Using an Ontology-Based Multi-Agent System for Decentralized Control of a Swarm of UAVs

Anatoly Gladun[1], Katerina Khala[1]

[1] *International Research and Training Center for Information Technologies and Systems under NAS and MES of Ukraine, 40 Acad. Glushkov av., Kyiv, 03187, Ukraine*

Abstract
One of the main challenges today is deploying a coordinated group of unmanned aerial vehicles (UAVs), especially those based on multi-agent systems (MAS), which offer the advantages of decentralized management. Key benefits of using MAS in UAV systems include enhanced joint problem-solving capabilities, improved system survivability, and increased availability and scalability when executing complex missions. Effective data exchange between drones requires interoperable information, ensuring unambiguous communication and a formalized understanding of each drone's role and function. Furthermore, such systems need to establish protocols for cooperation, role reassignment in cases of failure or loss, and reliable identification of other UAVs. Implementing an ontological model can help address these challenges by formalizing knowledge about system policies for various complex scenarios, thereby enhancing the MAS's ability to accomplish its objectives efficiently.

Keywords
Ontology, knowledge representation, swarm intelligence, drones, UAV, hierarchical control structure, adaptive ontology

1. Introduction

Today, UAVs (drones) are gaining significant attention across various fields, including military, state security, natural resource protection, and numerous civilian applications. For instance, the decreasing cost of drones has broadened their appeal for civilian uses such as precision agriculture [1,2], surveillance, environmental monitoring [3], and search and rescue operations [4]. UAVs have proven particularly effective in dynamically changing environments and hard-to-reach areas, though these scenarios often require specialized sensors to address specific challenges.

Recent advances in technologies like blockchain (distributed databases), artificial intelligence, and machine learning have enabled the development of UAV systems with enhanced capabilities. These improvements offer higher safety, reliability, and efficiency, increasing the UAVs' ability to perform complex tasks more successfully.

In some cases, relying on a single UAV can be limiting, such as in search and rescue or surveillance operations. Quickly deploying multiple drones can significantly improve the likelihood of successful mission completion. Small UAVs capable of operating cooperatively with minimal human intervention are particularly valuable, as they allow tasks that would have been assigned to a single drone to be divided and performed in parallel. As noted in [5], using groups of UAVs offers several advantages:

The overall cost of purchasing and maintaining several small commercial UAVs is lower than that of a single large UAV.

Scalability, an essential feature of UAV groups, is often lacking in single-UAV operations.

Fault tolerance is increased, as the malfunction of one drone has a limited impact on the overall group.

Operations are completed more quickly due to distributed tasking.

Improving the efficiency of multi-UAV coordination is essential for maximizing results while

ITS-2023: Information Technologies and Security, November 30, 2023, Kyiv, Ukraine
glanat@yahoo.com (A. Gladun); cecerongreat@ukr.net (K. Khala)
0000-0002-4133-8169 (A. Gladun); 0000-0002-9477-970X (K. Khala)

minimizing costs in cooperative tasks. Unlike single-UAV decision-making, multi-UAV coordination involves challenges such as intelligent decision-making [6], distributed cooperation [7], and formation control [8].

We proposed a multi-agent system (MAS) model to manage the coordination, deployment, and data exchange among multiple UAVs. This model is based on an ontological knowledge representation, enabling effective decision-making across various scenarios during task execution. MAS handles specific functions, such as coordinating joint efforts by dividing mission objectives into sub-tasks for each UAV, facilitating message exchange between UAVs during collaborative operations, and dynamically reassigning roles if a UAV is lost or fails.

For optimal UAV network operations, it is essential to model each UAV as an agent and create work plans that accommodate the distributed roles within the UAV network. To plan network operations effectively using MAS, five components are needed:

knowledge about the external environment;

information on the defined operational area;

knowledge of role distribution (e.g., leader, coordinator, executor);

a set of tasks to be performed (e.g., patrol, regroup, retreat);

mechanisms for task distribution and redistribution.

A key aspect is decomposing complex tasks into sub-tasks and organizing interactions between UAV agents and ground control centers. Also, supreme considerations include ensuring information security, reliable data transmission, mutual UAV identification, and improving calculation accuracy during task execution.

2. Organizational structure, management, and information exchange challenges in UAV groups

UAV groups generally are organized into two main structural categories: centralized and decentralized. In centralized groups, a central scheduler coordinates tasks. Under *centralized control*, a leader manages all individual nodes, while *hierarchical coordination* distributes tasks through multiple hierarchical levels. In contrast, decentralized swarms lack a single leader or central planner. With *coordination by consensus*, nodes collectively decide how to perform and coordinate tasks, often using methods like voting or an auction system. In *immediate coordination*, each node responds to its surrounding nodes [9].

Centralized groups quickly find satisfactory solutions, and their behavior can be planned in advance. However, they are sensitive to leader loss, involve computational complexity, and are slower in team allocation. Decentralized groups, on the other hand, are more scalable, have no single point of failure, and can operate in low-bandwidth environments. They excel at finding novel solutions to challenges and can achieve complex outcomes with simple system designs. However, decisions made by each node are based on localized information rather than on data collected at the group's global level [10].

The ability to make decisions is a crucial attribute in designing autonomous and intelligent systems. Real-time decision-making based on data collected by the swarm enhances decision efficiency and allows the swarm to remain resilient in the face of uncertainties and dynamic changes.

Managing a group of UAVs is a complex task with four primary management approaches:

switching between algorithms that dictate swarm behavior;

adjusting the parameters of the group management algorithm;

remotely controlling specific nodes (leaders); and

modifying the environment to influence group behavior [11].

For joint operations, it is critical that drones can gather accurate information and share it unambiguously with each other and with the command-and-control operator. In a complex system, effective information exchange enhances cooperation and coordinated actions among decentralized participants, thereby improving the achievement of shared goals.

Information can come from internal sources (like sensors) and external sources (such as weather forecasting systems). However, significant challenges arise when exchanging information between UAVs, especially due to data heterogeneity. Data from various sources can appear in different formats, depending on their type and origin. Even when UAVs use the same terminology, interpretations may vary. For instance, one drone might use the term "position" to indicate a geo-referenced local frame, while another uses it to refer to angular coordinates.

To address these issues, semantic compatibility within the system is essential for resolving data heterogeneity and enabling transparent information exchange. Additionally, machine-computed logic supports reasoning, knowledge discovery, and data integration across systems. To ensure unambiguous understanding by the drones and the operator, the information exchanged must carry rich semantics that effectively model and abstract data heterogeneity [12].

The use of ontologies is proposed to facilitate clear communication and mutual understanding between UAV agents and the operator. Ontologies perform as controlled dictionaries of logically and well-defined terms and are structured hierarchically through type-subtype relationships. These terms are used to label and semantically enrich various data types, enabling their integration within a computational environment. Ontologies allow the definition of functions, contexts, and situations needed for semantic information exchange, facilitating effective communication between agents or between an operator and an agent.

3. Ontological approach to ensuring communication of UAV group

An ontology provides a well-defined, unambiguous conceptualization. The TBox operator describes the system using controlled vocabulary, while the ABox operator contains facts or statements compatible with the TBox vocabulary. This structure facilitates information gathering, knowledge sharing, and potential reuse.

Ontologies are widely used to formally represent information across various fields, including the semantic web, smart homes, and healthcare. They allow reasoning about objects and their attributes within a domain. Notably, group-based systems operate in environments where individual information and knowledge can be specified as atomic concepts and correlated by semantic contents containing the core concepts needed for a shared semantic understanding among agents exchanging information. Four primary classes within the ontological model form the foundation of the UAV operational process:

information objects classes, such as flight and surveillance plans and other directives provided by operators or programmed into the UAV;

agents' classes, such as UAV operators and autonomous UAVs, that send, receive, and execute directives;

processes classes, such as flight, communication, and surveillance processes, provided by these directives and performed by these agents;

roles classes, such as commander, operator, host, and decoy, are assigned to participants in these processes and dictate the prescriptions for which each participant is responsible.

An important aspect of task allocation for multiple UAVs is the characterization of mission scenarios from various perspectives. These scenarios are implemented as constraints or objectives in the task allocation process for a few UAVs. Constraints and objectives relate not only to limited resources and UAV heterogeneity but also to the diversity of task requirements and the complexity of the task environment. Task requirements vary widely, for example, some tasks involve mobile or unknown targets, while others have strict time constraints. Environmental constraints may also play a role, as real-world settings often contain obstacles and hazards. Additionally, constraints can be unknown, dynamic, or even adversarial rather than static or well-defined. Although detailed constraints help create more realistic scenarios, they also increase the complexity of finding viable solutions.

A *scenario class* contains a set of instructions that coordinate a group of agents to achieve a common goal. A key feature of scenario ontology is its general application to group actions and actors, defined in terms of the roles played by group members and the rights, responsibilities, and constraints associated with these roles. It enables the scenario to be applied repeatedly, in different circumstances, or by different agent groups. Scenarios are distinct from general plans or other directive information, as they offer advantages in competitive contexts. They are primarily used to coordinate the actions of group members, assigning tasks and responsibilities according to the different roles each member performs.

An *action class* represents a directive information object that assigns an action as required, prohibited, or permitted, resulting from an activity that implements a specific authority role.

A *group class* consists of agents whose members are intentionally associated with performing assigned roles and responsibilities. This structure is necessary for achieving one or more goals through direct cooperation and distributed decision-making. When group members collaborate towards a common goal, they do so by dividing problems and responsibilities. Each member fulfills at least one group role defined by the set of responsibilities and rights they hold within the context of the group's functioning.

A *group role class* refers to a role assigned to an agent, the member of a swarm, based on a specific action assigned to that agent. The agent must apply this action within the appropriate contexts of the group.

Ontologically defined scenarios are self-explanatory and can be utilized by both operators and agents. It creates a shared understanding of the information contained in the scenario and the entities referred to within that information. It ensures that group scenarios provide a unified vocabulary for both intra-group and inter-group communication regarding various aspects of their current challenges.

4. Ontology-based multi-agent system for UAVs interaction

The application of MAS as an intelligent system offers the flexibility to enhance reliable and successful interactions among UAV groups. MAS is capable of solving coordination and optimization problems and is adaptable to the uncertainties present in complex systems. Agents within this framework can not only accept tasks but also take the initiative to request tasks by sharing and exchanging information with other agents for coordination, communication, or cooperation. This system is well-suited for complex and distributed tasks, reducing human involvement in the decision-making process.

At the core of the multi-agent approach is the concept of remote and intelligent software agents, which function as independent specialized computer programs or elements of artificial intelligence. These agents can independently determine the best methods to achieve their goals and execute their tasks, exhibiting properties such as autonomy, activity, proactivity, and social behavior [14].

Ontologies can significantly enhance MAS at various stages of development by providing a clear separation of the overall problem, facilitating the process of searching and reusing information, supporting analysis and manipulation of information, and enabling effective communication between agents.

The agent must receive, store, and process information about the current state of the subject area. An agent's knowledge of the environment, other agents, and itself is represented as an ontology. This approach effectively addresses the challenges of generalizing heterogeneous low-level data into relatively high-level concepts, and it facilitates data sharing, system interoperability, and software reuse. When agents operate with the same concepts, it helps solve numerous problems, including communication methods between agents and adaptation to new conditions.

Ontologies enhance the ability to perform semantic reasoning [15], providing functionalities such as consistency checking, concept enforcement, classification, and implementation. They also enable a shared understanding of information structures between humans and software agents, allowing for the reuse of core knowledge [16]. Integrating these semantic technologies into MAS improves the knowledge representation and reasoning capabilities of applications developed within these frameworks [17]. The application of ontologies in MAS opens opportunities for creating logical rules that can be applied to semantic reasoning and the derivation of new knowledge. However, the use of ontologies can complicate MAS invariance concerning the domain, necessitating changes to the ontology—at least for that specific domain.

The design of agent behavior and interaction in MAS primarily involves data exchange models, which govern how agents communicate with one another and with ground control centers. An agent's data exchange framework typically includes message content and message exchanges (e.g., the KQML protocol). The message content comprises two components [18]: a content language (providing the syntax or grammar of the content) and an ontology (constituting the semantics or vocabulary of the message). Figure 1 illustrates an ontology-based agent data exchange model.

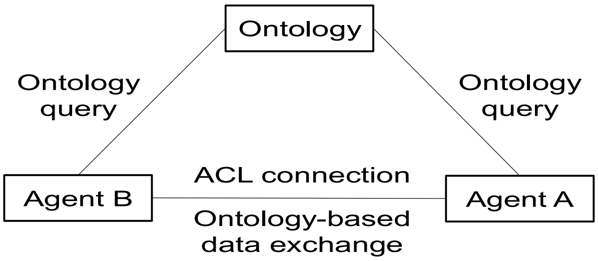

Figure 1: Model of data exchange between agents based on ontology [19]

Agents communicate through the exchange of messages, and the standard Foundation for Intelligent Physical Agents (FIPA) is commonly used to develop MAS [19]. The FIPA semantic language serves as a standard content language and is widely adopted in this context [20]. In FIPA, the ontology comprises a list of concepts, predicates, and actions that are specific to the communication domain. FIPA services provide the ontological agent with several ontology-related services to address the challenges associated with using multiple ontologies [18].

During the design of the MAS architecture, the syntax and semantics are introduced to define a common top-level ontology [21]. This type of ontology represents the common concepts used within the system, while the syntax and semantics for domain ontologies and agent-specific ontologies describe the purposes and functions of those ontologies [22, 23].

When developing the MAS architecture, it is necessary to highlight the following essential components:

- access service to provide access to agent attributes;
- message service that responded to the transmission of messages between agents or additional systems;
- agent library that contains information on the classification of agents within the MAS;
- agent interaction, which Manages the essential activities of agents, facilitating the loading and recording of agent properties while optimizing their resource usage;
- ontology - a knowledge base that encompasses information about the operating environment, performed actions, and self-knowledge of the agent (including updated data).

Figure 2 illustrates the interaction of the ontology-based multi-agent system with UAVs.

The relevance of ontology in MAS lies in its ability to address the issue of information overflow within the network [24]. The main problems that ontologies successfully solve include:

presentation of knowledge for logical inference applicable to requests made by users or agents;

filtering and classification of information;

indexing of collected information;

organization of a common terminology that agents and users can use for data exchange.

In this context, each UAV is modeled as an agent that must move, perceive its state and environment, follow a plan, achieve its goals, interact, and adapt its behavior. To account for the complex interactions between agents and to address the necessary heterogeneous properties, we have modeled the UAV agent using a Belief-Desire-Intention (BDI) architecture [24].

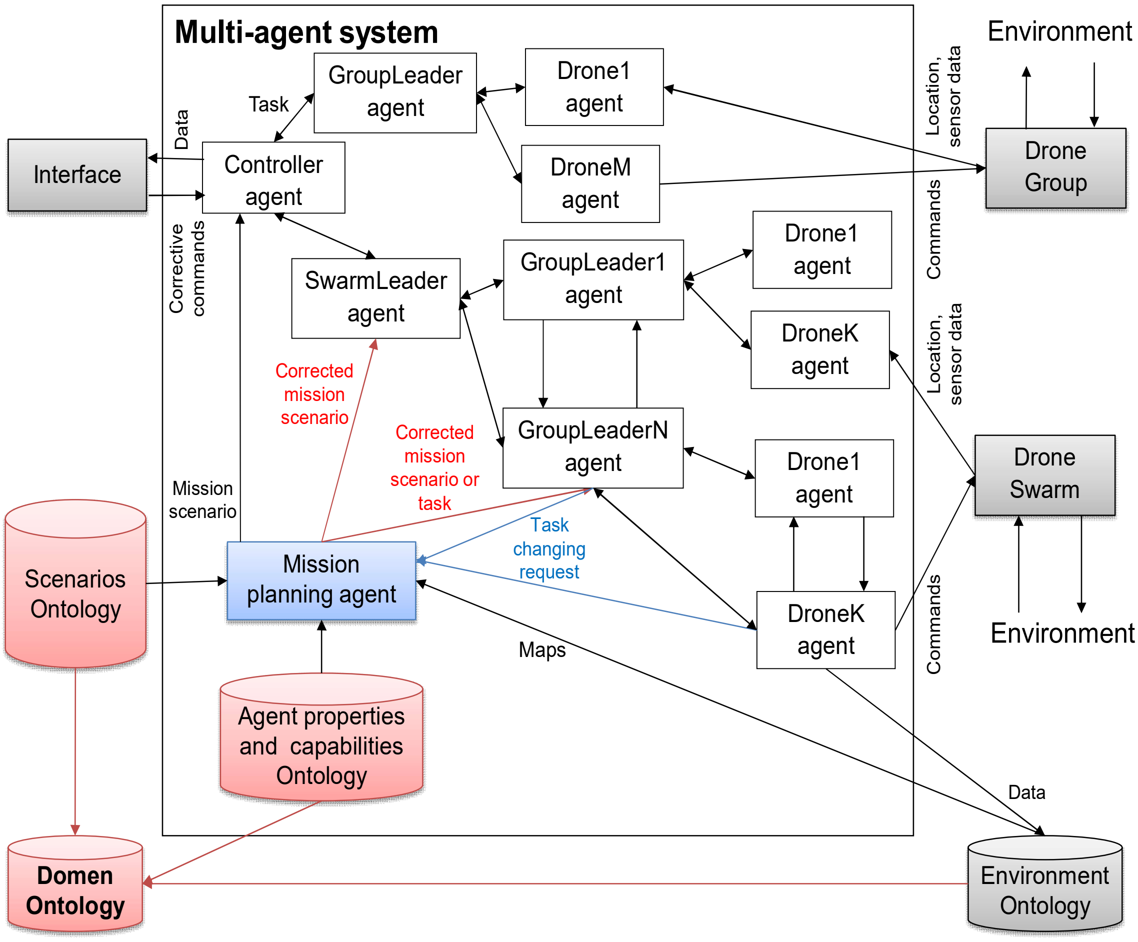

Figure 2: Interaction of ontology-based multi-agent system and UAV

A BDI architecture for an agent consists of a set of scenarios that define how the agent achieves its ultimate goals. Each scenario is composed of head, body, and tail labels that outline the agent's working algorithm. The body of the scenario contains sequences of actions that define the goals the agent must achieve and the conditions it must check. The head and tail labels represent intentions, which are drawn from a predefined list of intentions.

At any given moment, when an agent selects an execution scenario, the intentions can be either active or inactive. The agent will execute a scenario only when all the intentions in its head are active. After executing a scenario, the agent deactivates these intentions and activates all the tail intentions of the scenario. It is assumed that the current set of active intentions is not empty, although the body of scenarios and the set of tail intentions may be empty.

In this framework, an agent can be conceptualized as comprising a set of beliefs (B), plans (P), situations (S), actions (A), and intentions (I). When the agent perceives changes in the environment, it believes that an event (E) has occurred, which corresponds to certain situations from the environment (S). The registration of an event by the agent involves a change in its reasoning state, reflected in the selection of a belief from B. Based on this belief and its desires (which are determined by a plan from P), the agent fulfills certain intentions from I, which consist of a sequence of actions from A. These actions together form a plan for achieving the specified goal. Thus, the planned action is determined by the chosen plan and executed by altering the current situation in the environment.

Figure 3 illustrates a UAV agent that is based on ontology and is designed for movement, perception, plan execution, interaction, behavior adaptation, and skill control. Beliefs are linked to dispositions when addressing a problem and can be updated based on knowledge about the environment and the agent's self-knowledge. The UAV agent must verify each belief against logical rules derived from its self-knowledge and the assigned problem.

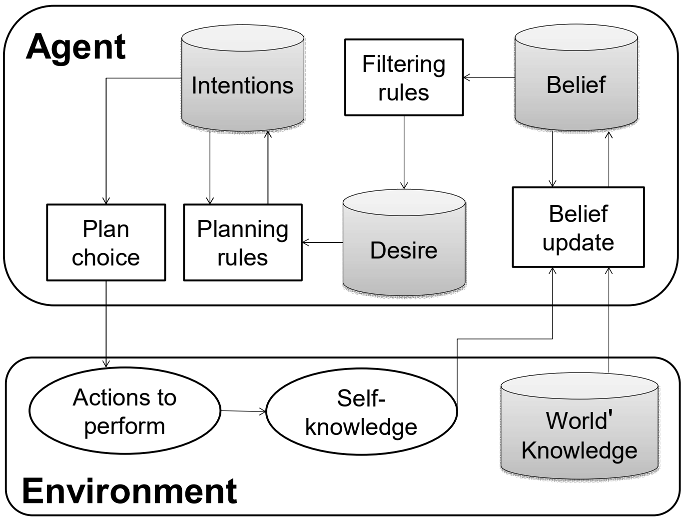

Figure 3: BDI model for UAV network agents

The UAV agent is motivated to achieve a specific goal, which is unattainable without a plan. The ontology of scenarios takes into account intentions and aspirations to determine the actions influenced by the environment. Given the complex and volatile nature of environments that can incapacitate or destroy agents, the agent's mission capability may change. Therefore, dynamic task allocation is crucial for enhancing the coordinated capabilities of MAS.

The proposed method involves creating roles to define the behavior of the agent model, facilitating hierarchical coordination. This model assumes partial connectivity with several nearby agents for a limited time. Each UAV agent can only interact with its nearest neighbors within a specified range. When UAV agents engage in interactions to decide on mission execution, it necessitates collaboration between agents and with the environmental ontology, a process that can be resource-intensive.

The hierarchical model includes roles that manage group heterogeneity and connectivity based on skill metrics, potential flight time, data sharing, and decision-making. For instance, a SwarmLeader or GroupLeader should take the lead in decision-making and communication while maintaining intermediate flight times. This role is crucial for making fundamental decisions that ensure mission success. Consequently, the GroupLeader agent must have efficient pathways for sending and receiving messages from other agents. For example, when a group of UAVs encounters an obstacle, both their trajectories and formation must be adjusted. The SwarmLeader or GroupLeader can implement a predetermined mission trajectory to reorganize the UAV group effectively.

In MAS, the BDI model is instrumental in managing mission information. Beliefs motivate agents to handle missions while considering their capabilities, limitations, and environmental factors. Capabilities include the UAV's movement, navigation, and location abilities derived from its hardware, as well as computing, processing, and communication capabilities associated with the drone agent. Desires represent the expected outcomes, such as the objective of executing a specific mission. Intentions outline the strategies for accomplishing the desired mission objectives. If a mission fails, the system consults with other agents to identify improvements, including when and how to modify formations.

An individual agent may diverge from the current common knowledge of the world, which can be resolved through information sharing and communication. Effective operational communication and interaction significantly impact the success rate of mission execution, which is why the Mission Planner was designed to facilitate the seamless distribution of tasks and information.

The MAS incorporates a deadline mechanism, an allocation confirmation mechanism, and a scale control mechanism to limit the scope of allocation. These features ensure real-time operation, prevent resource waste, and mitigate excessive pressure on communication and calculations, particularly when numerous agents are involved. When assigning missions, the order of execution is often more critical than maximizing the advantage of each individual agent. Ignoring this order can significantly reduce the feasibility of allocations and result in less efficient and optimal outcomes when missions are assigned without considering their interdependencies.

Before a mission begins, a comprehensive task plan is developed based on information derived from the environment ontology. These complex tasks are then decomposed into several subtasks and distributed among agents according to their capabilities. If the environment changes and agents encounter malfunctions during the execution of the mission, continuing with the previous plan may diminish effectiveness or render the mission impossible. Therefore, an online task assignment method must be employed to redistribute these subtasks in real-time, taking into account the capabilities of each agent.

Limitations often arise because certain missions require agents to operate in a coordinated logical sequence rather than as isolated entities. For example, blocking off a specific area necessitates teamwork among agents. The agent group mechanism is introduced to manage complex and interrelated missions. The mechanism can be separated into two types. One type requires agents in the MAS to engage in parallel data processing, such as joint monitoring. The other type demands strict adherence to spatial and temporal order, such as executing strikes following reconnaissance.

Different groups of agents carry out missions at varying costs, making it crucial to identify the most suitable group. However, this task complicates the optimization problem. Utilizing Particle Swarm Optimization (PSO) to determine the appropriate command structure offers an effective solution to this complex challenge involving substantial computations [25]. When selecting a group, its capabilities must align with the mission requirements while maintaining lower residual capacity and execution costs [26, 27].

In the PSO process, priority is given to parameters that better meet the requirements compared to execution costs and other alternatives. These priorities influence the particles, with the calculated fitness value of each particle being directly proportional to the efficacy of the command. Once a group is identified, the confirmation information is disseminated to all members. Only if all members accept the mission will the distribution information be sent to them; if any members reject it, the system will seek alternative agents with capabilities similar to those who declined or initiate a new round of group selection.

This intelligent redistribution allows the MAS to adapt to frequent mission changes, enhancing resource utilization and improving computational efficiency. The redistribution management mechanism is illustrated in Figure 4.

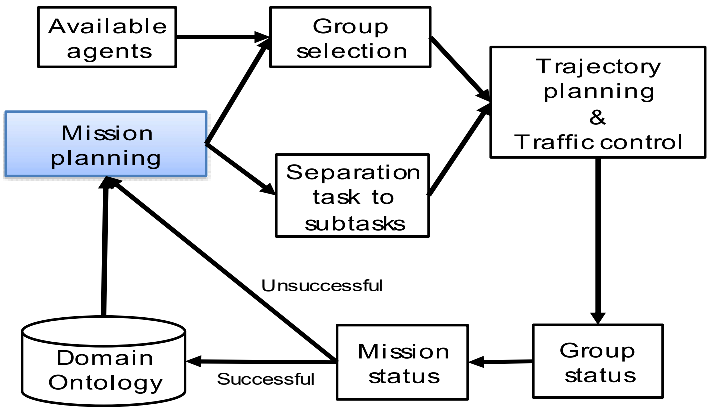

Figure 4: Mechanism of mission redistribution management

An initial group reacts to changes in the mission, which are then modeled and calculated to generate solutions based on the current availability of agents. To ensure that decisions are well-suited for the current group, the cost changes will be assessed, and updates will be made using domain ontologies. In this context, the domain ontology will integrate data from the agents' properties and capabilities, as well as from the scenarios and environment ontologies. This feedback will enable the system to adapt to mission changes and prevent excessive reallocation and misallocation of resources.

5. Conclusions

The proposed ontology-based multi-agent system (MAS) is designed for effective network interaction among a group of heterogeneous UAVs. This approach establishes a role-based hierarchy of UAVs, each equipped with designated travel routes, defined flight times, data-sharing capabilities, and the ability to make decisions to achieve a common goal.

An ontology enhances the data within the knowledge base by utilizing an agreed-upon structure of relationships and well-defined terms. This framework allows for logical conclusions to be drawn from data labeled with terms from the ontology.

Addressing security concerns within the MAS is essential, particularly regarding protection against unauthorized access and malicious code. Mobile agents from external sources pose various risks to the host system since they execute within its address space. To ensure security, each agent must undergo an authorization process before transferring control. This process involves verifying the agent's registration and determining whether it possesses the appropriate privileges to perform specific actions and access certain resources. The security system must effectively prevent any unauthorized actions by the agent.

To create the ontologies used in UAV MAS, we plan to develop new approaches to automate the formalized presentation of knowledge based on various information resources, whether open or closed. This approach relies on structuring the information field of the object through a taxonomic representation of selected characteristics of the object under study. This structuring is necessary for constructing a semantic model that addresses the recognition problem and adapts to the specifics of the problem being solved. The adaptation of the ontological models for the problem and the information object is based on their semantic proximity. The method involves applying weights to the concepts and relationships used in the recognition of information objects.

References

[1] M. Bacco, E. Ferro, A. Gotta, Radio propagation models for UAVs: what is missing?, in: Proceedings of the 11th International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2014, pp. 391-392.

[2] F. M. Bacco, E. Ferro, A. Gotta, UAVs in WSNs for agricultural applications: an analysis of the two-ray radio propagation model, in: IEEE SENSORS 2014 Proceedings. IEEE, 2014, pp. 130-133.

[3] G. Chmaj and H. Selvaraj, Distributed processing applications for UAV/drones: a survey, in: Progress in Systems Engineering. Springer, 2015.

[4] S. D'Auria, M. Luglio, C. Roseti, R. Strollo, F. Zampognaro, Real Time Transmission of Cultural Heritage 3D Survey in Case of Emergency, in: 3rd Inter-national Conference on Information and Communication Technologies for Disaster Management, Vienna, Austria, December 2016.

[5] I. Bekmezci, O. K. Sahingoz, S. Temel, Flying ad-hoc networks (FANETs): A survey, Ad Hoc Networks, vol. 11, no. 3, 2013, pp. 1254-1270.

[6] Y. Han, H. Wang, Z. Zhang, W. Wang, Boundary-aware vehicle tracking upon UAV, Electronics Letters, no. 56, 2020, pp. 873–876. doi:10.1049/el.2020.1170

[7] H. Jiang, D. Shi, C. Xue; Y. Wang, G. Wang, Y. Zhang, Multi-agent deep reinforcement learning with type-based hierarchical group communication, Appl. Intell., vol. 51, 2021, pp. 5793–5808. doi:10.1007/s10489-020-02065-9

[8] G. Zhan, Z. Gong, Q. Lv, Z. Zhou, Z. Wang, Z. Yang, D. Zhou, Flight test of autonomous formation management for multiple fixed-wing uavs based on missile parallel method, Drones, no. 6, 2022, p. 99.
[9] M. Brambilla, et al., Swarm robotics: a review from the swarm engineering perspective, Swarm Intelligence, vol. 7, 2013, pp. 2–3.
[10] M. Yogeswaran, S. Ponnambalam, Swarm robotics: an extensive research review, in: I. Fuerstner (Eds.), Advanced Knowledge Application in Practice, London, 2010, p. 259.
[11] A. Kolling, et al., Human interaction with robot swarms: a survey, in: IEEE Transactions on Human-Machine Systems, vol. 46, no. 1, 2016, p. 8.
[12] M. Zgurovsky, D. Lande, A. Boldak, Linguistic Analysis of Internet Media and Social Network Data in the Problems of Social Transformation Assessment, Cybernetics and Systems Analysis, vol. 57, issue 2, 2021, pp. 228–237. dio: doi.org/10.1007/s10559-021-00348-8.
[13] A. Gladun, K. Khala, R. Martinez-Bejar, Development of Object's Structured Information Field with Specific Properties for Its Semantic Model Building, in: CEUR Workshop Proceedings, Vol. 3241, 2021, pp. 102–111.
[14] A.Ya. Gladun, Intellektualnye agenty v korporativnyh sistemah. Korporativnye sistemy, No. 3, 2006, pp. 48-64. https://comizdat.com/index_.php?in=ksks_allarticles&CATEGORY=38.
[15] E. Sirin, B. Parsia, B.C. Grau, A. Kalyanpur, Y. Katz, Pellet: A practical OWL-DL reasoner, Web Semantics: Science, Services and Agents on the World Wide Web 5(2), 2007, pp. 51–53. doi:10.1016/j.websem.2007.03.004
[16] T.R. Gruber, A translation approach to portable ontology specifications, Knowledge Acquisition 5(2), 1993, pp. 199–220. doi:10.1006/knac.1993.1008.
[17] M. Hadzic, P. Wongthongtham, T. Dillon and E. Chang, Ontology-Based Multi-Agent Systems, Studies in Computational Intelligence, Springer, 2009. doi:10.1007/978-3-642-01904-3.
[18] V.M. Catterson, E.M. Davidson, S.D.J. McArthur, Issues in Integrating Existing Multi-Agent Systems for Power Engineering Applications, in: Proceedings of the 13th International Conference on, Intelligent Systems Application to Power Systems, Arlington, VA, USA, 6–10 November 2005, p. 6.
[19] FIPA. FIPA Ontology Service Specification; FIPA: Geneva, Switzerland, 2000.
[20] A. Borgida, Description logics in data management, in: IEEE Trans. Knowl. Data Eng. no. 7, 1995, pp. 671–682.
[21] M. Wooldridge, N.R. Jennings, D. Kinny, The Gaia Methodology for Agent-Oriented Analysis and Design. Auton. Agents Multi Agent Syst., no. 3, 2000, pp. 285–312.
[22] M. Uslar, M. Specht, S. Rohjans, J. Trefke, J.M. González, The Common Information Model. CIM: IEC 61968/61970 and 62325—A Practical Introduction to the CIM; Springer: Berlin, Germany, 2012.
[23] F. Bellifemine, G. Caire, D. Greenwood, Developing multi-agent systems with JADE, John Wiley & Sons, Ltd, 2007, p. 281.
[24] M.H. Dominguez, J.-I. Hernández-Vega, D.-G. Palomares-Gorham, C. Hernández-Santos, J.S. Cuevas, A BDI Agent System for the Collaboration of the Unmanned Aerial Vehicle. Res. Comput. Sci., vol. 121, 2016, pp. 113–124.
[25] J. Kennedy, R. Eberhart, Particle swarm optimization, in: Proceedings of ICNN'95-international conference on neural networks. vol. 4, 1995, pp. 1942–1948.
[26] Y. Rizk, M. Awad, EW. Tunstel, Decision making in multiagent systems: A survey. IEEE Transactions on Cognitive and Developmental Systems, 10(3), 2018, pp. 514–529. https://doi.org/10.1109/TCDS.2018. 2840971
[27] K. Osipov and G. Sukthankar, Forming effective teams from agents with diverse skill sets, in: Proceedings of the International Conference on Social Informatics (SocialInformatics '12), December 2012, pp. 44– 48.