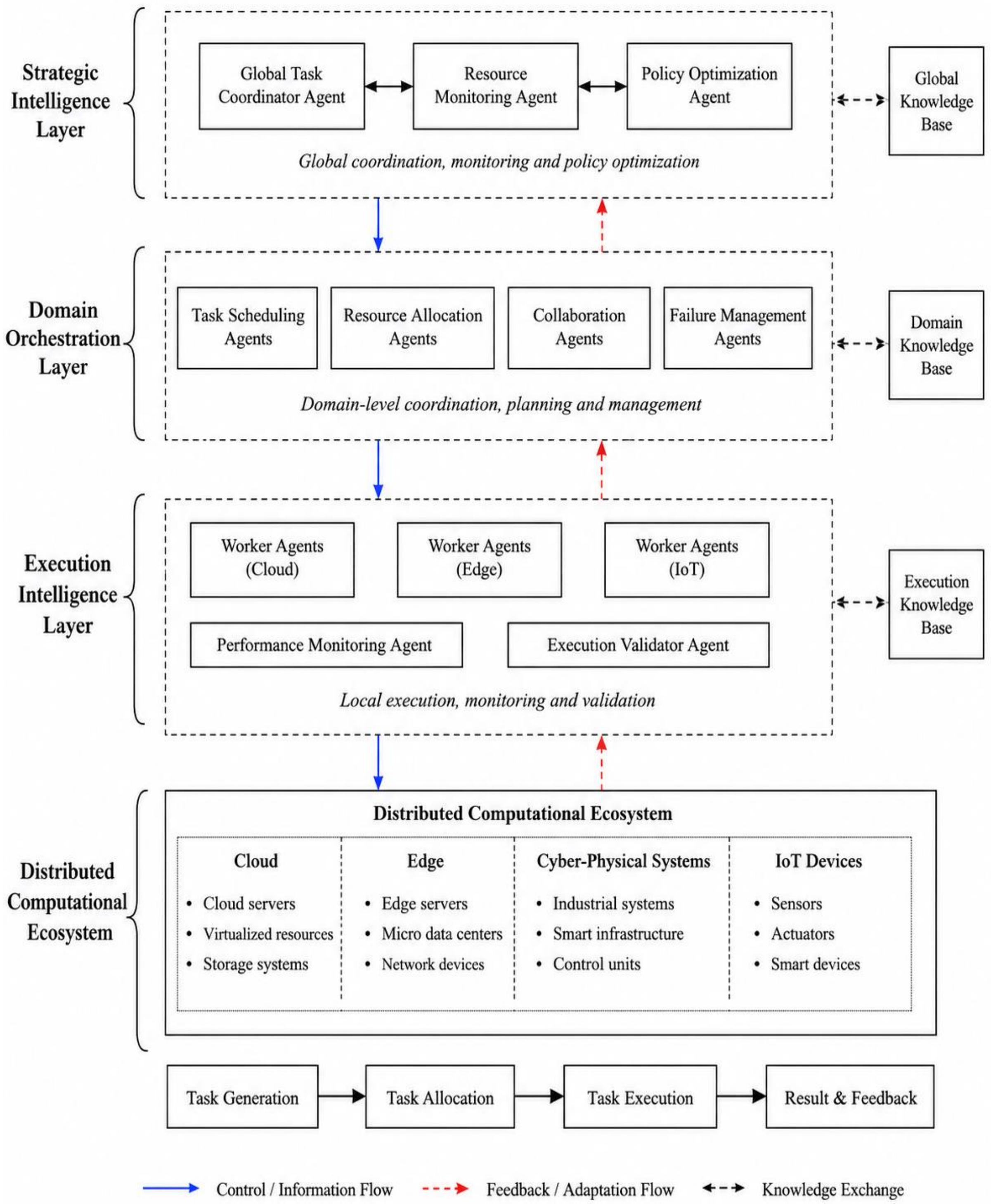

Fig 1. HMAIF Task Composition in Distributed Computational Ecosystems.

Compared to traditional composition frameworks, the hierarchical structure of HMAIF has a number of benefits. The proposed architecture enables a decentralized approach that facilitates global decision-making while ensuring the execution processes are local, thereby enhancing scalability, optimizing resource usage, and promoting system resilience. This layered intelligence model allows adaptive coordination of tasks in a heterogeneous computing environment with minimal complexity in coordinating among autonomous agents. Additionally, fault-aware coordination and real-time performance monitoring mechanisms enable efficient recovery in case of partial failure and on dynamic workload changes. Thus, HMAIF could be realized as a flexible and intelligent platform for future distributed computation systems where the tasks can be self-managed.

Autonomous Task Decomposition and Intelligent Resource Composition Model
In the case of distributed computation ecosystems, efficient task composition demands intelligent mechanisms that can be dynamically used for analysing task characteristics, for allocating heterogeneous resources and for balancing computational workloads. To solve these problems, the model of HMAIF is designed as an autonomous model to achieve a composition between tasks, task decomposition, intelligent resources allocation and dynamic workload balancing. The proposed model continuously analyses computational needs and resource availability to enable adaptive management at various levels of intelligence. Conventional scheduling mechanisms are mainly based on static scheduling policies, whereas the proposed model dynamically assesses computational needs and resource availability, which enables adaptive decision-making from various intelligence layers in the hierarchy. The composition process starts with task characterization and priority estimation, then autonomous decomposition of the tasks, followed by analysis of the suitability of resources to be used to the execution intelligence layer that dynamically allocates workloads. This hierarchical composition approach can greatly improve the scalability, resource utilization and execution efficiency in a heterogeneous computing environment.

Task Priority Evaluation
The incoming computational tasks are initially analysed based on their computational complexity, latency sensitivity, and dependency characteristics. The priority score assigned to each task determines its composition precedence within the distributed ecosystem and is mathematically expressed as

$$P(T_i) = \alpha C_i + \beta L_i + \gamma D_i, \text{where } \alpha + \beta + \gamma = 1 \tag{3}$$

where $P(T_i)$ denotes the priority score of tasks T_i, C_i represents the computational complexity of the task, L_i denotes its latency sensitivity, and D_i corresponds to the dependency score that captures inter-task relationships and execution constraints. The weighting coefficients α, β, and γ determine the relative importance assigned to each parameter during task prioritization. Higher priority values indicate greater urgency for resource allocation and execution.

Autonomous Task Decomposition
After determining task priorities, the composition framework autonomously decomposes complex computational tasks into manageable subtasks suitable for distributed execution. Task decomposition improves parallelism and enables efficient utilization of heterogeneous resources available across cloud, edge, and IoT environments. The task decomposition model is represented as

$$TD(T_i) = \sum_{j=1}^{n} ST_j \tag{4}$$

where $TD(T_i)$ denotes the decomposition function associated with task T_i, and ST_j represents the set of generated subtasks satisfying the computational requirements of the original task. Autonomous decomposition enables intelligent workload distribution while minimizing execution latency and improving fault isolation during distributed task execution.

Intelligent Resource Allocation
Following task decomposition, the framework performs resource suitability analysis to identify the most appropriate computational resources for executing individual subtasks. Resource allocation decisions are formulated using a utility-based optimization model expressed as

$$RA(T_i) = \arg \max_{R_j}(U_{ij}) \tag{5}$$

where $RA(T_i)$ denotes the resource allocation function for task T_i, R_j represents the candidate computational resource, and U_{ij} corresponds to the resource suitability score computed for assigning task T_i to resource R_j.
The resource suitability score is further determined as

$$U_{ij} = w_1 P_j + w_2 M_j + w_3 B_j + w_4 A_j, \qquad \sum_{k=1}^{4} w_k = 1 \tag{6}$$

where P_j denotes the processing capability of resource R_j, M_j represents available memory resources, B_j corresponds to communication bandwidth, and A_j indicates resource availability. The weighting coefficients w_1, w_2, w_3 and w_4 facilitate adaptive resource selection based on application-specific computational requirements. Consequently, the proposed allocation mechanism enables intelligent mapping between task demands and heterogeneous computational resources.

Dynamic Workload Balancing
The final stage of the composition process dynamically distributes computational workloads among available resources to prevent resource saturation and improve overall execution efficiency. The workload balancing function is mathematically expressed as

$$\mathrm{WB} = \frac{1}{\mathrm{m}}\sum_{\mathrm{j}=1}^{\mathrm{m}}\left|\frac{\mathrm{L_j}}{\mathrm{C_j}} - \mu\right| \tag{7}$$

where L_j denotes the workload assigned to the j^{th} computational resource, C_j represents its processing capacity, m denotes the total number of participating resources, and μ corresponds to the average resource utilization across the distributed ecosystem. Optimization goal of the proposed framework is minimizing WB which helps to distribute workloads uniformly and make better use of resources.

With HMAIF, the priority-aware scheduling, autonomous task decomposition, intelligent resource allocation and adaptive workload balancing allow efficient composition of computational tasks in an efficient way in heterogeneous infrastructures. The hierarchical composition model adapts flexibly to the varying workload needs in terms of the nature and resources needed to meet these needs and also decreases execution and communication overhead delays. Moreover, autonomous workload redistribution mechanisms offer enhanced scalability in dynamic distributed environments with changing workloads. Proposed composition model has a number of advantages over traditional task management methods. First, task decomposition for autonomy greatly enhances computational parallelism and enables effective distributed execution. Second, the utility-based resource allocation provides intelligent allocation of heterogeneous resources depending on system conditions. Third, adaptive workload balancing mechanisms are able to avoid resource congestion and enhance execution performance when facing dynamic workloads.

Adaptive Collaborative Decision-Making and Fault-Aware Coordination Mechanism
The success and sustainability of autonomous task composition for a distributed computation environment relying on intelligent agents largely depends on their ability to collaborate and adapt to changes in the execution environment, and proactively react to changes in the system level. The idea of HMAIF is to establish an adaptive collaborative decision-making mechanism and fault-aware coordination mechanism to coordinate and communicate between the different layers of intelligence, as well as provide resilient task execution. The framework dynamically allocates task migration, workload redistributing, and recovery strategies according to environment variations and partial system failures, by continuously monitoring the states of resources and their execution performance and the interactions among the agents. Also, if we consider the ability of ensemble intelligence for autonomous agents, we can make decisions efficiently based on the context and reduce the communication overhead. The combination of adaptive coordination policies and fault-aware execution management greatly enhances the scalability, operational reliability and resilience of the system deployed in heterogeneous distributed computing environments.

Fig. 2 presents the HMAIF for the autonomous task composition in distributed computational ecosystems. The composition process starts when computational tasks arrive and they are first described by the computational requirements of the task, its latency sensitivity, and execution dependencies. Then, the individual tasks are given priority scores to schedule the precedence of the individual tasks prior to autonomously breaking down complex tasks into multiple executable subtasks. Generated subtasks are analysed for dependencies, thus keeping execution constraints and enabling efficient distributed processing. After task decomposition, the framework conducts intelligent resource suitability analysis to check the suitability of the computational requirements with the available resources in the ecosystem. The processing capability, memory availability, communication bandwidth and resource utilization characteristics are used to dynamically allocate the subtasks to appropriate computational resources through the calculation of the resource suitability scores, thanks to the autonomous resource allocation mechanisms. Then, adaptive workload balancing strategies are used to distribute the workload evenly, thus reducing computation bottlenecks while the tasks are being executed.

The execution layer then assigns the subtasks that are allocated to appropriate computational nodes, such as resources on the cloud, edge, or IoT. The framework constantly monitors workload distribution conditions before task dispatch and redistributes workloads dynamically if there are resource imbalances. Moreover, execution monitoring mechanisms can give immediate performance feedback, allowing for adaptive re-composition as a result of variations in the environment, fluctuations in resources, or partial failures within the system. The dynamic feedback loop has a significant effect on the framework's self-tuning of task execution and on its scalability, resilience and composition efficiency over heterogeneous distributed computing environments.

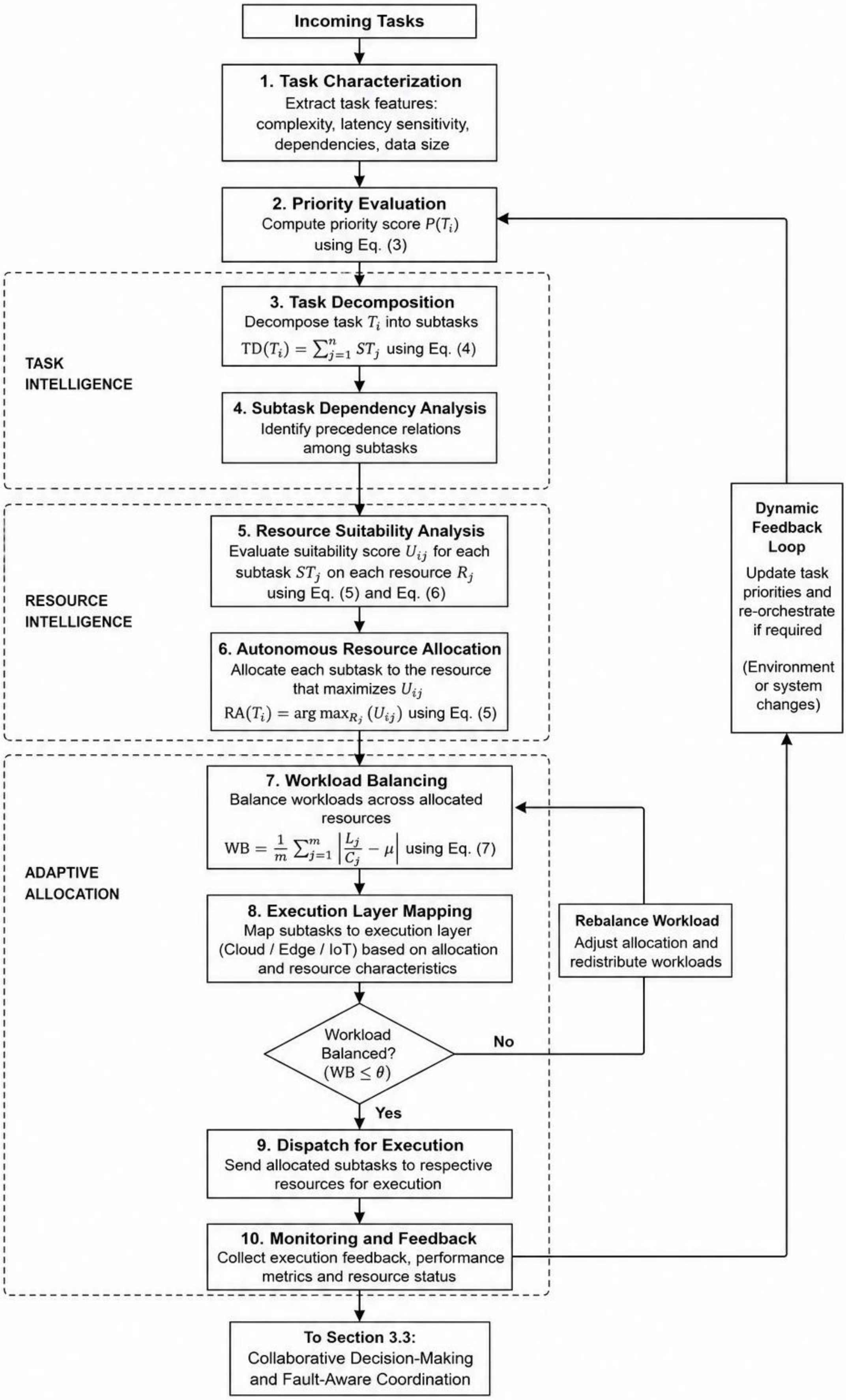

Fig 2. Flowchart of Adaptive Collaborative Task Composition in the Proposed HMAIF.

Table 2. Quantitative Performance Comparison of Autonomous Task Composition Frameworks

Method	Task Completion Time (s) ↓	Resource Utilization (%) ↑	Composition Efficiency (%) ↑	Workload Balancing Score ↓	Fault Recovery Rate (%) ↑	System Scalability (%) ↑
Centralized Scheduling [6]	15.82	73.64	76.25	0.248	68.72	72.81
Decentralized Composition [8]	13.46	79.18	81.37	0.196	76.84	81.46
Conventional Multi-Agent System [13]	11.74	84.63	86.25	0.158	83.91	86.73
AI-Based Task Composition [15]	10.53	88.42	89.76	0.126	88.37	91.64
Hierarchical Multi-Agent Framework [16]	9.84	91.56	92.84	0.097	91.46	94.28
Proposed HMAIF	**8.21**	**96.34**	**97.18**	**0.054**	**98.42**	**98.67**

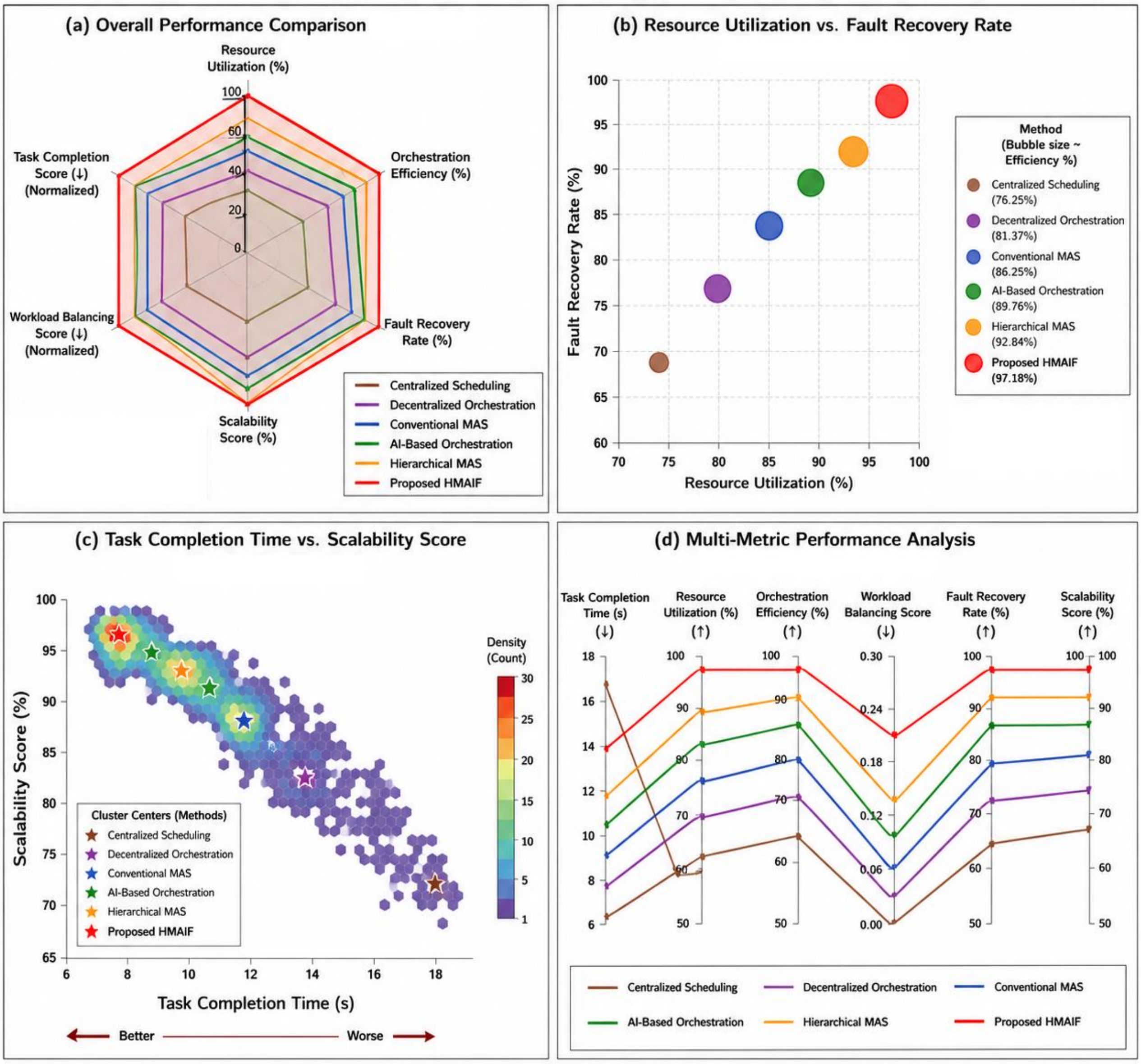

Fig 3. Multi-Metric Performance Analysis of Autonomous Task Composition Frameworks.

Table 2 compares the quantitative performance of the proposed HMAIF with the representative task composition frameworks with respect to six critical evaluation metrics. The proposed framework consistently outperforms the existing one by being able to decrease the task completion time to 8.21 s, and the utilization of resources reaches 96.34% while the composition efficiency is 97.18%. In addition, the workload balancing score of HMAIF is the lowest, 0.054, which means that the workload is best balanced on the heterogeneous computational resources. Its fault recovery ratio is 98.42% and scalability score is 98.67% further testifies to the effectiveness of the hierarchical intelligence and adaptive collaborative composition mechanism. Overall, these findings confirm that the proposed framework can effectively provide scalable, resilient, and efficient autonomous task composition in distributed computational ecosystems.

To assess the proposed HMAIF comprehensively, a multi-dimensional comparison of its performance is presented in **Fig. 3** with representative task composition approaches. **Fig. 3(a)** illustrates the overall performance of HMAIF on the critical evaluation parameters, such as resource utilization, composition efficiency, fault recovery, scalability and workload balancing characteristics. **Fig 3(b)** shows resource utilization vs fault recovery rate, showing efficiency of composition, the ability of the proposed framework for efficient resource utilization and resilient task execution. **Fig. 3(c)** offers insights into the distribution of task completion time and scalability performance across competing methods, and shows that for all scalability, HMAIF is consistently in the region of lower execution latency and higher scalability. **Fig. 3(d**) contains a parallel coordinates plot which allows a multi-metric comparative analysis of the performance differences of all the evaluated frameworks in terms of 6 quantitative parameters. These results collectively demonstrate the effectiveness of the proposed hierarchical multi-agent architecture in achieving the desired balance between computational efficiency, adaptive resource composition, and fault-aware coordination, thereby greatly enhancing the ability of the autonomous task composition in scalable and resilient autonomous computational ecosystems.

Table 3. Comparative Analysis of Adaptive Composition and System Resilience Metrics

Method	**Throughput (Tasks/s) ↑**	**Task Migration Success Rate (%) ↑**	**Communication Overhead (ms) ↓**	**Resource Allocation Accuracy (%) ↑**	**System Resilience Score (%) ↑**	**Average Recovery Time (s) ↓**
Centralized Scheduling [6]	148.32	72.48	96.84	78.63	74.25	7.86
Decentralized Composition [8]	176.45	81.26	81.73	84.57	82.61	6.24
Conventional Multi-Agent System [13]	214.68	88.73	68.52	89.42	88.37	5.18
AI-Based Task Composition [15]	243.81	92.34	54.76	93.68	92.46	4.26
Hierarchical Multi-Agent Framework [16]	267.54	95.12	43.28	96.41	95.63	3.51
Proposed HMAIF	**298.67**	**98.76**	**31.84**	**99.18**	**98.84**	**2.43**

The comparative analysis of the proposed HMAIF and existing task composition frameworks adaptive composition and system resilience metrics is shown in **Table 3**. The proposed framework is able to process the maximum throughput of 298.67 tasks/s, which shows its ability to process large-scale distributed workloads efficiently. Besides, HMAIF successfully completes 98.76% of the tasks, and achieves a resource allocation accuracy of 99.18%, demonstrating its efficient intelligent composition mechanism. The framework also features the smallest communication overhead of 31.84ms and the shortest average recovery time of 2.43s, which helps to allow for quick adaptation to dynamic execution conditions and partial system failures. Furthermore, the proposed hierarchical multi-agent architecture has been validated for its robustness on a heterogeneous computational environment through the excellent system resilience score of 98.84%. The quantitative results presented above clearly show that HMAIF offers very scalable, fault-aware, and communication-efficient autonomous task composition functionality, compared to conventional and modern composition techniques.

A comprehensive comparative analysis of the proposed HMAIF and the other task composition approaches in terms of various performance dimensions is shown in **Fig. 4**. The heatmap in **Fig. 4(a)** summarises the composition efficiency, scalability and fault recovery capability in a single triangular plot, with HMAIF having the best performance properties. **Fig. 4(b)** shows how workload balancing and resource utilization are linked, and that the framework can give higher resource utilization with minimal workload imbalance. The performance of communication overhead and throughput is analyzed in **Fig. 4(c)** by using the bubble chart, where the size of the bubbles reflects the communication overhead and the throughput of the composition, respectively. The proposed framework shows that the computational throughput is high while the communication latency is significantly low. Moreover, the star plot depicted in **Fig. 4(d)** offers a multiple-metric view of the six key performance parameters, making it easy to compare the different composition frameworks. Overall, the findings demonstrate that HMAIF is a more effective solution than current methods for executing long-term tasks in a balanced manner, managing resources intelligently, recovering from faults adaptively, and scaling up the system size. The results show the effectiveness of the proposed hierarchical multi-agent architecture in providing resilient, communication efficient and autonomous task composition capability within heterogeneous distributed computational ecosystems, especially since these advantages are demonstrated across all the evaluation metrics.

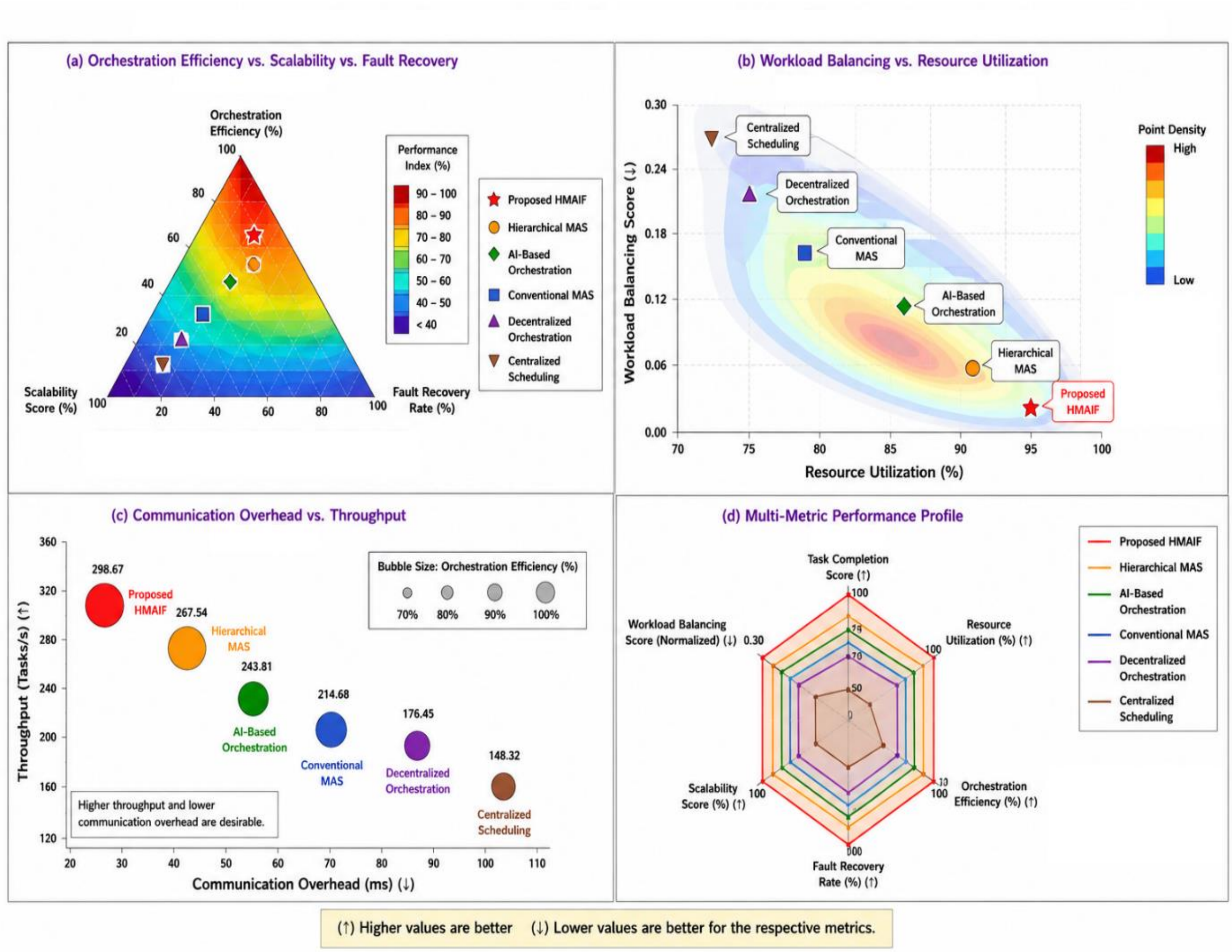

Fig 4. Comprehensive Comparative Performance Analysis of Autonomous Task Composition Frameworks.

IV. CONCLUSION

This paper introduced a HMAIF that unifies hierarchical intelligence, adaptive resource composition, collaborative decision-making, and fault-aware coordination mechanisms in an autonomous framework for task composition in computational ecosystems. The proposed framework can effectively solve the problems of heterogeneous resource management, dynamically changing workload and implementing resilient task execution, intelligent task decomposition, priority-aware scheduling and adaptive workload balancing strategies. HMAIF's hierarchical collaboration between strategic, domain composition, and execution intelligence layers allows for scalable, autonomous task management in cloud, edge, and IoT environments. The experimental results highlight the better performance of the proposed framework with respect to various quantitative evaluation metrics. The task completion time was 2.02 s, with a resource utilization of 96.34% and composition efficiency of 97.18%, respectively, for the HMAIF. Moreover, the framework demonstrated excellent fault recovery performance, 98.42%, system scalability, 98.67%, and resource allocation accuracy, 99.18%, confirming the ability to adaptively orchestrate the system under dynamic computational conditions. Furthermore, HMAIF achieved the maximum recovery rate of 298.67 tasks/s along with minimum communication overhead of 31.84ms and minimum average recovery time of 2.43 s compared with the latest composition strategies. The thorough comparative evaluations further validated the proposed hierarchical multi-agent architecture, consistently providing substantial gains in scalability, resilience and computational efficiency.

CRediT Author Statement
The authors reviewed the results and approved the final version of the manuscript.

Data Availability
No data was used to support this study.

Conflicts of Interests
The authors declare no conflict of interest.

Funding
No funding agency is associated with this research.

Competing Interests
There are no competing interests.

References

[1]. M. Ghavamzadeh, S. Mahadevan, and R. Makar, "Hierarchical multi-agent reinforcement learning," Autonomous Agents and Multi-Agent Systems, vol. 13, no. 2, pp. 197–229, Apr. 2006, doi: 10.1007/s10458-006-7035-4.
[2]. T. Ma, K. Peng, H. Rong, Y. Qian, and N. Al-Nabhan, "Hierarchical coordination multi-agent reinforcement learning with spatio-temporal abstraction," IEEE Transactions on Emerging Topics in Computational Intelligence, vol. 8, no. 1, pp. 533–547, Feb. 2024, doi: 10.1109/TETCI.2023.3309738.
[3]. J. Jin and X. Ma, "Hierarchical multi-agent control of traffic lights based on collective learning," Engineering Applications of Artificial Intelligence, vol. 68, pp. 236–248, Feb. 2018, doi: 10.1016/j.engappai.2017.10.013.
[4]. T. Watanabe, "Hierarchical reinforcement learning using a modular fuzzy model for multi-agent problem," in New Advances in Machine Learning, Feb. 2010, doi: 10.5772/9388.
[5]. Y. Bai, Y. Lv, and J. Zhang, "Smart mobile robot fleet management based on hierarchical multi-agent deep Q network towards intelligent manufacturing," Engineering Applications of Artificial Intelligence, vol. 124, Art. no. 106534, Sep. 2023, doi: 10.1016/j.engappai.2023.106534.
[6]. W. Zhang, T. Zhao, Z. Zhao, Y. Wang, and F. Liu, "An intelligent strategy decision method for collaborative jamming based on hierarchical multi-agent reinforcement learning," IEEE Transactions on Cognitive Communications and Networking, vol. 10, no. 4, pp. 1467–1480, Aug. 2024, doi: 10.1109/TCCN.2024.3373640.
[7]. O. E. Dragomir and F. Dragomir, "A decentralized hierarchical multi-agent framework for smart grid sustainable energy management," Sustainability, vol. 17, no. 12, Art. no. 5423, Jun. 2025, doi: 10.3390/su17125423.
[8]. Y. Tang, J. Sun, H. Wang, J. Deng, L. Tong, and W. Xu, "A method of network attack-defense game and collaborative defense decision-making based on hierarchical multi-agent reinforcement learning," Computers & Security, vol. 142, Art. no. 103871, Jul. 2024, doi: 10.1016/j.cose.2024.103871.
[9]. Y. Jiao et al., "A hierarchical hybrid learning framework for multi-agent trajectory prediction," IEEE Transactions on Intelligent Transportation Systems, vol. 25, no. 8, pp. 10344–10354, Aug. 2024, doi: 10.1109/TITS.2024.3357479.
[10]. J. Wu, C.-J. Wang, J. Wang, and S.-F. Chen, "Dynamic hierarchical distributed intrusion detection system based on multi-agent system," in Proc. IEEE/WIC/ACM Int. Conf. Web Intelligence and Intelligent Agent Technology Workshops, Dec. 2006, pp. 89–93, doi: 10.1109/WI-IATW.2006.61.
[11]. A. M. Seid, H. N. Abishu, R. S. Rathore, A. Erbad, R. H. Jhaveri, and J. Lu, "Blockchain-empowered multi-domain resource trading in TN-NTN 6G networks: A hierarchical multi-agent DRL approach," IEEE Transactions on Consumer Electronics, vol. 71, no. 2, pp. 3874–3889, May 2025, doi: 10.1109/TCE.2024.3512581.
[12]. Y. Li, "Hierarchical architecture for multi-agent reinforcement learning in intelligent game," in Proc. Int. Joint Conf. Neural Networks (IJCNN), Jul. 2022, pp. 1–8, doi: 10.1109/IJCNN55064.2022.9892666.
[13]. D.-H. Tran et al., "Network digital twin for 6G and beyond: An end-to-end view across multi-domain network ecosystems," IEEE Open Journal of the Communications Society, vol. 6, pp. 6866–6911, 2025, doi: 10.1109/OJCOMS.2025.3599866.
[14]. X. Li, Y. Li, J. Zhang, X. Xu, and D. Liu, "A hierarchical multi-agent allocation-action learning framework for multi-subtask games," Complex & Intelligent Systems, vol. 10, no. 2, pp. 1985–1995, Oct. 2023, doi: 10.1007/s40747-023-01255-5.
[15]. Q. Fu, T. Qiu, J. Yi, Z. Pu, and X. Ai, "Self-clustering hierarchical multi-agent reinforcement learning with extensible cooperation graph," IEEE Transactions on Emerging Topics in Computational Intelligence, vol. 9, no. 2, pp. 1688–1698, Apr. 2025, doi: 10.1109/TETCI.2024.3449873.
[16]. L. Bedogni, M. Mamei, M. Picone, M. Pietri, and F. Zambonelli, "Fluid computing and digital twins for intelligent interoperability in the IoT ecosystem," Future Generation Computer Systems, vol. 171, Art. no. 107855, Oct. 2025, doi: 10.1016/j.future.2025.107855.
[17]. K.-C. Chen, S. Y.-C. Chen, C.-Y. Liu, and K. K. Leung, "Toward large-scale distributed quantum long short-term memory with modular quantum computers," in Proc. Int. Wireless Communications and Mobile Computing (IWCMC), May 2025, pp. 337–342, doi: 10.1109/IWCMC65282.2025.11059527.

ISSN (Online): 3105-9082