Alpha C2—An Intelligent Air Defense Commander Independent of Human Decision-Making

QIANG FU, CHENG-LI FAN, YAFEI SONG, (Member, IEEE), AND XIANG-KE GUO
College of Air and Missile Defense, Air Force Engineering University, Xi’an 710038, China
Corresponding author: Cheng-Li Fan (ffq516@163.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61901512, and in part by the Natural Science Foundation of Shaanxi Province under Grant 2019JQ-013.

ABSTRACT The ultimate goal of military intelligence is to equip the command and control (C2) system with the decision-making art of excellent human commanders and to be more agile and stable than human beings. Intelligent commander Alpha C2 solves the dynamic decision-making problem in the complex scenarios of air defense operations using a deep reinforcement learning framework. Unlike traditional C2 systems that rely on expert rules and decision-making models, Alpha C2 interacts with digital battlefields close to the real world and generates learning data. By integrating the states of multiple parties as input, a gated recurrent unit network is used to introduce historical information, and an attention mechanism selects the object of action, making the output decision more reliable. Without learning human combat experience, the neural network is trained in fixed- and random-strategy scenarios based on a proximal policy optimization algorithm. Finally, 1,000 rounds of offline confrontation were conducted on a digital battlefield, whose results show that the generalization ability of Alpha C2 trained using a random strategy is better, and that it can defeat an opponent with a higher winning rate than an Expert C2 system (72% vs 21%). The use of resources is more reasonable than Expert C2, reflecting the flexible and changeable art of command.

INDEX TERMS Intelligent decision-making, deep reinforcement learning, Alpha C2, expert system.

I. INTRODUCTION

Battlefield is full of antagonism, complexity and uncertainty [1]. How to deal with uncertainty is significant to decision making in military operation. To lift the mystical veil over the war, many theories have been developed to handle uncertain information [2]–[5]. Command and control (C2) systems are still deterministic, i.e., a designated input produces a specific output. Such a system’s function is stable but rigid, and improving the “intelligence” level of a C2 system to assist decision-making would constitute a thorough innovation in the military field [6]. Decision-making intelligence has seen brilliant achievements in the application of game confrontation, such as Go [7], [8], StarCraft [9], [10], and DOTA [11], [12], but is still in the exploratory stage in the field of military confrontation. (Although ALPHA, developed by the University of Cincinnati, defeated human pilots in simulated air combat [13], it involved small-scale combat decisions, without considering complex system-confrontation strategy at the tactical level.) Military applications are limited for the following reasons: (1) The operational rules and evaluation criteria are difficult to express concisely. The rules of confrontation/judgment are clear in the game field, while abstract and numerous in the military field. (2) Effective means of verification and evaluation are lacking. The game platform itself is a good experimental environment, but verification is costly in the military field. (3) Minimal data are available. While data have emerged in the civilian field, operational data are lacking in the military field.

Because air combat is characterized by high antagonism, timeliness, a dynamic environment, a flexible combat style, and a large decision-making space, intelligent decision-making is needed to improve combat effectiveness. Weapon target assignment (WTA) is the core issue in air-combat command and decision-making. Its implementation rests on the establishment of a mathematical model based on situational awareness to minimize the probability of missing high-value targets [14], loss of defensive sites [15], and resources consumed for missile interception [16] and to maximize the effective kill probability [17], [18] while considering multiple constraints, to realize the effective utilization...
of air combat resources and avoid missing key targets and repeated shooting. This problem has been proved to be NP-complete [19] and can be divided into the categories of static WTA [20] and shoot-look-shoot-based dynamic WTA [21]. However, the opponent’s behavior is neither cooperative nor certain [22]. The lack of opponent strategy modeling makes it difficult for the idealized and regular WTA model to adapt to a changeable battlefield environment, or to reflect a commander’s flexible decision-making. More importantly, current research on WTA focuses on the establishment and solution of a mathematical model [23]–[25] with few means to validate the model itself.

In this study, a digital battlefield for air-to-ground confrontations close to actual combat was established. The red party’s Alpha C2 commanded several air-defense units to confront the air penetration launched by the blue party to defend strategic positions. Alpha C2 interacted with the digital battlefield online to generate learning data. It used no existing decision-making models (including WTA, situation awareness [26], [27] and sensor Assignment [28], [29]).

A deep reinforcement learning framework for Alpha C2 was constructed, integrating the states of the strategic position, fire unit, detected target, and attackable target as input, and used a gated recurrent unit network to introduce historical information, thus making the decision more accurate. In addition, an attention mechanism was used to choose the object of action, so that the network could give priority to hitting high-value targets. After 1,000 rounds of offline confrontation and deduction in a digital battlefield, the trained Alpha C2 defeated the blue party with a 72% winning rate, compared to the 21% winning rate of an expert C2 system, and its use of combat resources was more reasonable, showing more flexible tactics in the confrontation. The originality and contributions of this paper are summarized as follows.

- A decision-making method independent of human model is presented, which can be well used in military intelligence.
- A novel network structure for air defense operations is introduced, Improved the judgment ability of neural network in many-to-many scenarios.
- A digital battlefield for air-to-ground confrontations close to actual combat was established, the evaluation problem of military confrontation is solved.

The rest of this paper is organized as follows. In Section II, we present the Alpha C2 Network Structure and algorithm design. In Section III, the rules and scenario design of digital battlefield are introduced. In section IV, the training process of Alpha C2 under fixed strategy and random strategy are introduced. Section V presents the experimental comparison between expert C2 and Alpha C2. Finally, the conclusion and future work are discussed in Section VI.

II. DEEP REINFORCEMENT LEARNING IN ALPHA C2

A. ALPHA C2 NETWORK STRUCTURE

State space: the input information of Alpha C2 (neural network) can be divided into four categories: (1) The state of the strategic position defended by the red party, including the basic information such as position and type, and the state of the strategic position being attacked; (2) The state of the red party’s fire unit, including its current configuration, working state of the fire-control radar, working state of the missile-launching vehicle, state of the attacked fire-control radar, and information on enemy units that can be attacked by the fire unit; (3) The state of the discovered blue party units, including the basic movement information of blue party combat units and the situation attacked by the red party air-defense missiles; (4) The state of assailable blue party units, including the physical factors such as equipment capability, earth curvature and occlusion of surface features. The number of units of each kind of information varies with the change of battlefield situation.

The action of Alpha C2 has three parts: (1) choosing fire units; (2) choosing the type of missiles to launch; and (3) determining the blue party target to intercept. In theory, if the Alpha C2 network model can output these three kinds of data, it should be effective. However, there are many limitations in practice. The missiles launched must be provided with guidance information by fire-control radar. The number of missiles guided by fire-control radar is limited, as is the number of tracked targets. In addition, a number of blue party targets can be hit by red party fire units, and each blue party target can also be hit by multiple red party fire units. Therefore, in many-to-many scenarios, considering equipment capability and physical constraints, the neural network cannot effectively judge these conditions.

Action space: Two methods are considered to avoid the abovementioned phenomena: (1) Use the Alpha C2 neural network model to choose the red party fire unit and use a rule to choose a blue party target; (2) Use the Alpha C2 neural network model to choose a blue party target and use a rule to choose the red party fire unit to intercept. If the first method is adopted, then the threat of the target to be intercepted must be estimated manually and the interception sequence must be formed. Otherwise, the target posing the greatest threat is likely not attacked first, resulting in defense failure. If the second method is adopted, then Alpha C2 decides which blue party targets to intercept after considering all the incoming targets and the battlefield situation.

WTA should be driven by combat tasks and incoming targets [30]. Therefore, it is meaningless to use the first method to select fire units. It is most difficult to determine the blue party target of priority interception. There is a fixed routine to select fire units for interception using rules, such as the closest distance-based interception and the most missile-based interception. Therefore, the second method is adopted in this study.

Centralized command has higher operational efficiency and is the current mainstream command mode. Moreover, with a large number of combat units, the use of a multi-intelligent agent structure [31] will incur a great communication burden. Therefore, a command structure is adopted in this study in which an intelligent agent controls
multiple fire units to make air defense command decisions.

The Alpha C2 network structure is shown in Figure 1. It is composed of value and policy networks with an actor-critic structure [32]. The policy network makes decisions, which are evaluated by the value network (see next). Value network: Four states are input, feature extraction is processed through the two fully connected rectified linear units (FC-ReLU) [33] and then combined and put into the two FC-ReLUs, and put into fully connected (FC) layer. Finally, the value is output.

![FIGURE 1. Alpha C2 neural network structure.](image)

Policy network: Four states are input, processed through the two FC-ReLUs (feature extraction), and then combined and put into the FC-ReLU and the gated recurrent unit network (GRU). Finally, the optional actions (predicates of the actions) are output. In addition, after the attackable blue party target state is processed through the Max-pool, the target (object of the action) to be attacked is selected using an attention mechanism [36]. Considering that decision-making under the current state depends on full connection. When the number of target units (action objects) increases, the network structure must expand. Because there are many combat units, if both the subject and predicate are output by a neural network, then it cannot converge. Therefore, to determine the attack behavior, an attention mechanism [36] was used in this study to select the target unit, and then to choose the action subject with the best attack effect on the fire unit according to the rules in the environment.

Using the traditional fully connected form to select the target unit is equivalent to multi-classification with a fixed dimension. Such an approach would not be able to support the dynamic change of the target unit. The attention mechanism could choose among any multiple units, thus making the network more scalable [37].

An attention mechanism enabled intelligent agents to focus on certain information at a certain time and ignore other information, which allowed intelligent agents to make better decisions faster and more accurately in local areas. Given a task-related query vector \( q \), an attention variable \( z \in [1, N] \) was used to represent the index position of the selected information, i.e., \( N \) represented that the total number of input information, \( z = i \) represented that the \( i \)th input information was selected. To facilitate calculation, a "soft" attention mechanism was used. First, the probability of choosing the \( i \)th input information with given \( q \) and \( X \) was calculated by using (3)

\[
\alpha_i = \frac{p(z = i | X, q) = \text{soft max}(s(x_i, q))}{\sum_{j=1}^{N} \exp(s(x_j, q))}
\]

where \( \alpha_i \) is the attention probability distribution; and \( s(x_i, q) \) is an attention scoring function, calculated using an additive model as (4)

\[
s(x_i, q) = v^T \tanh(Wx_i + Uq)
\]

where \( W \), \( U \), and \( v^T \) are learnable network parameters, \( x_i \) denotes the eigenvector of the \( i \)th blue party target that
can currently be attacked, $X$ denotes the eigenmatrix of all blue party targets that can currently be attacked, and $q$ is the query vector output from the front part of the network, i.e., the hidden state obtained by the GRU. The input information was aggregated using the “soft” information selection mechanism as (5)

$$att(X, q) = \sum_{i=1}^{N} x_i \alpha_i$$

### D. ASYNCHRONOUS PROXIMAL POLICY OPTIMIZATION

The neural network was trained using a proximal policy optimization (PPO) algorithm [38]. Different from Q-learning and other value-based methods [39], it directly calculates the strategy gradient of the cumulative expected return by optimizing the strategy function, and then solves for the strategy parameters that maximize the overall return. There are different ways to define the loss function in PPO, such as no clipping or penalty, with clipping, and with Kullback–Leibler (KL) penalty. According to the multi-joint dynamics with contact experiment, the implementation of PPO with clipping is simple, and the effect is better than other PPO variants. Therefore, PPO with clipping was adopted in this study, aiming to minimize (6)

$$L_t(\theta) = \min(r_t^\ast(\theta)A_t, \ \text{clip}(r_t^\ast(\theta), 1-\epsilon, 1+\epsilon)A_t)$$

where $L_t(\theta)$ is the expected return at time $t$, $\epsilon$ is the clipping parameter, $r_t^\ast(\theta)$ is the ratio of the new policy function to the old. When $r_t^\ast(\theta) \notin [1-\epsilon, 1+\epsilon]$, the dominant function $A_t$ was clipped, so it was updated several times based on the old strategy function, and the updated strategy function deviated little from the original.

In each round of updates, the algorithm runs $N$ actors in parallel, and each actor runs $T$ steps. $NT$ steps of data were collected in total, and dominance was calculated in each step to estimate $A_1 \ldots A_T$. After data acquisition was completed, it was used to update policy parameters, where the objective function of the cumulative expected return was $L(\theta)$. $K$ iterations were performed in each round of updates, and a small batch of datasets was selected each time, with size $M \leq NT$. In this paper, asynchronous training mode is adopted, so the process of sampling and learning need not wait for each other, which makes the training more efficient.

### III. DIGITAL BATTLEFIELD ENVIRONMENT

Intelligent agents must interact with the environment in the training process, and this is the main restriction on the development of military intelligence. Therefore, the physical environment must be mapped to the virtual environment. Digital battlefields must be created accordingly, so as to provide basic support for Alpha C2 training.

The development of digital battlefields based on UE4 (Unreal Engine 4), which can generate countermeasure data in real time. At the end of each round, the system automatically records the number of battle damage and the outcome, we adopt the elevation digital map and consider the real physical constraints. In this study, the elements of the air-to-ground confrontation digital battlefield were set as follows.

#### A. RED PARTY FORCE SETTINGS AND CAPABILITY INDICATORS

The red party had two key defense sites (command post and airport), one airborne early warning aircraft (detection range 400 km), six long-range fire units and six short-range fire units.

Long-range fire unit consisted of one long-range fire-control radar vehicle (which could track eight blue party targets at the same time and guide 16 air-defense missiles with a maximum detection distance of 200 km and a sector of 120°), and eight long-range missile-launching vehicles (compatible with long- and short-range air-defense missiles; each launching vehicle loaded three long-range and four short-range air-defense missiles).

Short-range fire unit consisted of one short-range fire-control radar vehicle (which can track four enemy targets simultaneously and guide eight air-defense missiles with a maximum detection distance of 60 km and a sector of 360°) and three short-range missile-launching vehicles (each launching vehicle loaded four short-range air-defense missiles).

If the fire-control radar was destroyed, the fire unit lost its combat capability; in the guidance process, the radar needed to be turned on all the way; when it was turned on, it radiated electromagnetic waves, which could be captured by the opponent to expose its position; fire-control radar was physically restricted by earth curvature and occlusion of surface features, so there would be blind areas. Considering the refraction of the atmosphere to the wave, the radar’s limited visual range $R_{max}$ was calculated by using (7)

$$R_{max} = 4.12(\sqrt{H_T(m)} + \sqrt{H_R(m)})(km)$$

where $\sqrt{H_T}$ is the target altitude and $\sqrt{H_R}$ is the altitude of the radar antenna. In this study, $H_R$ was set to 4 m.

The flight trajectory of an air-defense missile was the best energy trajectory. It could intercept objects up to 160 km (long range) and 40 km (short range). For UAVs, fighter aircraft, bombers, anti-radiation missiles, and air-to-ground missiles, the high kill probability in the killing range was 75%, and the low kill probability was 55%. For cruise missiles, the high and low kill probabilities were 45% and 35%, respectively.

Four long-range and three short-range fire units were deployed in a fan shape to protect the red party’s command post. Two long-range and three short-range fire units were deployed to defend the red party’s airport. There were 12 fire units in total.

#### B. BLUE PARTY FORCE SETTING AND CAPABILITY INDICATORS

The blue party had 18 cruise missiles, 20 UAVs (each carrying two anti-radiation (AR) missiles and one air-to-ground (ATG) missile), 12 fighter aircrafts (each carrying six AR missiles
and two ATG missiles), four bombers, two Jammers (for long-distance jamming outside the defense area, with a jamming sector of 15°). After the red party’s radar was jammed, the kill probability was reduced according to the level of jamming).

The range of an AR missile was 110 km, and the hit rate was 80%; the range of an ATG missile was 60 km, and the hit rate was 80%.

C. CONFRONTATION OF THE TWO PARTIES

The blue party launched three rounds of attacks in total. In the first round, 18 cruise missiles were divided into two groups to strike a sudden attack at the command post and airport. As shown in Figure 2, a cruise missile attacked at an ultra-low altitude of 100 m. Influenced by the earth’s curvature, the range of the target intercepted by the red party’s fire-control radar was only about 40 km. Therefore, it was necessary for the red party to plan resources to ensure interception while minimizing ammunition consumption.

In the second round, 12 fighter aircraft attacked the defense sites and destroyed the exposed air defense positions under the cover of 20 UAVs, as shown in Figure 3.

For the second attack, which was more confrontational, the opponents of Alpha C2 were designed to be stronger: as shown in Figure 3(a), fighter aircraft penetrated under the cover of UAVs. The flight altitude of a UAV was 2000–3000 m, which induced the red party’s fire-control radar to start and fighter aircraft to penetrate at an ultra-low altitude of 100–150 m (protected by the earth’s curvature, they flew safely in the blind area of the fire-control radar). As shown in Figure 3(b), when the red party’s fire-control radar started, the fighter aircraft climbed up to the attack area, formed intervisibility with the red party’s fire-control radar, and launched an anti-radiation bomb attack. As shown in Figure 3(c), after the attack, the fighter aircraft descended and maneuvered to escape, entering a hunting state and waiting for another attack (for visualization, the size of the red and blue parties’ equipment was enlarged 50 times in the 3D sand table).

The defense pressure of the red party was as high as necessary to intercept not only UAVs and fighter aircraft but a large number of air-to-ground missiles and anti-radiation bombs launched by all combat aircraft. Due to the limited number of guidance missiles and targets tracked by the red party’s fire-control radar, resources were easily saturated. The third round was launched immediately after the second round, using four bombers to penetrate and bomb the defense sites.

IV. ALPHA C2 TRAINING

A. REWARD FUNCTION

Alpha C2 deserved the greatest reward if it won the confrontation, and the greatest punishment if it failed. Many factors were involved in the confrontation process. In addition to just winning or losing, the results should reflect the degree, i.e., big or small winning or losing. Therefore, it was necessary to design the reward for both parties’ battle damage.

After the confrontation, which party won or lost and the battle damage of the two parties determined whether the action in a game had a positive or negative effect, i.e., the feedback signal (reward) obtained by Alpha C2. This had two parts: victory and defeat, and the battle damage of both parties. The proportion of victory or defeat should be greater than that of battle damage, i.e., if a game is won, the reward must be greater than zero, and less than zero otherwise. The reward index should be as objective as possible. Based on the scenario of this study, by refining the key indicators using an analytic hierarchy, optimized by a simulation test, a reward
TABLE 1. Reward scores for Alpha C2.

| Victory                      | Defeat                                 | Command post (suffered one attack) | Airport (suffered one attack) | Long-range radar damage |
|------------------------------|----------------------------------------|------------------------------------|-------------------------------|------------------------|
| Score                        | −1                                     | −0.15                              | −0.05                         | −0.05                  |
| Short-range radar damage     | Consumption of long-range missiles     | Consumption of short-range missiles | Interception of UAV           | Interception of fighter aircraft |
| Score                        | −0.01                                  | 0.005                              | 0.0003                        | 0.01                   |
| Interception of bomber       | Interception of cruise missiles        | Interception of anti-radiation missiles | Intercetion of air-to-ground missiles | 0.1                    |
| Score                        | 0.1                                    | 0.002                              | 0.001                         | 0.002                  |

The reward function was determined by five military experts to obtain the reward score of each battle unit, as shown in Table 1.

B. TRAINING SCENARIOS

In this study, two training scenarios were designed, based on either a fixed or random blue party strategy. The training hardware was the same. The CPU for the simulation environment was an Intel Xeon E5-2678V3 with 88 cores and 256 GB memory; Two GPUs run neural network training, and the model was Nvidia GeForce 1080Ti, 72 cores, 11 GB video memory. In PPO, the hyper-parameter was $\varepsilon = 0.2$; the learning rate was $10^{-4}$; the batch size was 5,120; the number of hidden layer units in the neural network was 256, and 72 actors were run in parallel, iterated three times in each round.

1) Alpha C2 model 1: in scenario 1, the blue party’s attack route, formation size, and combat task were fixed. In the first round, 18 cruise missiles were divided into two groups to strike a sudden attack on a command post and airport; In the second round, 12 fighter aircraft attacked the defense sites and destroyed the exposed air-defense positions under the cover of 20 UAVs; the third round was launched immediately after the second round, using four bombers to penetrate and bomb the defense sites.

Rules for judging victory or defeat: (1) when the red party’s command post was attacked three times or the bomber approached the command post within a range of 10 km, the red party lost (the blue party won); (2) when the blue party lost more than 30% of its fighter aircraft, the blue party lost (the red party won); (3) if the red party lost more than 60% of its long-range radars, then the red party lost (the blue party won). Each actor was iterated 4,250 times to obtain Alpha C2 model 1.

Figure 5(a) shows the winning rate curve of Alpha C2 model 1. Alpha C2 initially had no strategy. Almost all fire units shot freely under the physical and equipment-capability constraints, and the winning rate was zero. This shows that disorderly interception cannot defeat powerful opponents. Victory occurred when the model was trained 200 times. The final winning rate was more than 70%. Figure 5(b) shows the mean reward curve, with the reward value increased from −0.85 to about 0.5. Winning or losing is one aspect. It also shows that Alpha C2’s use of resources was becoming increasingly reasonable. Yet it should be noted that even if the blue party adopted a fixed strategy, curve jitter was still obvious because there are many uncertainties in the battlefield, such as the kill probability, which inhibits the regular growth of the curve.

2) Alpha C2 model 2: To compare the generalization ability of models with different numbers of iterations, independent of the Alpha C2 model 1, the model was trained in scenario 1 as well. The hyper-parameter setting remained unchanged. Each actor was iterated 3,700 times to obtain Alpha C2 model 2.
As shown in Figure 6(a), after 3,700 iterations, the winning rate of model 2 almost reached 60%. As shown in Figure 6(b), the mean reward was close to 0.2. Due to the scenario’s randomness, the upward trend of model 2 differed from that of model 1.

(3) Alpha C2 model 3: When trained in scenario 2, it is impossible to accurately predict the opponent’s attack pattern in actual combat. To more closely resemble a real battlefield, more randomness was added. As shown in Figure 4, the overall direction of the blue party’s attack remained unchanged, but the route of penetration, arrival time, and formation of units became random, which better reflected the uncertainty of the battlefield. The size of force, capability index, and rules to determine victory or defeat in scenario 2 were the same as in scenario 1.

Figure 7(a)-(c) shows the three battlefield situations randomly selected from scenario 2, where the penetration route, arrival time, and formations of units of the blue party were all different. Thus the operational strategy and combat process had greater uncertainty. The hyper-parameter settings were the same. Each actor was iterated 3,250 times to obtain Alpha C2 model 3.

The light-blue curve in Figure 8(a) is the winning rate curve of Alpha C2 trained 3,250 times in a random environment. When it was iterated more than 400 times, victory occurred, and the winning rate increased sharply at around 2,000 iterations. This shows that Alpha C2 gradually adapted to the randomness of the blue party’s strategy, and the final winning rate was about 40%. As shown in Figure 8(b), even if Alpha C2 did not win the first 400 times, the mean reward still rose significantly, indicating the gradual improvement from big defeat to small defeat, and the increasingly rational use of resources. At about 2,000 times, the mean reward increased significantly, eventually rising from -0.82 to almost zero.

As shown in Figure 9, by comparing the mean reward of three versions of Alpha C2 in the training process, it can be found that the light-blue curve rises relatively slowly,
indicating that compared with a fixed strategy (red and dark-blue curves), the training of a random strategy is more difficult, especially in the early stage. In addition, model 1 has the most training iterations and the highest mean reward. However, this does not prove that model 1 is best. Their generalization ability in different scenarios also must be tested and compared with human knowledge.

V. EXPERIMENTS
A. EXPERT C2
To verify the decision-making quality of Alpha C2 requires one to benchmark with human experts. In complex and high-real-time air-defense scenarios, a human’s disadvantage is primarily in response time and not in decision-making. Without the help of a C2 system, human beings cannot effectively exercise situational awareness in a short time and make such a large number of WTAs (hundreds of decisions per minute, plus instructions). Therefore, human experience was regularized and modeled to form an expert decision scheme suitable for this scenario, which was embedded in the C2 system for confrontation verification.

Based on target threat estimation, WTA schemes mainly performed with eliminating targets with the maximum threat/highest value, the highest kill probability or the minimum resource consumption as the objective functions. Using modeling ideas proposed by Ahner and Parson [21], Xin et al. [29], and Bogdanowicz et al. [40] and consulting with five experts, a priority was determined according to scenario elements, and the final expert decision scheme was formed as follows:

(1) Target threat level (target value) was classified into levels from 0–10 according to the time when the target arrived at a defensive site, and the level was increased every 15 seconds.

(2) Priority was given to intercepting high-threat targets (to maximize damage to high-value targets).

(3) Interception strategies (indirectly considering the impact of target types on threat level) varied by types of targets. When the threat degree of the blue party’s AR missile and ATG missile reached 6, one air-defense missile was launched for interception and then entered the observation stage. If the target was not killed, then when the threat degree reached 10, two missiles were launched for interception. For the blue party’s aircraft, if the threat degree was more than 7, then a missile was launched for interception, and then entered the observation stage. If the target was not killed and shooting conditions were satisfied, then two missiles were launched to intercept a fighter aircraft, and one missile was launched to intercept a UAV. For bombers, when the threat degree was more than 4, one missile was launched for interception, and two missiles were launched when the threat degree reached 9.

(4) The self-defense strategy was preferred for antiradiation missiles, i.e., the attacked fire units were intercepted. When the fire unit resource was saturated, it was assisted by the nearest fire unit to intercept (to minimize operational losses).

(5) Short-range missiles were preferred for cruise missiles (because cruise missiles penetrated at an ultra-low altitude, and affected by the earth curvature, the killing zone of short-range missiles was basically the same as that of long-range missiles, which minimized the cost-effectiveness ratio).

(6) When intercepting targets, fire units with high kill probabilities should be used first (to maximize kill probability); when the kill probability is the same, priority should be given to firing units tracking the target (to minimize radar radiation time [41]); and when the kill probability was the same and the target was tracked simultaneously, fire units with more missiles were preferred (fire load was the most balanced).

B. WINNING RATE, REWARD, AND BATTLE DAMAGE COMPARISON
Alpha C2 was trained in the above two scenarios to obtain three Alpha C2 models. These and the Expert C2 system based on expert knowledge were used to combat the blue party with fixed and random strategies, and 1,000 rounds of offline confrontation were conducted with each. The experimental results (winning rate, reward, and battle damage of both parties) were recorded. The deduction process did not require high-configuration hardware, so an ordinary PC could be used, in this case an Intel i7 7800X.

| Scenario 1 (fixed strategy) | Scenario 2 (random strategy) |
|-----------------------------|------------------------------|
| Win rate | Mean reward | Win rate | Mean reward |
| model 1  | 63.1% | 0.3797 | 2.9% | −0.6242 |
| model 2  | 56.2% | 0.1188 | 29.5% | −0.1013 |
| model 3  | 72.1% | 0.4774 | 49.2% | 0.0908 |
| Expert C2 | 21.8% | −0.1008 | 22.3% | −0.1072 |

Figure 10 compares the winning rates of the three models of Alpha C2 and Expert C2 in different scenarios. In scenario 1, where the fixed strategy was adopted, the winning rate of Alpha C2 model 1 reached up to 63.1%, while that of Expert C2 was only 21.8%, hence the winning rate increased significantly, while in scenario 2, where the random strategy was adopted, the winning rate was less than 2.9%. This shows that the training had been over-fitted, and the model lacked good generalization ability to adapt to the change of the opponent’s strategy. In scenario 2, the winning rate of Alpha C2 model 2 reached 56.2%. The number of
iterations of model 2 was adjusted, and the over-fitting was insignificant. Therefore, in scenario 2, the winning rate still reached 29.5%. The effect of Alpha C2 model 3 trained in scenario 2 with a random strategy was the best. Facing the change of the blue party’s strategy, its winning rate could still reach 49.2%. In scenario 1, its winning rate was as high as 72.1%. Model 3 had the fewest iterations, but the highest winning rate in both scenarios. This shows that training under a random strategy scenario enables the model to better adapt to the adjustment of the blue party’s strategy, and the training effect is the best. The winning rate of Expert C2 in scenario 2 was 22.3%, which was not different from that in scenario 1. This shows that when the condition indices of both parties remain unchanged, human experience rules are insensitive to randomness, and a C2 system based on expert knowledge is stable, but its ability to cope with a complex battlefield environment is poor.

It should be noted that when each fire unit shot freely and did not follow unified command, the winning rate was almost zero, versus about 20% for Expert C2. This shows that Expert C2 is effective and can well avoid repeated shooting and omission of key targets, but it lacks a deeper strategy. Fundamental problems cannot be solved simply by adjusting the rules. According to the winning rate, the training of a neural network model can solve complex military confrontation decision-making problems, increase the randomness of a strategy, and effectively alleviate the over-fitting phenomenon, and the trained Alpha C2 can better adapt to the changing battlefield environment.

Figure 11 describes the three models of Alpha C2 and Expert C2 in different scenarios, including the damage of long-range/short-range radar, the number of defense sites being attacked, and the damage to UAVs and fighter aircraft. In terms of the damage of the red party’s fire units and key sites, Alpha C2 model 1 performed better in scenario 1. The number of killed high-value targets (fighter aircraft) it killed was larger, but the total number of targets killed was smaller. In scenario 2, the effect was obviously reduced. Its performance was worse than Expert C2 in two key indicators, i.e., the number of lost long-range radars and the number of killed fighter aircraft. The overall performance of Alpha C2 model 2 in scenario 1 was better than in scenario 2. It killed more fighter aircraft, better controlled the damage status of long-range radar and key sites. However, its performance in scenario 2 was not substantially worse than in scenario 1. Overall, Alpha C2 model 3 performed better in both scenarios, both in the number of killed targets and the party’s own damage in the battle. Model 3 showed the best overall performance when compared to model 1, model 2, and Expert C2. Model 3 performed slightly better in scenario 1 than in scenario 2 in all aspects. Expert C2 was the most stable in the two scenarios. Considering the randomness of the environment, there was no significant difference between scenarios 1 and 2 in various indicators. In terms of the number of damaged defense sites and destroyed fighter aircraft, its performance was very poor.

Figure 12 further analyzes the three models of Alpha C2 and Expert C2 in different scenarios, including the consumption of long-range/short-range missiles, and the number of intercepted anti-radiation and air-to-ground missiles. Alpha C2 model 1 consumed fewer missiles in scenario 1 and more missiles in scenario 2, and intercepted more target missiles in scenario 2. It should be noted that the large number of intercepted AR and ATG missiles is not necessarily good. It should seize the opportunity to eliminate the opponent before the opponent attacks. Alpha C2 model 2 consumed more long-range missiles in scenario 3, but other indicators were not significantly different from Scenario 2. Alpha C2 model 3 consumed the most long-range missiles in both scenarios, basically 144 each, reaching maximum...
usage, while consuming fewer short-range missiles. Expert C2 consumed the most missiles and used almost all of them. The number of AR and ATG missiles intercepted by Alpha C2 was relatively small, but combined with the number of radar and defense sites damaged, as shown in Figure 11, the interception timing of Alpha C2 was better than that of Expert C2. Before the blue party fired the missile, the blue party’s plane was eliminated.

Table 3 shows all aspects of battle damage. Alpha C2 model 3 performs best on key indicators. Two additional comparisons are made here: (1) In terms of the number of cruise missiles killed, Alpha C2 and Expert C2 can effectively intercept in all scenarios, but Alpha C2 is more efficient. (2) The numbers of bombers killed by Expert C2 are 1.26 and 1.16 in scenario 1 and scenario 2, respectively, while Alpha C2 kills very few, especially model 3, where it is zero. However, this does not mean that Expert C2 kills bombers more effectively. Basically, Alpha C2 had already killed 30% of the blue party’s fighter aircraft before the bombers approached the defense sites. Then, the blue party failed and retreated. Alpha C2 avoided fighting with the bombers as much as possible.

### C. COMPARISON OF COMBAT DETAILS

Through data comparison, it is found that Alpha C2 can better complete tasks, but the data cannot reflect its decision-making details. Because the whole confrontation process is complicated, only partial cases were selected to compare it with Expert C2.

(1) Alpha C2 shows better firepower synergy capability. In the fight against cruise missiles, as shown in Figure 13(a), Expert C2 paid attention to its own defensive tasks but did not share defense pressure when faced with cruise missiles attacking command posts. As shown in Figure 13(b), Alpha C2 shared interception resources to relieve pressure defending a command post, while effectively completing its own defense tasks. As shown in Figure 13(c), Expert C2 had
low overall interception efficiency, and did not complete the interception until the cruise missile almost reached the defense site. As shown in Figure 13(d), Alpha C2 usually intercepted all cruise missiles with a large distance remaining, which shows its more reasonable use of interception resources.

(2) Faced with the blue party’s diversion strategy, as shown in Figure 14(a), because of the premature interception of Expert C2, the fire units of the red party were exposed after the booting of the fire-control radar. Because of the long distance, the blue party evaded the missile. Other flying formations launched cooperative attacks, launching anti-radiation missiles to attack the red party’s fire-control radar. As shown in Figure 14(b), the red party showed a more passive defense to intercept AR and ATG missiles, and could not well attack the aircraft. As shown in Figure 14(c), after the blue party continued to rush forward, it opened a gap and attacked the red party’s defense sites to win the battle with only damages to the UAVs when possible. Under the command of Alpha C2, as shown in Figure 14(d), the red fire-control radar was in a silent state in the face of the blue party’s temptation, so as not to expose itself too early. When the blue party approached in a roundabout way, all fire-control radars were turned on almost at the same time for a silent ambush. As shown in Figure 14(e), it was more effective to kill the blue party’s targets, while retaining certain resources to intercept anti-radiation missiles and air-to-ground missiles. As shown in Figure 14(f), when the long-range fire unit experienced
greater damage, the short-range fire unit showed more initiative and attracted more blue party firepower, while the long-range firepower unit was more silent and ambushed under the condition of guaranteeing its own safety.

3. As shown in Figure 15(a), Expert C2 shot at every target in a relatively scattered and disordered shooting process. The dominant forces were not gathered to attack key targets. As shown in Figure 15(b), Expert C2 missed the bomber. In the later stage of the confrontation, both parties consumed many resources. The blue party used part of its resources to fight all the resources of the red party, so that the bomber could approach the defense sites. As shown in Figure 15(c), on some special occasions, Alpha C2 chose to sacrifice a small number of fire units to avoid inappropriate large-scale shooting. It can be seen from Figure 15(d) that the sacrifice was meaningful, further luring the blue party’s fighter aircraft and UAVs into an ambush circle, quickly organizing firepower to shoot and annihilate the blue party, which is more effective for intercepting high-value targets and more purposeful for fire coordination. In addition, Alpha C2 could accomplish combat tasks earlier, and effectively killed the blue party’s fighter aircraft before the bomber even attacked, thus winning the battle.

By comparing reward, victory/defeat, battle damage, and battle scenarios, it was found that Alpha C2 has a higher level of decision-making than Expert C2. Without considering the opponent’s confrontation strategy, command decision-making can be studied as an optimization problem. In reality, the opponent must be full of strategies, which will lead to the deterministic C2 system falling into passive decision-making.

VI. CONCLUSION AND FUTURE WORK
Operational rules and decision-making models are the summary and compaction of human experience. However, it is difficult to embody the flexible and changeable command art of commanders through only a set of objective functions or some priority criteria. Inspiration in human decision-making is difficult to describe mathematically. The problem of C2 system assistant decision-making may be solved by a neural network. In this study, different from the traditional approach to modeling and solving decision-making problems, a deep reinforcement learning framework for air defense Alpha C2 was proposed. A reasonable state space, action space, and reward function were designed based on the analysis of battlefield situations. A gated recurrent unit network and attention mechanism were introduced to improve the network’s decision-making ability. Through real-time confrontation in a digital battlefield, the problems of lack of confrontation data and difficulty evaluating a model in the military field were effectively solved. Experimental results show that without learning the existing models, rules, and operational experience of human beings, Alpha C2 after deep reinforcement learning training has a higher winning rate than a traditional Expert C2 system, and its use of resources is more reasonable. Alpha C2 shows a higher decision-making art in complex confrontations.

Alpha C2 can be used as a battle planning system to evaluate and optimize a battle scheme. It can also operate in parallel with the commander in the Observation-Orientation-Decision-Action (OODA) loop to provide high-quality real-time decision-making advice for the commander. AI cannot completely replace humans in military confrontation. Its task is to help them more effectively achieve combat objectives. Therefore, humans should always dominate, and machine cooperation can be used widely in the military field.

Many problems still require further study regarding Alpha C2.

- At the end of each round of confrontation, the intelligent agent gets a reward, and the reward signal is sparse. In theory, real-time reward feedback for each unit’s decision can better encourage effective behavior, but it may also result in over-fitting.

- Compared with games, the physical world has more obvious delayed-action response. When Alpha C2 performs an action, it cannot immediately obtain a response. For example, when a missile is launched, whether the missile can kill the target cannot be immediately known. Only after a long flight and target encounter can the result be obtained, which may lead to a decrease in the predictive ability of Alpha C2 in the course of a confrontation.

- To improve the algorithm’s capabilities, it is necessary to enrich the digital battlefield function, realize the intelligence of the blue party, and further enhance the decision-making level and generalization ability of Alpha C2 through powerful opponents.

REFERENCES
[1] Q. Fu, Y. Song, C.-L. Fan, L. Lei, and X. Wang, “Evidential model for intuitionistic fuzzy multi-attribute group decision making,” Soft Comput., vol. 24, no. 10, pp. 7615–7635, May 2020, doi: 10.1007/s00500-019-04389-2.
[2] C.-L. Fan, Y. Song, Q. Fu, L. Lei, and X. Wang, “New operators for aggregating intuitionistic fuzzy information with their application in decision making,” IEEE Access, vol. 6, pp. 27214–27238, 2018.
[3] Y. Song, Q. Fu, Y. Wang, and X. Wang, “Divergence-based cross entropy and uncertainty measures of Atanassov’s intuitionistic fuzzy sets with their application in decision making,” Appl. Soft Comput., vol. 84, Nov. 2019, Art. no. 105703.
[4] B. Zhang, M. Zhang, Y. Song, and L. Zhang, “Combining evidence sources in time domain with decision maker’s preference on time sequence,” IEEE Access, vol. 7, pp. 174210–174218, 2019.
[5] Y. Song, X. Wang, W. Quan, and W. Huang, “A new approach to construct similarity measure for intuitionistic fuzzy sets,” Soft Comput., vol. 23, no. 6, pp. 1985–1998, 2019.
[6] L. Peter, “Algorithmic warfare: Applying artificial intelligence to warfighting,” in Air Power Development Centre, 2018.
[7] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. van den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. L Santos, D. Dieleman, D. Grewe, J. Nham, K. Kalchbrenner, I. Sutskever, T. Lillicrap, M. Leach, K. Kavukcuoglu, T. Graepel, and D. Hassabis, “Mastering the game of go with deep neural networks and tree search,” Nature, vol. 529, no. 7587, pp. 484–489, Jan. 2016.
[8] D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. Baker, M. Lai, A. Bolton, Y. Chen, T. Lillicrap, F. Hui, L. Sifre, G. van Den Driessche, T. Graepel, and D. Hassabis, “Mastering the game of go without human knowledge,” Nature, vol. 550, no. 7676, pp. 354–359, Oct. 2017.
[9] O. Vinyals and I. Babuschkin, “Grandmaster level in StarCraft II using multi-agent reinforcement learning,” Nature, vol. 575, no. 7782, pp. 350–354, Nov. 2019.

[10] P. Peng, Y. Wen, Y. Yang, Q. Yuan, Z. Tang, H. Long, and J. Wang, “Multigait bidirectionally-coordinated nets: Emergence of human-level coordination in learning to play StarCraft combat games,” 2017, arXiv:1703.10069. [Online]. Available: http://arxiv.org/abs/1703.10069

[11] M. Hutson, “AI takes on video games in quest for common sense,” Nature, vol. 361, no. 6403, pp. 632–633, Aug. 2018.

[12] W. Looi, M. Dhalial, R. Alhajj, and J. Rokne, “Recommender system for items in dota 2,” IEEE Trans. Games, vol. 11, no. 4, pp. 396–404, Dec. 2019.

[13] N. Ernest and D. Carroll, “Genetic fuzzy based artificial intelligence for unmanned aerial vehicle control in simulated air combat missions,” J. Defense Manage., vol. 06, no. 01, 2016.

[14] O. Onay, “A mathematical approach to neuromarketing: A weapon—target assignment model,” Int. J. Academic Res. Bus. Social Sci., vol. 6, 1, pp. 164–173, 2016.

[15] G. Shang, Z. Zaiyue, Z. Xiaoru, and C. Cungen, “Immune genetic algorithm for weapon-target assignment problem,” in Proc. Workshop Intell. Comput. Oper. Res., no. 8, pp. 1689–1701, Dec. 2015.

[16] O. Kwon, D. Kang, K. Lee, and S. Park, “Lagrangian relaxation approach to the targeting problem,” Nav. Res. Logistics, vol. 46, no. 6, pp. 640–653, Sep. 1999.

[17] O. Karasakal, “Air defense missile-target allocation models for a naval task group,” Comput. Oper. Res., vol. 35, no. 6, pp. 1759–1770, Jun. 2008.

[18] T. Sikken, “Solving weapon target assignment problem with dynamic programming,” Indep. Res. Projects Appl. Math., vol. 32, 2008.

[19] S. Lloyd and H. Witsenhausen, “Weapons allocation is NP-complete,” in Proc. Summer Comput. Simulation Conf., 1986, pp. 1054–1058.

[20] R. K. Ahuja, A. Kumar, K. C. Jha, and J. B. Orlin, “Exact and heuristic algorithms for the weapon-target assignment problem,” Oper. Res., vol. 55, no. 6, pp. 1136–1146, Dec. 2007.

[21] D. K. Ahner and C. R. Parson, “Optimal multi-stage allocation of weapons to targets using adaptive dynamic programming,” Optim. Lett., vol. 9, no. 8, pp. 1699–1701, Dec. 2015.

[22] Q. Pan, D. Zhou, Y. Tang, and X. Li, “A novel antagonistic weapon-target assignment model considering uncertainty and its solution using decomposition co-evolution algorithm,” IEEE Access, vol. 7, pp. 37498–37517, 2019.

[23] D. K. Ahner and C. R. Parson, “Optimal multi-stage allocation of weapons to targets using adaptive dynamic programming,” Optim. Lett., vol. 9, no. 9, pp. 1699–1701, Dec. 2015.

[24] A. Kline, D. Ahner, and R. Hill, “The weapon-target assignment problem,” Comput. Oper. Res., vol. 105, pp. 226–236, May 2019.

[25] M.-Z. Lee, “Constrained Weapon-Target assignment: Enhanced very large scale neighborhood search algorithm,” IEEE Trans. Syst., Man, Cybern.-A, Syst. Humans, vol. 40, no. 1, pp. 198–204, Jan. 2010.

[26] Q. Liang and X. Cheng, “KUPS: Knowledge-based ubiquitous and persistent sensor networks for threat assessment,” IEEE Trans. Aeros. Electron. Syst., vol. 44, no. 4, pp. 1060–1069, Jul. 2008.

[27] T. Shujuan, B. Duyan, Y. Xu, and W. Weijia, “Dynamic sensor management algorithm based on improved efficacy function,” in Proc. 12th IEEE Int. Conf. Electron. Meas. Instrum. (ICEMI), Jul. 2010, pp. 145–148.

[28] B. Xin, Y. Wang, and J. Chen, “An efficient marginal-return-based constructive heuristic to solve the sensor–weapon–target assignment problem,” IEEE Trans. Syst., Man, Cybern., vol. 49, no. 12, pp. 2536-2547, Dec. 2019.

[29] D. P. Bertsekas, M. L. Homer, D. A. Logan, S. D. Patek, and N. R. Sandell, “Missile defense and interceptor allocation by neurodynamic programming,” IEEE Trans. Syst., Man, Cybern.-A, Syst. Humans, vol. 30, no. 1, pp. 42–51, Jan. 2000.

[30] A. Viseras and R. Garcia, “DeepGP: Multi-robot information gathering with deep reinforcement learning,” IEEE Robot. Autom. Lett., vol. 4, no. 3, pp. 3059–3066, Jul. 2019.

[31] V. Nair and G. Hinton, “Rectified linear units improve restricted Boltzmann machines,” in Proc. 27th Int. Conf. Mach. Learn., 2010, pp. 807–814.

[32] K. Cho, B. Van Merriënboer, C. Gulcehre, and D. Bahdanau, “Learning phrase representations using RNN encoder-decoder for statistical machine translation,” in Proc. Conf. Empirical Methods Natural Lang. Process., 2014, pp. 1724–1734.

[33] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” Neural Comput., vol. 9, no. 8, pp. 1735–1780, 1997.

[34] M. Deniz, L. Bazzani, H. Larochelle, and N. de Freitas, “Learning where to attend with deep architectures for image tracking,” Neural Comput., vol. 24, no. 8, pp. 2151–2184, 2012.

[35] Q. Liang and X. Cheng, “KUPS: Knowledge-based ubiquitous and persistent sensor networks for threat assessment,” IEEE Access, vol. 7, pp. 90497–90506, 2019.

[36] G. Sam, M. Craig, and K. CETIN, “Distillation strategies for proximal policy optimization,” Mach. Learn., to be published.

[37] J. Hu and M. P. Wellman, “Nash Q-learning for general-sum stochastic games,” J. Mach. Learn. Res., vol. 4, pp. 1039–1069, Nov. 2003.

[38] Z. R. Bogdanowicz, A. Tolo, 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.

[39] H. Liu, L. Yu, H. Yang, Z. Zhou, A. Traffic Control, and C. Navigation CollegeAir Force Engineering University Xi’an 710051, “Self-adapting radiation control method for RFS in tracking,” J. Syst. Eng. Electron., vol. 29, no. 2, pp. 244–255, Apr. 2018.

Qiang Fu was born in Shaanxi, China, in 1988. He received the Ph.D. degree from Air Force Engineering University. He is currently a Lecturer with the College of Air and Missile Defense, Air Force Engineering University. His research interests include intelligent information processing, intelligent control, and fuzzy sets.

Cheng-Li Fan was born in Sichuan, China, in 1988. She received the Ph.D. degree from Air Force Engineering University, in 2015. She has been working as a Postdoctoral Researcher with Air Force Engineering University, since January 2017, where she is currently a Lecturer with the College of Air and Missile Defense. Her research areas include decision making, operational research, fuzzy sets, and intelligent information processing.

Yafei Song (Member, IEEE) was born in Henan, China, in 1988. He received the Ph.D. degree from Air Force Engineering University, in 2015. He has been working as a Postdoctoral Researcher with Air Force Engineering University, since April 2017, where he is currently a Lecturer with the College of Air and Missile Defense. His research interests include pattern recognition, intelligent information processing, and evidential reasoning.

Xiang-Ke Guo was born in Henan, China, in 1980. He received the M.S. degree from Air Force Engineering University, in 2007, and the Ph.D. degree from the School of Electronics and Information Engineering, Beihang University, in 2018. His current research interests include target tracking and intelligent information processing.