

Article

A Cooperative Decision-Making Approach Based on a Soar Cognitive Architecture for Multi-Unmanned Vehicles

Lin Ding^{1,2}, Yong Tang^{3,4}, Tao Wang^{1,2,*}, Tianle Xie^{1,2}, Peihao Huang^{1,2} and Bingsan Yang^{1,2}

- ¹ School of Intelligent Systems Engineering, Sun Yat-sen University, Shenzhen 517108, China; dinglin7@mail2.sysu.edu.cn (L.D.); tsetlok@mail2.sysu.edu.cn (T.X.); huangph26@mail2.sysu.edu.cn (P.H.); yangbs3@mail2.sysu.edu.cn (B.Y.)
- ² Southern Marine Science and Engineering Guangdong Laboratory, Zhuhai 519000, China
- ³ School of Civil Aviation, Northwestern Polytechnical University, Xi'an 710072, China; tangyonguas@163.com
- ⁴ Unmanned Aerial System Co., Ltd., Aviation Industry Corporation of China (Chengdu), Chengdu 610091, China
- * Correspondence: wangt339@mail.sysu.edu.cn

Abstract: Multi-unmanned systems have demonstrated significant applications across various fields under complex or extreme operating environments. In order to make such systems highly efficient and reliable, cooperative decision-making methods have been utilized as a critical technology for successful future applications. However, current multi-agent decision-making algorithms pose many challenges, including difficulties understanding human decision processes, poor time efficiency, and reduced interpretability. Thus, a real-time online collaborative decision-making model simulating human cognition is presented in this paper to solve those problems under unknown, complex, and dynamic environments. The provided model based on the Soar cognitive architecture aims to establish domain knowledge and simulate the process of human cooperation and adversarial cognition, fostering an understanding of the environment and tasks to generate real-time adversarial decisions for multi-unmanned systems. This paper devised intricate forest environments to evaluate the collaborative capabilities of agents and their proficiency in implementing various tactical strategies while assessing the effectiveness, reliability, and real-time action of the proposed model. The results reveal significant advantages for the agents in adversarial experiments, demonstrating strong capabilities in understanding the environment and collaborating effectively. Additionally, decision-making occurs in milliseconds, with time consumption decreasing as experience accumulates, mirroring the growth pattern of human decision-making.

Keywords: multi-unmanned vehicle; cooperative decision-making; cognitive architecture



Citation: Ding, L.; Tang, Y.; Wang, T.; Xie, T.; Huang, P.; Yang, B. A Cooperative Decision-Making Approach Based on a Soar Cognitive Architecture for Multi-Unmanned Vehicles. *Drones* **2024**, *8*, 155. <https://doi.org/10.3390/drones8040155>

Academic Editor: Abdessattar Abdelkefi

Received: 28 February 2024
Revised: 1 April 2024
Accepted: 16 April 2024
Published: 18 April 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Multi-unmanned systems have demonstrated significant applications across various fields under complex or extreme operating environments. Research on decision-making and adversarial scenarios for multi-unmanned systems is predominantly centered around machine learning and artificial intelligence algorithms, using deep learning networks [1–6] and bio-inspired heuristic algorithms [7–12]. However, these methods encounter many challenges, such as difficulties understanding human decision processes, poor time efficiency, and reduced interpretability, thus limiting their applicability in security contexts. In contrast, methods based on planning strategies demonstrate higher stability and interpretability, with enhanced safety. Nonetheless, their low scalability poses challenges in adapting to complex and dynamic environments [13,14]. Therefore, scholars adopted a strategy of integrating two methods to overcome respective limitations. For example, they proposed collaborative human-ml decision-making and expert-machine collaborative decision-making methods. These methods involve task division, with humans providing domain knowledge and experience, while machine learning systems handle data analysis

and pattern recognition, thus jointly supporting decision-making [15]. Another strategy involves making decisions using expert systems and machine learning together and integrating the final decision through credibility settings [16,17]. Additionally, leveraging explainable AI for enhanced decision-making involves training enhanced interpretable algorithms using samples evaluated by humans [18,19]. Nonetheless, these methods still fail to fully consider the process of human cognitive modeling. Therefore, this paper proposes an approach that simulates human cognition, constructing a real-time online collaborative decision-making model that is both interpretable and scalable. The aim is to address the aforementioned issues encountered in unknown, complex, and dynamic environments.

Soar, standing for State-Operator-Action-Result, embodies a versatile cognitive architecture designed to emulate human cognitive processes. It employs knowledge-based reasoning to analyze input data, enabling it to tackle a wide array of problems effectively. Following its abbreviation, Soar's problem-solving approach revolves around state transitions where the initial state corresponds to the problem posed and the resolution of the problem corresponds to reaching the goal state. In Soar, the result of state transition is achieved through proposing operators and applying the action of proposed operators [20–22]. Specifically, Soar begins by constructing state spaces with MEs (Memory Elements) through the application of domain-specific planning knowledge. MEs consist of long-term memory elements and short-term memory elements. Long-term memory elements represent domain knowledge. Short-term elements entail information from the current environment, serving for temporary storage and processing. When short-term memory elements meet all conditions of long-term elements, it is termed as the current situation matching the state space associated with the rule. Next, Soar retrieves the operators corresponding to the state space and stores these operators in the operator candidate set. After traversing all the state spaces, the operator candidate set will contain all available operators. Each operator carries an attribute called Preference, where a higher preference indicates a higher priority for the operator. Then Soar selects an operator based on their preferences, executes corresponding actions in the operator, orchestrates transitions between states, and repeats the above steps until the problem is resolved, culminating in the desired results [23,24]. Through its rule-based reasoning mechanism, Soar can derive decisions from interpretable rules, further enhancing its problem-solving capabilities.

Notably, Soar not only offers interpretability but also incorporates learning mechanisms for continuous improvement. Chunking, an experiential learning technique, monitors the historical problem-solving process, capturing relevant scenarios and final decisions to derive new rules. These new rules are promptly activated upon encountering similar scenarios, thereby reducing decision retrieval time [25]. The RL (Reinforcement Learning) mechanism updates operators' preferences based on rewards from historical behaviors, adjusting the probability of operator selection and application to maximize future rewards [26]. Episodic learning records historical problem-solving processes and acquires experience by comparing historical and current scenarios based on activation [27]. These learning capabilities enable Soar to make decisions even in the absence of matching rules, gradually enhancing and optimizing decisions to meet the collaborative cognitive gaming requirements of multi-unmanned systems in complex and dynamic environments.

Historically, Soar applications have primarily focused on a single intelligent agent, encompassing a range of fields such as robotics [28–32], target identification and allocation [33,34], and autonomous learning systems [35–38]. However, multi-agent models are also employed, albeit in a smaller proportion. Examples include the TacAir-Soar project [39,40] and MOUTBots project [41].

In Soar-based multi-agent projects, extensive knowledge bases are established to meticulously decompose potential scenarios. Tasks are abstracted into hierarchical structures, and actions are encoded layer by layer, culminating in the completion of the entire project [39–41].

This paper models collaborative and adversarial behaviors under complex and dynamic environments, establishing a multi-unmanned adversarial system based on Soar, and it validates the model in Unity.

Section 1 introduces the research objectives and content while briefly outlining the basic principles of the Soar cognitive architecture and related work. Section 2 elaborates on the modeling of multi-unmanned systems based on the Soar architecture and knowledge establishment. Section 3 shows the adversarial environment and game settings based on Unity. Section 4 analyzes the simulation results. Section 5 is a conclusion section, providing an overview of the research achievements and future prospects.

2. System Design Based on Soar Cognitive Architecture

2.1. Overall Framework Design

Implementation of the Soar framework hinges on knowledge establishment, which enables the system to thoroughly comprehend information, and the interface facilitates an accurate interaction with the environment. Soar interprets the knowledge, compares it with the environmental information, develops reasoning, and finally makes decisions that should be taken and implemented. The overall framework design is illustrated in Figure 1.

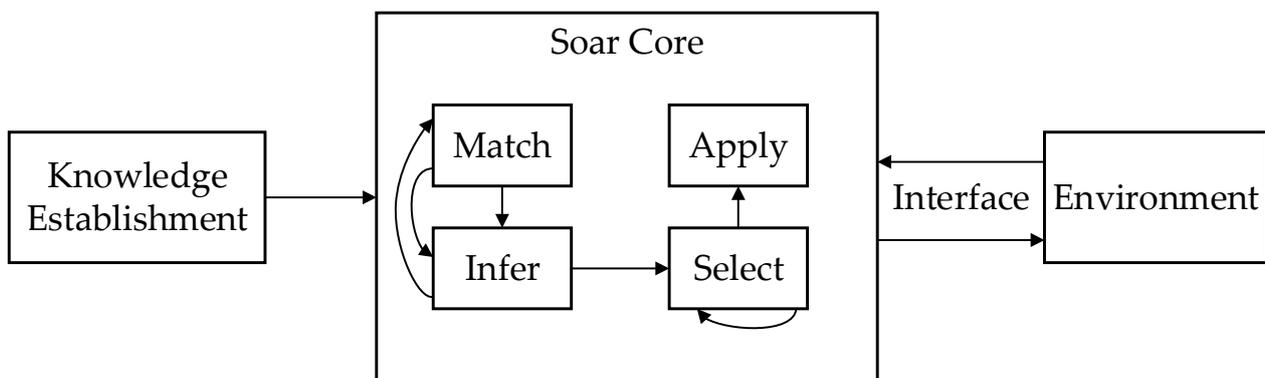


Figure 1. Overall framework design.

2.2. Interface

The environmental information is provided by Unity in this paper. Soar interacts with the external world through the SML interface and handles tasks such as receiving environmental information and outputting decisions [42]. Unmanned systems transmit information obtained from the Unity environment to a Java program through sockets. This paper presents a Java program that organizes symbolic information through a series of procedural steps, ensuring that the feature symbols conform to the requirements of the Soar kernel. The processed information is then sent to the Soar kernel. Soar receives and integrates the information, conducts knowledge reasoning to generate decisions, and finally outputs the decision results to the Java program. The Java program packages the decisions, utilizes sockets to transmit them back to the unmanned systems, and the unmanned systems interpret the decisions and execute corresponding actions, forming the OODA loop, as shown in Figure 2.

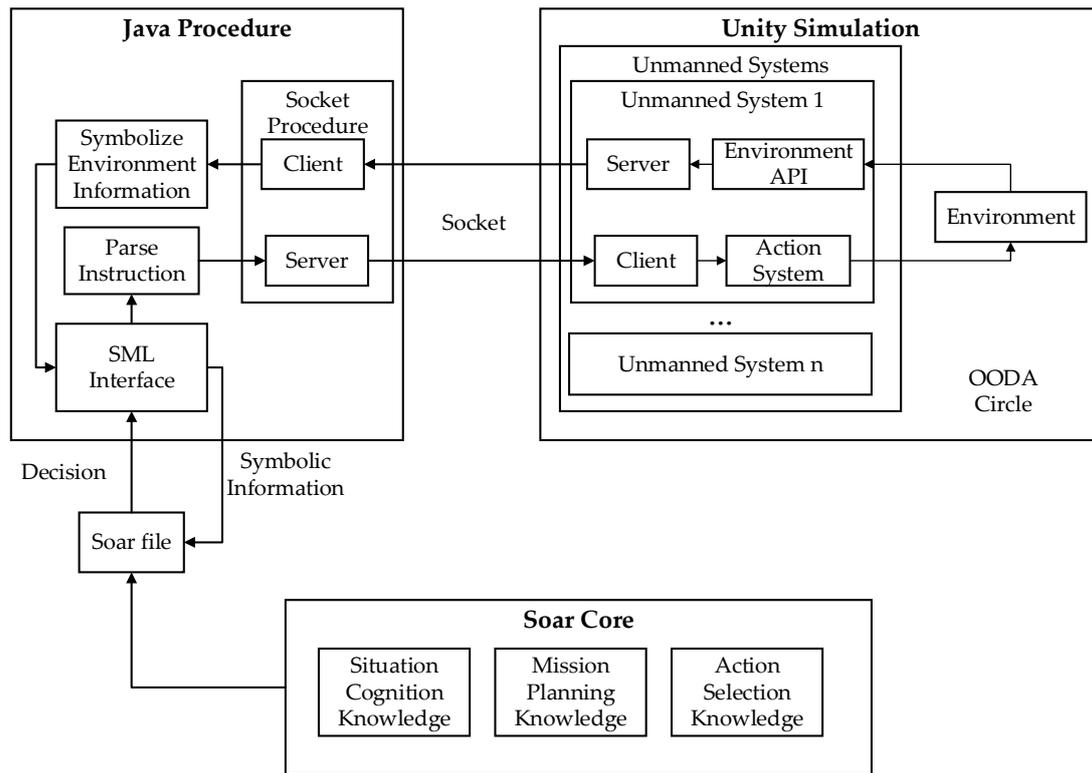


Figure 2. Project structure.

2.3. Knowledge Establishment

Knowledge establishment is a crucial step in implementing the Soar framework. In the construction of multi-unmanned systems, it involves conceptualizing, encoding, and integrating various types of knowledge to enable them to function effectively within the Soar cognitive architecture. Generally, the cognitive process can be divided into two main stages. Firstly, it utilizes domain knowledge to summarize and contemplate the initial information, forming a deeper cognitive foundation. This process does not generate output commands to the environment but is akin to the internal thought process in the human mind, creating new cognitive elements. Secondly, it integrates new cognitive information with environmental information, reasons, and then formulates appropriate action plans. This dual-stage cognitive processing approach adapts more effectively to complex and dynamic scenarios.

This paper categorizes knowledge into three types: Situation Cognition Knowledge, Mission Planning Knowledge, and Action Selection Knowledge. These knowledge categories will guide unmanned systems in making appropriate decisions.

Situation Cognition Knowledge helps the system to develop a deeper understanding and cognition of the information, which is a type of internal reasoning knowledge, as shown in Figure 3. It effectively integrates complex information into a comprehensive cognitive structure, which includes terrain analysis, orientation perception, capability awareness, distance judgment, role inference, threat assessment, and comprehensive value assessment. These elements permeate the entire cognitive process, ensuring that unmanned systems can understand and respond to various scenarios as well as provide a cognitive foundation for subsequent planning decisions.

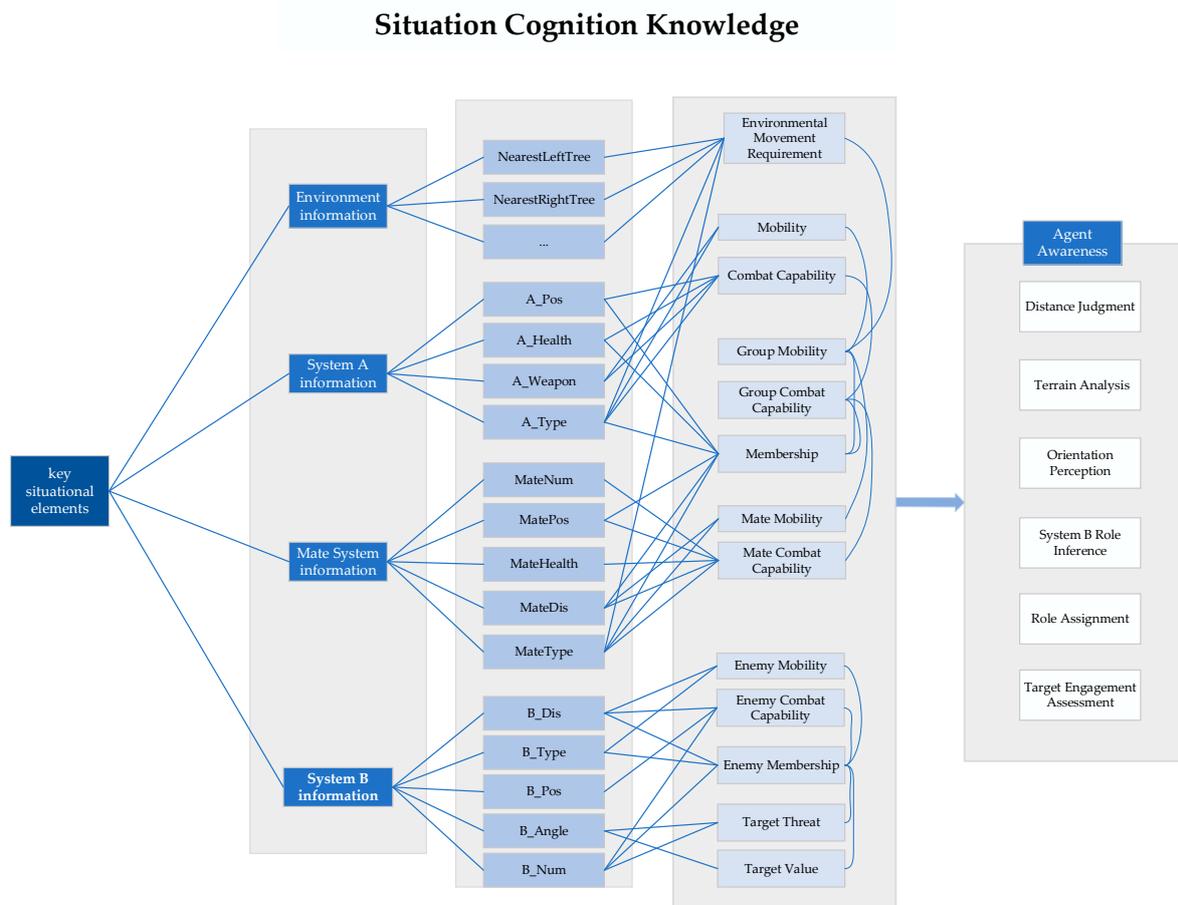


Figure 3. The key design analysis of Situation Cognition Knowledge involves refining rules in Soar to endow unmanned systems with situational awareness.

Mission Planning Knowledge encompasses various factors and strategies that are considered during mission execution. It represents decision-oriented knowledge that generates sequences of autonomous decisions. Soar’s subgoal structure is utilized to decompose the overall mission into a combination of different types of tasks, such as reconnaissance, strike, etc., which are shown in Figure 4.

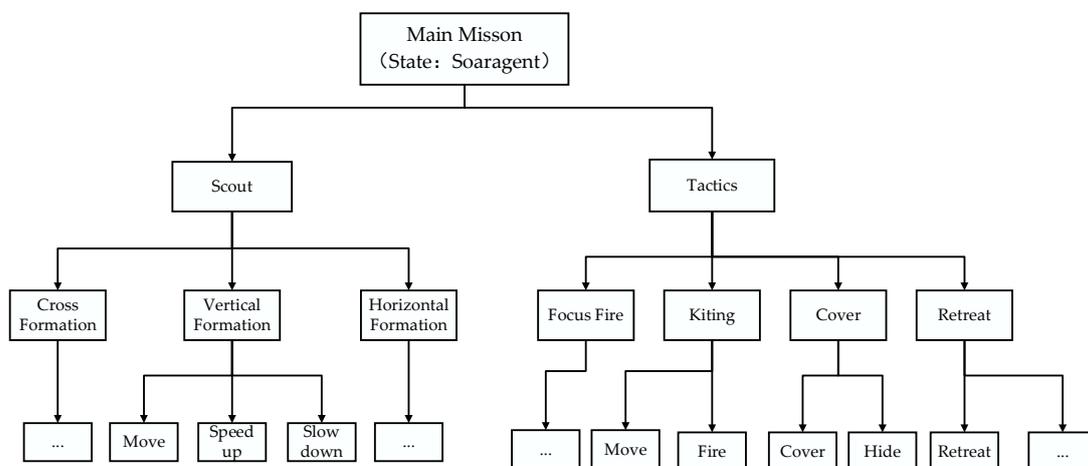


Figure 4. Subgoal structure for hierarchical task decomposition.

Generally, a task can also be subdivided into combinations of various actions, as clearly defined in Table 1.

Table 1. Description of action and tactic.

Type of Action	Description
Cross formation	When the team advances to a narrow road, a formation is designed to ensure safety both in the front and on both sides.
Vertical formation	When the team advances to a very narrow pathway allowing only single-person passage, a formation is designed for safe progression.
Horizontal formation	When the team advances to a wide area, a formation is designed to ensure the full utilization of firepower.
Focus fire	Both sides of the team maintain a horizontal alignment and launch a joint attack.
Kiting	The kiting tactic involves launching attacks by leveraging the advantages of range and speed.
Cover(tactic)	The cover strategy protects the attacker and completes the assault.
Move	Autonomous navigation to the corresponding point.
Retreat	Quick location of nearby large obstacles and resistance to rival attacks.
Fire	Firing at the target, but only one target can be attacked at a time. There is a certain cooldown period after firing.
Hide	Seeking cover while changing ammunition (no offensive capability).
Cover(action)	When there is not enough cover on the front lines, roles are switched to provide cover for teammates.
Speed up	When team members are distant, members positioned at the rear of the team accelerate to catch up.
Slow down	When team members are widely spaced, members positioned at the front of the team slow down.

Action Selection Knowledge involves the detailed modeling of individual behavior, primarily addressing two issues: the predictability problem caused by entities tending to choose only the best behavior in similar situations, and the problem of behavior merits and demerits. This knowledge models multiple behaviors for each scenario to ensure behavioral variability, and it assigns priority values to these behaviors. The priority values are closely related to the probability of behavior selection: the higher the priority value, the greater the likelihood of selection. Table 2 outlines these preferences. It helps systems to make more flexible and intelligent decisions in various situations.

Table 2. Preferences of tactics.

Situation	Focus Fire	Cover	Kiting	Retreat
Both teams have similar configurations.	4	3	2	1
Team 1 has a significant speed advantage.	3	2	4	1
Team 1 possesses a distinct power advantage.	3	4	2	1
Team 1 has critically low health.	1	2	3	4

Upon receiving environmental information, Soar first utilizes Situation Cognition Knowledge to form cognitive elements. Based on this cognitive understanding, the system further decomposes tasks and selects execution based on Mission Planning Knowledge using the subgoal structure. In this paper, a comprehensive understanding of elements is not entirely achieved at the initial stage. Since the system prioritizes different cognitive elements under different tasks, differentiated cognitive strategies are formed during the subtask stage to optimize system performance and reduce resource costs. Finally, the system formulates appropriate decisions and outputs corresponding behaviors to the external environment. The entire process is shown in Figure 5, presenting an organic and coherent cognitive and decision-making flow.

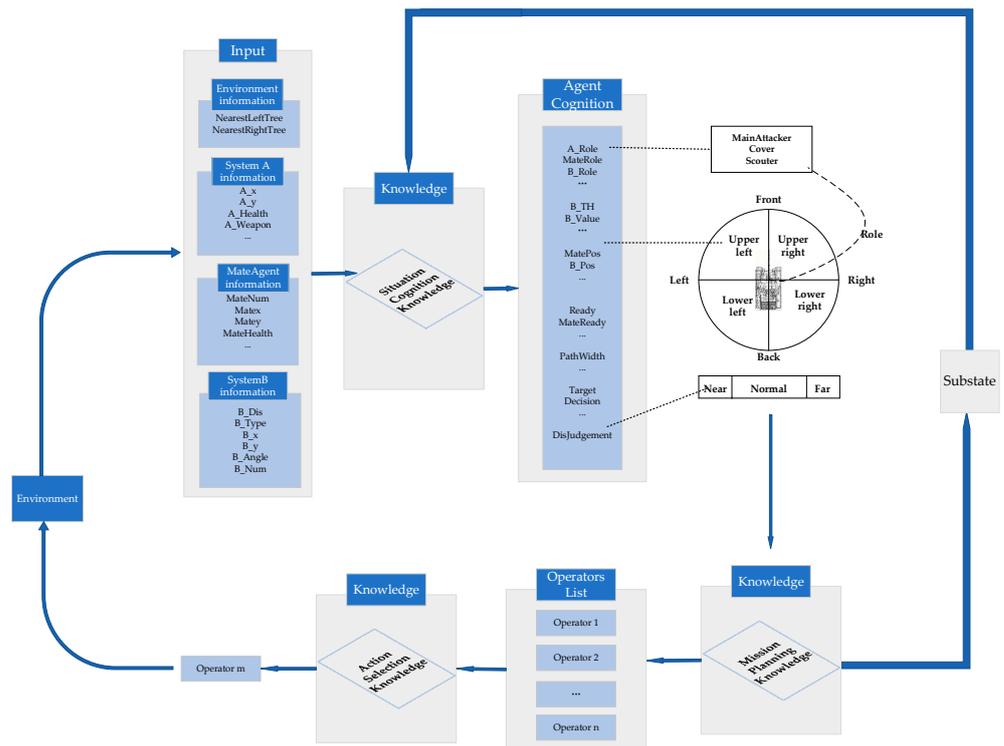


Figure 5. The process where an unmanned system based on Soar generates cognition from knowledge and outputs decisions.

3. Game Setting

3.1. Scene

The scenario draws inspiration from the game Tank Wars. Unlike the two-dimensional game, this paper implements a three-dimensional scene in Unity, which includes various types of unmanned systems and obstacles as elements. The view of each unmanned system is limited to increase local observability, while the movement characteristics and action orientations of unmanned systems are expanded to make the game more complex and dynamic. The environment is shown in Figure 6.



Figure 6. Three-dimensional Tank Battle game interface, Team 1’s spawn point is located at the bottom of the map. Team 2 has two possible spawn points, located on the right and left sides of the map, respectively. Different scenes are set based on different surrounding environments. Area 1 is a jungle passage with a constantly changing width, and it provides a necessary path to Area 2. Team 1 needs to make appropriate formation changes here to ensure safe progress. Area 2 is the confrontation area near Team 2’s spawn point 1, with many obstacles. Area 3 is the confrontation area near Team 2’s spawn point 2, which is open with no cover to block attacks.

Area 1, Area 2, and Area 3 are designed to verify the capabilities of the unmanned systems, including terrain awareness, an understanding of adversarial scenarios, and combat capabilities.

3.2. Team Configuration

This paper aims to investigate the collaboration and adversarial confrontation among multiple unmanned systems. Team 1 adopts the Soar intelligent collaborative confrontation rules, while Team 2 uses single-player confrontation rules. Team 1 possesses robust collaborative perception capabilities and shares information; if any member detects rival information, they publish it to the platform which shares information to other members, ensuring that all team members can promptly access dynamic rival information, even if they have not directly detected the rival's position themselves. Team 2 relies solely on the members' own perceptual information.

Furthermore, Team 1 dynamically calculates the comprehensive value and threat within the team, choosing to attack the Team 2 member with the highest comprehensive value to quickly weaken the opponent's strength. Team 2, on the other hand, focuses on attacking the nearest Team 1 member, ensuring safety.

In addition, this paper tests multi-roles and asymmetric scenarios. The attribute design for different roles is shown in Table 3.

Table 3. Character attributes table.

Type	ATK	HP	CD	Attack Range	Field of View	Speed Range	Tag
HT	500	5000	5	220	350	15–25	Team 1/Team 2
LT	300	3000	3	280	350	25–35	Team 1/Team 2

3.3. Game Flow

Figure 7 illustrates the cyclic interaction process between the environment and unmanned systems, which involves the following steps:

- (1) Initialize: Initialize scene and unmanned system properties, establish the connection between the client and server, send information to the server, and start client reception.
- (2) Step: Unmanned systems choose actions to execute at each time step, affecting the environment and various properties of all unmanned systems.
- (3) Monitor: Monitor the environment and various data to determine whether the termination condition is met.
- (4) Game Done: The combat experiment terminates when the number of unmanned systems for either side reaches 0 or reaches the maximum number of combat steps (10,000). If one side achieves victory, record one victory for the winning side. If both sides fail to eliminate each other, then record one draw.
- (5) Reset: Reset the game. Set the step count back to 0, kill all systems, and regenerate two teams of unmanned systems.

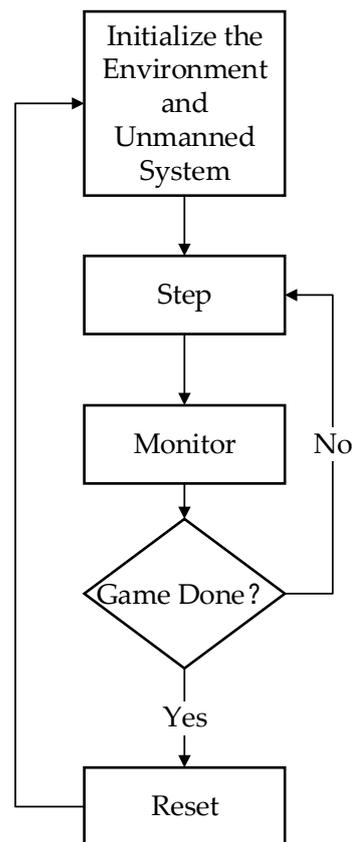


Figure 7. Game flow.

4. Result Analysis

This article adopts the scale of a two-drone formation to investigate scenarios of 2V2 distributed unmanned system conflicts. In order to validate the feasibility and effectiveness of the system in this study, independent decision models are equipped for each unmanned system. The rationality of the decision-making approach is verified through simulation results.

4.1. Real-Time Analysis

The promptness of decision-making during confrontation is a crucial aspect of unmanned system games. This section records the average execution time of a single decision to comprehensively evaluate the system's performance, assessing whether the system can make decisions promptly to meet real-time requirements in the game.

As the initial run may involve some additional initialization work, such as model loading and connection establishment, the time required for the first run far exceeds that of subsequent single decisions. This section calculates the execution time of multiple decision cycles. The average execution time for a single decision is displayed in Figure 8. The outcomes show that the average time for a single decision, excluding the initial run, with the model incorporating all decision rules is 2.5–7.0 milliseconds, which fully satisfies the rapid response requirements in online real-time decision-making.

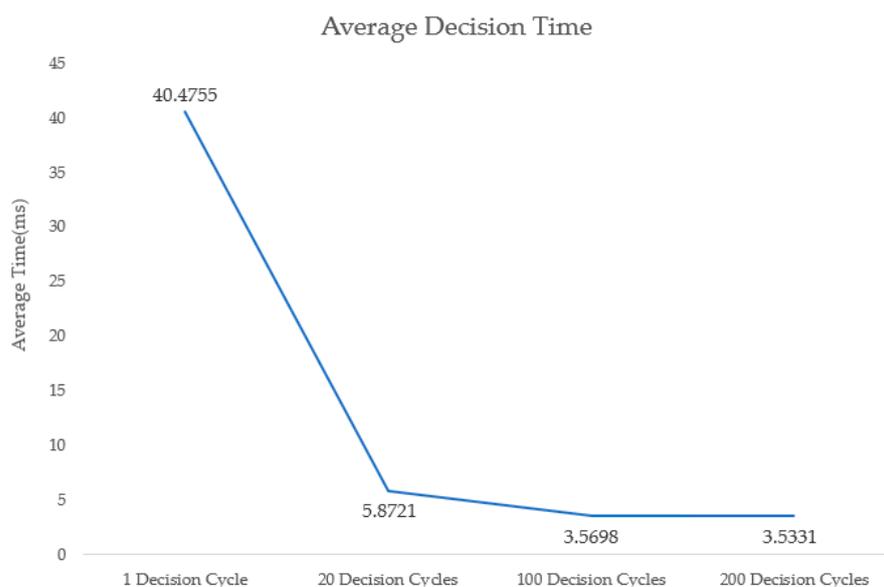


Figure 8. Average response time for single decision.

Furthermore, the changing outcomes of decision-making time align with the growth pattern observed in human decision-making, where the cycle gradually shortens as more events (computational cases) are experienced.

4.2. Decision Analysis

In Area 1, unmanned systems are capable of autonomously adapting their formation while traversing a narrow forest passage. As depicted in Figure 9, the unmanned systems depart from the starting point and sequentially undergo formation changes, transitioning from a vertical to a cross-shaped and finally to a horizontal formation, based on their surroundings. They assess the road width ahead and autonomously adjust their formation relative to the team's position. The internal cognitive computations and decision outcomes during this process are also illustrated in Figure 9. These results demonstrate the unmanned systems' environmental perception and their ability to autonomously adapt their formation based on team position.

Area 2 represents a dense scenario, primarily testing the Cover strategy during the confrontation. In this strategy, unmanned systems calculate distances to obstacles, which are sufficient for block firing, and autonomously choose the nearest cover for evasion, as depicted in the action sequence in Figure 10.

Area 3 is an open and spacious scenario where confrontation primarily tests three strategies.

In the Cover strategy, unmanned systems A and B, based on their type, health status, and the condition of the teammates, choose one member as a cover. In the scenario with a lack of cover, the cover member acts as bait to attract rival fire, while another member serves as the main output. During the confrontation, the roles of the members flexibly adjust according to the changing situation. The decision outputs and action sequences are illustrated in Figure 11.

In the Focus Fire strategy, A and B are arranged in a horizontal formation to maximize frontal confrontation capabilities, as depicted in Figure 12.

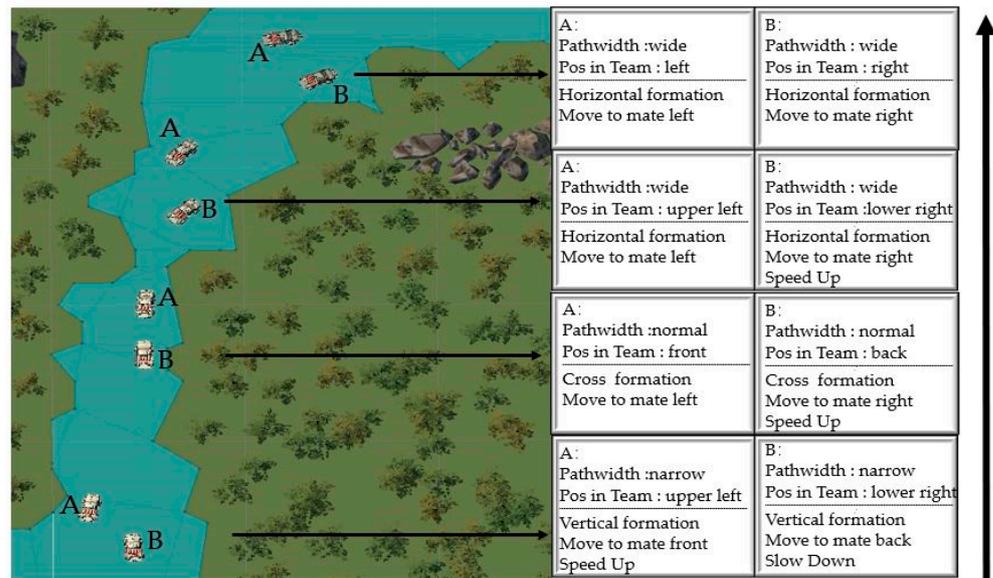


Figure 9. The green parts on the left of Figure represent the forested region of Area 1, while the blue section indicates the navigable area for unmanned systems where formation changes take place. The terrain perception, orientation recognition, and output decisions of the unmanned systems in this area are illustrated on the right side of this figure.

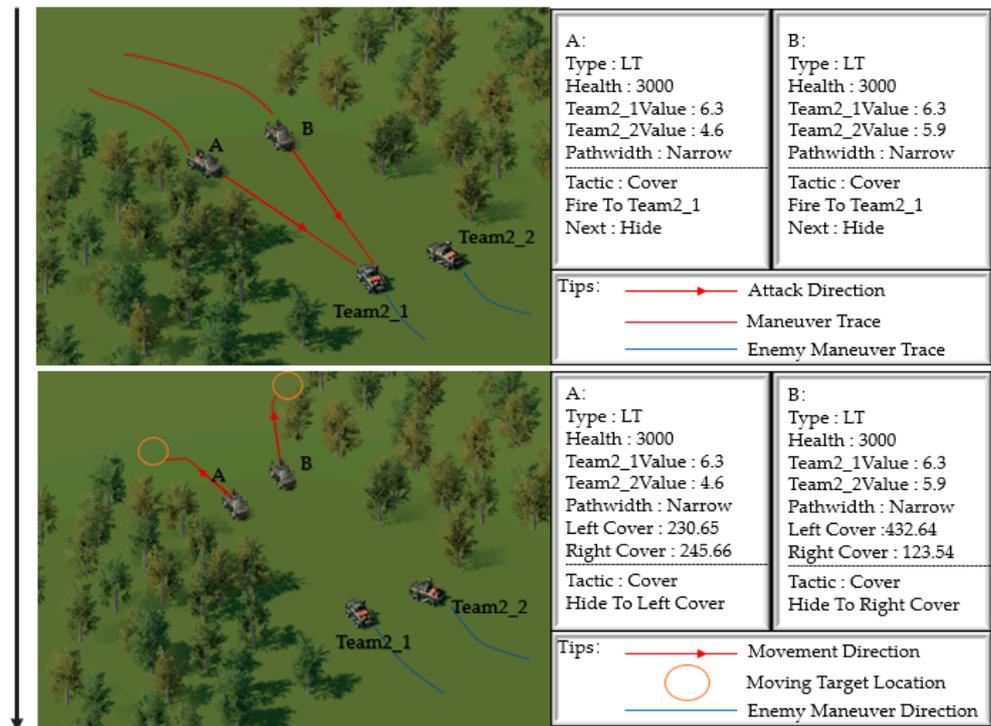


Figure 10. The Soar multi-system deduces, through road-width reasoning, that this location is densely populated with cover. It employs the Cover strategy, where, after an attack, it selects the nearest cover capable of blocking attacks for evasion.

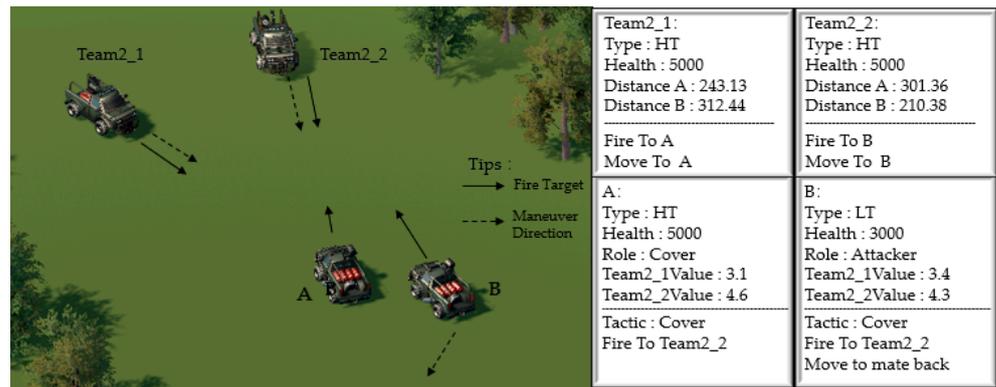
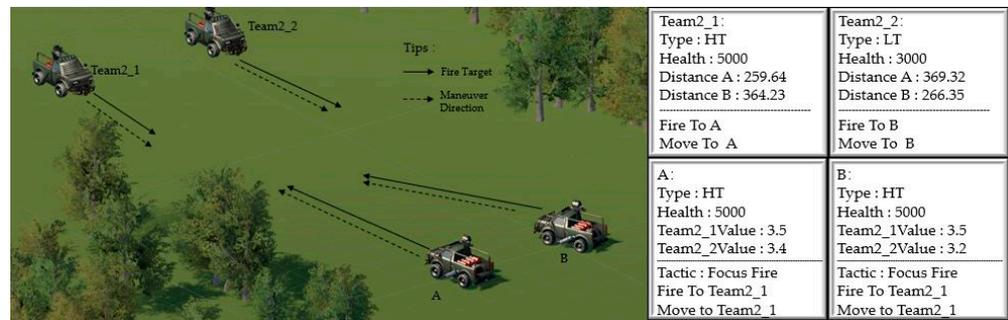
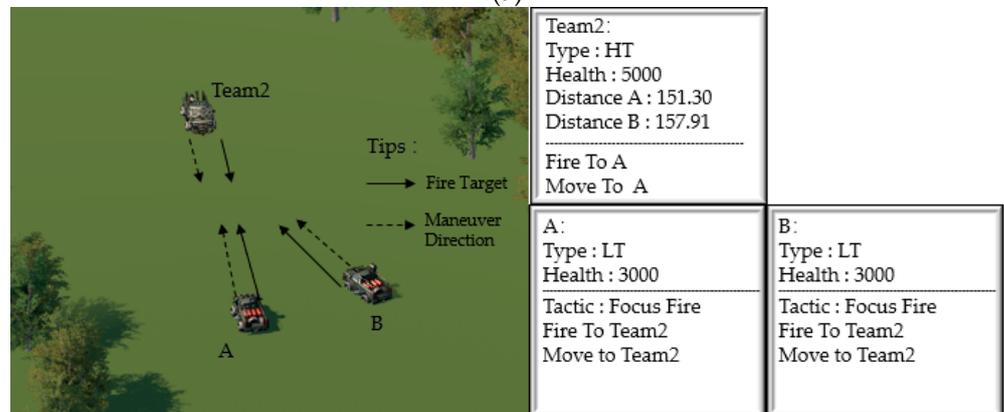


Figure 11. In the Cover strategy, A acts as the cover fire support, protecting B during the attack. After the attack, when entering the cooldown period, B moves behind A.



(a)



(b)

Figure 12. In the Focus Fire strategy, Team 1 members are aligned in a horizontal formation, targeting the highest-value objective. The solid line represents the direction of the attack, while the dashed line indicates the maneuvering direction. (a) Testing 2V2; (b) Testing 2V1.

Kiting is a strategy that involves launching attacks while systematically withdrawing. When the unmanned system’s firepower is ready, it selects the target with the highest value. During the cooldown period, it swiftly withdraws, moving away from the rival towards a safe zone. This tactic aims to fully leverage the advantages of range and speed, ensuring self-preservation while engaging the rival, as illustrated in Figure 13.

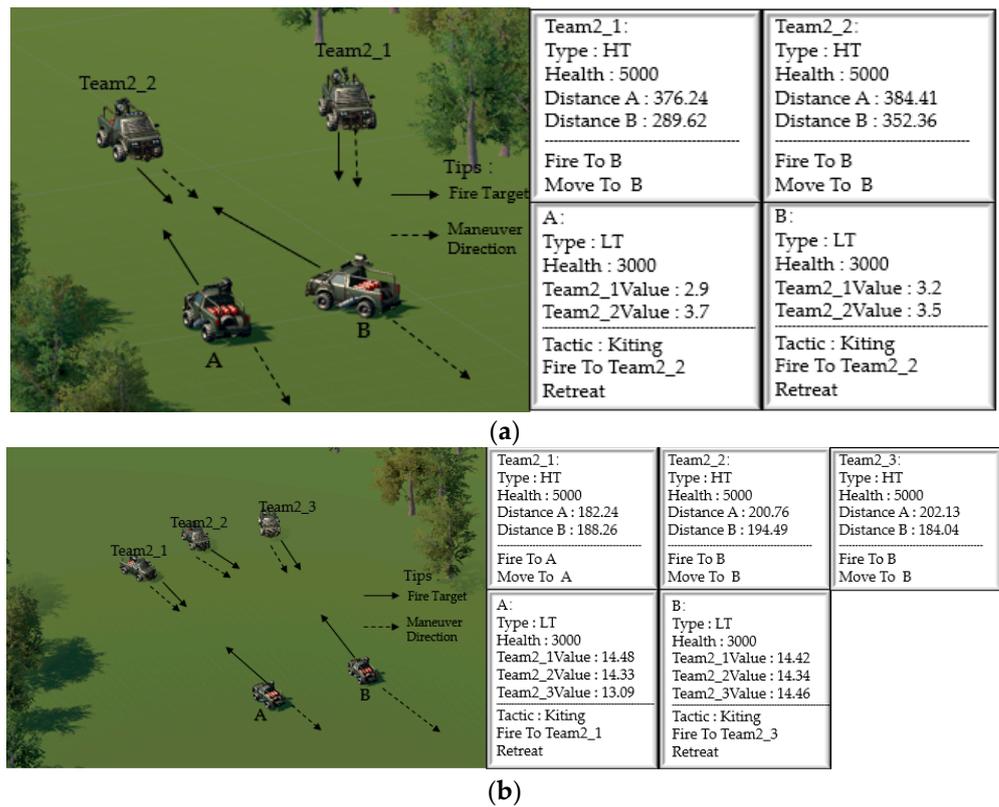


Figure 13. Testing the Kiting strategy. Team 1 members engage in attacks while retreating. The diagram illustrates the rival pursuing Team 1. The solid line represents the direction of the attack, while the dashed line indicates the maneuvering direction. (a) Testing 2V2; (b) Testing 2V3.

4.3. Performance from Past Outcomes

This section presents numerous asymmetric confrontation experiments involving two conflict scenarios. The experimental results were systematically collected and organized and are presented in Table 4.

Table 4. Results of asymmetric confrontation experiments in different scenarios. Team 2’s configuration is represented by a preceding “VS”, while Team 1’s configuration by a succeeding “VS”. The strategies chosen by Team 1 are indicated in parentheses.

Confrontation Scenario	Both Parties’ Configurations and Strategies	Team 1 Wins	Team 2 Wins	Draws	Team 1’s Win Rate
Area 2	2HT VS 2LT (Cover)	36	16	2	66.67%
Area 3	2HT VS 2LT (Kiting)	39	17	1	68.42%
	HTLT VS 2LT (Kiting)	33	18	4	60%
	2HT VS HTLT (Kiting)	28	16	0	63%
	HTLT VS 2HT (Focus Fire)	42	9	0	82.35%%
	HTLT VS 2HT (Cover)	43	0	2	95.56%
	HT VS 2LT (Focus Fire)	25	0	0	100%
	3HT VS 2LT (Kiting)	28	15	7	56%

The experiments demonstrate that the Soar multi-unmanned system can autonomously plan tasks and engage in adversarial games across diverse environments. This validates the effectiveness of the Soar cognitive architecture in decision-making for multi-unmanned system confrontations and provides crucial insights for autonomous decision-making and cooperative operations in complex environments. The success of the collaborative decision-

making model also provides robust support for the future application of unmanned systems in more complex scenarios.

4.4. Performance during the Game

The experimental findings were as follows:

1. In Area 2, when applying the Cover strategy, obstacles not only block rival attacks but sometimes hinder Team 1's attacks as well, resulting in a longer overall testing time. Further refinement of the attack actions is needed, including the addition of criteria to assess whether there are obstacles in the attack direction.
2. In Area 3, when facing the situation of 2HT VS 2LT, the outcome of the confrontation is closely related to the positions of Team 1 and Team 2. When the distance between the two sides is close, the advantage of the Kiting tactic is not fully utilized because Team 1 remains within the rival's attack range for an extended period. This may lead to a decrease in Team 1's win rate.
3. When dealing with the situation of HTLT VS 2HT in Area 3, adopting the Cover strategy has a higher win rate than the Focus Fire strategy. Furthermore, statistics show that during confrontations, the Focus Fire strategy can lead to victories but is also accompanied by mate losses. The probability of losing one mate system constitutes 90% of the total victories. In contrast, when employing the Cover strategy, the loss of one mate system accounts for only 20% of the total victories.
4. In tests with asymmetric quantities, the Focus Fire strategy is employed when Team 1 has a numerical advantage, while the Kiting tactic is chosen in case of a numerical disadvantage. In tests where there is a numerical disadvantage, the frequency of draws increases due to multiple instances of reaching the maximum time steps.
5. There is a correlation between the decision speed of the Soar system and the number of rules. In this paper, the knowledge set was around 200 rules, and, excluding the first decision, the maximum time for a single decision was kept within 7 milliseconds, meeting real-time requirements. However, when dealing with a higher magnitude of rules, it is crucial to focus on testing and controlling decision-making times.

5. Conclusions and Future Work

This paper proposes a method for multi-unmanned system adversarial games based on the Soar cognitive architecture, accomplishing the following tasks:

1. Constructing knowledge of cooperative and adversarial cognitive decision models for multi-unmanned systems by designing Situation Cognition Knowledge, Mission Planning Knowledge, and Action Selection Knowledge to assist unmanned systems in adversarial tasks. Situational Awareness Knowledge assists systems in internal cognition, Mission Planning Knowledge utilizes hierarchical thought to decompose adversarial tasks into subtasks related to strategies and defines the execution actions under each strategy, helping system devise fully hierarchical autonomous task planning, while Action Selection Knowledge assists in selecting appropriate strategies for application.
2. Providing a complex forest simulation environment based on Unity, along with communication interfaces for connecting and testing cognitive decision models. The decision-making outcomes are intuitively displayed in the visualized scenarios. This establishes a foundation for the subsequent development of multi-domain collaborative unmanned systems.
3. The positive performance validates the feasibility and effectiveness of Soar cognitive architecture application to multi-unmanned systems. The system demonstrated the decision-making capabilities of Soar in complex and dynamic environments, showcasing its ability to make appropriate decisions in various complex scenarios. These results confirm the effectiveness of the Soar architecture in collaborative decision-making for multi-unmanned systems.

This study focuses on multi-unmanned systems confrontation scenarios, with decision-making primarily concentrated on the strategic selection of combat objectives. Future research will include more unmanned systems and focus on building more complex adversarial environments. As the number of unmanned systems increases, tactical rules will also expand. The next step will focus on optimizing and incorporating tactical and reinforcement learning rules, altering the selection probability of operators through rewards, and reinforcement learning mechanisms. This system will learn from historical experience and acquire rewards to determine which decision-making approach will achieve the highest reward value.

Author Contributions: Conceptualization, L.D., Y.T. and T.W.; methodology, L.D., Y.T. and T.W.; software, L.D.; validation, L.D. and T.W.; formal analysis, L.D., B.Y. and P.H.; investigation, L.D. and T.W.; resources, Y.T. and T.W.; writing—original draft preparation, L.D. and T.X.; writing—review and editing, L.D., T.W., and T.X.; visualization, L.D., B.Y. and T.W.; supervision, T.W.; project administration, T.W.; funding acquisition, T.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The original contributions presented in the study are included in the article material, further inquiries can be directed to the corresponding author.

Acknowledgments: The authors thank everyone that provided suggestions and support for this paper.

Conflicts of Interest: Author Yong Tang was employed by the Aviation Industry Corporation of China (Chengdu). The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

- Gao, J.; Wang, G.; Gao, L. LSTM-MADDPG multi-agent Cooperative Decision Algorithm based on asynchronous cooperative updating. *J. Jilin Univ. (Engin. Technol. Ed.)* **2022**, *7*, 1–9.
- Vinyals, O.; Ewalds, T.; Bartunov, S.; Georgiev, P.; Vezhnevets, A.S.; Yeo, M.; Makhzani, A.; Küttler, H.; Agapiou, J.; Schrittwieser, J.; et al. StarCraft II: A New Challenge for Reinforcement Learning. *arXiv* **2017**, arXiv:1708.04782.
- Ecoffet, A.; Huizinga, J.; Lehman, J.; Stanley, K.O.; Clune, J. First return, then explore. *Nature* **2021**, *590*, 580–586. [[CrossRef](#)] [[PubMed](#)]
- Silver, D.; Singh, S.; Precup, D.; Sutton, R.S. Reward is enough. *Artif. Intell.* **2021**, *299*, 103535. [[CrossRef](#)]
- Cong, C. Research on Multi-Agent Cooperative Decision Making Method Based on Deep Reinforcement Learning. Master's Thesis, University of Chinese Academy of Sciences, Beijing, China, 2022.
- Shi, D.; Yan, X.; Gong, L.; Zhang, J.; Guan, D.; Wei, M. Reinforcement learning driven multi-agent cooperative combat simulation algorithm for naval battle field. *J. Syst. Simul.* **2023**, *35*, 786–796.
- Zhang, J.; He, Y.; Peng, Y.; Li, G. Path planning of cooperative game based on neural network and artificial potential field. *Acta Aeronaut. Astronaut. Sin.* **2019**, *40*, 228–238.
- Xing, Y.; Liu, H.; Li, B. Research on intelligent evolution of joint fire strike tactics. *J. Ordnance Equip. Eng.* **2021**, *42*, 189–195.
- Xu, J.; Zhu, X. Collaborative decision algorithm based on multi-agent reinforcement learning. *J. Ningxia Norm. Univ.* **2023**, *44*, 71–79.
- Ge, F. Swarm Cooperative Solution Algorithm Based on Chaotic Ants and Its Application. Ph.D. Thesis, Hefei University of Technology, Hefei, China, 2012.
- Song, W.X.; Zhang, F.; Liu, R.T. Mathematical models in research of metapopulation theory. *J. Gansu Agric. Univ.* **2009**, *44*, 133–139.
- Alfonso, R.H.; Pedro, J.T. Effects of diffusion on total biomass in simple metacommunities. *J. Theor. Biol.* **2018**, *3*, 12–24.
- Schmitt, F.; Schulte, A. Mixed-initiative mission planning using planning strategy models in military manned-unmanned teaming missions. In Proceedings of the 2015 IEEE International Conference on Systems, Man, and Cybernetics, Hong Kong, China, 9–12 October 2015; IEEE: Piscataway, NJ, USA, 2015; pp. 1391–1396.
- Yang, J.H.; Kapolka, M.; Chung, T.H. Autonomy balancing in a manned-unmanned teaming (MUT) swarm attack. In *Robot Intelligence Technology and Applications 2012*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 561–569.
- Puranam, P. Human–AI collaborative decision-making as an organization design problem. *J. Org. Des.* **2021**, *10*, 75–80. [[CrossRef](#)]
- Aickelin, U.; Maadi, M.; Khorshidi, H.A. Expert–Machine Collaborative Decision Making: We Need Healthy Competition. *IEEE Intell. Syst.* **2022**, *37*, 28–31. [[CrossRef](#)]

17. Maadi, M.; Khorshidi, H.A.; Aickelin, U. Collaborative Human-ML Decision Making Using Experts' Privileged Information Under Uncertainty. In Proceedings of the AAAI 2021 Fall Symposium on Human Partnership with Medical AI: Design, Operationalization, and Ethics (AAAI-HUMAN 2021), Virtual Event, 4–6 November 2021.
18. Zytek, A.; Liu, D.; Vaithianathan, R.; Veeramachaneni, K. Sibyl: Understanding and Addressing the Usability Challenges of Machine Learning In High-Stakes Decision Making. *IEEE Trans. Visual. Comput. Graph.* **2022**, *28*, 1161–1171. [[CrossRef](#)] [[PubMed](#)]
19. Doshi-Velez, F.; Kim, B. Towards A Rigorous Science of Interpretable Machine Learning. *arXiv* **2017**, arXiv:1702.08608.
20. Laird John, E., III; Robert; Wray, E.; Yongjia, W.; Nate, D.; Andrew, M.N.; Samuel, W.; Marinier, I.I.R.P.; Nicholas, G.; Joseph, X. *The Soar Cognitive Architecture*; The MIT Press: Cambridge, MA, USA, 2015.
21. Laird, J.; Newell, A.; Rosenbloom, P. Soar: An Architecture for General Intelligence. *Artif. Intell.* **1987**, *33*, 1–64. [[CrossRef](#)]
22. Wray, R.E.; Jones, R.M. An Introduction to Soar as an Agent Architecture. In *Cognition and Multi-Agent Interaction: From Cognitive Modeling to Social Simulation*; Sun, R., Ed.; Cambridge Univ. Press: Cambridge, UK, 2005; pp. 53–78.
23. Laird, J.E. Introduction to SOAR. *arXiv* **2022**, arXiv:2205.03854. Available online: <http://arxiv.org/abs/2205.03854> (accessed on 8 May 2022).
24. Laird, J.E. Intelligence, Knowledge & Human-like Intelligence. *J. Artif. Gen. Intell.* **2020**, *11*, 41–44. [[CrossRef](#)]
25. Kennedy, W.G.; De Jong, K.A. Characteristics of Long-term Learning in Soar and Its Application to the Utility Problem. In Proceedings of the 20th International Conference on Machine Learning, Washington, DC, USA, 21–24 August; ICML: Honolulu, HI, USA, 2003; pp. 337–344.
26. Nason, S.; Laird, J.E.; Soar, R.L. Integrating Reinforcement Learning with Soar. *Cogn. Syst. Res.* **2005**, *6*, 51–59. [[CrossRef](#)]
27. Nuxoll, A.M.; Laird, J.E. A Cognitive Model of Episodic Memory Integrated with a General Cognitive Architecture. In Proceedings of the International Conference on Cognitive Modeling, Pittsburgh, PA, USA, 30 July–1 August 2004; ICCM: Warwick, UK, 2004; pp. 220–225.
28. Hanford, S.D. A cognitive robotic system based on the Soar cognitive architecture for mobile robot navigation, search, and mapping missions. In *Dissertations & Theses Gradworks*; The Pennsylvania State University: State College, PA, USA, 2011.
29. Gunetti, P.; Dodd, T.; Thompson, H. Simulation of a Soar-Based Autonomous Mission Management System for Unmanned Aircraft. *J. Aerosp. Comput. Inf. Commun.* **2013**, *10*, 53–70. [[CrossRef](#)]
30. Laird, J.E.; Yager, E.S.; Hucka, M.; Tuck, C.M. Robo-Soar: An integration of external interaction, planning, and learning, using Soar. *IEEE Robot. Auton. Syst.* **1991**, *8*, 113–129. [[CrossRef](#)]
31. Van Dang, C.; Tran, T.T.; Pham, T.X.; Gil, K.-J.; Shin, Y.-B.; Kim, J.-W. Implementation of a Refusable Human-Robot Interaction Task with Humanoid Robot by Connecting Soar and ROS. *J. Korea Robot. Soc.* **2017**, *12*, 55–64. [[CrossRef](#)]
32. Pfeiffer, S.; Angulo, C. Gesture learning and execution in a humanoid robot via dynamic movement primitives. *Pattern Recognit. Lett.* **2015**, *67*, 100–107. [[CrossRef](#)]
33. Wu, T.; Sun, X.; Zhao, S. Application of Soar in the construction of Air defense Decision Behavior Model of Surface Ship CGF. *Command Control Simul.* **2013**, *2*, 108–112.
34. Zhao, Y.; Derbinsky, N.; Wong, L.Y.; Sonnenshein, J.; Kendall, T. Continual and real-time learning for modeling combat identification in a tactical environment. In Proceedings of the NIPS 2018 Workshop on Continual Learning, Montréal, QC, Canada, 3–8 December 2018.
35. Luo, F.; Zhou, Q.; Fuentes, J.; Ding, W.; Gu, C. A Soar-Based Space Exploration Algorithm for Mobile Robots. *Entropy J.* **2022**, *24*, 426. [[CrossRef](#)] [[PubMed](#)]
36. Chen, W.; Wu, H.; Tang, L.; Wang, W. An intrusion prevention system with cognitive function. *J. Henan Univ. Sci. Technol. (Nat. Sci. Ed.)* **2017**, *38*, 49–53+6. [[CrossRef](#)]
37. Czuba, A. Target Detection in Changing Noisy Environment Using Coherent Radar Model Integrated with Soar Cognitive Architecture. In Proceedings of the 2022 IEEE 21st International Conference on Cognitive Informatics & Cognitive Computing (ICCI*CC), Toronto, ON, Canada, 8–10 December 2022; pp. 64–71. [[CrossRef](#)]
38. Miner, A.; Laird, J.E. A Demonstration of Compositional, Hierarchical Interactive Task Learning. *Proc. AAAI Conf. Artif. Intell.* **2022**, *36*, 13203–13205. [[CrossRef](#)]
39. Jones, R.M.; Laird, J.E.; Nielsen, P.E.; Coulter, K.J.; Kenny, P.; Koss, F.V. Automated Intelligent Pilots for Combat Flight Simulation. *AI Mag.* **1999**, *20*, 27–41.
40. Laird, J.E. Toward cognitive robotics. In Proceedings of the SPIE, Orlando, FL, USA, 14–15 April 2009.
41. Wray, R.E.; Laird, J.E.; Nuxoll, A.; Stokes, D.; Kerfoot, A. Synthetic Adversaries for Urban Combat Training. *AI Mag.* **2005**, *26*, 82–92. Available online: <https://search.ebscohost.com/login.aspx?direct=true&db=edsbl&AN=RN176366008&site=eds-live> (accessed on 20 February 2024).
42. Available online: <https://soar.eecs.umich.edu/articles/articles/soar-markup-language-sml/78-sml-quick-start-guide> (accessed on 15 August 2014).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.