Computer Modeling in
Engineering & Sciences

Tech Science Press

https://doi.org/10.32604/cmes.2026.084691

ARTICLE

Optimal Task Assignment in Holonic Multi-Agent Systems by Resolving Performative Inconsistencies

Awais Qasim[1], Aniqa Iftikhar[1], Hanaa Nafea[2], Nay Chi Moe Oo[3] and Byung-Seo Kim[4,*]

[1]Department of Computer Science, GC University, Lahore, Pakistan
[2]Department of Computer Science, College of Computer Science and Engineering, Taibah University, Al-Madinah Al-Munawwarah, Saudi Arabia
[3]Sirindhorn International Institute of Technology, Thammasat University, Pathum Thani, Thailand
[4]Department of Software and Communications Engineering, Hongik University, Sejong-si, Republic of Korea

*Corresponding Author: Byung-Seo Kim. Email: jsnbs@hongik.ac.kr

Received: 27 April 2026; Accepted: 15 June 2026; Published: 27 July 2026

ABSTRACT: Optimal task assignment in holonic multi-agent systems has emerged as a pivotal problem in modern distributed systems. Despite substantial gains in agent coordination, many large-scale systems still suffer from poor job allocation, resulting in performance bottlenecks and resource waste. Effective task assignment is critical for these systems since it influences individual agent performance and overall adaptability. A significant challenge within holonic multi-agent systems is ensuring optimal task assignment while resolving performative inconsistencies, such as role conflicts and coordination failures among agents. This research proposes a novel optimization framework to address these inconsistencies, enabling more efficient task allocation in holonic multi-agent systems. An objective function that minimizes task completion time, resource usage, and task priority while accounting for performative inconsistencies is presented. The effectiveness of the approach is demonstrated through real-world scenarios of smart transportation systems. Simulation results show that the proposed task allocation framework enables holons to achieve improved overall system performance through optimal distribution of task sets and effective resolution of performative inconsistencies.

KEYWORDS: Holonic multi agent system; task assignment; optimization; resource management; intelligent transport system

1 Introduction

Optimal task assignment in Holonic Multi-Agent Systems (HMAS) has emerged as a pivotal problem in modern distributed systems, improving both resource efficiency and system scalability. Despite substantial gains in agent coordination, many large-scale systems still suffer from poor job allocation, resulting in performance bottlenecks and resource wastage. Effective task assignment is critical for these systems since it influences individual agent performance and overall adaptability. Using advanced optimization techniques, HMAS can guarantee that tasks are distributed to maximize productivity and minimize system costs, making it a key focus in the study and application of complex systems. In MAS (Multi-Agent System), an autonomous agent makes decisions influenced not only by its perceptions of the environment but also by prior knowledge and interaction with other agents, such as a set of predefined actions established during the design process [1,2].

HMAS is a specialized form of MASs where the system is hierarchically organized into nested agents known as holons, forming a recursive structure. Each agent has a certain amount of autonomy, which allows it to make local decisions based on its own knowledge and expertise while creating dynamic linkages to manage the system. Holon integrates both local and global perspectives, as well as individual and collective viewpoints. A Holon is a self-similar structure made up of other Holons, and when these are organized hierarchically, this is known as holarchy, as shown in Fig. 1. This holarchy consists of autonomous, self-sufficient agents, which may include independent agents, all referred to as Holons. These HMAS have been successfully implemented in a variety of complex systems and have proven to be an efficient solution to a number of issues related to hierarchical and self-organizing structures [3]. For example, it has been applied to transportation, distributed sensor management, supply chain management, health organizations, biological network simulation, complex software systems, and the,smart grid market to manage electricity agents in Smart grid operation [4,5].

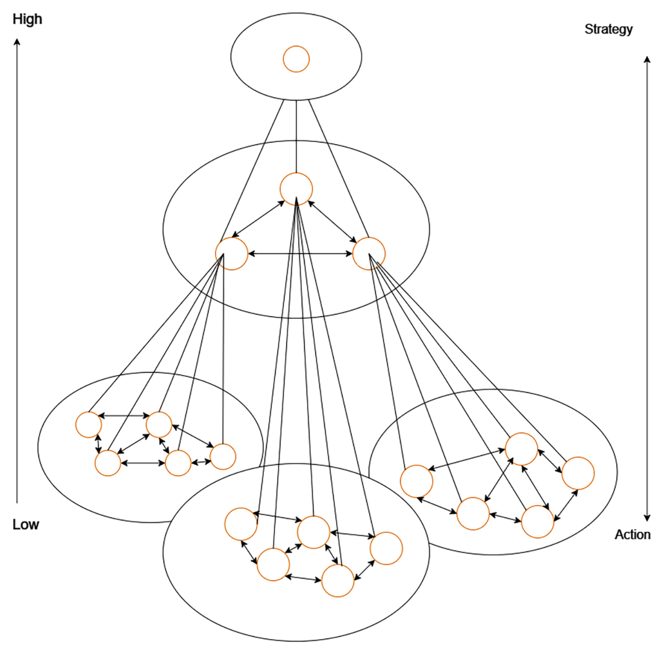


Figure 1: Abstract structure of three levels of holons.

The Optimal task assignment mechanism is important for HMAS because it enhances resource efficiency and reduces system costs. Optimization in task assignment involves determining which agents perform which task, how agents should communicate, and how their behaviors should be structured to achieve optimal performance [6,7]. Efficient task assignment must ensure adaptability and resilience to fluctuations.

MAS are heavily dependent on standardized communication mechanisms to enable coordination, cooperation, and negotiation among distributed agents. The Foundation for Intelligent Physical Agents (FIPA) introduced the FIPA-ACL (Foundation for Intelligent Physical Agents-Agent Communication Language) standard to provide a structured communication language based on speech-act theory, supporting performatives such as request, inform, propose, and confirm for interoperable agent interaction [8]. The Java Agent Development Framework (JADE) implements FIPA-compliant communication protocols and has been widely adopted for developing distributed intelligent systems due to its support for agent messaging, ontology management, and behavioral coordination [9]. Wooldridge and Jennings further emphasized that effective communication and cooperation are fundamental characteristics of intelligent agents operating in dynamic environments, particularly in distributed task allocation and autonomous decision-making scenarios [10]. These communication principles form the foundation of the proposed HMAS framework for resolving performative inconsistencies during optimal task assignment. The performatives specify how agents can communicate with one another and how their messages are structured. However, the diversity of communicative acts among different agents can lead to an increased risk of misunderstanding. Due to these performative inconsistencies, holons with similar capabilities exhibit varying performance under different conditions. These inconsistencies stem from factors like resource limitations, environmental changes, and communication delays. Traditional approaches of optimal task allocation in holons often fail to adapt to these dynamic conditions, resulting in sub-optimal task allocation. This leads to the research problem of how to optimally assign tasks to holons that account for and resolve performative inconsistencies, improving system performance, adaptability, and scalability across complex, real-world applications.

2 Related Work

The work of Inal, Sel, Aktepe, Türker, and Ersöz investigates the influence of performative inconsistencies on system efficiency, underlining the importance of resilient task allocation techniques to optimize resource utilization and boost overall production. However, their work focuses primarily on optimizing scheduling performance using reinforcement learning but does not address communication-level inconsistencies among agents [11]. Fernando et al presented a framework that blends a top-down hierarchical structure with decentralized control, enabling efficient decision-making in multi-agent systems. But their approach mainly concentrates on structural organization and negotiation scalability; however, it lacks mechanisms to detect conflicting performatives and manage inconsistent commitments between agents [12]. Esmaeili, Mozayani, Motlagh, and Matson suggested self-organizing methods for holonic multi-agent systems that build a holonic structure using a bottom-up approach and common social notions. Although their work evaluates performance in a task allocation environment, the work mainly focuses on organizational formation and self-organization, not on the correctness of interaction semantics between agents [13].

The study of Khazaei and Mozayani argues that holonic organization can minimize multi-agent system complexity by simulating huge SIP networks. However, the approach primarily targets resource management and load balancing in telecommunications infrastructure and lacks mechanisms to detect or resolve inconsistent performatives or interaction semantics [14]. Pahwa et al proposed a goal-based HMAS for power distribution systems, which organizes agents around operational goals. The work primarily addresses power system control and optimization, and not the correctness of the interaction semantics between agents [15]. The study of Zhang et al. investigated negotiation protocols and their role in resolving conflicts and reaching consensus among autonomous agents. However, the model mainly focuses on protocol optimization and performance metrics and does not address contradictory proposals, commitments, or cancellations [16].

Roshanzamir, et al. have shed light on the application of swarm intelligence approaches for task allocation, highlighting its adaptability to deal with complicated dynamic task contexts, but the approach assumes that agent interactions and message interpretations remain consistent [17]. Wang et al. analyze a cooperative control technique that includes intermittent communication and leadership competition to improve job scheduling and stability in large-scale applications. However, the approach mainly treats communication as a connectivity and control stability issue, rather than addressing the content and correctness of the exchanged messages [18]. Poudel and Moh explained that central task assignment can streamline coordination but may be less adaptable to dynamic environments and can be a potential point of failure. However, its focus is mainly on algorithmic performance, network constraints, and mission efficiency [6]. The work of

Akbari and Unland focuses on designing a holonic multi-agent front-end for medical diagnostic support, but it does not address task coordination issues caused by communication semantics [19]. Missaoui et al. proposed a new norm model for HMAS, introducing internal and external norms to manage both internal organization and external interactions, thus addressing challenges in social control and interoperability, but did not analyze the semantic correctness of individual communication acts [20]. The study of Abdoos et al. describes a specialized algorithm developed particularly for HMAS to solve the problem of hierarchical communication amongst holons by specifying abstract data flows for state estimation, action selection, and reward calculation. However, coordination issues that arise from communication inconsistencies are not considered [21].

Beheshti et al. examine negotiation in holonic multi-agent systems (HMAS), emphasizing their distinct combination of competitive and cooperative characteristics where agents balance their personal utility with that of their holon while accommodating potential. However, the impact of performative mismatches on task assignment reliability is not examined [22]. Abdoos highlights the use of Holonic HMASs for traffic signal control, emphasizing how holonic structures may ensure consistent performance by proper coordination, but the work mainly evaluates system performance and stability, assuming that communication between agents is correct and consistent [23]. The work of Mazdin and Rinner, Qasim et al., Amrani, et al. focuses on coalition formation and communication-aware task allocation and handling temporal aspects of standardized FIPA messages in agent-based systems. However, they do not discuss task assignment optimization in holonic MAS structures or analyze whether exchanged performatives lead to conflicting commitments or inconsistent decisions [24–26].

Existing optimization approaches in HMAS focus primarily on improving task allocation efficiency, reducing execution cost, and balancing agent workload. Linear programming, auction-based coordination, and heuristic optimization methods have demonstrated effectiveness in resource management and distributed scheduling, but these approaches generally assume consistent inter-agent communication and do not take into account the performative inconsistencies arising from FIPA-ACL interactions. As a result, communication conflicts may propagate through the holonic hierarchy, degrading system reliability and coordination efficiency. The proposed approach addresses this research gap by integrating performative inconsistency resolution with optimal task assignment in a dynamically adaptive HMAS framework. Table 1 summarizes the limitations of existing work and how the proposed work overcomes these limitations.

Table 1: Limitations of existing research.

References	Model/Framework Used	Common Limitation w.r.t. Proposed Work
(Inal et al., 2023) [11], (Roshanzamir et al., 2017) [17], (Abdoos et al., 2015) [21], (Poudel and Moh, 2022) [6]	Reinforcement Learning/Optimization Model for Holonic Multi-agent Systems.	All of these works focus on optimization convergence rather than communication correctness or coordination conflicts that might lead to inefficient working of agents.

(Continued)

Table 1 (continued)

References	Model/Framework Used	Common Limitation w.r.t. Proposed Work
(Fernando et al., 2019) [12], (Pahwa et al., 2015) [15], (Beheshti et al., 2016) [22], (Abdoos, 2018) [23]	Agent Negotiations in Holonic structures/Socially-Inspired Self-Organizing Holonic Model	All of these works focus on the organization/negotiation of agents in a holonic organization but ignore the issues that might arise because of performative conflicts.
(Khazaei and Mozayani, 2017) [14], (Zhang et al., 2019) [16], (Wang et al., 2020) [18], (Mazdin and Rinner, 2021) [24]	Load Balancing/Cooperative Control Model/Coalition Formation in Holonic structures	All of these works focus on improving efficiency by optimal task assignment in Holonic agent-based systems, but there is no focus on the,correct interpretation of messages.
(Akbari and Unland, 2022) [19], (Missaoui et al., 2017) [20], (Qasim et al, 2020) [25], (Amrani et al., 2020) [26]	Norm Implementation/Temporal Constraints/Interoperability in Holons	All of these works focus on system governance or interoperability rather than resolving interaction inconsistencies in Holonic systems.

3 Proposed Optimal Task Assignment in Holonic Agents Framework

Fig. 2 shows the working of our proposed Optimal Task Assignment in Holonic Agents. In our system, a holon can be any autonomous unit that is both a whole in itself and a part of a larger system. It can operate independently, but it is also part of a larger hierarchy of holons. The concept of holons is integrated into the framework of MAS, where each holon will act as an agent with its own autonomy, goals, and responsibilities. These holonic agents will then cooperate, coordinate, and negotiate with each other to achieve complex goals. There are multiple levels, and on each level, there can be multiple holons, and each holon may contain multiple agents. It should be noted that when a task is assigned to holons optimally, then new holons might be created dynamically. Additionally, a single agent can be part of multiple holons, as shown by the dotted oval on level 1. These agents are already part of a holon, but due to optimal task assignment, when a new holon is created, they are also part of it. This type of structure is well-suited for the modelling of complex systems such as manufacturing, transportation, healthcare, and smart grids. The holons can reorganize themselves, create new holons, or dissolve existing ones as needed to respond to dynamic requirements. The complete system can grow or shrink in complexity without losing its overall functionality.

3.1 Performative Inconsistencies in HMAS

Performative inconsistencies in HMAS refer to situations where agents or holons exhibit unexpected or contradictory behaviors during task execution or communication, leading to sub-optimal performance. These inconsistencies can arise due to the following:

- Conflicting roles: When different agents or holons interpret or perform their assigned tasks in ways that conflict with the overall system goals.
- Coordination failures: Inadequate synchronization between agents, causing overlapping or redundant actions.
- Resource constraints: When agents exceed or under-utilize resources, affecting performance.
- Misaligned objectives: Agents may follow different local objectives that deviate from the global system goals.

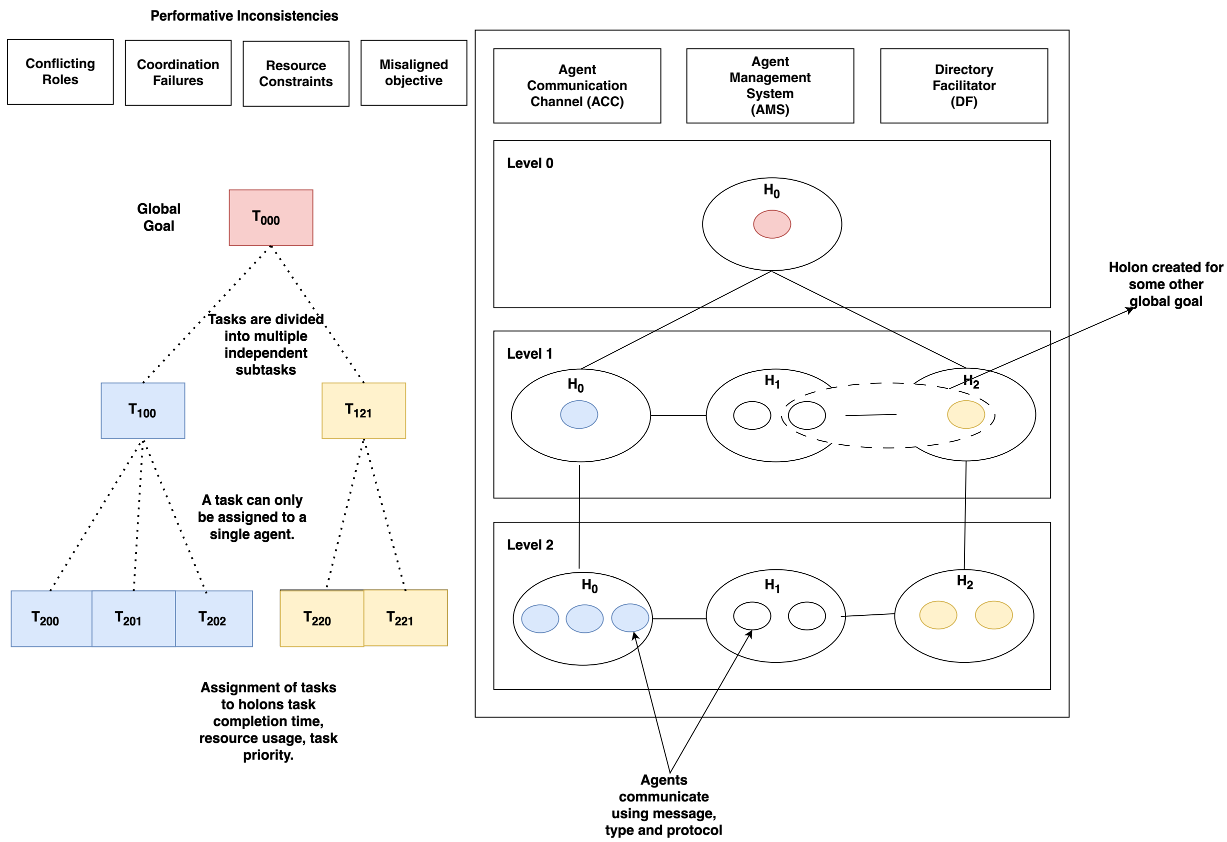


Figure 2: Proposed optimal task assignment in holonic agents framework.

The designer of the system can include any or all of these as optimization constraints depending on the requirements and complexity of the system.

3.2 Formal Representation of Holonic Multi-Agent Systems

HMAS are formally specified for a precise specification of the structuring of agents in holons. The structure and behavior of holons, as well as the relationships between them, are also defined.

3.2.1 Definition of Holon

A holon H can be formally defined as a tuple as shown in Eq. (1).

$$H = \langle ID, A, C, R, M, P \rangle \tag{1}$$

- ID: Unique identifier for the holon.

- A: Set of agents $\{a_1, a_2, ..., a_n\}$ that constitute the holon.
- C: Abilities or skills that the holon possesses.
- R: Set of roles that the holon can play within the system.
- M: Set of methods or plans that the holon can execute to achieve goals.
- P: Set of policies or rules that govern the behavior of the holon.

These holons are organized in a hierarchical structure, where a holon can be part of a higher-level holon, forming a recursive structure. The hierarchical relationship is defined in Eq. (2).

$$H_i \in H_j \implies H_i \; is \; a \; subholon \; of \; H_j \tag{2}$$

The hierarchy of holons can be represented as a tree or graph where each node represents a holon, and each edge represents a "part-of" relationship. The three sets required for optimal task allocation modeling are defined below.

- Let $H = \{H_0, H_1, \ldots, H_m\}$ represent the set of holons in the system.
- Let $T = \{T_0, T_1, \ldots, T_n\}$ represent the set of tasks to be assigned.
- Let $L = \{L_0, L_1, \ldots, L_k\}$ represent the hierarchical levels, where L_1 is the highest level.

3.2.2 Communication

Holons interact with each other through communication and cooperation mechanisms. The communication between two holons H_i and H_j can be modeled as

$$Comm(H_i, H_j) = \langle Msg, Type, Protocol \rangle$$

where:

- Msg: Message content.
- Type: Type of interaction (e.g., request, inform, negotiate or any FIPA performative).
- Protocol: Communication protocol or pattern used (e.g., Contract Net Protocol).

For standardization, our framework only allows messages to be communicated among agents/holons in FIPA performatives.

3.3 Optimal Task Assignment in Holons

A task represents a discrete unit of work, and the objective function represents the global goal that needs to be achieved. It can be minimizing total time, cost, energy consumption, or maximizing performance metrics. Constraints represent different limitations of the systems that should be kept in mind while optimally assigning tasks to holons. In the proposed framework, performative inconsistencies are modelled as constraints. Optimal task assignment thus refers to the process of allocating tasks to agents in a way that optimizes a certain objective depending on the scenario. The framework also incorporates the event-triggered approach so instead of recomputing agent assignments continuously, the system will dynamically assign tasks only when important events occur.

3.3.1 Objective Functions

- Task Completion Time: $f_1(H_i, T_j)$ is the time taken by holon H_i to complete task T_j.
- Resource Usage: $f_2(H_i, T_j)$ is the resource consumption (e.g., energy, cost) of holon H_i when performing task T_j.
- Task Priority: $p(T_j)$ represents the priority of task T_j.

- Event Triggered Approach: Task reassignment occurs only when the inconsistency exceeds the threshold.

  $$E(t) = 1 \quad \text{if} \quad \phi(t) > \delta$$

  where $E(t)$ = event indicator, δ = threshold, $\phi(t)$ = system inconsistency metric

3.3.2 Decision Variables

Let x_{ij} be a binary variable, where:

$$x_{ij} = \begin{cases} 1 & \text{if task } T_j \text{ is assigned to holon } H_i, \\ 0 & \text{otherwise.} \end{cases}$$

3.3.3 Constraints

The first two constraints are related to the performative inconsistencies, and the other two are of general holon constraints.

- **Conflicting Roles:** No two conflicting tasks should be assigned to the same agent:

  $$x_{i,k} + x_{j,k} \leq 1 \quad \forall i, j \in \text{ConflictingTasks}, \forall k$$

- **Coordination Failures:** Ensure that dependent tasks are executed in sequence:

  $$t_i + d_i \leq t_j \quad \text{where} \quad d_i \quad \text{is the duration of task} \quad T_i$$

- **Task Assignment Constraint:** Each task must be assigned to exactly one holon:

  $$\sum_{i=1}^{m} x_{ij} = 1 \quad \forall j \in \{1, 2, \ldots, n\}$$

- **Capacity Constraint:** Each holon H_i can handle a limited number of tasks, depending on its level in the hierarchy:

  $$\sum_{j=1}^{n} x_{ij} \leq C_i \quad \forall i \in \{1, 2, \ldots, m\}$$

  where C_i is the capacity of holon H_i.

- **Fault Tolerance Constraint:** To improve system resilience, a fault-tolerance constraint is incorporated into the task assignment process. The proposed mechanism ensures that all the critical tasks have at least one backup agent.

  $$\sum_{i} b_{ij} \geq 1 \quad \forall j \in T_c$$

  where b_{ij} indicates that agent i is assigned as a backup agent for task j, and T_c denotes the set of critical tasks requiring fault tolerance.

- **Data Dimension Constraint:** The data dimension constraint shows that the proposed framework also considers scalability and heterogeneity of task data, making it generalizable to all domains. The constraint is defined as the total data workload assigned to an agent, which must not exceed its processing or communication capacity.

$$\sum_j D_j x_{ij} \leq C_i \quad \forall i$$

where D_j represents the dimension or size of data associated with task j, x_{ij} denotes the assignment of task j to agent i, and C_i is the maximum data processing capacity of agent i.

The designer of the system can implement all or some of the constraints.

3.3.4 Optimization Problem

The goal is to minimize the total cost function, which is a weighted sum of the objectives:

$$\text{Minimize } Z = \sum_{i=1}^{m} \sum_{j=1}^{n} \left(w_1 \cdot f_1(H_i, T_j) + w_2 \cdot f_2(H_i, T_j) - w_3 \cdot p(T_j) \right) \cdot E(t) \cdot x_{ij}$$

where:

- n is the number of tasks
- m is the number of holons
- w_1, w_2, w_3 are weights reflecting the importance of time, resource usage, and task priority, respectively.
- $f_1(H_i, T_j)$ and $f_2(H_i, T_j)$ are costs (or negative factors) that need to be minimized. Both of these values should be as low as possible for optimal performance of holons. Hence, their weighted sums $w_1 \cdot f_1$ and $w_2 \cdot f_2$ are added, contributing to the total cost that should be minimum.
- $p(T_j)$ (Task Priority) is treated as a positive factor because higher priority tasks should be favored. The higher the priority, the more beneficial it is to assign and complete the task. So, when $w_3 \cdot p(T_j)$ is subtracted, the value of Z is effectively increased for lower priority tasks and decreased for higher priority tasks. This drives the optimization to prefer high-priority tasks. By subtracting $w_3 \cdot p(T_j)$, the objective function is adjusted to favor tasks with higher priorities while minimizing time and resource consumption.

3.4 Dynamic Adjustment of Hierarchies

In our system, tasks will be initially distributed based on a pre-defined policy. It can be that the tasks are allotted to the nearest available holon or the most capable holon. However, as tasks progress, the hierarchy of holons can dynamically adjust. Holons can request help from higher-level holons or delegate tasks to lower-level holons. The system continuously updates and optimizes task assignment based on the current state of the system. After an initial task allocation, the hierarchy is adjusted based on performance metrics $P(H_i)$, where:

$$P(H_i) = \frac{\sum_{j=1}^{n} \text{Success}(H_i, T_j) \cdot x_{ij}}{\sum_{j=1}^{n} x_{ij}}$$

here, $\text{Success}(H_i, T_j)$ is a measure of how well holon H_i performed task T_j. If $P(H_i)$ exceeds a certain threshold, H_i may be promoted to a higher level in the hierarchy.

3.5 Computational Complexity and Scalability

Let n denote the number of tasks and m denote the number of holonic agents participating in the assignment process. The task allocation decision variables are represented by x_{ij}, where each variable corresponds to the assignment of task j to holon i. Consequently, the optimization problem contains approximately $O(mn)$ decision variables and assignment constraints. The computational complexity depends primarily on the selected solver. The worst-case computational complexity for interior-point LP solvers is generally polynomial and can be approximated as $O((mn)^3)$. However, due to the hierarchical decomposition of the proposed HMAS structure, the optimization problem is partitioned among tactical and operational holons. This will significantly reduce the effective search space and improve scalability compared to centralized optimization approaches. Additionally, the event-triggered optimization mechanism reduces unnecessary recomputation overhead by activating re-optimization only when an event of interest happens. Therefore, the practical runtime complexity is lower than continuous global optimization approaches. As the number of tasks and agents increases, local optimization subproblems are solved independently within holonic clusters, thereby reducing communication overhead and computational bottlenecks.

4 Application of the Proposed OTAHA Framework

In this section, the application of the proposed OTAHA framework is demonstrated using an example of a Smart Transportation System for Urban Traffic Management. As shown in Fig. 3, a city has deployed a smart transportation system in which various agents, including autonomous vehicles, traffic signals, and traffic monitoring drones, work together to manage traffic efficiently. Each of the entities' vehicles, traffic lights, and drones is modeled as a holon, and collectively they form a multi-agent system to achieve traffic management goals. In this scenario, tasks such as dynamic rerouting, road condition monitoring, incident reporting, and emergency vehicle prioritization are complex and require efficient task allocation. The following three types of holons are identified.

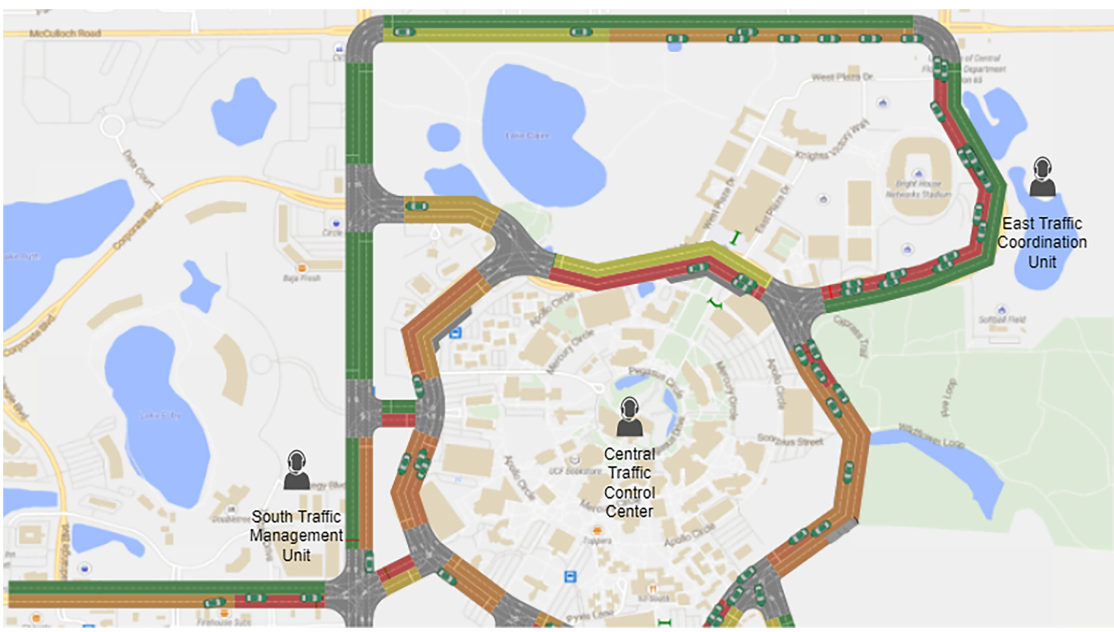


Figure 3: Holons working for urban traffic management.

Top-Level Holon (Traffic Control Center): This holon acts as the highest decision-making entity, coordinating large-scale traffic tasks like optimizing traffic flow across the entire city.

Mid-Level Holons (District Traffic Managers): These holons will manage smaller zones within the city, dealing with local traffic issues like optimizing traffic light cycles or rerouting vehicles around accidents.

Low-Level Holons (Autonomous Vehicles, Drones, and Traffic Lights): These holons will perform tasks such as following optimized routes, reporting real-time road conditions, and adjusting traffic lights based on local congestion.

4.1 Scenario: Rerouting of Traffic Due to Accident

An accident occurs on a major road, and the control center must reroute traffic. The top-level holon identifies the affected areas and assigns district managers (mid-level holons) to prioritize rerouting vehicles away from the congested area. Each district manager dynamically allocates tasks to local vehicles (low-level holons), instructing them to follow alternate routes based on real-time traffic data.

- **Holons:**
  - H_1: Central Traffic Control Center (Capacity 2 tasks)
  - H_2: South Traffic Management Unit (Capacity 1 task)
  - H_3: East Traffic Coordination Hub (Capacity 1 task)
- **Tasks:**
  - T_1: Monitor Traffic (Priority 3, Medium)
  - T_2: Control Traffic Lights (Priority 2, Low)
  - T_3: Reroute Vehicles (Priority 5, High)
  - T_4: Emergency Vehicle Coordination (Priority 7, Very High)
- $H = \{H_1, H_2, H_3\}$ be the set of holons present at different hierarchies of the system.
- $T = \{T_1, T_2, T_3, T_4\}$ be the set of tasks that need to be optimally assigned to holons.

Now, the decision variables for the case study are defined. Let x_{ij} be a binary variable where:

$$x_{ij} = \begin{cases} 1 & \text{if task } T_j \text{ is assigned to holon } H_i, \\ 0 & \text{otherwise.} \end{cases}$$

For this problem, 3 holons and 4 tasks are identified, resulting in the following decision variables:

$x_{11}, x_{12}, x_{13}, x_{14}, x_{21}, x_{22}, x_{23}, x_{24}, x_{31}, x_{32}, x_{33}, x_{34}$

4.2 Objective Function

The objective function is designed to minimize the total cost.

$$\text{Minimize } Z = \sum_{i=1}^{3} \sum_{j=1}^{4} \left(w_1 \cdot f_1(H_i, T_j) + w_2 \cdot f_2(H_i, T_j) - w_3 \cdot p(T_j) \right) \cdot x_{ij}$$

where:

- $f_1(H_i, T_j)$ is the time taken by the holon H_i to complete the task T_j.
- $f_2(H_i, T_j)$ is the resource consumption of holon H_i for task T_j.
- $p(T_j)$ is the priority of task T_j.
- w_1, w_2, w_3 are weights that reflect the importance of time, resource usage, and task priority, respectively.

The time taken by different holons to perform the four tasks is defined below. The unit time is in minutes.

$$f_1(H_1, T_1) = 10, \quad f_1(H_1, T_2) = 8, \quad f_1(H_1, T_3) = 15, \quad f_1(H_1, T_4) = 12$$
$$f_1(H_2, T_1) = 12, \quad f_1(H_2, T_2) = 9, \quad f_1(H_2, T_3) = 17, \quad f_1(H_2, T_4) = 11$$
$$f_1(H_3, T_1) = 11, \quad f_1(H_3, T_2) = 10, \quad f_1(H_3, T_3) = 14, \quad f_1(H_3, T_4) = 13$$

The resource usage of different holons to perform the four tasks in energy units is given below.

$$f_2(H_1, T_1) = 6, \quad f_2(H_1, T_2) = 5, \quad f_2(H_1, T_3) = 7, \quad f_2(H_1, T_4) = 8$$
$$f_2(H_2, T_1) = 5, \quad f_2(H_2, T_2) = 4, \quad f_2(H_2, T_3) = 6, \quad f_2(H_2, T_4) = 7$$
$$f_2(H_3, T_1) = 6, \quad f_2(H_3, T_2) = 5, \quad f_2(H_3, T_3) = 7, \quad f_2(H_3, T_4) = 8$$

The following are the priorities of the four tasks in the system.

$$p(T_1) = 3, \quad p(T_2) = 2, \quad p(T_3) = 5, \quad p(T_4) = 7$$

The weights of different functions are specified below. These weights can be adjusted to increase the importance of a specific task in the system. Currently, more weight is assigned to the time taken by a holon to perform a task.

$$w_1 = 0.4, \quad w_2 = 0.3, \quad w_3 = 0.3$$

4.3 Constraints

Now, the constraints are specified for the smart traffic management system according to the OTAHA framework.

1. Task Assignment Constraint: Each task must be assigned to exactly one of the three holons. For each task j:

   $$x_{1j} + x_{2j} + x_{3j} = 1 \quad \forall j \in \{1, 2, 3, 4\}$$

   This ensures that every task is assigned to one and only one holon.
2. Capacity Constraint: Each holon H_i can handle a limited number of tasks, depending on its capacity. For holon H_1 (capacity 2 tasks), holon H_2 and H_3 (capacity 1 task each):

   $$x_{11} + x_{12} + x_{13} + x_{14} \leq 2$$
   $$x_{21} + x_{22} + x_{23} + x_{24} \leq 1$$
   $$x_{31} + x_{32} + x_{33} + x_{34} \leq 1$$

4.4 Solving Linear Optimization Problem under Normal Traffic Conditions

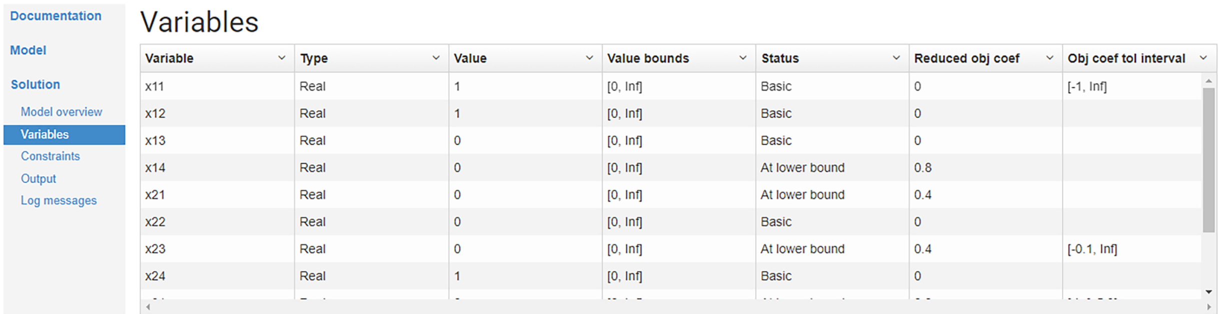
For the normal traffic conditions, the objective function becomes:

$$\begin{aligned} Z = {} & 0.4 \times (10x_{11} + 8x_{12} + 15x_{13} + 12x_{14} + 12x_{21} + 9x_{22} + 17x_{23} + 11x_{24} + 11x_{31} + 10x_{32} + 14x_{33} + 13x_{34}) \\ & + 0.3 \times (6x_{11} + 5x_{12} + 7x_{13} + 8x_{14} + 5x_{21} + 4x_{22} + 6x_{23} + 7x_{24} + 6x_{31} + 5x_{32} + 7x_{33} + 8x_{34}) \\ & - 0.3 \times (3x_{11} + 2x_{12} + 5x_{13} + 7x_{14} + 3x_{21} + 2x_{22} + 5x_{23} + 7x_{24} + 3x_{31} + 2x_{32} + 5x_{33} + 7x_{34}) \end{aligned}$$

After defining the objective function and the constraints, the optimization problem can be solved using any LP (Linear Programming) solver. In this case study, https://online-optimizer.appspot.com/ LP solver was used for normal traffic conditions that returned the assignments shown in Fig. 4.

The assignments result in the following values:

- Holon H_1 handles tasks T_1 and T_2, taking 10 min to monitor traffic and 8 min to control traffic lights.
- Holon H_2 handles the task T_4, taking 11 min to coordinate emergency vehicles.
- Holon H_3 handles the task T_3, taking 14 min to reroute vehicles.



Variable	Type	Value	Value bounds	Status	Reduced obj coef	Obj coef tol interval
x11	Real	1	[0, Inf]	Basic	0	[-1, Inf]
x12	Real	1	[0, Inf]	Basic	0	
x13	Real	0	[0, Inf]	Basic	0	
x14	Real	0	[0, Inf]	At lower bound	0.8	
x21	Real	0	[0, Inf]	At lower bound	0.4	
x22	Real	0	[0, Inf]	Basic	0	
x23	Real	0	[0, Inf]	At lower bound	0.4	[-0.1, Inf]
x24	Real	1	[0, Inf]	Basic	0	

Figure 4: Result of LP solver.

This linear programming problem can also be solved using the Python PuLP library. Since it is exhausting to add linear programming data in a graphical environment, the use of the PuLP library makes it easier to input complex problems. Figs. 5 and 6 show the output of the PuLP-based program that matches the results of the graphical LP solver.

```
Status     : Optimization terminated successfully. (Optimal)
Minimum Z  : 19.6
======================================================
Assignment Matrix (1 = assigned):
      Task 1  Task 2  Task 3  Task 4
----------------------------------------
Holon 1 |   1     1     0     0
Holon 2 |   0     0     0     1
Holon 3 |   0     0     1     0
======================================================
Detailed Assignments:
======================================================
Holon 1 → Task 1
 Time=10, Resource=6, Quality=3
 Cost contribution = 0.4×10 + 0.3×6 - 0.3×3 = 4.9000
Holon 1 → Task 2
 Time=8, Resource=5, Quality=2
 Cost contribution = 0.4×8 + 0.3×5 - 0.3×2 = 4.1000
Holon 2 → Task 4
 Time=11, Resource=7, Quality=7
 Cost contribution = 0.4×11 + 0.3×7 - 0.3×7 = 4.4000
Holon 3 → Task 3
 Time=14, Resource=7, Quality=5
 Cost contribution = 0.4×14 + 0.3×7 - 0.3×5 = 6.2000
Total Z = 19.6
======================================================
```

Figure 5: Result of solving the LP problem using PuLP.

Summary of the optimal solution:

Assignment	Time	Resource	Quality	Contribution
Holon 1 → Task 1	10	6	3	4.90
Holon 1 → Task 2	8	5	2	4.10
Holon 2 → Task 4	11	7	7	4.40
Holon 3 → Task 3	14	7	5	6.20
Total Z				**19.60**

Figure 6: Summary of tasks assignment using PuLP.

4.5 Solving Linear Optimization Problem under Emergency Traffic Conditions

For emergency traffic conditions, the weights are adjusted to reflect the urgency of the tasks. In the following, the adjusted values for the emergency scenario are set.

- Higher weight in time $w_1 = 0.6$ (since time is more critical).
- Lower weight on resource usage $w_2 = 0.2$ (less emphasis on resource consumption).
- Lower weight on task priority $w_3 = 0.2$ (higher priority for certain tasks).

For the emergency traffic condition, the adjusted objective function is:

$$Z_{\text{emergency}} = \sum_{i=1}^{3}\sum_{j=1}^{4}\left(0.6 \cdot f_1(H_i, T_j) + 0.2 \cdot f_2(H_i, T_j) - 0.2 \cdot p(T_j)\right) \cdot x_{ij}$$

Substituting the real values for $f_1(H_i, T_j)$, $f_2(H_i, T_j)$, and $p(T_j)$:

$$\begin{aligned} Z_{\text{emergency}} = {} & 0.6 \times (10x_{11} + 8x_{12} + 15x_{13} + 12x_{14} + 12x_{21} + 9x_{22} + 17x_{23} + 11x_{24} + 11x_{31} + 10x_{32} + 14x_{33} + 13x_{34}) \\ & + 0.2 \times (6x_{11} + 5x_{12} + 7x_{13} + 8x_{14} + 5x_{21} + 4x_{22} + 6x_{23} + 7x_{24} + 6x_{31} + 5x_{32} + 7x_{33} + 8x_{34}) \\ & - 0.2 \times (3x_{11} + 2x_{12} + 5x_{13} + 7x_{14} + 3x_{21} + 2x_{22} + 5x_{23} + 7x_{24} + 3x_{31} + 2x_{32} + 5x_{33} + 7x_{34}) \end{aligned}$$

After solving the problem using the LP Solver for emergency traffic conditions, the solver will return the following assignment.

In this emergency scenario:

- Holon H_1 handles tasks T_1 and T_4, prioritizing emergency vehicle coordination, which takes 10 min to monitor traffic and 12 min for emergency Vehicle coordination.
- Holon H_3 handles the task T_2, taking 5 min to control the traffic lights.
- Holon H_2 handles the task T_3, taking 17 min to reroute vehicles.

The total cost function Z for this scenario is adjusted according to the high priority of the task T_4, ensuring that emergency services are coordinated quickly.

5 Results and Discussion

Fig. 7 shows a graph comparing the two traffic scenarios (normal and emergency) according to the objective function. The Python library *matplotlib* was used for visualization.

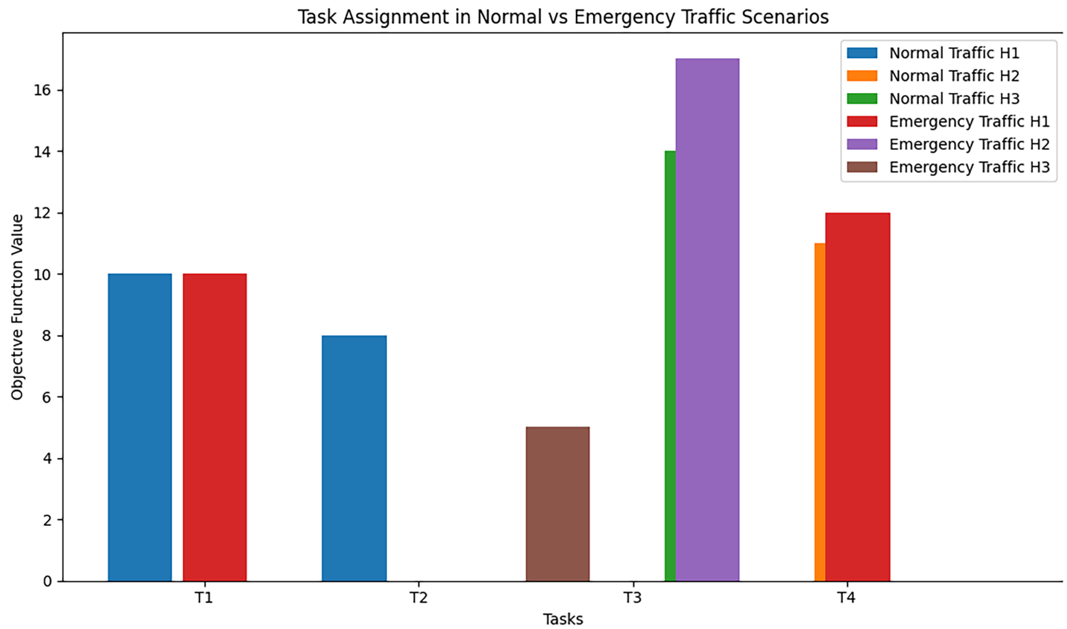


Figure 7: Tasks completion time for different holons.

Each bar represents the objective function value (cost value) of a task assigned to a holon in minutes. The graph allows us to visually compare the impact of different traffic conditions on task assignment. Following this step-by-step approach, the optimization problem for Dynamic Hierarchical Task Allocation (DHTA) in holons was solved for two different scenarios (normal traffic and emergency conditions). The LP solver provided the optimal task assignment to minimize the overall cost function subject to various constraints. Hierarchical Task Allocation is essential to efficiently distribute tasks across multiple agents at different levels of control. Dynamic Task Allocation ensures that tasks are reallocated in response to changing conditions, such as accidents in the transportation scenario or equipment malfunctions in the warehouse scenario. Moreover, our proposed approach allows for real-time adjustment and optimization of tasks, leading to increased efficiency, scalability, and resilience in both complex and dynamic environments.

Table 2 provides a comparison of the proposed approach with the previous works, including heuristic approaches (task assignment based on rules or approximations like greedy algorithms) and game-theoretic models (multi-agent systems where holons are viewed as players, and task allocation is determined through competition or cooperation).

Table 2: Methodology-based comparison between OTAHA and previous approaches.

Criteria	OTAHA Framework	Heuristic Approaches	Game-Theoretic Approaches
Methodology	Optimization-based task allocation using linear programming	Greedy task allocation based on priority	Game theory model of holons as players
Objective	Minimize cost, time, and resource usage	Maximize task throughput	Maximize utility in competitive settings

(Continued)

Table 2 (continued)

Criteria	OTAHA Framework	Heuristic Approaches	Game-Theoretic Approaches
Decision Variables	Binary decision variables for task assignment	Static assignment based on task urgency	Dynamic task allocation through bidding
Complexity	Medium (due to optimization)	Low (simple greedy algorithm)	High (due to competition/collaboration)
Adaptability	High (through optimization and dynamic reassignment)	Medium	Medium-High

A metrics-based comparison of the proposed framework with the performance of the previous approaches is provided in Table 3. It can be seen from the table that the OTAHA framework has adaptability compared to previously used approaches. This stems from the fact that the framework has the ability to dynamically change the hierarchy and levels of the holons as per need. Additionally, the proposed solution has high scalability and the lowest resource consumption as tasks are always assigned optimally.

Table 3: Performance metrics comparison between OTAHA and previous approaches.

Performan ce Metrics	OTAHA Framework	Heuristic Approaches	Game-Theoretic Approaches
Task Completion Time	Lower due to optimal allocation	Moderate (depends on task ordering)	Moderate (depends on cooperation)
Resource Consumption	Lower due to resource-aware allocation	Higher	Moderate (depends on bidding)
Scalability	High (due to dynamic allocation)	Low	Moderate
Resilience	High (accounts for failures)	Low (no dynamic handling)	High (due to cooperation mechanisms)
Task Priority Handling	Weighted in optimization	Static assignment	Dynamic (based on utility function)

6 Conclusion and Future Work

In this research, a framework for optimal task assignment in Holonic Multi-Agent Systems is proposed that addresses performative inconsistencies. These inconsistencies can arise due to conflicting roles, coordination failures, and capacity limitations. Traditional approaches of optimal task allocation in holons often fail to adapt to these dynamic conditions, resulting in sub-optimal task allocation. After formally modelling the holons, agents, objective function, and constraints, it was shown how the problem can be modeled as a linear optimization problem, effectively assigning tasks across holons while ensuring system coherence and efficiency. The application of the proposed framework was demonstrated using a real-world case study of a smart transportation system. Multiple levels of holons were defined, each having a separate responsibility, and then the objective function was formulated based on task completion time, resource usage, and task priority. The problem was then solved using a real-world LP solver, and the optimally assigned tasks were visually presented. Through these evaluations, it was shown that the proposed approach has better adaptability and low resource consumption compared to previous work.

The limitation of the proposed work is that while designing the objective function for the holonic agents, it is assumed that the goals of all agents are aligned. In the case where agents belonging to different holons have completely opposite goals, this will increase the complexity. Hence, in the future, the work can be extended by incorporating multi-objective optimization. This will allow the optimization model to include additional objectives like energy consumption, cost, or environmental impact, reflecting a broader range of system goals. Moreover, machine learning techniques can also be integrated to predict task complexities, resource requirements, or agent behavior, which will enhance the efficiency of task allocation in real-time.

Acknowledgement: Not applicable.

Funding Statement: The authors received no specific funding for this study.

Author Contributions: The authors confirm contribution to the paper as follows: Conceptualization, Awais Qasim and Aniqa Iftikhar; methodology, Awais Qasim, Aniqa Iftikhar and Hanaa Nafea; software, Awais Qasim, Aniqa Iftikhar and Nay Chi Moe Oo; validation, Awais Qasim, Aniqa Iftikhar, Hanaa Nafea and Nay Chi Moe Oo; formal analysis, Awais Qasim and Byung-Seo Kim; investigation, Awais Qasim, Aniqa Iftikhar and Byung-Seo Kim; resources, Byung-Seo Kim; writing-original draft preparation, Awais Qasim, Aniqa Iftikhar and Nay Chi Moe Oo; writing—review and editing, Awais Qasim and Byung-Seo Kim; visualization, Awais Qasim and Aniqa Iftikhar; supervision, Byung-Seo Kim; project administration, Awais Qasim and Byung-Seo Kim; funding acquisition, Byung-Seo Kim. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: Not applicable.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Ansari J, Gholami A, Kazemi A. Multi-agent systems for reactive power control in smart grids. Int J Electr Power Energy Syst. 2016;83:411–25. doi:10.1016/j.ijepes.2016.04.010.
2. Hassan T, Hussain I, Haque HMU, Mirza HT, Ali MN, Byung-Seo K. Semantic knowledge based reinforcement learning formalism for smart learning environments. Comput Mater Contin. 2025;85(1):2071.
3. Cintuglu MH, Martin H, Mohammed OA. An intelligent multi agent framework for active distribution networks based on IEC 61850 and FIPA standards. In: Proceedings of the 2015 18th International Conference on Intelligent System Application to Power Systems (ISAP); 2015 Sep 11–16; Porto, Portugal. p. 1–6.
4. Hassan T, Hussain I, Haque HMU, Mirza HT, Ali MN, Kim BS, et al. Multi-agent reinforcement learning based context-aware heterogeneous decision support system. Comput Mater Contin. 2026;87(3):1–10.
5. Malik FH, Lehtonen M. A review: agents in smart grids. Electr Power Syst Res. 2016;131:71–9. doi:10.1016/j.epsr.2015.10.004.
6. Poudel S, Moh S. Task assignment algorithms for unmanned aerial vehicle networks: a comprehensive survey. Veh Commun. 2022;35:100469. doi:10.1016/j.vehcom.2022.100469.
7. Ali MN, Imran M, Din MSU, Kim BS. IFCNS: intelligent forwarding and computation node selection in mobile edge computing. IEEE Sens J. 2025. doi:10.1109/JSEN.2025.3564414.

8. FIPA Standard SC00061G. FIPA ACL message structure specification. Geneva, Switzerland: Foundation for Intelligent Physical Agents; 2002. Archived specification.
9. Bellifemine F, Caire G, Poggi A, Rimassa G. JADE: a software framework for developing multi-agent applications. Lessons learned. Inf Softw Technol. 2008;50(1–2):10–21.
10. Wooldridge M, Jennings NR. Intelligent agents: theory and practice. Knowl Eng Rev. 1995;10(2):115–52.
11. İnal AF, Sel Ç, Aktepe A, Türker AK, Ersöz S. A multi-agent reinforcement learning approach to the dynamic job shop scheduling problem. Sustainability. 2023;15(10):8262. doi:10.3390/su15108262.
12. Fernando S, Premasiri D, Dias V, Kohomban U, Welikala Y, Subasinghe H. Decentralizing decision making process of simultaneous bilateral negotiations of large multiagent systems in e-commerce applications using holonic structures. In: Proceedings of the PAAMS Collection: International Workshops of PAAMS 2019; 2019 Jun 26–28; Ávila, Spain. p. 137–51.
13. Esmaeili A, Mozayani N, Motlagh MRJ, Matson ET. A socially-based distributed self-organizing algorithm for holonic multi-agent systems: case study in a task environment. Cogn Syst Res. 2017;43:21–44.
14. Khazaei M, Mozayani N. Overload management with regard to fairness in session initiation protocol networks by holonic multiagent systems. Int J Netw Manag. 2017;27(3):e1969. doi:10.1002/nem.1969.
15. Pahwa A, DeLoach SA, Natarajan B, Das S, Malekpour AR, Alam SS, et al. Goal-based holonic multiagent system for operation of power distribution systems. IEEE Trans Smart Grid. 2015;6(5):2510–8. doi:10.1109/TSG.2015.2404334.
16. Zhang J, Wang G, Song Y. Task assignment of the improved contract net protocol under a multi-agent system. Algorithms. 2019;12(4):70. doi:10.3390/a12040070.
17. Roshanzamir M, Balafar MA, Razavi SN. Empowering particle swarm optimization algorithm using multi agents' capability: a holonic approach. Knowl-Based Syst. 2017;136:58–74. doi:10.1016/j.knosys.2017.08.023.
18. Wang B, Chen W, Zhang B, Zhao Y, Shi P. Cooperative control-based task assignments for multiagent systems with intermittent communication. IEEE Trans Ind Inform. 2020;17(10):6697–708. doi:10.1109/TII.2020.3044950.
19. Akbari Z, Unland R. A powerful holonic and multi-agent-based front-end for medical diagnostics systems. In: Handbook of artificial intelligence in healthcare: Vol 1—Advances and applications. Berlin/Heidelberg, Germany: Springer; 2022. p. 313–52.
20. Missaoui E, Mazigh B, Bhiri S, Hilaire V. A normative model for holonic multi-agent systems. In: Proceedings of the 2017 IEEE 29th International Conference on Tools with Artificial Intelligence (ICTAI); 2017 Nov 6–8; Boston, MA, USA. p. 1251–8.
21. Abdoos M, Mozayani N, Bazzan AL. Towards reinforcement learning for holonic multi-agent systems. Intell Data Anal. 2015;19(2):211–32.
22. Beheshti R, Barmaki R, Mozayani N. Negotiations in holonic multi-agent systems. In: Recent advances in agent-based complex automated negotiation. Berlin/Heidelberg, Germany: Springer; 2016. p. 107–18.
23. Abdoos M. Robustness analysis of holonic multi-agent systems: application to traffic signals control. Tehnički Vjesn. 2018;25(5):1306–13.
24. Mazdin P, Rinner B. Distributed and communication-aware coalition formation and task assignment in multi-robot systems. IEEE Access. 2021;9:35088–100. doi:10.1109/ACCESS.2021.3061149.
25. Qasim A, Iqbal S, Aziz Z, Kazmi SAR, Munawar A, Gilani BA, et al. Handling temporal constraints in interaction protocols for intelligent multi-agent systems. Int J Smart Sens Intell Syst. 2020;13(1):1–15.
26. Amrani NEA, Youssfi M, Bouattane O, Abra OEK. Interoperability between heterogeneous multi-agent systems recommended by fipa: towards a weakly coupled approach based on a network of recurrent neurons of the lstm type. In: Proceedings of the 2020 3rd International Conference on Advanced Communication Technologies and Networking (CommNet); 2020 Sep 4–6; Marrakech, Morocco. p. 1–6.