DDCL: A Dual Decision-making Continuous Reinforcement Learning Method Based on Sim2Real

Wenwen Xiao[1], Xinzhi Wang[1], Xiangfeng Luo[1,†] , Shaorong Xie[1]
[1]School of Computer Engineering and Science
[1]Shanghai University, Shanghai, 200444, China
{xww, wxz2017, luoxf, srxie}@shu.edu.cn

***Abstract*— Continuous reinforcement learning carries potential security risks when applied in real-world scenarios, which could have significant societal implications. While its field of application is expanding, the majority of applications still remain confined to virtual environments. if only a single continuous learning method is applied to an unmanned system, it will still forget previously learned experiences, and retraining will be required when it encounters unknown environments. This reduces the learning efficiency of the unmanned system. To address these issues, some scholars have suggested prioritizing the experience playback pool and using transfer learning to apply previously learned strategies to new environments. However, these methods only alleviate the speed at which the unmanned system forgets its experiences and do not fundamentally solve the problem. Additionally, they cannot prevent dangerous actions and falling into local optima. Therefore, we propose a dual decision-making continuous learning method based on Simulation to Reality (Sim2Real). This method employs a knowledge body to eliminate the local optimal dilemma, and corrects bad strategies in a timely manner to ensure that the unmanned system makes the best decision every time. Our experimental results demonstrate that our method has a 30% higher success rate than other state-of-the-art methods, and the model transfer to real scenes is still highly effective.**

KEYWORDS: Social Computing; Continuous learning; Reinforcement learning; Simulation to reality(Sim2Real).

I. INTRODUCTION

Continuous reinforcement learning plays a crucial role in our society by enabling the development of social software that enhances interactions within groups and improves the efficiency of human social activities [1] [2]. Continuous reinforcement learning involves acquiring skills through continuous interaction with a complex environment and building higher-level skills based on previously learned ones. The continuous reinforcement learning process is analogous to how babies learn to walk, where crawling is mastered first, followed by standing and eventually walking, as illustrated in Fig. 1. In essence, each new skill is built upon the old ones.However, there are still at least two challenges in continuous learning: avoiding catastrophic forgetting caused by neural networks and ensuring the malleability and stability of the models for migration to real-world scenarios.

The current approach to mitigate the forgetting problem of agents during the learning process is primarily through the priority experience replay mechanism [3]. However, this method requires frequent parameter adjustments and can result in reduced learning efficiency of the agent.

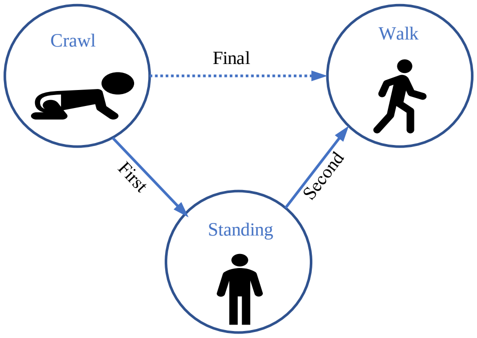


Fig. 1: Diagram of continuous learning for Babies

To ensure that the trained model of an agent in a virtual scene is steadily migrated to a real scene, the current mainstream approach uses adjusting while training to optimize the policy parameters of the agent in real time. However, this method requires a large amount of human resource cost and will become ineffective in the face of some complex scenarios.

In this paper, we propose a dual decision-making continuous learning method based on Simulation to Reality, which consists of three main stages: perception stage, decision stage, and execution stage.

In the perception stage, virtual data is first transformed into real data using existing data generation tools. Then, real data from the real scene is transformed into virtual data, and the resulting data is fused. Next, the fused data is passed through a semantic segmentation extractor to obtain a feature map, and object detection techniques are employed to extract entity category information in the scene. Finally, a semantic knowledge graph is constructed based on the feature map and entity category information, which serves as the agent's prior knowledge during the decision stage. This prior knowledge can reduce the agent's exploration of the scene and improve the efficiency of its decision-making.

In the decision stage, the agent's decision-making action is controlled by a dual decision-making mechanism consisting of continuous learning and body-of-knowledge control methods. The agent learns different strategies through trial and error with various environments, with each strategy

DOI reference number:10.18293/SEKE2023-053 †Corresponding author forming a skill. To prevent forgetting of the learned skills, periodic updating of the training environment is necessary. Body-of-knowledge control is utilized to help the agent escape from local optimum in policy learning. The decision to apply body-of-knowledge control is based on the evaluation results of the discriminator, which consists of success rate, error rate, and reward value indicators.

In the execution stage, the agent's actuators receive the optimal decision-making action determined by the discriminator evaluation to ensure compliance and safety, enabling the agent to quickly complete the task.

In summary, we propose a dual decision-making continuous learning method based on Simulation to Reality, which can effectively mitigate the problem of skill forgetting during the learning process of an agent. Moreover, this method is crucial for ensuring the stable migration of strategies learned by an agent in a virtual scene to a real scene.

The main contributions of our paper are summarized as follows:

1) We propose a dual decision-making continuous learning method based on Simulation to Reality, which effectively circumvents bad strategies and ensures the ability of continuous learning of the agent.

2) We improve the efficiency of the agent's search for unknown environments by introducing semantic knowledge graphs as prior knowledge in the perception stage.

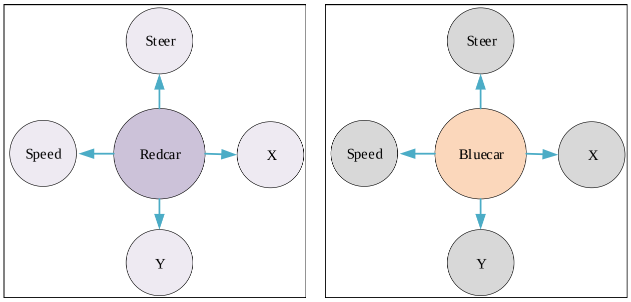
3) We migrated the trained model in the virtual scene to the real unmanned vehicle defensive scene, and the mobile vehicle can still complete the task smoothly.

II. Related Work

A. Continuous reinforcement learning

Continuous reinforcement learning is a method to address catastrophic forgetting in neural network learning, which is critical in improving the efficiency of agents in social activities[4] [5]. This approach allows agents to learn different skills at different times, improving their continuous learning ability and avoiding the need to retrain the agent for new tasks. As a result, the agent's learning efficiency is improved.

The mainstream methods are regularization, memory playback, parametric isolation, and integrated methods. Regularization methods are achieved by adding regular terms to the homeopathic function during training a new task and modifying the ratio of old and new data to reduce the rate of forgetting the agent. The memory replay method is to reuse the data that has been used before to reduce forgetting. The parameter isolation method is to assign different model parameters in different tasks of the agent training and freezes some model parameters in time according to the performance of the agent to ensure that the old model parameters occupy the majority[6]. The combined approach combines the above two approaches to form a new approach. For example, Buzzega combines regularization and memory replay to propose dark experience replay[7].



(a) Red vehicle knowledge graph (b) Blue vehicle knowledge graph

Fig. 2: Semantic knowledge graph

B. Mobile vehicle

Mobile vehicles have been applied across diverse industries, such as hospitals, factories, supermarkets, and hotels, profoundly impacting the social life of people today[8]. Among them, ground-guided vehicles stand out for their quick response time, fast speed, and high carrying capacity. However, most current mobile vehicles rely on rule-based control methods that are limited to simple scenarios[9] [10]. In complex scenarios, these methods exhibit poor performance and may even fail. Vision-based navigation and radar slam navigation are the two main rule-based control methods, both of which require environmental map information and suffer from low decision-making efficiency and poor migration[11].

With the rapid development of deep learning technology, continuous interaction between mobile vehicles and the environment via deep reinforcement learning has become a popular research direction for enabling autonomous decision-making[12]. To address the challenges of performance degradation and skill forgetting when trained models of mobile vehicles are migrated from virtual to real scenes, we propose a dual decision framework that successfully completes the red and blue vehicle defense task in real scenes with zero-shot transfer.

III. Methods and Analysis

A. Dual Autonomy Decision Framework

The dual autonomous decision-making framework plays a very important role in improving the continuous learning ability of mobile vehicles and reducing the speed of skill forgetting, laying the foundation for large-avoidance swarm intelligence, as shown in Fig. 3. The framework is mainly divided into the perception stage, decision stage and execution stage, and the role of each stage is also different. In the perception stage, it focuses on how to obtain feature maps and reduce the state space of the ground mobile vehicle; In the decision stage, it focuses on how to improve the autonomous decision-making ability of the ground mobile vehicle; In the execution stage, it focuses on how to execute the output actions of the decision stage smoothly.

Perceptual Stage: The perception stage is the basis of the dual autonomous decision-making framework. The main

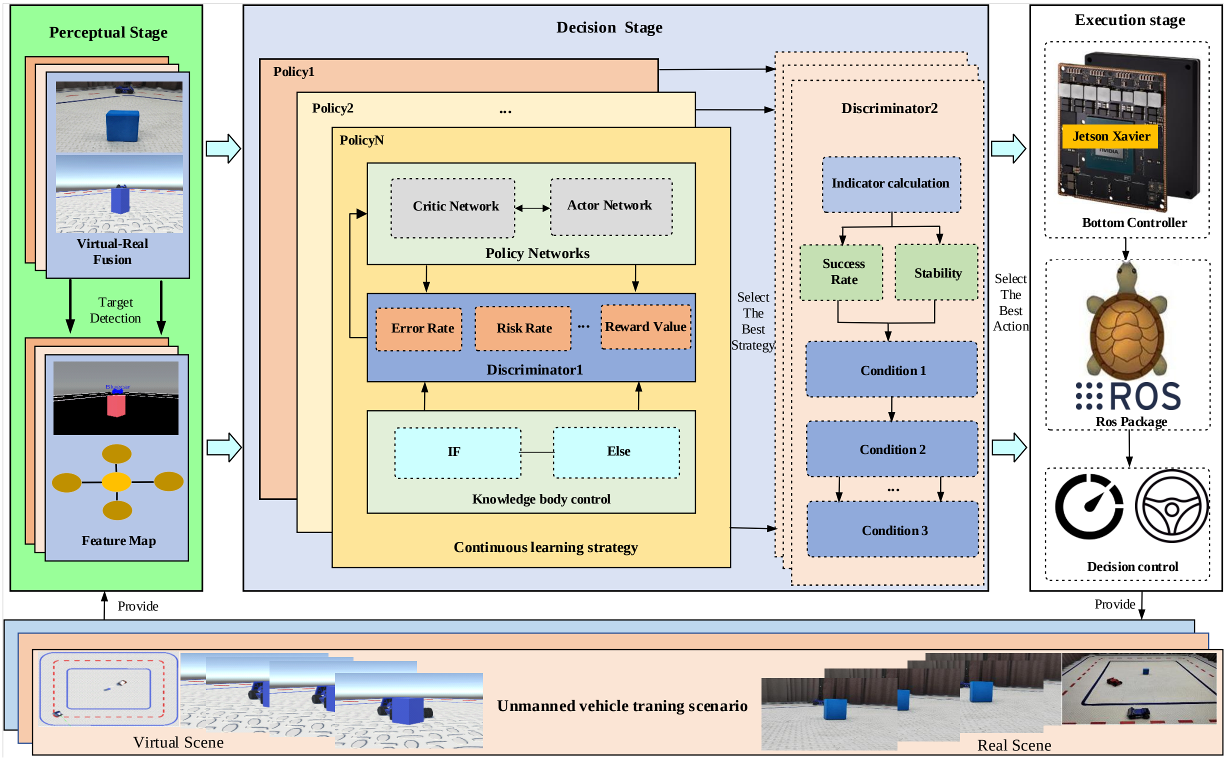


Fig. 3: The overall framework of dual continuous autonomous decision-making

work of the perception stage has two parts, namely image generation and feature map construction, as shown in Fig. 2. Image generation uses a generative adversarial network to generate images in the virtual scene into images in the real scene, generate virtual scene images from the original images in the real scene, and then mix the virtual images and real images for training to obtain an image generation model.

Decision Stage: The decision stage is similar to the brain of the dual autonomous decision-making framework, providing continuous learning capabilities for mobile vehicles. The decision stage mainly consists of a policy network, a body of knowledge controllers and two discriminators. First, the mobile vehicle simultaneously learns multiple continuous learning policies, and each policy network is dependent on the critic and actor networks. Then use the evaluation index in discriminator 1 to evaluate the policy output in the policy network. If the evaluation result is not good, it will directly switch to the body of knowledge controller to correct its policy parameters. Finally, the optimal action is obtained through the evaluation conditions and indicators in discriminator 2.

Execution Stage: The main function of the execution stage is to ensure that the mobile vehicle can control the movement of the mobile vehicle smoothly according to continuous action. The execution stage comprises the underlying controller, PID module(Proportional, Integral, Differential), robot operating system(ROS) and API interface. Through the above modules, it can be seen that the torque of the underlying motor is converted into continuous speed and direction data, where the value range of speed and direction is -1 to between 1.

B. Dual Decision Continuous learning algorithm (DDCL)

We proposed the dual-decision continuous learning method based on the Proximal Policy Optimization(PPO) algorithm[13]. The pseudocode of DDCL algorithm is shown in algorithm 1. Because only relying on this method when facing some complex scenes, it is easy to forget the previously learned experience or fall into the problem of local optimum, which makes the mobile vehicle unable to perform continuous learning. Therefore, to solve the above problems, we also use the knowledge body control method to assist the decision-making of mobile vehicles. On the one hand, it can improve the efficiency of decision-making, and on the other hand, it can avoid the training of mobile vehicles from scratch. The policy network structure of this method is shown in Fig. 4. The network update method of the actor and critic depends on the PPO algorithm[13].

To ensure that the red car agent can learn defensive and patrol strategies, it is necessary to set the reward function skillfully. The reward function formula is as follows:

$$R_{\text{total}} = \begin{cases} +10 & \text{when } R_{car} \text{ is intercepted} \\ +5 & \text{when } R_{car} \text{ is on patrol} \\ -10 & \text{when } B_{car} \text{ is attacked the target} \end{cases} \tag{1}$$

where R_{total} is the reward function for interacting with the environment.

According to the evaluation index in discrimination 1, the comprehensive strategy evaluation value P_{total} is obtained,

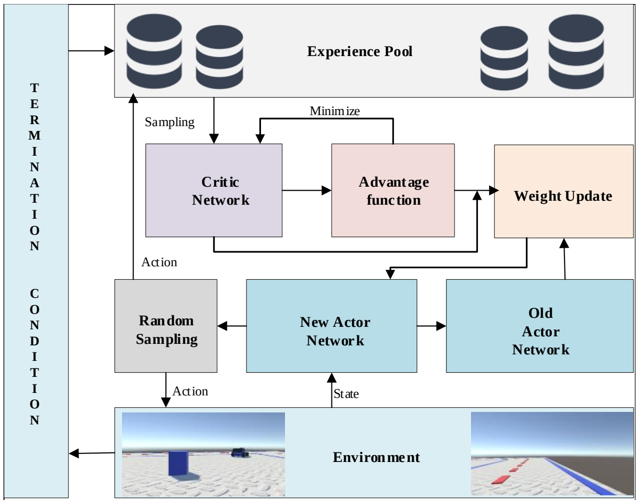


Fig. 4: Policy Network Structure

Algorithm 1 Dual Decision Continuous learning

```
Input: Initialize strategy, $S_1$, $S_2$, ... $S_i$. Initialize actor parameters: $A_1$, $A_2$, ... $A_i$. Initialize actor parameters:$B_1$, $B_2$, ... $B_i$.Initialize hyperparameters $T_1$and $T_2$.
Choose a strategy at random.
for $n$ episode do
    Update Actor Network Parameters:
    $a_i^{'} \leftarrow a_i - T_1 \Delta_{a_i} \mathcal{N}_{Actor}(a_i)$
    Update Critic Network Parameters:
    $b_i^{'} \leftarrow b_i - T_2 \Delta_{b_i} \mathcal{M}_{critic}(b_i)$
end for
Calculate the strategy evaluation value, see formula 2.
Calculate the action evaluation value, see formula 3.
```

TABLE I: Performance comparison in virtual scene

Algorithm	Success Rate	Risk Rate	Stablity
Random	8%	92%	BAD +
SAC[15]	60%	40%	GOOD -
DDCL(Ours)	**90%**	**10%**	**GOOD +**

and the optimal strategy is selected. The specific formula is as follows:

$$P_{\text{total}} = W_1 \cdot P_e + W_2 \cdot P_r + W_3 \cdot P_w \tag{2}$$

where Error rate P_e is represents the stationary time of the moving vehicle in each round; The risk rate P_r is represents the number of extreme actions of the moving vehicle in each round; The reward value P_W is represents the average reward value in each round.

According to the evaluation index in discrimination 2, the comprehensive action evaluation value A_{total} is obtained, and the optimal action is selected. The specific formula is as follows:

$$A_{\text{total}} = \eta_1 \cdot A_g + \eta_2 \cdot A_s \tag{3}$$

Where the action success rate A_g represents the number of times the vehicle swings in each round; The stability A_{sss} represents the offset of the moving vehicle from the centerline of the track in each round.

In summary, to prevent the mobile vehicle from forgetting the previously learned skills and improve the transferability of the model, it is necessary to adjust the strategy in time according to the actual situation of the comprehensive action and the evaluation value of the strategy. If the evaluation shows poor results, it's necessary to switch directly to the knowledge-based control mode.

IV. EXPERMENTS

A. Scenario description and tools

Scenario description: The goal of the red and blue vehicle defensive task is to allow the red vehicle to intercept the blue vehicle in time through the double continuous learning decision-making algorithm to ensure the safety of the guard target. The main entities in the red and blue offensive and defensive scene have a red car, a blue car, and a target. Among them, the agent controls the red vehicle through the double continuous learning algorithm, the knowledge body controls the blue vehicle, and some fixed decision-making mechanisms are artificially set. The defensive interception scenes of the red and blue sides construct the same scene in virtual and real, respectively, assuming that the dynamic models of the moving vehicles in the virtual and real scenes are the same or similar.

Red and blue of vehicle defensive tasks: Blue vehicle decision-making mode: blue team vehicles drive at a constant speed in the outer lane of the scene and perform patrol tasks. It launches an attack on the target every 5s. If it is intercepted by a red vehicle, it will exit the attack mode and continue to execute the patrol mechanism. Red vehicle decision-making mode: The red vehicle pays close attention to the movement of the blue vehicle in real-time. If the blue vehicle has already driven to the blue inner circle, the red vehicle will start to intercept until the blue vehicle exits the inner blue area.

Scenario tools: We used the unity virtual engine [14] to create a virtual scene of the game between red and blue. The GPU in the server is an NVIDIA GeForce RTX3090 graphics card, and the CPU is Inter core i7-9700. The control driver of the mobile vehicle is ROS 18.04 LTS, and the controller is Jetson Xavier NX. The flow chart of real car model migration is shown in Fig. 9.

B. Unmanned vehicles train in virtual scenarios

Feature map construction: The feature map is to obtain the feature vector through the method of feature extraction from the original image, which can process the original high-dimensional image information into a low-dimensional feature vector, thereby improving the decision-making efficiency of the mobile vehicle. The feature map is obtained by data processing the images in the virtual and real scenes through the existing yolov5 method. The feature map consists

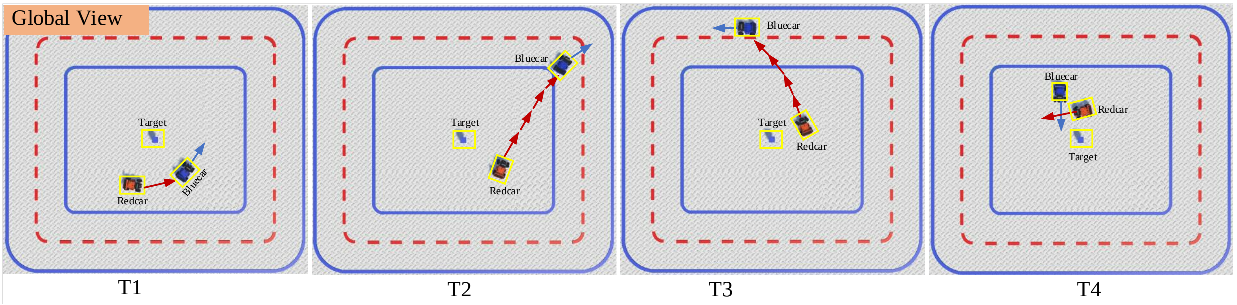
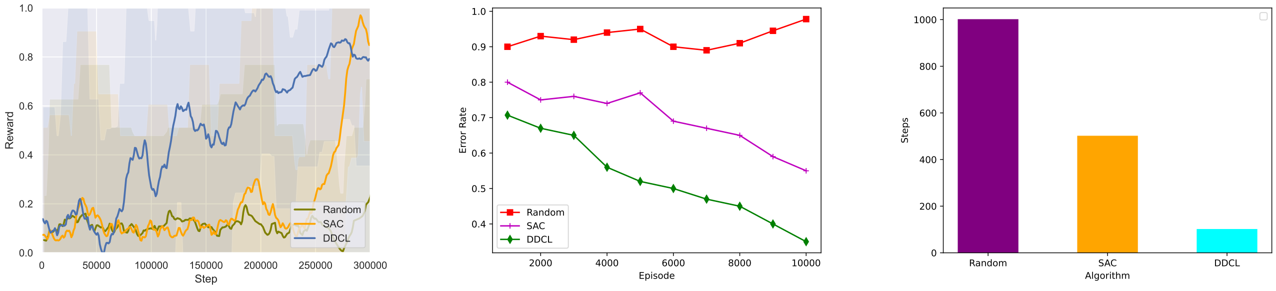


Fig. 5: The overall vision of the red and blue of vehicle defensive task in the virtual scene



(a) Average reward value

(b) Error Rate

(c) Episode Steps

Fig. 6: Red vehicle performance comparison

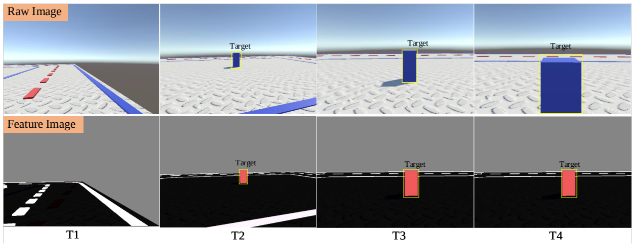


Fig. 7: The original and feature maps of the red vehicle at different times

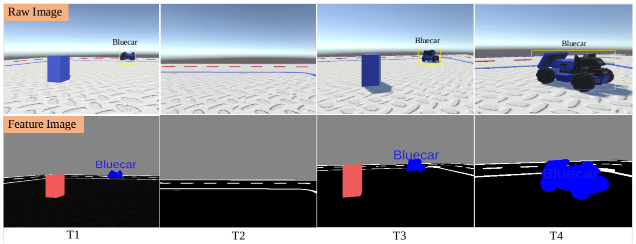


Fig. 8: The original and feature maps of the blue vehicle at different times

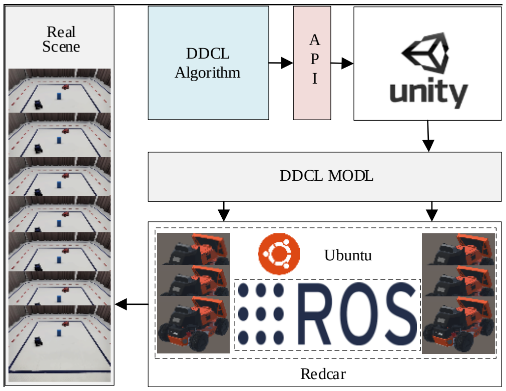


Fig. 9: Dual continuous learning algorithm model migration process

TABLE II: Performance comparison in real scene

Algorithm	Success rate	Risk Rate	Stablity
Random	1%	98%	BAD +
SAC[15]	20%	40%	BAD -
DDCL(Ours)	**60%**	**40%**	**GOOD -**

of a semantic segmentation map and an entity knowledge graph.The original and feature maps of the red and blue vehicles at different moments are shown in Fig. 7 and Fig. 8, respectively.

Decision Model Training: We connect the dual continuous learning algorithm to unity in the virtual engine for accelerated training. The overall picture of the red and blue of defensive vehicle tasks trained at different moments is shown in Fig. 5. Compared with the state-of-the-art algorithm, our proposed dual continuous learning algorithm has greater advantages in average reward value, error rate, success rate, risk rate and episode steps, as shown in Fig. 6 and Table I.

C. Unmanned vehicles verified in real scenarios

Migrate the model trained in the virtual scene to the real scene with zero-shot. The migration process is shown

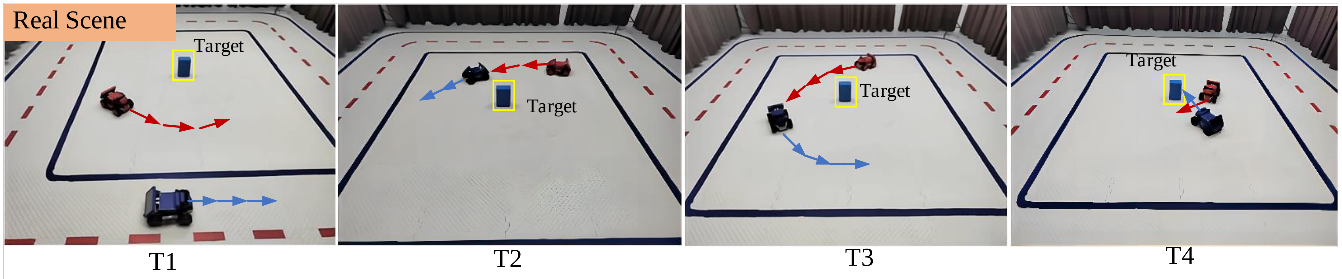


Fig. 10: The overall vision of the red and blue of vehicle defensive task in the real scene

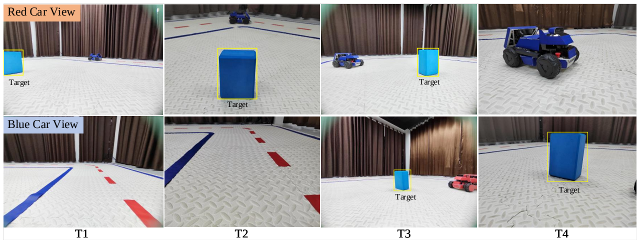


Fig. 11: Front view images of red and blue cars in real scenes

in Fig. 9. It can be seen from Table II that the DDCL algorithm proposed by us has a success rate of 60% in real red and blue of defensive vehicle tasks, which is better than state-of-the-art algorithms. The risk rate reaches 40%, mainly caused by the kinematic differences of real vehicles, the ground's friction coefficient and the light's intensity. Fig. 10 shows the running trajectories of the red vehicle and the blue vehicle in the real scene of the mobile vehicle at different times. The view of the red vehicle is shown in the upper row of Fig. 11, and the view of the blue vehicle is shown in the lower row of Fig. 11.

V. Conclusions

The continuous reinforcement learning method in social computing is of great assistance in improving the efficiency of our social life. Therefore, we first developed a dual-decision framework to enhance the autonomous learning capability of mobile vehicles and ensure stable performance when they are applied in real-world scenarios. We then propose a dual decision-making continuous reinforcement learning method based on Simulation to Reality, which enables the mobile vehicle to avoid bad strategies and maintain continuous learning ability. Our experimental results demonstrate significant improvements in red and blue defensive vehicle tasks, and successful migration of the model to realistic scenarios with zero-shot. The mobile vehicle was able to complete the task smoothly. In the future, we aim to construct more complex game confrontation scenarios and introduce additional mobile vehicles to realize intelligent group games.

VI. ACKNOWLEDGMENT

The work reported in this paper was supported in part by the Ministry of Industry and Information Technology project of the Intelligent Ship Situation Awareness System under the grant No.MC-201920-X01, and National Natural Science Foundation of China under the grant No.61991415.

References

[1] X. Wang, L. Kou, V. Sugumaran, X. Luo, and H. Zhang, “Emotion correlation mining through deep learning models on natural language text,” *IEEE transactions on cybernetics*, vol. 51, no. 9, pp. 4400–4413, 2020.

[2] X. Wang, H. Zhang, and Z. Xu, “Public sentiments analysis based on fuzzy logic for text,” *International Journal of Software Engineering and Knowledge Engineering*, vol. 26, no. 09n10, pp. 1341–1360, 2016.

[3] T. Schaul, J. Quan, I. Antonoglou, and D. Silver, “Prioritized experience replay,” *arXiv preprint arXiv:1511.05952*, 2015.

[4] Y. Y. Y Y Lin, F Du, “Continuous learning: A review of research,” *Journal of Yunnan University (Natural Science Edition)*, vol. 45, pp. 1–14, 2023.

[5] X. Wang, V. Sugumaran, H. Zhang, and Z. Xu, “A capability assessment model for emergency management organizations,” *Information Systems Frontiers*, vol. 20, pp. 653–667, 2018.

[6] S. Lee, J. Ha, D. Zhang, and G. Kim, “A neural dirichlet process mixture model for task-free continual learning,” *arXiv preprint arXiv:2001.00689*, 2020.

[7] P. Buzzega, M. Boschini, A. Porrello, D. Abati, and S. Calderara, “Dark experience for general continual learning: a strong, simple baseline,” *Advances in neural information processing systems*, vol. 33, pp. 15920–15930, 2020.

[8] X. Xiao, B. Liu, G. Warnell, and P. Stone, “Motion planning and control for mobile robot navigation using machine learning: a survey,” *Autonomous Robots*, vol. 46, no. 5, pp. 569–597, 2022.

[9] S. Liang, Z. Cao, C. Wang, and J. Yu, “Hierarchical estimation-based lidar odometry with scan-to-map matching and fixed-lag smoothing,” *IEEE Transactions on Intelligent Vehicles*, 2022.

[10] J. Jin, N. M. Nguyen, N. Sakib, D. Graves, H. Yao, and M. Jagersand, “Mapless navigation among dynamics with social-safety-awareness: a reinforcement learning approach from 2d laser scans,” in *2020 IEEE international conference on robotics and automation (ICRA)*, pp. 6979–6985, IEEE, 2020.

[11] D. S. Chaplot, R. Salakhutdinov, A. Gupta, and S. Gupta, “Neural topological slam for visual navigation,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12875–12884, 2020.

[12] J. Ye, D. Batra, A. Das, and E. Wijmans, “Auxiliary tasks and exploration enable objectgoal navigation,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 16117–16126, 2021.

[13] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, “Proximal policy optimization algorithms,” *arXiv preprint arXiv:1707.06347*, 2017.

[14] A. Juliani, V.-P. Berges, E. Teng, A. Cohen, J. Harper, C. Elion, C. Goy, Y. Gao, H. Henry, M. Mattar, and D. Lange, “Unity: A general platform for intelligent agents,” *arXiv preprint arXiv:1809.02627*, 2020.

[15] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, “Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor,” in *International conference on machine learning*, pp. 1861–1870, PMLR, 2018.