Design and implementation of intelligent game system based on Reinforcement Learning——wargaming as an example

YUXIANG SUN* and BO YUAN, School of Electronics, Computing and Mathematics, University of Derby, China
BIN LI, School of Management and Engineering, Nanjing University, Nanjing 210023, China
YIHUI PENG, School of Management and Engineering, Nanjing University, Nanjing 210023, china
BOJIAN TANG, School of Management and Engineering, Nanjing University, Nanjing 210023, china
XIANZHONG ZHOU, Nanjing University, China

The field of intelligent game confrontation has become one of the hot areas of current research. Taking the construction of intelligent wargaming system as a typical example of intelligent game, this paper analyzes the modeling elements of a wargaming system, including wargame elements, wargame rules, and intelligent interface design, and constructs the overall architecture of intelligent wargaming system. The intelligent algorithm of reinforcement learning based on A3C is studied, and the reward setting of the reinforcement learning training process is improved. The state input, algorithm driving process and action output process of intelligent wargame environment are defined. Finally, the theory and work of the system proposed in this paper are verified by the self-developed intelligent wargaming system. This work provides a feasible path for the design and implementation of intelligent game system based on reinforcement learning, and provides a basic platform for the future research of intelligent game confrontation based on reinforcement learning.

CCS Concepts: • **Computer systems organization** → **Embedded systems**; *Redundancy*; Robotics; • **Networks** → Network reliability.

Additional Key Words and Phrases: Intelligent game, intelligent wargame, reinforcement learning and system design

ACM Reference Format:
Yuxiang Sun, Bo Yuan, Bin Li, Yihui Peng, Bojian Tang, and Xianzhong Zhou. 2018. Design and implementation of intelligent game system based on Reinforcement Learning——wargaming as an example. *Proc. ACM Meas. Anal. Comput. Syst.* 37, 4, Article 111 (August 2018), 12 pages. https://doi.org/10.1145/1122445.1122456

1 INTRODUCTION

In recent years, artificial intelligence technology has made great progress, especially in the field of intelligent games. In 2016, AlphaGoand Lee Sedol had a go game that attracted people's attention. In the end, the AI AlphaGo won the human race with a 4:1 result, which set off a wide range of hot discussions in the society and promoted another development wave of AI technology [14][11]. Then, AlphaGo's development team Deepmind struck while

*Both authors contributed equally to this research.

Authors' addresses: Yuxiang Sun, sunyuxiangsun@126.com(Y.S.); Bo Yuan, b.yuan@derby.ac.uk, School of Electronics, Computing and Mathematics, University of Derby, China, 43017-6221; Bin Li, School of Management and Engineering, Nanjing University, Nanjing 210023, China; Yihui Peng, School of Management and Engineering, Nanjing University, Nanjing 210023, china; Bojian Tang, School of Management and Engineering, Nanjing University, Nanjing 210023, china; Xianzhong Zhou, Nanjing University, China, zhouxz@nju.edu.cn(X.Z.).

2476-1249/2018/8-ART111 $15.00
https://doi.org/10.1145/1122445.1122456

the iron was hot and made a further significant breakthrough in the "StarCraft" game, and successfully developed Alphastar [15][16]. China Tencent AI Lab uses deep reinforcement learning technology to build "Awareness AI" in the virtual environment of "King's glory" game, and develops a high expansion and low coupling reinforcement training system, so that AI can have the ability to attack, induce, defend, cheat and continuously release skills [18][19]. Although intelligent game system has made remarkable achievements, there are still many problems to be further studied. Although the concept of artificial intelligence was put forward as early as 1956, due to the lack of computer performance and theoretical basis, artificial intelligence is far from being able to challenge human thinking [10]. With the gradual deepening of intelligent research, the realization of various algorithms and the emergence of AlphaGo in Go[7][3][2], it is a trend to study intelligent game system. Moreover, intelligent assistant decision-making is a bottleneck problem that restricts the upgrading of intelligent game system, which can not be ignored or even need to be solved every minute. Due to the characteristics of intelligent game system, the algorithm effect of deep learning or reinforcement learning still has a lot of room to improve. Taking the most classic game system "wargaming" as an example, this paper briefly describes the construction idea and simulation verification of the intelligent game system based on reinforcement learning. The algorithm model designed in this study also provides a way for other researchers to design an intelligent wargame system suitable for complex environment. Firstly, the general architecture of intelligent wargame system is established, and the function of each module is explained. Aiming at the core module of intelligent wargame system, the intelligent decision algorithm model is established, and the modeling idea is verified by typical experimental environment. Among them, the intelligent decision-making model is driven by reinforcement learning represented by A3C algorithm, which verifies the feasibility of intelligent decision-making algorithm model in intelligent wargaming system from principle and practice.

2 ENVIRONMENT MODELING OF INTELLIGENT WARGAME SYSTEM

2.1 Design of components of intelligent wargame system

The composition of intelligent wargame system must include the basic elements. In order to ensure the normal input-output of wargame system and the orderly advance of intelligent game, wargame system should include four basic elements: wargame system chess pieces, wargame system map, wargame system confrontation rules and wargame system scenario.

(1)Wargame system chess pieces

In the wargame system, the purpose of chess pieces is to represent the actual game unit, which can represent the game unit in the game, or take the formation as the basic unit, and the relevant parameters of the game unit need to be noted, such as unit number, unit number, attack ability, protection ability, mobility value and other main information.

(2)Wargame system map

The main function of wargame system map is to simulate the actual geographical situation, which needs to truly reflect the geographical conditions and confrontation scenarios of plains, highways, mountains, jungles, lakes, oceans, rivers and so on [12]. At present, there are many ways to draw maps. One is to map the actual geographical environment and restore it according to the scale. The other is to grid the map and abstract it to a certain extent. Because the gridded map can be easily understood by the machine, and can better restore the actual geographical conditions and confrontation scenes, while the map scaled by scale is not easy for the machine to understand, the map of the electronic wargame system generally chooses the gridded map. In this case, the grid in the gridded map is also called the chess grid, In wargame system, the minimum unit of chess action is chess grid, and the minimum unit of geographical conditions and confrontation scene simulation is also chess grid. At present, the grid map basically adopts hexagon as the choice of grid [8], the reason is that it can be closer to the real situation. In the real confrontation environment, the movement rules of confrontation units

are not limited by the grid shape, but can choose the direction of 360 degrees. Therefore, in order to be close to the real situation, the grid shape should support more movement directions of chess pieces. As we know, only regular triangles, squares and regular hexagons can cover the whole plane without gaps. When they are used as chess grids, the number of moving directions they can choose is three, four and six respectively. Therefore, it is more appropriate to choose regular hexagons as map grid shapes.

(3)Rules of wargame system confrontation

If chess pieces and maps are the flesh and blood and skeleton of wargame system, then wargame rules are the soul of the game and the most important elements of the game. The movement of all chess pieces and the use of maps are inseparable from wargame rules [5]. The main function of rules is to standardize the wargaming and make it orderly. Rules can make the two sides involved in wargaming carry out a series of actions such as maneuver and confrontation under a set of clear and specific provisions. There are two main ways to formulate the content of wargame rules: one is summarized from the past historical experience, the other is from the abstract summary of confrontation and simulation data, which is the concentrated reflection of a large number of research results on game confrontation. In most cases, wargame rules are divided into two parts, one is deduction rules, the other is adjudication Rules [4]. The application scope and effect of these two parts of rules are not the same. Deduction rules focus on standardizing the behavior of the game, explaining how to play the game, such as the attack rules of the chess pieces, the mobility rules of the chess pieces, the rules of the chess pieces getting on and off the bus, the rules of the chess pieces hiding and masking, and so on. All these belong to the deduction rules of the game. The ruling rules focus on defining the basis of confrontation ruling of chess fighting, ruling on the damage caused by chess pieces of both sides in the process of fighting, and finally determining the battle damage of both sides' forces and the victory or defeat of the battle after the end of the fighting process [6].

(4)Wargame system scenario

The main connotation of scenario is to conceive in advance of the situation of deduction, the target of confrontation between the two sides, the action plan of confrontation and the development of the process, etc., and to divide the scenario of filing a case, the basic scenario and the supplementary scenario. What needs to be clear is that the scenario of wargaming should be based on chess pieces, maps and rules, describe the scenario background, give the initial confrontation situation, game objectives, research plans, etc., and judge the action first and then, the number of deduction and the final victory or defeat.

2.2 Basic rule design of deduction

Rules are a system to limit and regulate the behavior of chess pieces in game confrontation, which will greatly affect the path selection and game process. The decision made in the rule system is an important factor to determine the outcome of the deduction [16]. Therefore, the design of intelligent wargame system must be based on the support of basic rules. This paper takes wargaming system as an example, encapsulates the corresponding basic functions into corresponding basic function functions through program functions, and then realizes the algorithm of intelligent wargame through the call of basic function functions, and finally realizes the establishment of intelligent engine. The main functions are as follows.

(1)Moving function Initialize the starting position, assign the value in the scenario, calculate the X and Y coordinates of each chess piece, obtain the coordinates of the surrounding hexagonal grid, and then select one of the obtained hexagonal grid coordinates to assign the value, and then move the coordinates. The moving direction includes seven directions: East, West, Northeast, Northwest, Southeast, Southwest and Static. In the process of moving, the organic power loss, the specific loss value refer to the table of power loss, take the tank chessman as an example. Each tank has two maneuvers per round. (2)Shooting reward integral function Fire the enemy's chess pieces, obtain the coordinates of the enemy's chess pieces, and then judge whether the enemy's chess pieces exist after shooting. If they exist and the coordinates correspond to the coordinates of the enemy's chess

Table 1. power loss of different terrain

Terrain	Consumption value
Flat ground	-1
Road	-0.5
Forest	-2
Hide	-1

Table 2. Shooting rules

Number of rules	Rule details
Rule 1	Before shooting, judge whether to shoot according to the intervisibility rule in the whole situation
Rule 2	The longest firing distance of the tank is 8 squares, each round, the tank fires directly according to the pieces in turn
Rule 3	Distance: influence level – 1-5: 3 | 6-8: 2
Rule 4	If the target elevation difference is > = 40, the random level is - 3 for the attacker and + 2 for the defender= 30, attacker - 2, Defender + 1; > 3= 20, > = 10, attacker-1
Rule 5	If the defender is in the forest, the attacker's random number is - 2
Rule 6	When there are four barriers around in special terrain, the random number on the attacking side is - 1

pieces, the corresponding reward points will be obtained. Otherwise, no points will be scored. (3)Firing function Get the coordinate position of the chess pieces, judge whether the enemy chess pieces can be observed by calling the visual function, if the observed distance can be shot, set the strike effect according to the distance between the enemy and the target. Moreover, shooting will be affected by random number to simulate the randomness of confrontation. Shooting is affected by distance, terrain, intervisibility, randomness and other aspects, which are reflected in random numbers. See table 2 for details.

(4)Get adjacent coordinate function Input the X and Y coordinates of the chessmen, representing the coordinates of the hexagonal lattice, and output the list to represent the coordinates of the surrounding hexagonal lattice in the form of a list.

(5)Query the distance between two hexagonal lattices

Enter the coordinates of x0, Y0, X1 and Y1 as int, which represents the coordinates of the starting hexagonal lattice and the ending hexagonal lattice, and output the distance between the two hexagonal lattices.

(6)Function of getting chess piece state information

The current coordinates of the chess pieces and the turn maneuver state are obtained through the function.

(7)Check whether the pieces can observe the opponent's pieces

Enter the status information of the opponent's chess pieces. The opponent's chess pieces can be observed and output true, but not output false. The confrontation rule of the whole intelligent wargame is that the red and blue sides confront each other. The pieces of both sides can move, cover, direct fire and indirect fire. The maneuver refers to inputting x, y coordinates, representing the coordinates of adjacent hexagonal grid, outputting the effect, and moving the pieces. Shadowing is to ensure that the chess pieces enter the hidden state, which is not conducive to being attacked. Direct fire is to input the coordinates of the enemy's pieces, output the corresponding shooting

effect and shoot the enemy's pieces. Input x, Y represents the target hexagonal grid coordinates, output effect, aiming at the target hexagonal grid.

(8)Rules of adjudication

Rules of adjudication. After each shooting, make a ruling and choose according to the result: invalid, damaged, suppressed. Each tank unit has three teams, and the damage result is selected from 1-3. If it is suppressed, the opponent's pieces will not be able to move and shoot for one turn and return to normal in the next.

After the deduction of each game, the total score of one party = task completion score + parameter a × the resulting score (damage score + strike score), and judge the outcome. Set a parameter a to adjust the proportion of task score and achievement score in the total score. When the proportion of task points is high, the calculation effect of the algorithm model is different from that of the result points. The former results in a higher winning rate, while the latter results in a higher battle loss ratio. At present, for the current research, the winning rate is the most important evaluation factor, so the task completion score is relatively high.

2.3 Core interface design of intelligent game engine

(1)Environment loading interface The intelligent wargame interface should include environment loading module to ensure that each loading scenario can start the relevant environment.

(2)Environment reset interface

The interface ensures that the environment can be reset, the game can be restarted, and the observations can be returned in each epic. In reinforcement learning algorithm, agents need to constantly try, accumulate experience, and then learn good actions from experience. An attempt is called a track or an epoch. Every attempt has to reach the end state. After an attempt, the agent needs to start from the beginning, which requires the agent to have the function of reinitialization. This is what the function reset () does.

(3)Environment Rendering interface

In each step, env.render() refreshes the screen. The render() function plays the role of the image engine here. The two essential parts of a simulation environment are physical engine and image engine. The motion law of objects in the environment is simulated by the physical engine. The image engine is used to display the object image in the environment. In fact, for the reinforcement learning algorithm, this function can not. However, in order to display the state of objects in the current environment, it is necessary for image engine. In addition, it is convenient for us to debug code by adding image engine.

(4)Execute action interface

Env. step (action) reinforcement learning algorithm needs to return the state information after the action is executed. According to the Bellman equation, it needs to return the state and return value. The state-space of wargaming can be defined as the position state coordinates x and y, including the real-time state (maneuver, concealment and Design) of chess pieces, forming the state space of wargaming. Among them, the mobile directions of chess pieces are south, north, northeast, northwest, southeast, southwest and static, which are defined as $0 \sim 6$ respectively. The shooting state of chess pieces in one of the squares is shooting or not shooting, so it can provide the necessary basic requirements for the use of deep reinforcement learning technology.

3 ARCHITECTURE CONSTRUCTION OF INTELLIGENT WARGAMING DEDUCTION SYSTEM

The environment of the intelligent wargaming system designed in this paper is as follows, mainly represented by hexagonal grid-specific terrain. The higher the terrain elevation is, the darker the color is.Black represents second-class highway, red represents the first-class highway, and shadow represents urban residential area, which is conducive to concealment.

The intelligent wargaming system consists of three layers: basic support layer, simulation platform layer and typical application layer. The overall architecture of the intelligent wargame simulation platform is mainly

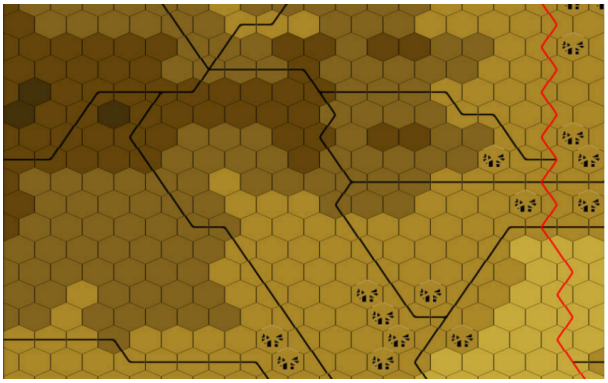

Fig. 1. Scenario display of intelligent wargame system

supported by AI wargame model system and database, with map editing , chess editing , rule editing , scenario editing , confrontation planning management , wargame situation display as the specific functions of the simulation platform, and the intelligent push engine interface supports two types of typical applications of wargame simulation platform. The framework mainly includes three main layers: typical application, simulation

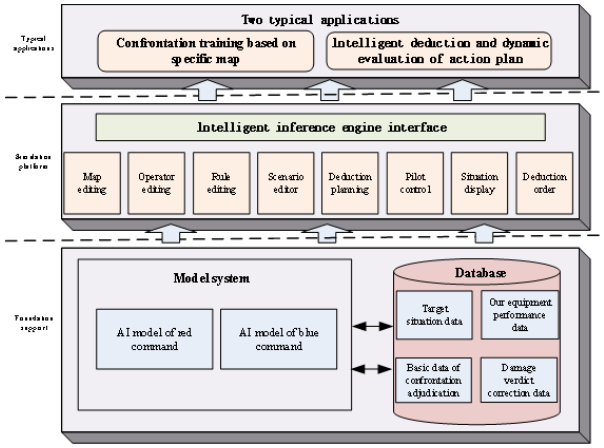

Fig. 2. Framework of intelligent wargaming system

platform and foundation support,The overall construction of intelligent wargame system architecture.Among them, the database system is mainly composed of target situation data, equipment performance data, confrontation decision basic data set and damage decision correction data. Target situation data is responsible for acquiring confrontation situation target data, and preparing for subsequent input data processing after acquiring opponent situation data [17]. The performance data of our equipment includes the assignment of the performance index data of our chess pieces, which is used as the data support for intelligent decision-making. The basic data of confrontation adjudication is used to evaluate the results of confrontation between the two sides in the game, and the random data value is introduced. The rule of adjudication is the basic criterion and core for the implementation of wargaming, and is the summary and induction of past combat experience. The Adjudication

Table 3. Status information settings are shown in the table below

Input information	Corresponding status information
0-5	The elevation of 6 directions around the current chess piece
6-11	Map types of 6 directions around current chess pieces
12-15	Current remaining health of our chess pieces
16-17	Current map capture point
18-19	Current enemy's chess HP
20	Whether the current chessman has visibility with the enemy

Rules in wargaming are the adjudication methods and regulations based on historical data and the principle of probability and statistics. The damage judgment correction data is used to further modify the relevant results after the end of the game.The model system is mainly based on the intelligent decision algorithm driven by reinforcement learning, which is the core of the intelligent game deduction system[13].

3.1 Design of intelligent decision model

With the proposal of reinforcement learning method and continuous deepening of research, it has gradually changed from the sample learning mode based on massive data to the evolution mode of autonomous learning, and adopted the "left and right fighting" technology in the small sample or no sample environment [1]. In this section, an intelligent decision-making model based on reinforcement learning is designed to realize autonomous decision-making and autonomous game confrontation of intelligent wargame system. The intelligent wargaming system designed in this paper includes the selection module of reinforcement learning algorithm, which can select the intelligent algorithm to make intelligent decision. Therefore, it is necessary to filter the state variable space and action variable space of the simulation environment to obtain the state space S = {S1, S2,..., Sn} and action space A = {A1, A2,..., Am} suitable for the input and output of the algorithm. Intelligent algorithm includes three typical types: DQN algorithm driven, A3C algorithm driven and PPO algorithm driven.This paper takes A3C algorithm as an example.

In the intelligent wargame environment, the input state is the (OBS) observation that can be observed by our chess pieces. The OBS is input into the neural network of reinforcement learning algorithm in the form of a list, which is used to train the neural network and get the estimated value. For reinforcement learning, we need to input the global OBS of wargame, which is represented by $\gamma(s_i)$ in this paper, including the height around the chess pieces, map type around the chess pieces, the remaining blood of both chess pieces, the number of both sides, double position information, the position information of control points, whether the chess pieces are intervisibility, global state observation quantity $\gamma(s) = \gamma(s_1) \bigcup \gamma(s_2) \bigcup \cdots \bigcup \gamma(s_n)$ that is, the union of all local state observations.

4 ALGORITHM DRIVEN PROCESS

The training model of multi-agent is shown in the figure. The action decision-making of chess pieces is based on actor network and critical network, and the chess pieces are named agent$_i$(I=1,2,. . . ,n).Local state observation is a set of situation information that can be observed by each chess piece. The input of each critical network takes into account the action AI of the corresponding chess piece and the global state measurement after the action, and each chess piece has its own reward value. When the actor network of each chess piece is updated, the state estimation difference of critical network output will be input to update, so as to adjust the actor network.

Table 4. Action information table

Input information	Corresponding status information
0	The pieces move to the left
1	The pieces move to the right
2	The pieces move up and left
3	The pieces move down to the left
4	The pieces move up and right
5	The pieces move down and to the right
6	Pieces fire at enemy pieces
7	The chess pieces are still
8	Chess concealment

In the intelligent wargame platform designed in this paper, our algorithm adopts the action decision algorithm of distributed execution and centralized training, as shown in the figure below. When training, critical and actor are trained by centralized learning, and actor only needs to know local information to execute. At the same time, each agent trains multiple strategies and optimizes them based on the overall effect of all strategies to improve the stability and robustness of the algorithm. Each actor and critical has an eval net and a target net. The Q value generated by critical state estimation network and the value calculated by Bellman equation generated by state reality network is subtracted to calculate the loss value. Then the loss value is used to update the critical network parameters in reverse, and then the critical is used to guide the optimization of actor in reverse. Finally, the actor is used for action and output. The purpose of updating actor network is to adjust the probability of action output to get higher value.

4.1 Action output

The output action in wargame can obtain new situation information and return value after executing action, and then feed back to Q network to further update network parameters. The specific output actions include moving, shooting, stationary and hiding.

Due to the uncertainty of action decision and firing object, and the complexity of scenario map state space, the convergence speed of training is slow, and it is likely to be difficult to win for a long time [9], resulting in a large number of meaningless training. According to the characteristics of the above environment, this paper formulates the tactical decision rules as shown in the figure. In this paper, based on the decision generation mechanism of comprehensive rules and Multi-Agent Reinforcement learning algorithm, an online evaluation system σ of Multi-Agent Reinforcement Learning Algorithm in the process of confrontation is constructed. After obtaining the current situation information, A3C algorithm outputs the corresponding decision scheme, and then uses σ to evaluate whether the current action is effective. If the value is greater than σ, the action is output according to the reinforcement learning A3C method. If the value is less than σ, the action is output according to the scheme in the expert rule base. σ value calculation includes R_{win} win return value.S_{tl} red tank survival score. G_{tl} Red capture hold point score.K_{tl} red destroy opponent's tank score. R_{lose} win return value. S_{t2} blue tank survival scoreG_{t2} blue capture hold point score.K_{t2} blue correction coefficient destroy opponent's tanker score,α_1 correction coefficient respectively. According to the experts' experience, the critical value of α is determined. If α exceeds the critical value, the reinforcement learning A3C decision algorithm is selected. Otherwise, the action is selected according to the experts' rules. This can ensure that the action output will not be difficult to converge for a long time, and avoid a lot of meaningless training.The formula is as follows.

Table 5. Reward setting table

Awards	Reward value
Capture (victory)	+100
Closer to the key point	+1
Away from the key point	-0.1
Every move	-0.01
Attack enemy pieces	+0.5 * enemy blood loss
Hit by enemy pieces	-0.6 * self blood loss
Annihilate enemy pieces	+50
Be annihilated by enemy pieces	-60
The other side won	-200

$$\sigma = \begin{cases} \sigma + \alpha_1|R| + \alpha_2|S_{tl}| + \alpha_3|G_{tl}| + \alpha_4|K_{tl}|, & Win \\ \sigma - \alpha_1|R| - \alpha_2|S_{tl}| - \alpha_3|G_{tl}| - \alpha_4|K_{tl}|, & Loss \end{cases} \tag{1}$$

4.2 Reward value stting

In reinforcement learning, reward plays the role of supervising the training process, and the agent optimizes the reward according to the reward. In the simulation environment discussed in this paper, wargaming environment only makes rule judgment and engagement decision for actions, and does not provide any reward information after maneuver or engagement, Only when our chess pieces reach the capture control point or completely annihilate the enemy's chess pieces, the victory information will be sent; when the enemy's chess pieces reach the capture control point or completely annihilate our chess pieces, the victory information will be sent; when the enemy's chess pieces reach the capture control point or completely annihilate our chess pieces, the failure information will be sent; that is, there is no reward in every step of the training process [20]. However, this situation will lead to the training process most of the time is no reward, this sparse reward will lead to the training result is difficult to converge, training efficiency is very low. In view of this situation, this paper adds an additional reward mechanism, that is, the closer the chess piece is to the control point, the higher the reward value will be, and the farther the chess piece is to the control point, the lower the reward value will be. In order to prevent the chess piece from infinite movement and difficult to converge, this paper deducts a small amount of reward value for each movement to prevent the situation of unable to converge. The specific reward settings are shown in the table.

5 SIMULATION AND VERIFICATION OF INTELLIGENT WARGAMING SYSTEM

Combined with the elements of wargame, wargame rules, system architecture and intelligent decision-making model, this paper constructs an intelligent wargaming system, conducts intelligent game confrontation in the system, generates confrontation data, and verifies the feasibility of the design idea of this intelligent game system. Scenario Description: the intelligent game wargame system is mainly divided into red and blue sides. The winning rule is that one side reaches the control point first, or destroys all tanks of the other side. As the basic unit of the map, each hexagon has a number and elevation. The higher the elevation, the darker the color. The red solid line represents the first-class highway, and the black solid line represents the second-class highway. The shadow part in the hexagonal grid represents the urban residential area. Tanks in the urban residential area are not conducive to the other party's discovery, but also conducive to concealment and improve the survival rate. In the game

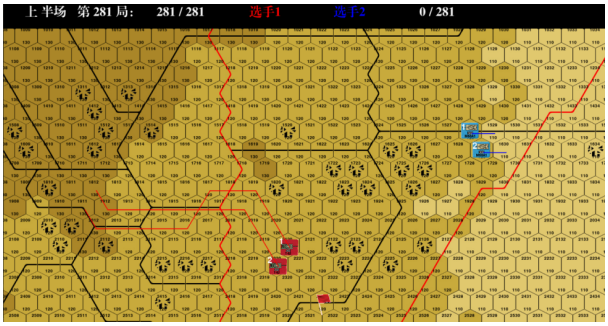

Fig. 3. Deduction effect of intelligent wargaming system

deduction, the red and blue sides are used to conduct the deduction. The blue side is driven by reinforcement learning algorithm, while the red side is driven by rules. Take 100 innings as the unit to count the winning rate. The detailed winning rate is shown in the figure 5.

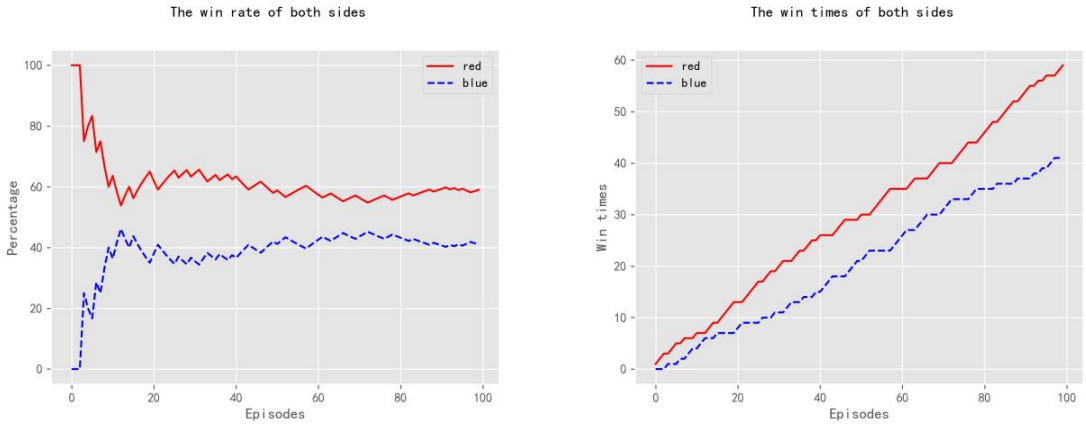

Fig. 4. Winning rate(a)(b)

(a)Win rate:the red side is the AI of DQN intelligent algorithm and the blue side is rule-based AI.(b)Win times: the red side is the AI of DQN intelligent algorithm and the blue side is rule-based AI; The winning rate and the number of wins for the red and blue sides. The first round wins so one side starts from 1 and the other from 0. In the course of 100 games, the winning rate of the blue side controlled by A3C reinforcement learning intelligent algorithm is 59%, and the winning rate of the red side based on knowledge base and rules is 41%. The details of the confrontation between red and blue are shown in the figure. Through the game against the details score can better verify the feasibility of the game system design. The survival score was 1930 in red and 970 in blue. The red side get goal scored 4821 points and the blue side scored 3250 points. The red side scored 4330 points and the blue side 2850 points for destroying enemy pieces. On the whole, the blue side AI controlled by reinforcement learning algorithm mainly wins by seizing the control points, which shows that reinforcement learning algorithm is more inclined to the fast and efficient way of winning, while the red side AI based on rules wins by attacking the other side. On the whole, reinforcement learning algorithm has more advantages.(1) The get goal score of both sides (Red: A3C); (2) the kill score of both sides (Red: A3C); (3) the survive score of both sides(Red: A3C).

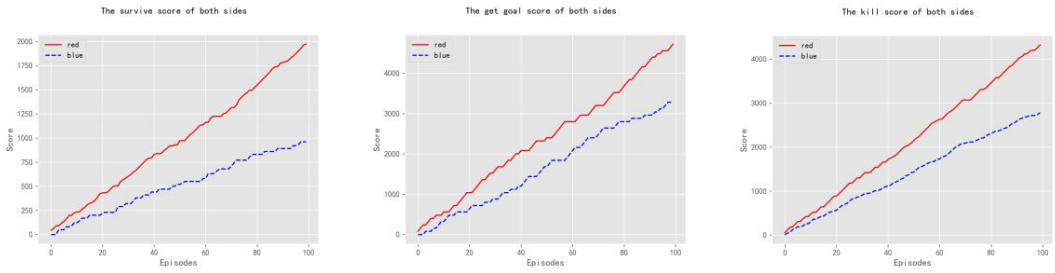

Fig. 5. Winning rate (1)(2)(3)

6 CONCLUSION

Intelligent game has become a hot issue in current research. In this paper, taking intelligent wargaming as an example, the basic requirements for environment modeling of intelligent wargaming system are introduced in detail. This paper analyzes the basic elements needed to build the intelligent wargame system and the basic rules of wargaming, establishes the core engine interface of the intelligent wargaming system, and establishes the architecture of the intelligent wargaming system. In this paper, the core of intelligent decision-making model of intelligent wargaming is analyzed in detail, and the reinforcement learning intelligent deduction engine based on A3C is constructed. Finally, the work is verified by the experimental simulation of intelligent wargaming system. The work of this paper can provide a feasible path for the construction of intelligent game countermeasure deduction system, and provide basic work for the countermeasure research in the field of intelligent game.

REFERENCES

[1] Noam Brown and Tuomas Sandholm. 2019. Superhuman AI for multiplayer poker. *Science* 365, 6456 (2019), 885–890.
[2] Xiangrui Chao, Gang Kou, Tie Li, and Yi Peng. 2018. Jie Ke versus AlphaGo: A ranking approach using decision making method for large-scale data with incomplete information. *European Journal of Operational Research* 265, 1 (2018), 239–247.
[3] Yutian Chen, Aja Huang, Ziyu Wang, Ioannis Antonoglou, Julian Schrittwieser, David Silver, and Nando de Freitas. 2018. Bayesian optimization in alphago. *arXiv preprint arXiv:1812.06855* (2018).
[4] Vincent Corruble, Charles AG Madeira, and Geber L Ramalho. 2002. Steps toward Building of a Good AI for Complex Wargame-Type Simulation Games.. In *GAME-ON*.
[5] Stephen L Dorton, LeeAnn R Maryeski, Lauren Ogren, Ian T Dykens, and Adam Main. 2020. A wargame-augmented knowledge elicitation method for the agile development of novel systems. *Systems* 8, 3 (2020), 27.
[6] Charles Grant. 1979. *Wargame tactics*. Hippocrene Books.
[7] Scott R Granter, Andrew H Beck, and David J Papke Jr. 2017. AlphaGo, deep learning, and the future of the human microscopist. *Archives of pathology & laboratory medicine* 141, 5 (2017), 619–621.
[8] Lawrence G Jones and Anthony J Lattanze. 2001. *Using the architecture tradeoff analysis method to evaluate a wargame simulation system: A case study*. Technical Report. CARNEGIE-MELLON UNIV PITTSBURGH PA SOFTWARE ENGINEERING INST.
[9] Zhang Yongliang Chen Tiande Li Chen, Huang Yanyan. 2021. Multi agent decision making method based on actor critical framework and its application in wargame [J]. Systems engineering and electronic technology. *Systems engineering and electronic technology* 43, 3 (2021), 755–762.
[10] R Mitchell, J Michalski, and T Carbonell. 2013. *An artificial intelligence approach*. Springer.
[11] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. 2015. Human-level control through deep reinforcement learning. *nature* 518, 7540 (2015), 529–533.
[12] David O Ross. 2003. Designing a system-on-system wargame. In *Enabling Technologies for Simulation Science VII*, Vol. 5091. International Society for Optics and Photonics, 149–153.
[13] Tongfei Shang, Kun Han, Jianfeng Ma, and Ming Mao. 2019. Research on self-gaming training method of wargame based on deep reinforcement learning. In *Proceedings of the 2019 International Conference on Artificial Intelligence and Computer Science*. 251–254.

[14] David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. 2016. Mastering the game of Go with deep neural networks and tree search. *nature* 529, 7587 (2016), 484–489.

[15] Oriol Vinyals, Igor Babuschkin, Wojciech M Czarnecki, Michaël Mathieu, Andrew Dudzik, Junyoung Chung, David H Choi, Richard Powell, Timo Ewalds, Petko Georgiev, et al. 2019. Grandmaster level in StarCraft II using multi-agent reinforcement learning. *Nature* 575, 7782 (2019), 350–354.

[16] Oriol Vinyals, Timo Ewalds, Sergey Bartunov, Petko Georgiev, Alexander Sasha Vezhnevets, Michelle Yeo, Alireza Makhzani, Heinrich Küttler, John Agapiou, Julian Schrittwieser, et al. 2017. Starcraft ii: A new challenge for reinforcement learning. *arXiv preprint arXiv:1708.04782* (2017).

[17] Brian Wade. 2018. The Four Critical Elements of Analytic Wargame Design. *Phalanx* 51, 4 (2018), 18–23.

[18] Deheng Ye, Guibin Chen, Wen Zhang, Sheng Chen, Bo Yuan, Bo Liu, Jia Chen, Zhao Liu, Fuhao Qiu, Hongsheng Yu, et al. 2020. Towards playing full moba games with deep reinforcement learning. *arXiv preprint arXiv:2011.12692* (2020).

[19] Deheng Ye, Guibin Chen, Peilin Zhao, Fuhao Qiu, Bo Yuan, Wen Zhang, Sheng Chen, Mingfei Sun, Xiaoqian Li, Siqin Li, et al. 2020. Supervised Learning Achieves Human-Level Performance in MOBA Games: A Case Study of Honor of Kings. *IEEE Transactions on Neural Networks and Learning Systems* (2020).

[20] Zhang Yongliang Chen Tiande Zhang Zhen, Huang Yanyan. 2021. Game confrontation algorithm of combat entity based on near end strategy optimization. *Journal of Nanjing University of technology* 45, 1 (2021), 77–83.