Combine Intent Recognition with Behavior Modeling in Teaching Competition Military Simulation Platform

Yi Zhang, Shuilin Li, Chuan Ai, Yong Peng and Kai Xu
College of Systems Engineering, National University of Defense Technology, Changsha 410073, China

Keywords: Simulation and Application, Evolution Behavior Tree, Symbolic Plan Recognition, Decision-Reasoning.

Abstract: Intent recognition refers to obtaining the observations of an agent and then using the observations to reason its current state and to predict its future actions. Behavior modeling, describing the behavior or performance of an agent, is an important research area in intent recognition. However, few studies have combined behavior modeling with intent recognition to investigate its real-world applications. In this paper, we study behavior modeling for intent recognition for cognitive intelligence, aiming to enhance the situational awareness capability of AI and expand its applications in multiple fields. Taking the combat environment and tanks as the research object, based on the behavior tree and SBR recognition algorithm, this paper designs the framework and experiments for behavior modeling and intent recognition. Firstly, uses the evolution behavior tree algorithm to autonomously generate the behavior model adapted to the environment. Secondly uses the SBR algorithm to effectively recognize actions and plan paths of enemy tank to guide self-tank actions in the TankSimV1.20 simulation platform. The results show that the tank survival rate increases by 80% under the guidance of the intent recognition results, and the method in this paper can provide effective guidance for the intent recognition behavior modeling, which has a broad application prospect.

1 INTRODUCTION

Intent recognition, the ability to recognize the activities, plans and goals of other agents, enables the observer to reason about the current state of the recognized agent and predict its future action(Mirsky et al., n.d.). In practical research, intent recognition contains three types: goal recognition, plan recognition, and activity recognition. Among these, activity recognition is the least abstract level of inference, goal recognition is the most abstract, and plan recognition lies somewhere between them.

Behavior modeling, describing the behavior or performance of an identified person, is an important research area in intent recognition. Behavior modeling is mainly divided into cognitive behavior modeling and physical behavior modeling. Physical behavior modeling refers to the direct physical modeling of the external environment. Cognitive behavior modeling refers to simulating the various thinking processes of the recognized agent.

In military simulation, behavior trees are often utilized to guide the Computer-Generated Forces (CGFs) for simulating combat processes(Fu Yanchang, 2019). While Behavior Trees can effectively manage and organize a series of predefined behavior patterns, the CGFs lack the ability to flexibly respond and make independent decisions based on actual situations(Jie Yang, 2021). If the integration of intent recognition and behavior trees can be achieved and applied in military simulation, it would enable CGFs to possess more

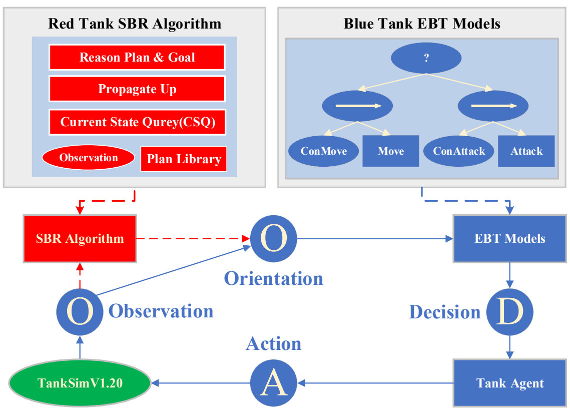

Figure 1: The framework of combine intent recognition with behavior modeling in teaching competition military simulation platform (TCMSP).

Zhang, Y., Li, S., Ai, C., Peng, Y. and Xu, K.
Combine Intent Recognition with Behavior Modeling in Teaching Competition Military Simulation Platform.
DOI: 10.5220/0012869400003758

In *Proceedings of the 14th International Conference on Simulation and Modeling Methodologies, Technologies and Applications (SIMULTECH 2024)*, pages 456-463
ISBN: 978-989-758-708-5; ISSN: 2184-2841

advanced decision-making capabilities, thereby improving the realism and practicality of military simulations. This work seeks to combine intent recognition with behavior modeling in military simulation platform, and thus allow a decision-making agent.

Figure 1 shows the framework of this paper. The circle consisting of four modules formulates a cognition loop(Xu et al., 2019). Specifically, the TankSim V1.20 firstly feed the observation into EBT Models, which is a behavior model that we generate using the Evolutionary Algorithm. The EBT models guides blue tank actions, making them more closely resemble real-world military forces(Jie Yang, 2021). Then the blue tank makes real-time decisions. Finally, the red tank makes decisions based on actions of blue tank using the SBR algorithm(Avrahami-Zilberbrand & Kaminka, 2005).

The rest of this paper is arranged as follows: Section 2 introduces related work. Section 3 introduces the algorithms we used. Section 4 introduces the experiments and results of this work. Section 5 introduces conclusions and future work.

2 RELATED WORK

2.1 Intent Recognition Algorithms

Since the formal definition of planning recognition process was proposed in 1978(Schmidt et al., 1978), scholars have started to study methods for plan recognition. A method of generalized planning recognition was proposed in 1986(Kautz & Allen, 1986), which describes task plan recognition with a planning graph, represents the decomposition of the task with the vertices of the graph, and proposes to use the graph overlay for the solution of the problem, which to some extent laid the foundation for subsequent research. This method, although efficient, assumes that the top-level goal of plan is unique and does not consider the different priori probabilities of different goals.

The Symbolic Plan Recognition (SBR) method was proposed in 2005 (Avrahami-Zilberbrand & Kaminka, 2005), which efficiently implements plan recognition using tagging and back propagating, and can quickly give partial solutions thus applying to multiple aspects, but the efficiency decreases when multiple plans are run concurrently. Further, the authors proposed Utility-based Plan Recognition (UPR)(Avrahami-Zilberbrand & Kaminka, 2007), which can recognize multiple plans in overlapping and interleaved contexts. The SBR family of algorithms runs efficiently and can produce results in each time and is usually used as a frequent choice of recognition method by researchers.

A probabilistic planning recognition algorithm based on planning tree grammar was proposed in 2009(Geib, 2009), which regarded plan recognition as the parsing of the grammar tree, which effectively solved the plan recognition in the case of multiple concurrent plans, but it needs to construct a complete parsing set. In the same period, a Planning Recognition as Planning (PRaP) approach was proposed in 2009(Ramírez & Geffner, 2009), which used planning techniques to solve the goal recognition problem by comparing the marginal cost, which is the difference between the consistency of a given observation and the best plan, between different plans for the same goal through multiple invocations of the AI planning system. This method is limited by the fact that it can only reason about one plan at a time and is computationally expensive.

Since then, attention has been paid to improving the performance of the recognizer. A cost-based goal recognition method that improves the speed of the recognizer compared to PRaP, but is limited to the path planning domain(Masters & Sardina, 2017). A method of sampling the parse space using Monte-Carlo tree search can significantly improves the speed of solution compared to full parse (Kantharaju et al., 2019).

2.2 Behavior Modelling Algorithms

Currently, commonly used behavior modeling methods include, but are not limited to, the behavior tree (BT), the finite state machines (FSM), and the dynamic script (DS).

A rule-based behavior decision-making algorithm for unidirectional two-channels by combining fuzzy inference with a finite state machine was proposed in 2023(WANG Liang et al., 2023). A vehicle-level expected functional safety hazard recognition method based on a model of finite state machine was proposed in 2023(XIONG Lu et al., 2023). Due to the special structural characteristics, finite state machines can only save a finite number of steps of state transfer, so it is difficult for the FSM system to monitor the historical execution flow of the transfer from the initial state to the final state, and vice versa.

Dynamic Scripts (DS) is a reinforcement learning technique based on rule scripts proposed in 2006(Spronck et al., 2006). In 2015, an Evolution Dynamic Script (EDS), which embeds an evolution approach to discovering new rules during DS learning was proposed in 2015(Kop et al., 2015). The representation of rule scripts improves the comprehensibility of the model, but the quality of the rule base greatly affects the quality of the generated model.

As a mathematical model, behavior tree describes the transfer between finite tasks in a modular way, which allows the creation of complex tasks with simple tasks without considering the execution process of the underlying simple tasks(XIAO Zichao, 2020), and is commonly used for the execution of tasks in fields such as computer science, control systems, robotics, and video games.

A behavior tree-based CGF behavior modelling was conducted in 2019(Fu Yanchang, 2019), which decomposed complex mission objectives into a hierarchical structure represented by behavior subtrees, and optimized the behavior tree by introducing machine learning methods to assist the modeling. In 2018, an integrated framework including a case-based reasoning evolution behavior tree, and reinforcement learning to facilitate CGF autonomous behavior modelling was proposed(Zhang et al., 2018).

Based on the behavior tree, a paper designs an autonomous generation framework of CGF decision model, which used an evolution behavior tree algorithm based on static constraints to efficiently generate behavior tree decision model according to expert domain knowledge(Jie Yang, 2021). Since this paper is very inspiring, it will be used as an important paper for our reference.

Most of these scholars are limited to one type of research direction. In fact, it is the merging of behavior modeling and intent recognition research that can advance the development of cognitive intelligence in AI. In this paper, behavior modeling and intent recognition are merged and simulated and experimented in the TSMCP, which can provide more comprehensive and accurate solutions for practical applications and promote the development of related fields.

3 BACKGROUND APPROACHES

This section provides some background on simulation environment, behavior tree and SBR. It includes behavior tree basics and SBR-related representations.

3.1 Simulation Environment

TankSimV1.20 simulation environment is a TCMSP developed by the College of Systems Engineering, National University of Defense Technology in 2020. In this environment, the user can generate tank platoons and assign behavior tree rules to each tank. Under the guidance of the behavior tree rules, the tanks can roam around the map and complete the annihilation mission. Figure 2 shows the simulation environment.

Figure 2: Simulation Environment TankSim V1.20.

The environment is set as follows: based on the TankSimV1.20 simulation environment, the tank platoons of both sides, red and blue, meet each other on a certain plain terrain, which consists of 14*14 squares. Both sides need to carry out annihilation tasks within a certain period. In this scenario, both sides carry a certain number of missiles to attack each other and can open the shield to defend against missiles. Both tanks are equipped with a series of sensors to detect each element on the map. The tank that defeats the opponent or has the most resources remaining in a limited time wins.

3.2 Evolution Behavior Tree

Behavior tree is used to control the actions of tanks, writing different behavior tree rules for tanks can make tanks realize different actions. Tanks with different behavior trees achieve different results in the competition, for example, tanks that only attack will not avoid the missile attack of enemy tank and will suffer heavy losses in the ambush; tanks that only escape will not attack the enemy tanks and will not win the war. Therefore, the tank behavior tree rule determines the victory or defeat of tanks in combat simulation.

Evolution algorithms, inspired by the evolution mechanisms of biological populations in nature, can search for optimal solutions to optimization problems and are known as intelligent methods, highly robust and adaptive, as well as being implicitly parallel and self-learning. Inspired by this paper(Jie Yang, 2021), we used the proposed evolution behavior tree algorithm to generate an agent behavior model, shown in Algorithm 1.

Data: Parameters
Output: Best behavior tree model
for *i* ∈ epoch
    Selection
    Generate New
    Crossover
    Mutation
end
return Behavior Tree Model

Algorithm 1: Evolution Behavior Algorithm.

3.3 SBR Algorithm

Symbolic Plan Recognition (SBR) was proposed in 2005 inspired by automated planning methods, which are more efficient at runtime at the expense of solution completeness and can produce partial results in each time. Therefore, this method is chosen for this paper for intelligent body intent recognition (Avrahami-Zilberbrand & Kaminka, 2005).

The SBR algorithm, shown in Algorithm 2, consists of the following three steps: describing the behavior of the intelligent body using the Plan Library (PL); efficiently matching the observations with the plans in the PL(CurrentStateQuery); and inferring the plan path (PropagateUp).

Data: Observations, PL
Output: Plan Path
for *o* ∈ *Observations*
    CurrentStateQuery()
    PropagateUp()
end
return Plan Path

Algorithm 2: Symbolic Plan Recognition.

4 EXPERIMENTAL STUDY

In the experimental study, train against manually set behavior tree scripts to generate superior behavioral tree rules by EBT algorithm. The generated behavior tree was used to guide the blue decision and the simple behavioral tree was used to guide the red decision. The two are played against each other under the above rules, and the survival rate and the energy remaining are used as the recognition evaluation metrics.

4.1 Behavior Tree

Firstly, based on the specific requirements of TankSimV1.20, we design the specific conditions and action nodes. Then the script was manually written as an adversarial script model. Finally, the results were analysed by evolution behavior tree algorithm.

Set the Base Node of the Behavior Tree
We set the basic nodes of the behavior tree like Figure 3. The meaning of each node in the figure is its English meaning, where the ellipse represents the condition, and the square represents the action.

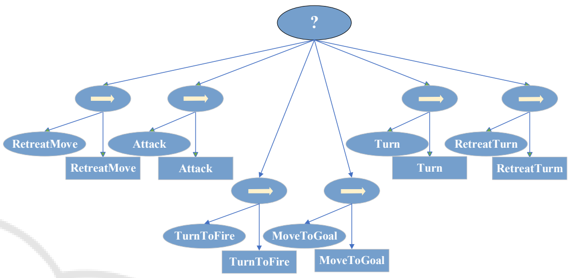

Figure 3: The behavior tree base nodes.

Set Parameters of EBT Algorithm
The population size is set to 100; the rest of the parameters are set as listed in the Table 1.

Table 1: The parameters for the evolution behavior tree.

Parameter	Value
Population Size	100
Evolution Epoch	100
Initial minimum length	14
Initial maximum length	18
Crossover probability	0.8
Mutation probability	0.05
Selection probability	0.1
New unit probability	0.05

Set Confront Script Model
The confront model we used to train the tank shows in Figure 4.

EBT Generates Model
With a crossover probability of 0.6, the EBT performance is optimal, and the population has the highest average fitness value. Under the crossover probability of 0.5 and 0.6, the EBT population converges more slowly, and it starts to converge only in 40 and 50 generations, respectively. The fitness

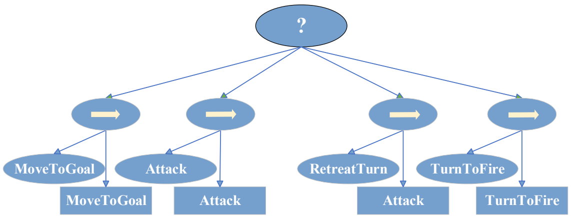

Figure 4: The confront behavior tree nodes.

to more than 0, which indicates that the behavior tree generated by the EBT under this probability is sufficiently strong to cope with the environmental changes. Figure 5 illustrated the fitness curve at different probabilities.

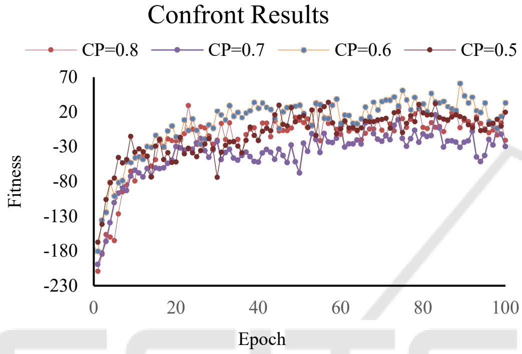

Figure 5: Fitness curve at different probabilities.

The evolution behavior tree corresponding to the highest average fitness value is selected as the final more excellent behavior tree, and the evolution with the highest average fitness value is the 50th generation of evolution, with a fitness value of 32. Figure 6 shows the best behavior tree model of the tank.

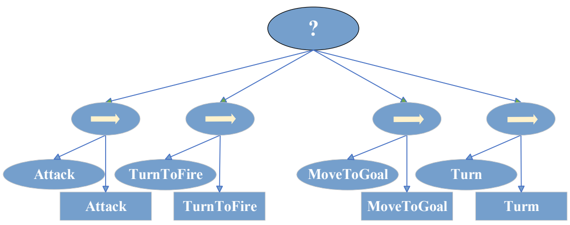

Figure 6: The best behavior tree model of the tank.

4.2 SBR Recognition Effect

This section analyses the recognition effects of two types of behavior trees given to red tank, one is a simple behavior tree with only roaming on the map, and the other is a complex behavior tree with both roaming and attacking functions. The opposing tank has complex behavior tree rules generated by EBT.

4.2.1 Simple BT

The roaming simple behavior tree plan library is shown in Figure 7(Mirsky et al., 2022). From this plan library, the tank can perform include simple actions such as steering to turn on the radar, turning backward, etc. Solid lines represent how each node is decomposed into other nodes and finally to actions. Dashed lines show the order by which the nodes should appear.

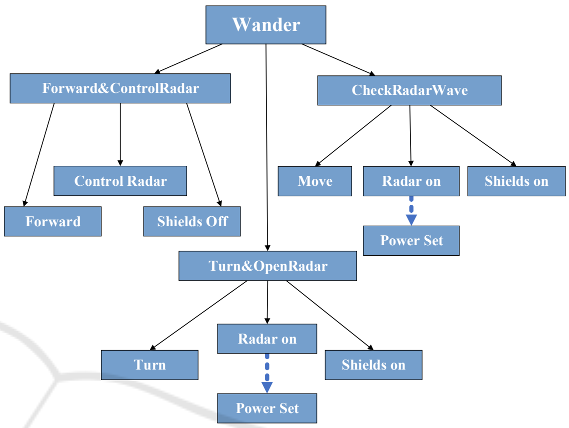

Figure 7: The simple BT example.

Firstly, to facilitate the analysis, only add the red and blue sides a tank respectively. Both sides of the tanks randomly appeared in the map when the simulation beginning. Because the red side has no attack ability, the end of the run shows that the blue tank wins. The recognition effect is as shown Table 2, which shows SBR has a perfect 100% recognition rate for simple behavior trees.

Table 2: The simple BT recognition results.

Observation	Recognition path	Rate
Turn	Turn & Open Radar-Turn	100
Forward	Forward & Control Radar-Forward	100
Shields off	Forward & Control Radar-Shields off	100

4.2.2 Complete BT

To verify the correctness of the SBR algorithm in recognizing complex behavior tree, 10 groups of experiments are carried out. The initial tank location was randomized for each set of experiments and 10 steps of the actions of tank were analyzed for each experiment.

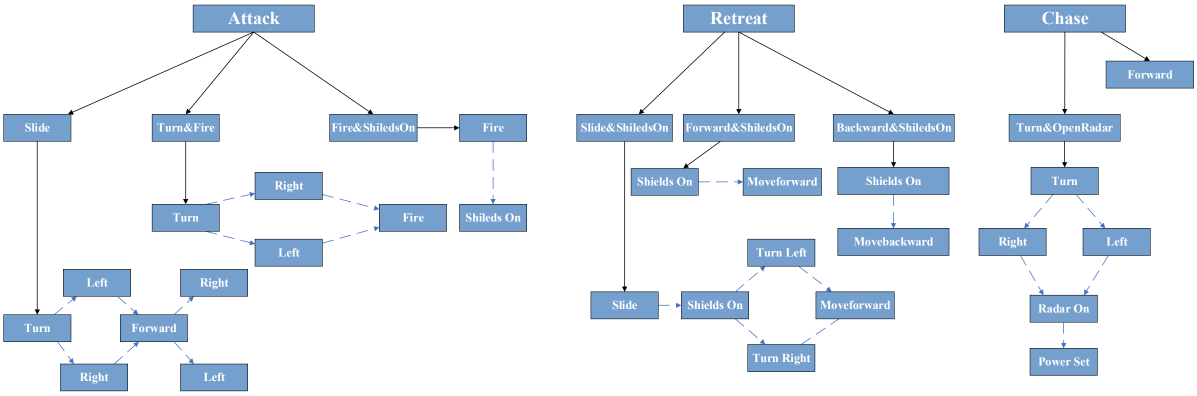

Figure 8: The complete BT example.

Since each step of the tank executes three or two actions of the top-level plan, and there is overlap in actions between the top-level plans, the tank plan library can be set to be fully distinguishable or partially distinguishable. The fully distinguishable case is like a simple behavior tree, with a recognition rate of 100%. Therefore, our experiments focus on analyzing partially distinguishable cases.

Plan library partially distinguishable means the same action isn't described with additional distinction of different top-level plans, e.g., turn right belongs to the same Wander and Chase top-level plans, but no additional information is made for each. In this case, the results of conducting 10 sets of experiments are shown in Table 3.The experimental results show that in the partially distinguishable case, the algorithm recognizes all possible outcomes, and cannot infer the exact action without additional information.

When the recognized action is Fire, since only the Attack top-level plan contains it, the recognition rate is 100%. However, when the recognized action is Shields on, the recognition algorithm is unable to determine exactly which plan it is since all three top-level plans, Attack, Wander and Retreat, contain this node. Nonetheless, if in practical applications, even if there are multiple possible recognition results, these results do not affect the correctness of the final decision, we can still consider that the algorithm performs well at the decision level.

4.3 Combination SBR with BT

In this section, the blue tank has the behavior rules generated by the behavior tree, and the red side has the simple rules. Use SBR to recognize actions of blue and guide decision of red making based on the recognition results. Analyze resource surplus of red and its survival time.

Table 3: Partially distinguishable experiments.

Group	Trial	Recognition	Actually
1	1	Attack/Wander/Retreat	Attack
	2	Attack-Fire	Attack
	3	Wander/Attack/Chase	Wander
	4	Wander	Wander
	…		
5	1	Wander/Chase/Retreat	Wander
	2	Wander	Wander
	4	Attack/Wander/Retreat	Attack
	5	Attack-Fire	Attack
	…		
10	1	Attack/Wander/Retreat	Attack
	2	Attack-Fire	Attack
	3	Wander	Wander
	4	Wander	Wander
	5	Wander/Attack/Chase	Wander
	…		

4.3.1 Compare Survival Time

To compare survival rates, we set up ten sets of experiments, each comparing decision making with/without SBR guidance. With SBR guidance, the red tank could reason the action of blue, and change self-state to protect self from the annihilate of blue, while without SBR the red tank could only make decisions based on the environment.

We set up the initial condition as a small range annihilation battle, and the blue side will annihilate the enemy in a limited step length and analyze the decision-making and survival time of the red side, and the results show in Figure 9.

The experimental results show that with SBR guidance, the red tank easily escapes from the blue tank encirclement, while the survival time is greatly improved. This result suggests that in a combat environment where the self-party is at a disadvantage, it can be assisted in escaping by making a recognition decision through observation of the enemy.

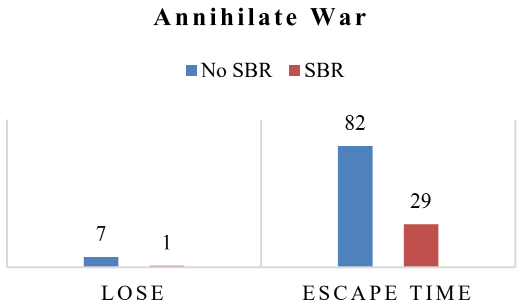

Figure 9: The loss number and escape time results.

4.3.2 Compare Energy Changes

To compare energy changes, we set up 10 sets of experiments, each with 60-time stamp data comparing decisions with/without SBR guidance. Without setting initial conditions for the tanks, we analyze the resource changes (in this case energy changes) of the red tanks as they roam through the environment, and the results show in Figure 10.

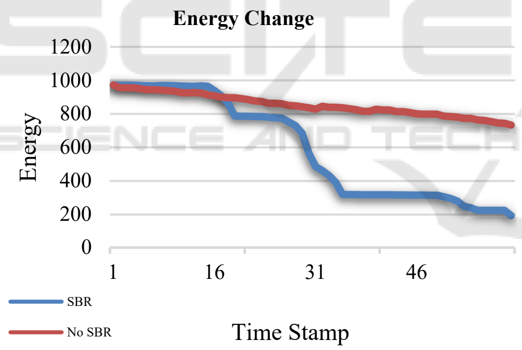

Figure 10: The energy change results.

The experimental results show that with SBR guidance, the red tank consumes less resources. Since it can recognize enemy movements and make decisions, it can reduce the consumption of unnecessary energy to detect the enemy. This result suggests that in a combat environment where the mission of the party is to explore the environment and avoid the enemy, having SBR guidance reduces energy consumption and allows the self-party to explore deeper and find resources.

5 CONCLUSIONS

In this paper, we have combined intent recognition with behavior modeling in TCMSP. Two methods, including behavior modeling of EBT, intent recognition of SBR are used. The framework of application above is simple but inspiring.

Specifically analyze recognition effects of two kinds of behavior rules of the tanks: simple roaming and complex attack. Experiments show that the SBR algorithm can recognize correctly up to 100% in the case of fully distinguishable plan library, while in the partially distinguishable libraries, affected by the representation of plan library and the number of the same actions, the average recognition correctness is 70% (in this paper).

At the same time, in the process of simple rules possessed by the red tank, the results of SBR recognizing the blue are used to assist the decision-making of the red side. Experiments show that under annihilation war, the red side survives longer and escapes easily; under resource war, the red side consumes less energy. These results show that effective recognition of complex behaviors of tank can provide strong support for intelligent decision-making.

Though this work is quite suggestive but still simple, problems are more complex and adversarial are still open in the future. The proposed approach could be enhanced by the following future work:

(1) Obtain the PL of the observed Agent automatically by learning algorithms and LLM. Solving the PL Representation Problem by a priori probability.

(2) Apply modeling and recognition methods to more complex systems like multiagent.

REFERENCES

Avrahami-Zilberbrand, D., & Kaminka, G. (2005, July 30). *Fast and Complete Symbolic Plan Recognition*. International Joint Conference on Artificial Intelligence.

Avrahami-Zilberbrand, D., & Kaminka, G. (2007, July 22). *Incorporating Observer Biases in Keyhole Plan Recognition (Efficiently!)*. AAAI Conference on Artificial Intelligence.

Fu Yanchang. (2019). *Research on CGF Behavior Model Based on Behavior Tree* [M.E., National University of Defense Technology].

Geib, C. (2009). *Delaying Commitment in Plan Recognition Using Combinatory Categorial Grammars*. Proceedings of the 21st International Joint Conference on Artificial Intelligence.

Jie Yang. (2021). *Research on Generation Method of Behavior Tree Decision Model for Computer Generated Forces(CGFs)* [M.E., National University of Defense Technology].

Kantharaju, P., Ontanon, S., & Geib, C. W. (2019). Scaling up CCG-Based Plan Recognition via Monte-Carlo Tree Search. *2019 IEEE Conference on Games (CoG)*, 1–8.

Kautz, H. A., & Allen, J. F. (1986). *Generalized plan recognition*. Proceedings of the Fifth AAAI National Conference on Artificial Intelligence, 32–37.

Kop, R., Toubman, A., Hoogendoorn, M., & Roessingh, J. J. (2015). *Evolutionary Dynamic Scripting: Adaptation of Expert Rule Bases for Serious Games.* Current Approaches in Applied Artificial Intelligence, 53–62.

Masters, P., & Sardina, S. (2017). *Cost-Based Goal Recognition for Path-Planning.* Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems, 750–758.

Mirsky, R., Galun, R., Gal, K., & Kaminka, G. (2022). *Comparing Plan Recognition Algorithms Through Standard Plan Libraries*. Frontiers in Artificial Intelligence, 4.

Mirsky, R., Sarah, K., & Christopher, G. (n.d.). *Introduction to Symbolic Plan and Goal Recognition*.

Ramírez, M., & Geffner, H. (2009, July 11). *Plan Recognition as Planning*. International Joint Conference on Artificial Intelligence.

Schmidt, C. F., Sridharan, N. S., & Goodson, J. L. (1978). *The plan recognition problem: An intersection of psychology and artificial intelligence.* Artificial Intelligence, 11(1), 45–83.

Spronck, P., Ponsen, M., Sprinkhuizen-Kuyper, I., & Postma, E. (2006). *Adaptive game AI with dynamic scripting.* Machine Learning, 63(3), 217–248.

WANG Liang, SU Dong-xu, YANG Xing-long, MA Wen-feng, & WANG Zi-jun. (2023). *Behavioral Decision Based on Fuzzy Inference and Finite State Machine*. Value Engineering, 42(18), 133–135.

XIAO Zichao. (2020). *Application Research of Behavior Tree in Self-driving Behavior Planning Strategy* [M.E., Lanzhou University].

XIONG Lu, JIA Tong, CHEN Junyi, XING Xingyu, & LI Bo. (2023). *Hazard Identification Method for Safety of the Intended Functionality Based on Finite State Machine*. JOURNAL OF TONGJI UNIVERSITY(NATURAL SCIENCE), 51(4), 616–622.

Xu, K., Zeng, Y., Zhang, Q., Yin, Q., Sun, L., & Xiao, K. (2019). *Online probabilistic goal recognition and its application in dynamic shortest-path local network interdiction.* Engineering Applications of Artificial Intelligence, 85, 57–71.

Zhang, Q., Yao, J., Yin, Q., & Zha, Y. (2018). *Learning Behavior Trees for Autonomous Agents with Hybrid Constraints Evolution.* Applied Sciences, 8(7), 1077.