IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 28 June 2022, accepted 7 July 2022, date of publication 14 July 2022, date of current version 25 July 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3190972

APPLIED RESEARCH

Dynamic Multitarget Assignment Based on Deep Reinforcement Learning

YIFEI WU, YONGLIN LEI, ZHI ZHU, XIAOCHEN YANG, AND QUN LI
College of Systems Engineering, National University of Defense Technology (NUDT), Changsha 410073, China
Corresponding author: Yonglin Lei (yllei@nudt.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 62003359.

ABSTRACT Dynamic multi-target assignment is a key technology that needs to be supported in order to improve the strike effectiveness during the coordinated attack of the missile swarm, and it is of great significance for improving the intelligence level of the new generation of strike weapon groups. Changes in ballistic trajectory during the penetration of multi-warhead missiles may cause the original target assignment scheme to no longer be optimal. Therefore, reassigning targets based on the real-time position of the warhead plays an important role in improving the effectiveness of the strike. In this paper, the dynamic multi-target assignment decision modeling method combining combat simulation and deep reinforcement learning was discussed, and an intelligent decision-making training framework for multi-target assignment was designed based on deep reinforcement learning. In conjunction with the typical combat cases, the warhead combat process was also divided into the penetration phase and the multi-target assignment phase, the model framework and reward function against the multi-target assignment of the missile were devised, and the SAC algorithm was employed to conduct application research on intelligent decision modeling for multi-target assignment. Preliminary test results suggest that the intelligent decision-making model based on deep reinforcement learning provides better combat effects than the traditional decision model based on knowledge engineering.

INDEX TERMS Deep reinforcement learning, combat simulation, intelligent decision-making, multi-target assignment.

I. INTRODUCTION

Target assignment is an activity in which mission planners formulate the optimal weapon attack plan according to the target strike mission received in the war planning stage or during the combat process. Dynamic target assignment is an online decision-making activity that adjusts the strike assignment plan in real-time according to changes in the battlefield situation during the flight of the strike weapon group. Typical examples are the dynamic multi-target assignment problem of cruise missile swarm cooperative attack, the dynamic multi-target re-assignment problem of ballistic missile multi-warhead cooperative attack, etc. Its attack mode has expanded from a single attack and defense operation to a many-to-many group cooperative confrontation and game. A coordinated multi-projectile attack can make full use of the scattered combat resources and information sharing, effectively improving the striking and penetrating capabilities [1].

In the problem of dynamic multi-target assignment of multi-warhead, their respective strike targets and flight trajectories are usually determined in advance. However, in the case of interception by the enemy's multiple defense systems, multiple maneuvering penetrations are required, which will make it difficult for the original assignment scheme to achieve the expected effect, while reassigning multiple targets after penetration can improve the combat effect. The penetration decision and target re-assignment of multi-warhead in the combat process directly determine the systemization capabilities, which is one of the key technologies for dynamic multi-target assignment.

The traditional method is mainly based on the combat model of the offensive and defensive sides, and makes target assignment by using mathematical programming. For

The associate editor coordinating the review of this manuscript and approving it for publication was Jesus Felez.

example, the relative movement information of the projectile and target can be used to evaluate the difficulty of guidance and attack performance, while the threat degree of the target can be evaluated on the basis of its own value and movement characteristics [2], [3], and then assigns the target. Shukan Liu *et al.* [4] comprehensively made use of the expert systems and neural networks to construct a basic model of behavioral decision-making, and then developed an intelligent command system to optimize target assignment. Another typical method is to convert the assignment problem into a mathematical programming problem, which then can be solved by the enumeration method, the branch and bound method, or integer programming [5], [6]. Nonetheless, with the increase of the scale of attack and defense, the complexity of optimization will increase sharply, resulting in an exponential increase in the computational time [7]. In consequence, the intelligent optimization methods that are flexible, strongly adaptive, and has relatively simple calculation have great advantages in solving complex multi-target decision-making and assignment, which are typically represented by genetic algorithm and particle swarm optimization (PSO) algorithm [8]. The PSO algorithm adopts the memory and learning of the individual movement position and the overall optimal position in the swarm to move in the optimal direction in the solution space, which, compared with the genetic algorithm, has higher computational efficiency. However, it lacks the fineness and global search capability [9], [10].

The highly dynamic cluster attack and defense puts forward extremely high demands for the optimality and effectiveness of decision-making, of which its complex and changeable attack and defense situation requires multiple online decision-makings and assignments. Its characteristics are that the state space is based on the original target position distribution and weapon types, adding state variables such as weapon group position, speed, remaining fuel, battlefield uncertainty, and the complexity of problem solving sharply increased. However, the above-mentioned optimization methods have their shortcomings in computational efficiency, global optimality, and inheritance of multiple decisions. Deep reinforcement learning combines deep learning with perception ability and reinforcement learning with decision-making ability, which can well solve decision-making problems under complex conditions and has achieved remarkable results in the industrial and military fields [11]–[13].

This paper explores the use of deep reinforcement learning methods to solve, the missile platform is intelligently trained based on a general weapon equipment combat simulation system (WESS) [14] and deep reinforcement learning to enable intelligent multi-target decision-making and assignment of the missile, including the construction of a deep reinforcement learning training framework based on combat simulation, the design of discretized action space, state space and reward function, and the application research of the SAC algorithm.

II. RELATED WORK

The weapon target assignment (WTA) studies how to assign weapon units to strike enemy targets to achieve the best strike effect. Currently, the research on WTA problem is mainly focused on the model and algorithm, that is, how to establish the optimal model, propose a reasonable solution algorithm, and generate a weapon target allocation scheme [15].

And its problems can be divided according to static WTA and dynamic WTA. For static WTA, the parameters for weapon and target are known, and the defender can analyze the countermeasures in the light of the weapon type and prediction point of the attacker, and gives the optimal assignment for the defense target, which is the main focus of the current research on WTA problems. For example, in article [16], the previous model research was analyzed and made a systematic summary of the WTA problem, based on this, the basic model of the WTA problem was established, in article [17], the concept of value into the model was introduced creatively, enriching the WTA model. In article [18], a firepower assignment model was constructed, of which its overall objectives are the combat efficiency and the cost ratio of missile weapons and is constrained by the target destruction. The genetic algorithm was also utilized to solve the firepower assignment problem. Meanwhile, in article [19], an improved multi-target particle swarm optimization algorithm was designed to optimize the assignment, and the particles were updated with improved learning factors and inertia weights, which enabled results with higher accuracy than the general Pareto front solution to be obtained. In the rapidly changing battlefield environment, research on static WTA provides a certain reference value, but it is not applicable. Therefore, considering the practical problems, research on dynamic WTA which are based on the static WTA model and focus on possible random events in the assignment process and processing them in time, enter the picture. In article [20], multi-stage weapon-target assignment was studied and the changes in state of each specific time period were analyzed in detail. Meanwhile, in article [21], the realistic effect was improved by leveraging the dynamic WTA method based on Markov decision process optimization in conjunction with dynamic assignment strategy and static WTA model. In article [22], a dynamic WTA model of tank warfare under the specific background of tank warfare was established, and it has been solved based on the improved model. And in article [23], when establishing the dynamic WTA model, the idea of dynamic programming is used to improve the arbitrary time algorithm by redefining the termination conditions, and the model is solved, which can reasonably ensure the timeliness and effectiveness of the allocation.

The process of solving the WTA problem not only needs to complete the corresponding model establishment, but also needs to select an appropriate algorithm to solve it. Intelligent algorithms are inspired by human beings from natural phenomena or processes, and create new solutions to problems by

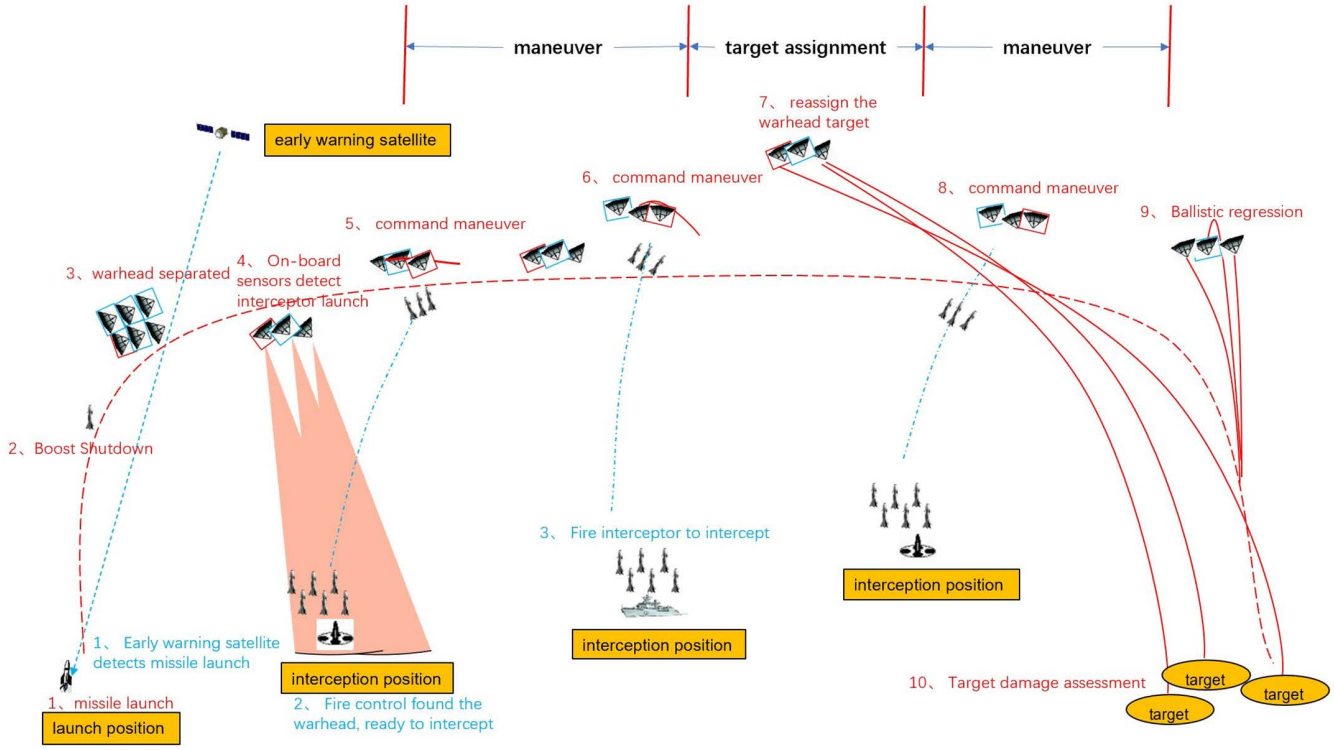

FIGURE 1. Multi-target assignment operational process.

simulating these natural phenomena or processes. At present, the methods of solving the WTA problem through intelligent algorithms include genetic algorithm [24], artificial fish swarm algorithm [25], particle swarm algorithm [26], neural network [27], game theory [28], Hungarian algorithm [29], new Non-dominated sorting algorithm [30] and so on. This paper explores the use of the deep reinforcement learning algorithm SAC to solve the problem of multi-target allocation. In the third part, combined with the typical problem, a decision-making modeling method for multi-target allocation based on deep reinforcement learning is proposed, including the modeling framework and training method. In the fourth part, the application of the SAC algorithm in target assignment is studied, and the model input and output and reward function are designed. In the fifth part, the effectiveness of deep reinforcement learning in the application of multi-target assignment is demonstrated by comparing the experimental results of rule-based and SAC-based algorithms.

III. METHOD FOR MULTI-TARGET ASSIGNMENT DECISION MODELING BASED ON DEEP REINFORCEMENT LEARNING

A. PROBLEM DESCRIPTION

Consider the problem of multi-warhead penetration and striking multi-target. Given that the red team has N warheads, the blue team has M important targets and assigned n interception positions on the way of the red team's warheads. After N warheads are launched by the red team, they will fly to M important targets according to the predetermined trajectory. Each interception position of the blue team will launch m interceptors to intercept each warhead of the red team. When the red team's warhead detects the interceptors at a certain distance, it will maneuver. After a total of n^*m maneuvering and penetrations, the red team's missile platform will reassign targets according to the information of warhead state. The applicability of the intelligent agent can be increased by changing the position of the blue team's interception position.

In the process of hitting targets with multi-warhead, it will go through the stages of maneuvering penetration and target re-assignment. The penetration decision includes whether to maneuver, that is, the timing of maneuvering; the maneuvering duration, that is, the distance of maneuvering and the maneuvering direction. The target, whether to maneuver, the maneuvering duration, and the maneuvering direction can be changed based on information of the real-time state. Then, finally, the effectiveness of the experiment is analyzed based on the hit of each target [31]. Since the missile formation may be subject to one or more interceptions, when encountering the interception of the enemy's defense system, the formation members can obtain the information of he intercepted missile and damage target according to the coordinated data transmission. Defense strategy and guidance instructions are selected to avoid enemy interception and complete the attack on the target under the condition of the least fuel consumption.

B. CONCEPTUAL FRAMEWORK OF REINFORCEMENT LEARNING MODELING FOR INTELLIGENT DECISION-MAKING

Missile penetration and attack are different from the common game simulation, unmanned vehicle driving, and other operations, and they rely on much manpower and funds for research and experiment. In addition, since the missile penetration object is uncontrollable, it is impossible to complete an actual experiment. Reinforcement learning relies on a continuous trial and error method to find the optimal strategy, which requires a large number of database samples to train the intelligent agent and is unacceptable for carrying out actual missile experiments. Therefore, it is of great importance to build a realistic battlefield environment based on combat simulation technology to simulate the offensive and defensive confrontation of missiles, thereby supporting the learning, training, and simulation of the intelligent penetration decision-making algorithm in terms of environment.

The modeling framework of autonomous decision-making for intelligent equipment based on combat simulation and reinforcement learning is shown in the figure, which is divided into a weapon equipment combat behavior model and a weapon equipment simulation model. The weapon equipment simulation model is responsible for generating the sample database required for reinforcement learning, while the weapon equipment combat behavior model is responsible for the decision-making of the weapon equipment in the simulation operation. In the behavior model, the decision on combat behavior is implemented by a python script, the conventional action decision by rule setting, and the intelligent decision-making action by calling the intelligent decision-making network to output the decision action.

Meanwhile, training intelligent decision-making models based on deep reinforcement learning methods requires the help of combat simulation operations to generate sample databases. Off-policy algorithms allow combat simulations to be run in parallel, and the training samples for each episode are written in parallel into the sample database for each round. The off-policy reinforcement learning training algorithm incorporates the generated sample database into the replay-buffer for continuous sampling and training, and updates the intelligent decision-making network. The updated intelligent decision-making network will be loaded during the subsequent combat simulation operation to reflect the decision-making loop and affect the combat decision-making behavior.

C. METHOD FOR INTELLIGENT DECISION-MAKING NETWORK TRAINING

Due to the complexity of combat simulation computation, reinforcement learning training directly based on randomly initialized neural networks is often difficult to quickly converge, or the number of combat simulation samples required for convergence is unacceptable. To this end, the pre-training may be carried out based on imitation learning, the traditional knowledge-based methods (such as production rules) are made full use for decision modeling, the rapid parallel experiment traversal of the entire scenario set is carried out by designing different decision rule schemes, from which the combat decision rule scheme with ideal combat effect is selected, and imitation learning is performed based on its corresponding round sample database. The intelligent decision-making network obtained through imitation learning can provide a better combat effect in the training scenario, based on which a step-by-step reinforcement learning training can be conducted to quickly achieve the purpose of network convergence. In short, the training and application process of the intelligent maneuvering penetration algorithm is mainly divided into the pre-training phase, iterative training phase, and intelligent testing phase, with each phase following the previous one, as shown in the figure.

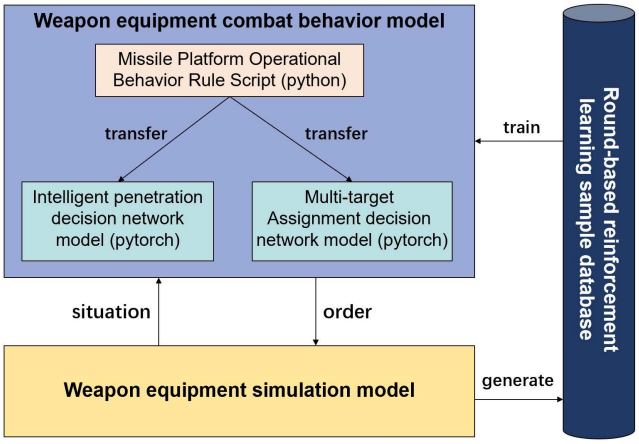

FIGURE 2. Modeling framework of intelligent decision-making for the missile based on deep reinforcement learning.

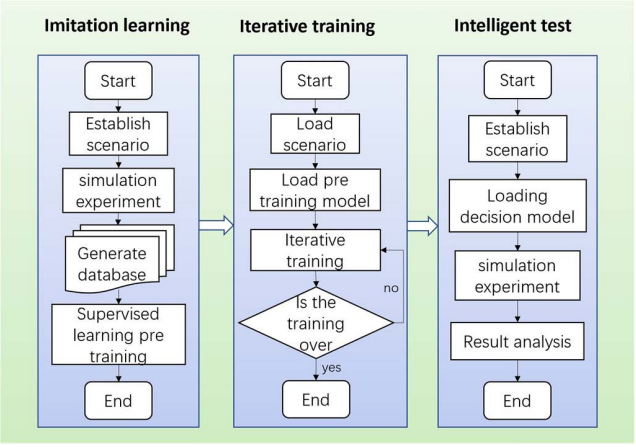

FIGURE 3. Flow chart of intelligent maneuver training of the missile.

In the pre-training phase, the multi-task scenario is firstly designed based on the scenario editor to generate scenario files. Then, the simulation experiments are conducted in

parallel to obtain the experimental database. Finally, the pre-training model is obtained by pre-training through supervised learning.

The output strategy of the decision network is learned and improved during the continuous interaction and iteration with the environment, and hence, it is imperative to conduct iterative training experiments. Based on the pre-training, iterative training must first load the pre-training model for initialization, and then build an empirical data pool of the SAC algorithm. After the iterative training starts, a batch of round data is extracted from the experience data pool for training, and the sample data generated by the iterative training and the environment must also be stored in the experience data pool for the SAC algorithm to learn and update. The results are recorded upon the completion of and before exiting the training. The training is completed on the condition that the multi-target assignment of the missile can achieve satisfactory combat results in all multi-mission scenarios.

After the iterative training, the intelligence test of the intelligent maneuvering penetration decision-making algorithm must be performed. In this phase, firstly, new task scenarios are edited and designed based on the scenario editor and generate test scenario files. Then, the network model based on the iterative training is generated, and the initialization parameters of the penetration decision network are loaded through the behavior script of the missile model. Finally, the simulation experiment is run in parallel, and the missile penetration data is collected to analyze the results.

IV. FRAMEWORK DESIGN OF MULTI-TARGET ASSIGNMENT DECISION MODEL OF THE MISSILE

A. MULTI-TARGET ASSIGNMENT BASED ON SAC

In a bid to cope with the high sampling complexity and convergence vulnerability of reinforcement learning, the SAC algorithm based on the Actor-critic (AC) offline strategy of the maximum entropy framework is mainly used in continuous control tasks, which adds the maximum entropy to the reward, encouraging it to explore all possible optimal paths, and updates the policy algorithm using the off-policy [32]. Meanwhile, SAC-Discrete accounts for some algorithm-level changes that need to be taken into account in the application of the SAC algorithm in discrete actions. For example, the architectural output of policy is no longer a Gaussian distribution in continuous control, but a discrete distribution of n actions. Overall, SAC-Discrete is better suited for multi-warhead penetration and target assignment decision-making, and as a result, the SAC-Discrete algorithm is employed to train the intelligent penetration network. The training goal of conventional reinforcement learning is to maximize the cumulative action reward value and the action state value function $Q_\pi(s, a)$. The action a when Q_π is the largest can be directly selected as the strategy, that is, the optimal strategy is expressed as:

$$\pi^* = \underset{\pi}{\operatorname{argmax}}\, E_{(s_t,a_t)\sim\rho_t}[\sum\nolimits_t R(s_t, a_t)] \quad (1)$$

In the SAC algorithm, the target function is composed of reward and α-weighted policy entropy H, and the introduction of entropy value makes the policy more random.

$$\pi^* = \underset{\pi}{\operatorname{argmax}}\, E_{(s_t,a_t)\sim\rho_t}[\underbrace{\sum\nolimits_t R(s_t, a_t)}_{reward} + \underbrace{\alpha H(\pi(\cdot|s_t))}_{entropy}] \quad (2)$$

In the multi-target assignment decision, different multi-target assignment results lead to different combat outcomes. As the combat effectiveness is associated only with the information of the current and future state, but not with that of the past, the multi-warhead penetration and multi-target assignment decision conforms to the Markov decision process. According to the requirements of the SAC method for deep reinforcement learning, it is necessary to build an intelligent assignment model according to the actual combat task and design the state and action space and reward function. The intelligent multi-target assignment process based on the SAC algorithm is shown in the figure below.

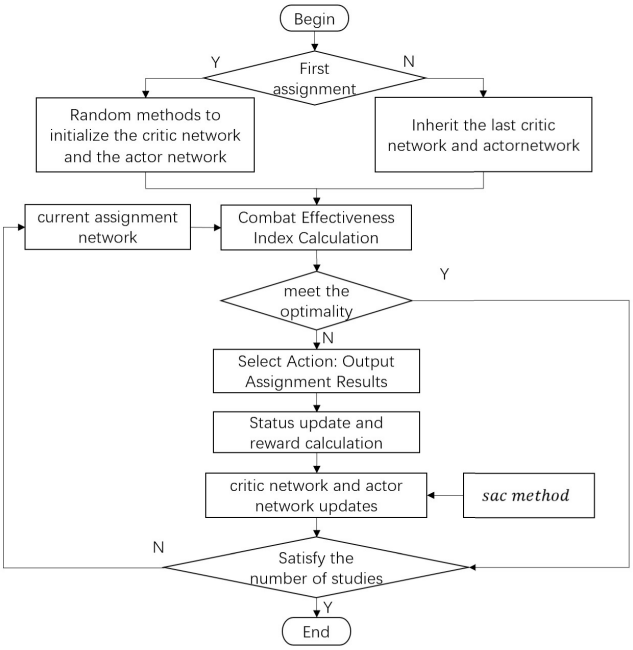

FIGURE 4. Iterative flowchart of intelligent decision-making based on the SAC algorithm.

B. DESIGN OF NETWORK MODEL

1) PENETRATION DECISION MODEL

The description of the missile state space should attach importance to the state variables that affect the final penetration effect of the offensive and defensive sides according to the actual confrontation process between the missile and the enemy's interception system, and form variables directly related to the action space and the reward function, which serve as inputs of the network, by processing the indirectly related information of the state space. The state space of the red team's attack warhead is described

TABLE 1. List of warhead state space variables.

State variable	Name of variable	Dimension	Pre-processing
Remaining fuel of warhead	f_{remain}	1	Remaining fuel/Total fuel
Distance from target point	d_{wt}	1	Distance/Earth radius
Relative position to interceptor	$\Delta_x, \Delta_y, \Delta_z$	3	Relative position vector /Maximum detection range of warhead
Relative velocity with interceptor	v_x, v_y, v_z	3	Relative velocity component /Maximum flight speed of interceptor
Angle between velocity and line of sight	q_{wi}	1	Angle/ π
Angle between interceptor velocity and line of sight	q_{iw}	1	Angle/ π

TABLE 2. List of warhead action space variables.

Action variable	Type of variable	Value range	Description
Whether to maneuver	bool	0,1	0 represents not maneuvering, 1 represents maneuvering
Maneuver time	int	1,…,6	Warhead fuel consumption are considered when the warhead maneuver is 1

as s= {*RemainingFuel*, Δ_x, Δ_y, Δ_z, d_{wt}, v_x, v_y, v_z, q_{wi}, q_{iw}, where *RemainingFuel* represents the fuel quantity of warhead, (Δ_x, Δ_y, Δ_z) represents the relative position vector of the interceptor and the warhead, d_{wt} represents the distance between the warhead and the target point, (v_x, v_y, v_z) represents the relative velocity vector of the interceptor and the warhead, q_{wi} represents the angle between the direction of the speed of warhead and the line of sight, and q_{iw} represents the angle between the direction of the speed of interceptor and the line of sight. The dimensionless processing of each variable in the missile state space is shown in Table 1.

The reinforcement learning action space should be designed in combination with the maneuver penetration task. In this paper, the warhead employed a large overload maneuver to complete penetration. When approaching the interceptor, it decides whether to maneuver and the duration of maneuver. The unit of maneuver time is the step length in the simulation operation. After comprehensively considering the maneuvering distance and fuel consumption of the warhead, the value range of the maneuver time is set to 1,2,…,6. Whether to maneuver is represented by 0 and 1. Therefore, the designed action space is shown in Table 2

TABLE 3. Multi-target assignment network input.

State variable	Symbol	Ranges	Dimension
Distance between each warhead and each target point	d_{xy}	0~1200 km	M*N
Deflection angle between each warhead and each target point	α_x	-90°~90°	M*N
Pitch angle between each warhead and each target point	β_x	-90°~90°	M*N
Speed of each warhead	V_n	0~+∞	N*1
Remaining fuel of each warhead	f_{remain}	0~60 kg	N*1

2) MULTI-TARGET ASSIGNMENT DECISION MODEL

For the strike of N warheads (N is a positive integer greater than 1) on M target points (M is a positive integer greater than 1), as shown in the table above, the network input is the input of the overall environment variable, and the sum is (3M +2)*N-dimensional state input. Here, the distance determines whether the warhead will reach the target point, the deflection angle and pitch angle determine whether an adjustment in direction is required, the speed determines whether the adjustment can be made, and the amount of remaining fuel gives consideration to how far to make target assignments. When the remaining fuel is insufficient, the maneuver cannot be performed and the mission will fail.

The output of the network is the multi-target assignment result, and the representation of the assignment result may be in the form of a matrix, a vector, or a numerical number. Here, for ease of presentation, numerical numbers are used to represent the assignment results. If N warheads hit N targets, the assignment result can be expressed as a tuple "$x_1x_2 \ldots x_i \ldots x_n$" representing the i-th warhead hitting the x_i-th target. In this experimental scenario, six warheads are set to hit six targets, that is, both N and M are six.

C. DESIGN OF REWARD FUNCTION

Reinforcement learning is the learning of a mapping from situations to actions so as to maximizes a scalar reward or reinforcement signal. The learner does not need to be directly told which actions totake, as in most forms of machine learning, but instead must discover which actions yield the most reward by trying them. Informally, the goal of an agent is to maximize the total reward it receives. This means the agent's objective is to maximize the reward that it receives over the long run but not the current gain [33]. In general, the reward design is jointly determined by state S, action A, and the next state S'. Mathematically, it can be expressed as the following formula.

$$R : S \times A \times S' \rightarrow R \tag{3}$$

The characteristic of multi-target assignment lies in that the quality of assignment is only known after an experiment has been completed, that is, the agent can be rewarded only in the last state and only one valid database need to be obtained from an episode. Therefore, the design

of the reward function mainly considers the mission task completion reward, including whether the warhead lands, the amount of the warhead misses the target, and whether it hits the target. The reward value is set according to the completion of the final mission. If the target is successfully hit, the reward value in [40] and [50] interval will be given according to the distance of the warhead and the target. If it fails to hit the target, the reward value in the range of [5] and [10] will be given according to the distance of the warhead from the target. The reward is calculated as follows (4), as shown at the bottom of the page, where, r_{kill} represents the killing radius of the warhead, and its value range is [1,100] kilometer. $d_{warhead}$ represents the distance the warhead deviates from the target, and its value range is [0, $+\infty$]. When $d_{warhead}$ is less than r_{kill}, the warhead is considered to be able to damage the target. The factor $r_{kill}/d_{warhead}$ and $(1-d_{warhead}/r_{kill})$ is set to make R is negatively correlated with $d_{warhead}$, and the two factors' value range is [0,1]. P represents the number of missed targets among N warheads, and $0 \leq P \leq N$ (P is a positive integer).

V. EXPERIMENTAL PROGRESS

A. IMITATION LEARNING NETWORK TRAINING

In terms of multi-target assignment initialization network based on supervised learning training, in the training, the data set is the input state of the optimal rule experiment, and the label is the assignment scheme adopted by the optimal rule. For use, the current state is input, and the network can output the solution that should be adopted under the current environmental state. The learning rate is an important hyperparameter in supervised learning and deep learning, which determines whether the target function can converge to the local minimum and when it converges to the local minimum. If the learning rate is too large, the network training will be insufficient, and the degree of intelligence will not high; if the learning rate is too small, the network will be difficult to converge. After many debugging and improving network parameters, the learning rate ''lr'' is set to 0.0001, the hidden layer is set to 256*256, and the interval return value is set to 2, that is, the current value is returned every two steps to provide data for subsequent observation of experimental changes. Since the multi-target assignment problem is essentially a classification problem, after comprehensive consideration, the cross-entropy loss function 'CrossEntropyLoss()' is used for network training. When the supervised learning script runs, it retrieves data from the specified database, generates an initialized network model, and saves it in the specified directory as the initial model for subsequent reinforcement learning training. The intermediate result generated during the run is shown as follows in the train_loss chart.

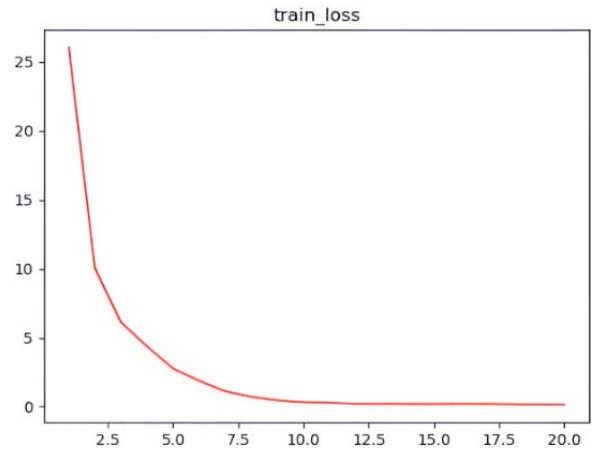

FIGURE 5. Supervised learning loss function of multi-target assignment.

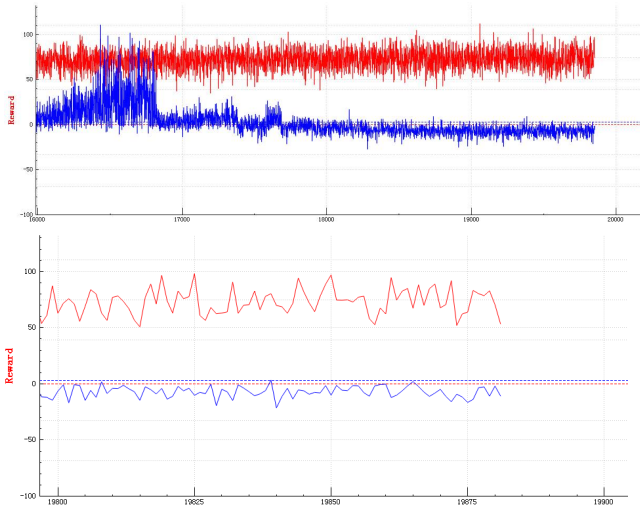

FIGURE 6. Reward convergence curve of iterative training of deep reinforcement learning.

x-label represents epoch, y-label represents loss function value. The figure above shows that with a learning rate of 0.0001, the loss value in training drops to 0.1558 after 20 epochs, which means that the network training has converged. The trained initialization network possessed an initial decision-making capability. However, it has a narrow applicable scenario and weak generalization capability.

B. ITERATIVE TRAINING OF REINFORCEMENT LEARNING

Before the training begins, the sample data needs to be stored in the empirical data pool as the initial database for iterative training, and the pre-training model is loaded to initialize the

$$R = \begin{cases} \left[5 + 5 * \frac{r_{kill}}{d_{warhead}}\right] * P, & (Target\ miss) \\ \left\{40 + 10 * \left(1 - \frac{d_{warhead}}{r_{kill}}\right)\right\} * (N - P), & (Target\ hit\ successfully) \end{cases} \quad (4)$$

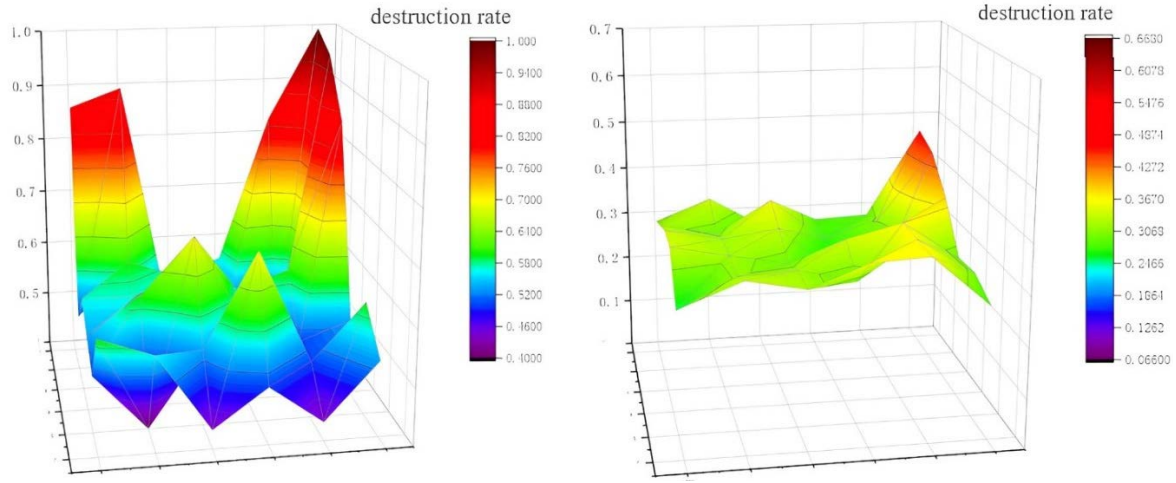

FIGURE 7. Comparison of intelligent testing results.

decision network. The factor value range and the number of factor levels are determined, and the iterative training tool of reinforcement learning is utilized to train 19,881 times to obtain the reward graph and the final convergence graph as follows:

The red curve represents the reward value, and the blue curve represents the loss function value. According to the curve recorded in the iterative training process, it can be known that the reward value has been stable in the [50, 100] interval, indicating that the average success rate of the useless intelligent decision-making network in the iterative training is about 30%. The convergence of the loss function means that the intelligent decision-making network converges during the iterative training process, which meets the requirements of network training. In the next section, the trained intelligent decision-making network will be simulated and tested, and compared with the rule-based decision-making experiment.

C. INTELLIGENT TESTING

After saving the network obtained from the iterative training of deep reinforcement learning, the intelligent decision and the rule-based decision are tested and compared under a new experimental scenario in a bid to better show the adaptability and generalization of reinforcement learning. The rule-based decision experiment employs the optimal rules selected by the full-space experiment to make decisions, while the intelligent decision experiment makes decisions by loading the trained reinforcement learning network. Experiments were run, and the experimental results were collected and analyzed, and is exported as follows:

The X-axis and Y-axis of Figure 6 represent the deployment of the enemy's interception positions. The image on the left shows the intelligent test results of the deep reinforcement learning network, the image on the right shows the test results of the optimal rules. Judging from the distribution of the images, for different enemy interception positions, the left image has reached the medium level or above in many places, while the right image has only one close to the medium level, indicating that the deep reinforcement learning training agent is more capable generalization ability, able to adapt to more scenarios. From the results of the image, left one where all destruction rates are above 40%, that is, the attack targets destroyed 40% of the total number of targets, or 90% and even 100% in better cases. Right one where the destruction rates are generally below 40% and only go as high as 50%. The data of intelligent decision network test with the results of the optimal rule testing is compared, and it is obtained that:

TABLE 4. Intelligent test experimental design of multi-target assignment.

Experimental results	Intelligent decision network testing	Optimal rule testing
Maximum destruction rate	100%	50%
Minimum destruction rate	40%	10%
Average destruction rate	60.09%	32.05%

The table suggests that the intelligent decision network has a destruction rate much higher than that of the optimal rule decision, of which the average destruction rate is nearly twice that of the optimal rule test. It is shown that the generalization and adaptability of the decision model trained by the deep reinforcement learning method are better than the optimal rules, which proves the effectiveness of the deep reinforcement learning algorithm in multi-target assignment.

VI. CONCLUSION

The penetration and target assignment of multi-warhead is an important means for missiles to improve survivability and effectiveness of strikes in combat missions. With the continuous development of multi-level, multi-stage, and multi-modal anti-missile defense systems, the difficulty of missile penetration continues to rise, and the effectiveness of

multi-warhead against multiple targets decreases. Research on how to increase the operational effectiveness of missiles through independent decision-making on missiles is of practical significance to strengthening China's strategic deterrence. Starting from the actual situation of missile combat tasks, in this paper, a multi-target assignment decision modeling method based on combat simulation and deep reinforcement learning was proposed for multi-warhead penetration and target assignment, and a missile multi-target assignment decision-making algorithm based on deep reinforcement learning was investigated and designed. The state space, action space, and reward function of missile decision were also designed and proposed, and the SAC algorithm was improved and applied in the decision-making of missile penetration and multi-target assignment. Finally, a case implementation is performed, and the feasibility and effectiveness of the intelligent decision-making algorithm are proved by comparing the results.

As the application research of deep reinforcement learning algorithm, the research on dynamic multi-target assignment of multiple warheads in this paper is not deep and comprehensive enough, and the next step needs to be improved and perfected, including the following three aspects:

(1) Comprehensive comparison of multiple intelligent decision-making methods. Due to the limited time and energy, I only carried out a comparison between the intelligent decision-making algorithm based on deep reinforcement learning and the rule-based decision-making method. The comprehensive comparison with other intelligent decision-making methods such as expert systems and genetic algorithms can effectively verify the intelligence of the algorithm.

(2) The decision space needs to be expanded. This paper only studies the maneuver decision-making and target assignment in the course of combat. Other decisions, including electronic jamming, target detection, and coordinated operations, have a greater impact on modern information warfare and are worthy of further research.

(3) Multi-agent collaborative combat decision-making. The multi-warhead target assignment problem studied in this paper only involves strike assignment, and there is no coordinated division of labor. The emergence of new combat concepts such as multi-projectile coordination and cluster penetration requires urgent research on collaborative decision-making among multi-agents. The emergent characteristics of clusters also lead to higher complexity of multi-agent problems.

REFERENCES

[1] Z. Ren, D. Guo, X. W. Dong, and Q. D. Li, "Research on the cooperative guidance and control method and application for aerial vehicle swarm systems," *Navigat. Position Timing*, vol. 6, no. 5, pp. 1–9, May 2019.

[2] Z. R. Bogdanowicz, A. Tolano, K. Patel, and N. P. Coleman, "Optimization of weapon–target pairings based on kill probabilities," *IEEE Trans. Cybern.*, vol. 43, no. 6, pp. 1835–1844, Dec. 2013.

[3] S. T. Lu, *Cooperative Guidance & Control of Missiles Autonomous Formation*, 1st ed. Beijing, China: National Defense Industry Press, 2015, pp. 88–96.

[4] S. K. Liu, H. Y. Wang, and F. X. Lu, "Online target assignment for cooperative attack of anti-ship of multiple missiles," *Command Control Simul.*, vol. 38, no. 1, pp. 38–40 and 52, Jan. 2016.

[5] Z. Ming, Z. Lingling, S. Xiaohong, M. Peijun, and Z. Yanhang, "Improved discrete mapping differential evolution for multi-unmanned aerial vehicles cooperative multi-targets assignment under unified model," *Int. J. Mach. Learn. Cybern.*, vol. 8, no. 3, pp. 765–780, Jun. 2017.

[6] Y. F. Ding, L. Q. Yang, J. Y. Hou, G. T. Jin, and Z. Y. Zheng, "Multi-target collaborative combat decision-making by improved particle swarm optimizer," *Trans. Nanjing Univ. Aeronaut. Astronaut.*, vol. 35, no. 1, pp. 181–187, 2018.

[7] J. Sun and C. Liu, "Finite-horizon differential games for missile–target interception system using adaptive dynamic programming with input constraints," *Int. J. Syst. Sci.*, vol. 49, no. 2, pp. 264–283, Jan. 2018.

[8] W. N. Wu, "Research on distributed mission planning for multiple unmanned aerial vehicles," Harbin Inst. Technol., Harbin, China, Tech. Rep., 2018.

[9] W.-N. Chen, J. Zhang, H. S. H. Chung, W.-L. Zhong, W.-G. Wu, and Y.-H. Shi, "A novel set-based particle swarm optimization method for discrete optimization problems," *IEEE Trans. Evol. Comput.*, vol. 14, no. 2, pp. 278–300, Apr. 2010.

[10] A.-G. Fei, L.-Y. Zhang, G. Liu, and Y. Wang, "The technique for air-to-air missile guidance superiority handover based on particle swarm auction hybrid algorithm," *J. Astron.*, vol. 34, no. 3, pp. 340–346, Mar. 2013.

[11] Y. Lu, X. Xu, and X. L. Zhang, "Hierarchical reinforcement learning for autonomous decision making and motion planning of intelligent vehicles," *IEEE Access*, vol. 8, pp. 209776–209789, 2020.

[12] X. Peng, R. Chen, J. Zhang, B. Chen, H.-W. Tseng, T.-L. Wu, and T.-H. Meen, "Enhanced autonomous navigation of robots by deep reinforcement learning algorithm with multistep method," *Sensors Mater.*, vol. 33, no. 2, p. 825, 2021.

[13] X. Li, W. Pu, and X. Zhao, "Towards learning behavior modeling of military logistics agent utilizing profit sharing reinforcement learning algorithm," *Appl. Soft Comput.*, vol. 112, Nov. 2021, Art. no. 107784.

[14] Y. L. Lei, J. Yao, N. Zhu, Y. F. Zhu, and W. P. Wang, "Weapon effectiveness simulation system (WESS)," *J. Syst. Simulation.*, vol. 29, no. 6, pp. 1244–1252, 2017.

[15] S. J. Shao, "Research on weapon target assignment based on intelligent algorithm," Harbin Inst. Technol., Harbin, China, Tech. Rep., 2019.

[16] R. H. Day, "Allocating weapons to target complexes by means of nonlinear programming," *Oper. Res.*, vol. 14, no. 6, pp. 992–1013, 1966.

[17] O. Kwon, K. Lee, D. Kang, and S. Park, "A branch-and-price algorithm for a targeting problem," *Nav. Res. Logistics*, vol. 54, no. 7, pp. 732–741, Oct. 2007.

[18] Z. J. Wang, "Research on decision-making method of multi-target interceptor," Harbin Inst. Technol., Harbin, China, Tech. Rep., 2020.

[19] Y. Yan and J. Huang, "Cooperative output regulation of discrete-time linear time-delay multi-agent systems under switching network," *Neurocomputing*, vol. 241, pp. 108–114, Feb. 2017.

[20] Y. P. Wang, B. Xin, and J. Chen, "Modeling and optimization of multi-stage sensor-weapon-target assignment," *Control Theory Appl.*, vol. 36, no. 11, pp. 1886–1895, Jul. 2019.

[21] B. Clark, D. Patt, and H. Schramm, "Mosaic warfare exploiting artificial intelligence and autonomous systems to implement decision-centric operations," CSBA, Center Strategic Budgetary Assessments, Washington, DC, USA, Tech. Rep., 2020, pp. 27–39.

[22] Z. Y. Wang and Y. J. Tan, "A solution to dynamic weapon-target assignment in the tank warfare," *J. Nat. Univ. Defense Technol.*, vol. 25, no. 6, pp. 56–61, 2003.

[23] Y. Zhang, L. Zhang, and M. Li, "A dynamic decision-maker of anytime algorithm for tasks decision-making problem," presented at the 7th IHMSC, 2015. [Online]. Available: https://ieeexplore-ieee-org-s.nudtproxy.yitlink.com/document/7334721

[24] Z. Song, F. Zhu, and D. Zhang, "A heuristic genetic algorithm for solving constrained weapon-target assignment problem," Presented at the ICIS, 2009. [Online]. Available: https://kns-cnki-net-s.nudtproxy.yitlink.com/KCMS/detail/detail.aspx?filename=IEEE200911001074&dbname=IPFDLAST2012

[25] Y. Chang, Z. Li, and Y. Kou, "A new approach to weapon-target assignment in cooperative air combat," *Math. Problems Eng.*, vol. 2017, pp. 1–17, Oct. 2017.

[26] M. Chen and F. X. Zhou, "Shipborne weapon target assignment based on improved particle swarm optimization," *Fire Control Command Control*, vol. 43, no. 11, pp. 72–76, 2018.

[27] T. Long, Z. Y. Liu, R. H. Shi, and S. Y. Wang, "Neural network based air defense weapon target intelligent assignment method," *Air Space Defense*, vol. 4, no. 1, pp. 1–7, 2021.
[28] F. Ma, Z. Y. Cao, and H. Liu, "Construction and search of strategy space of target assignment based on game theory," *J. Syst. Eng. Electron.*, vol. 32, no. 9, pp. 1941–1945, 2010.
[29] C. Leboucher, H. S. Shin, and P. Siarry, *A Two-Step Optimisation Method for Dynamic Weapon Target Assignment Problem* (Recent Advances on Meta-Heuristics and Their Application to Real Scenarios). Rijeka, Croatia: Intech, Jan. 2013.
[30] Y. Li, Y. Kou, and Z. Li, "An improved nondominated sorting genetic algorithm III method for solving multiobjective weapon-target assignment part I: The value of fighter combat," *Int. J. Aerosp. Eng.*, vol. 2018, pp. 1–23, Jun. 2018.
[31] B. Herd, S. Miles, P. Mcburney, and M. Luck, "Verification and validation of agent-based simulations using approximate model checking," *Multi-Agent-Based Simul.*, vol. 8235, pp. 436–442, Apr. 2013.
[32] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," *Statistics*, vol. 2, pp. 1467–5463, Jun. 2018.
[33] R. Sutton and A. Barto, *Reinforcement Learning* (Reinforcement Learning: An Introduction), 1st ed. Cambridge, MA, USA: MIT Press, 1998, pp. 4–7.

YIFEI WU was born in 1998. He received the bachelor's degree in simulation engineering, in 2020. He is currently pursuing the master's degree in electronic information with the National University of Defense Technology (NUDT), Changsha, China. His research interests include intelligent decision-making of weapon equipment and weapon effectiveness simulation and evaluation.

YONGLIN LEI was born in 1978. He received the Ph.D. degree in systems engineering from the National University of Defense Technology, in 2006. He was a Visiting Scholar at Arizona State University. He is currently a Professor with the National University of Defense Technology. His research interests include complex system modeling and simulation, model-driven engineering, and simulation composability.

ZHI ZHU was born in 1989. He received the M.S. degree in control science and engineering and the Ph.D. degree in military equipment from the National University of Defense Technology, in 2013 and 2018, respectively. He was a Visiting Ph.D. Student at Arizona State University. He is currently an Assistant Professor with the National University of Defense Technology. His research interests include language-driven development and system engineering.

XIAOCHEN YANG was born in 2002. He is currently pursuing the bachelor's degree in management science and engineering with the National University of Defense Technology (NUDT), Changsha, China. His research interest includes intelligent decision making of weapon equipment.

QUN LI was born in 1989. He received the M.S. degree in control science and engineering and the Ph.D. degree in military equipment from the National University of Defense Technology, in 2013 and 2018, respectively. He was a Visiting Ph.D. Student at Arizona State University. He is currently an Assistant Professor with the National University of Defense Technology. His research interests include language-driven development and system engineering.