

Deep Reinforcement Learning in Competing Cognitive Tactical Networks

Yi Chen*

Dept. of Mathematics, Wuhan University, 299 Bayi Rd., Wuhan, Hubei, China 430072

* Corresponding Author Email: 2019300001096@whu.edu.cn

Abstract. These instructions give you guidelines for preparing papers for DRP. Use this document as a template if you are using Microsoft Word 6.0 or later. Otherwise, use this document as an instruction set. The electronic file of your paper will be formatted further at DRP. Paper titles should be written in uppercase and lowercase letters, not all uppercase. Avoid writing long formulas with subscripts in the title; short formulas that identify the elements are fine (e.g., "Nd-Fe-B"). Do not write "(Invited)" in the title. Full names of authors are preferred in the author field, but are not required. Put a space between authors' initials. The abstract must be a concise yet comprehensive reflection of what is in your article. In particular, the abstract must be self-contained, without abbreviations, footnotes, or references. It should be a microcosm of the full article. The abstract must be between 100 - 300 words. Be sure that you adhere to these limits; otherwise, you will need to edit your abstract accordingly. The abstract must be written as one paragraph, and should not contain displayed mathematical equations or tabular material. The abstract should include three or four different keywords or phrases, as this will help readers to find it. It is important to avoid over-repetition of such phrases as this can result in a page being rejected by search engines. Ensure that your abstract reads well and is grammatically correct.

Keywords: cognitive tactical network, multi-agent reinforcement learning, deep reinforcement learning, Actor-Critic method.

1. Introduction

Communication is one of the paramount aspects in combat situation. Successful transmission of data meaning faster update rate of the Observe-Orient-Decide-Act cycle, enabling the force to faster reach advantageous positions. Distinguished from normal civil communication scenario, tactical communication systems must confront with adversarial attacks, for instance, jamming and node destructions. Such demand urges the network to become not only efficient, but also robust and compatible. A cognitive strategy can significantly increase the transmission capacity and jamming efficiency of a tactical network in a competitive environment [1]. Guided by game theory, multi-agent reinforcement learning has demonstrated prominent superiority [2]. Deep reinforcement learning techniques are employed in a hypothetical blue-force versus red force with traditional access strategy. This paper aims to design a cognitive-based strategy with deep reinforcement learning techniques for information transmission within tactical communication networks, refining the comprehensive capabilities of the network and meanwhile, discuss the effect of decentralized command and control under competitive circumstances. Based on the drastic development in deep reinforcement learning within the last decade, we are likely to develop a more efficient, accessible, compatible and robust network.

Traditionally, available spectrum within the combat space is usually divided into a set of separate channels, where the competitive forces' communication apparatuses strive to transmit useful information and interrupt the transmission of the other, respectively. These channels are incised into slots as time eclipses. Before communication strategies were introduced to tactical communication, the most pervasive ways of spectrum sharing are static introduction and random introduction. The former means that an apparatus is statically functioning on one specific channel and the latter means that it randomly picks a channel among the spectrum at the beginning of a time slot, using introduction algorithms like ALOHA [3], and function for the present time slot. As reinforcement learning was introduced to improve performance of tactical communication network, researchers started to

perceive the network in a more succinct perspective: adversarial networks can be deemed as two groups of intelligent agents involved in a competing game, aiming to maximize certain performance index that denotes the valid transmission of information. S.Dastangoo et al. [4] adopted classic reinforcement algorithms, Q-learning and MAB, corresponding to ‘state-aware’ situation and ‘state-agnostic’ situation, to build cognitive tactical network and discover the dramatic advantage of cognitive algorithms.

2. Experimental Models

2.1. Channel Access Models

In this research, it would not be unjust to classify access models according to whether the radio can receive and action based on historically transmission data into the cognitive and the noncognitive. Conventional multiple-access radio systems are based on static or random channel access model. Since in these patterns the information of the channel status would not feedback to the radio, these methods are noncognitive, making themselves vulnerable under adversarial circumstances.

In the research presented by Y.Gwon et al. [5], where one side of the game adopted cognitive algorithms to optimize its strategies while the other insisted with conventional access model, the cognitive side eventually reach to a point that its average channel reward forced the conventional force’s reward to converge to 0 as iterations proceeded. Thus, it is cataclysmically crucial to equip tactical network with cognitive algorithms in order to guarantee its competitiveness.

Then consider a cognitive channel access model. Supposed that considerable information of channels’ status is given to agents of two forces, which is the so-called ‘state-aware’ approach. Based on this premise, we refine the classic Q-learning algorithms with deep reinforcement techniques. We define a set of discrete system states and provide a plausible means to compute optimum strategies via optimizing the artificial quality function Q^* . Classic Q-learning can be achieved by calculating the state table. But as deep neural networks were introduced to reinforcement learning, it become feasible to match the authentic quality function Q^* through neural network computation. Comparing with traditional Q-learning, these DQN enable the structure of the network to be more flexible, compatible and robust.

2.2. Game Setting

As described in the above paragraphs, it is reasonable to simplify a complicated combat transmission situation to a multi-agent gaming process. Hypothetically, we assume that the available spectrum in the game can be divided into 10 separate channels. Each channel locates at the center frequency f_i with bandwidth B_j . A transmission opportunity is represented by a tuple $\langle f_i, B_j, t_k, T \rangle$, which defines a time-frequency slot at the i th channel and time t_k with time duration T . A typical transmission opportunity list is shown in Table 1. Since usually the duration of a time slot is fixed, we can simplify this notation.

Table 1. Transmission opportunity.

$\langle f_1, B_1, t_1 \rangle$	$\langle f_1, B_1, t_2 \rangle$...
$\langle f_2, B_2, t_1 \rangle$	$\langle f_2, B_2, t_2 \rangle$...
$\langle f_3, B_3, t_1 \rangle$	$\langle f_3, B_3, t_2 \rangle$...

Two forces, denoted by color red and blue respectively, participates in the game. Their ultimate goal is to maximize valid information transmission within the given time. Each force is equipped with the following apparatus: 2 communication nodes (radio), 2 jammers. Additionally, blue force has an extra control agent. Communication node transmit information if it takes action at a transmission opportunity while the jammers try to interrupt the transmission of the adversarial communication nodes. Meanwhile, these nodes and jammers will transmit their reward and action to the control agent. Control agents are designated to compute actions for all comm nodes and jammers that want to act

on channels for a given time slot. This is the structure of a centralized network, where a control agent is required. In a decentralized scenario, comm nodes and jammers are equipped with decision-making modules, thus they're capable to take action based on their own observation.

The coexistence of the two opposing kinds of signals (i.e., comm and jammer) in two forces breaks the game into two subgames, namely antijamming and jamming games. These two games are intertwined and form complicated relationships shown in Figure 1. In the antijamming game, take blue comm nodes for an example, the node needs to avoid red force jammers' deliberate jamming, channel collision with another comm nodes (blue force and red force) and misjamming from a blue jammer. These factors may cause a transmission fail. In the jamming game, the blue-force jammers attempt to minimize the red-force data transmission by picking the best channel to jam. Additionally, if a blue-force jammer successfully jams the red-force control agent, then in the next time slot, since the actions within the red-force network would be incoordinate, all red-force transmission and transmission would be perceived as invalid. The transmission will not be restored until the following time slot.

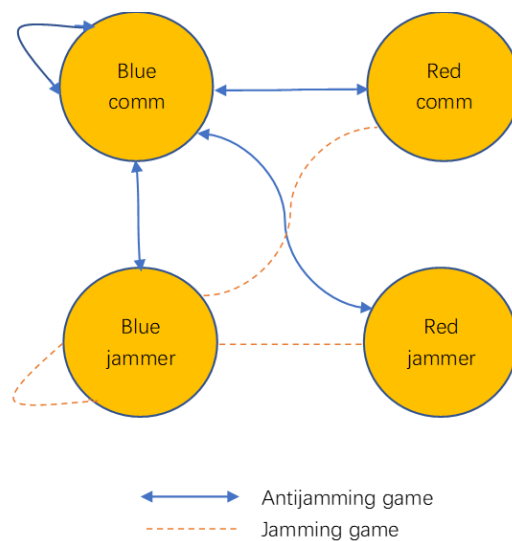


Figure 1. Jamming and antijamming relationship.

To depict a reward model for in single-channel scenario, different situations and their corresponding reward are given. It is notable that all these plots happen in one time slot. Table 2 offers a clear rule of reward based on behavior of two forces. Since control agent uniquely belong to blue force, situations involving blue control agent is not shown in the graph. If the control agent (need 1 channel to transmit) is interrupted red jammers, the red force receives 1 point of positive reward.

Table 2. Reward models of different situations.

Red-Force Action		Blue-Force Action		Outcome	Reward	
Comm	Jam	Comm	Jam		Blue	Red
Activate	-	-	-	Red-force transmit success	1	0
-	Jam	Activate	-	Red-force jamming success	1	0
Activate	-	Activate	-	Channel collision	0	0
-	Jam	-	Jam	Jam collision	0	0
Activate	Jam	-	-	Red-force misjams	0	0
Activate	-	-	Jam	Blue-force jam success	0	1

2.3. Reward Model

Last but not least, it is necessary to mention the reward model. The ultimate payback metric is used to assess the performance of the cognitive tactical network. When a comm node successfully transmit information within a duration of a time slot, it receives a reward of B. Jammers do not create positive reward by themselves. Their reward comes from successful interruption of the adversary's

transmission. If a Blue-force jammer block a red-force comm node from transmitting during n time slots, then we consider it has made a reward of n. The cognitive algorithms are applied to maximize the expected reward from channels.

3. Deep Q-learning

3.1. Notations

Before introducing the theory of deep Q-learning, certain terminologies need to be clarified. We denote the blue-force state in a terser representation, which is a tuple $\langle IC, ID, JD \rangle$, where IC denotes the number of blue-force control channels blocked, ID represents the number of blue-force data transmission (comm node) channel collided, JD denotes the number of interrupted red-force data transmission. Since control agent is unique to blue force, when it comes to describing the red-force state, to calculate its reward corresponding to its action, we need an extra indicator J_c to denote whether the red-force jammer disturb the transmission of the blue control agent. Meanwhile, IC is not needed for red-force evaluation.

3.2. Rationale of Q-learning

The quality of the actions chosen by the control agent can be described as a Q function. Usually, a Q function needs two input variables, states and actions. Nevertheless, it is noticeable that a specific Q function can be calculate only if the strategy of the agents is settled. Here, strategy $\Pi(a, s)$ is in fact a probability distribution of the action space, all possible actions, under the condition states. In Q-learning, we pursue to calculate the authentic Q function of the optimum strategy. To accentuate this, it is no harm to denote this special Q function as Q^\wedge .

In conventional Q-learning [6], the calculation is based on the theory of time-differential algorithms. In this case, the Q function, whatever strategy adopted, observes the Bellman equations [7]. During training, the reward is given from an ongoing training game, so theoretically it conveys certain information about the game. So, this reward is a more reliable data source that should be involved to make the calculation more reasonable. Through iterations, we are likely to witness the convergence. In multiagent gaming, since the agents reward are interlinked, it would be wise to adaptively redesign the Q function according to game theory. Classic adaptations include minimax Q, Nash Q and FFQ [8,9].

3.3. Deep Q Network (DQN)

Since the last decade, neural network has become a major propellant in machine learning, reinforcement learning is not excepted. Deep neural network offers vast opportunity for researchers to match complicated function through training. Thus, DQN algorithms [10] were introduced to researches. Though its interpretability still remains a conundrum for scientists, deep neural network clearly contributed profound accomplishment in reinforcement learning, one of the renowned products is AlphaGo.

In this research, we are going to adopt a deep reinforcement learning technique to modify the classic Q learning. Furthermore, an Actor-Critic [11] method is introduced to strategy optimization. A basic calculation iteration of Actor-Critic method is demonstrated as Figure 2. Similar with conventional Q-learning, Actor-Critic needs to train a neural network to match the authentic Q^\wedge for supervision, which is called "critic". However, Actor-Critic method extend a network to match strategy, which is called "actor". The training process of a single iteration is based on gradient descend. θ is the training parameter in the policy network while w is the training parameter in the value network. These two-network train one another and intertwine with dataflow. With the aid of TensorFlow, compared with conventional Q-learning, the code needed is far shorter.

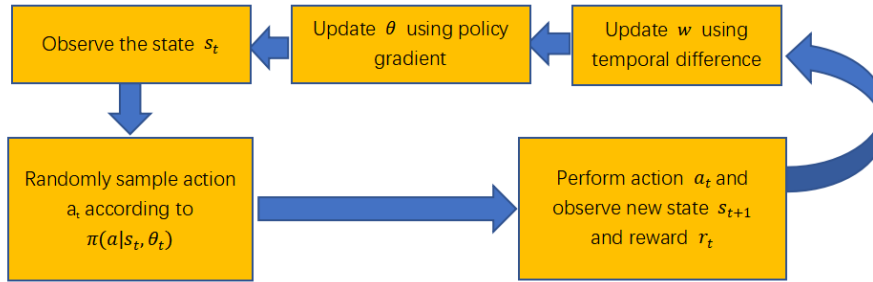


Figure 2. A typical Actor-Critic iteration cycle.

Typically, there are two types of basic perspective in reinforcement learning, value-based learning and strategy-based learning. The former pursues to calculate or to match the authentic Q function of the game while the latter focus on optimizing agents’ policies. Actor-Critic mechanism attempt to combine these two perspectives by co-training two networks, corresponding to the Q function and the strategy. The data flows and intertwines between two networks, mutually refine their own accuracy.

4. Simulation Parameter Configuration

Similar to research performed in the field of competitive cognition, we proceeded the following simulation tests. In the contest, blue-force adopts a centralize decision making configuration, where a control agent exists, and is trained by the actor-critic method along with deep Q-learning. To simplify the experiment, we restricted that both sides only equip 2 comm nodes and 2 jammers. But, in fact, as the number of comm nodes and jammers increase, the unrivalled advantage of cognitive algorithms would be even more self-evidently magnified [1]. Hypothetically, we assume the red-force radios and jammers adopt static access strategy.

5. Simulation Result Analysis

5.1. Result Analysis

To verify the superiority of our cognitive algorithms, we ran a simulation test. The accumulative averaged reward (per channel) variation is shown in Figure 3. It is very clear that our algorithm has enabled blue force to reach an advantageous condition in early iterations and progressed gradually. It is observed that the static access strategy made red force suffer great loss in the early stage, this is because the channel space is relatively crowded (10 channels, 9 agents altogether) and its data transmission is quickly interrupted due to multiple reasons including channel collision and misjamming. The crowded, scarce spectrum resources also made blue-force suffer, corresponding to the relatively flat stage while reward of the red force is confronting with great loss. After adjustments through iterations, blue force gradually overcome this difficulty and gained observable increase of reward.

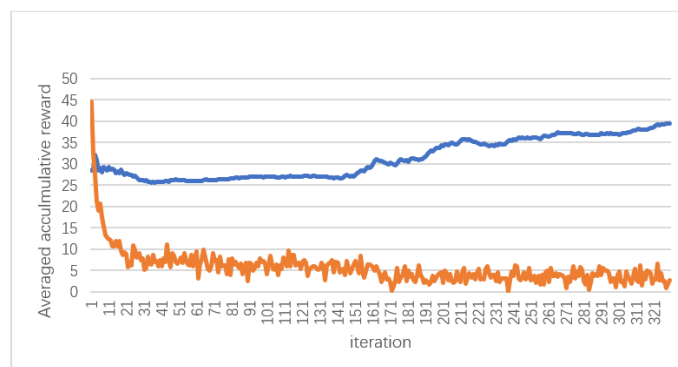


Figure 3. Simulation result demonstration.

5.2. Future Work

In this research, we mainly focus on the scenario that the tactical network with a central control agent that make decisions for every agent in the network. However, such structure could be systematically fragile in competitive environment, since once the control agent is jammed or destroyed, the whole network would be uncoordinated. To deter such threat, decentralized control mechanism should be studied and discussed. It is mentioned that, decentralized control, due to misjamming and channel collision, would cause transmission efficiency decrease [2]. But this tradeoff is worthwhile and necessary. And further study will be proceeded on the measurement of the robustness of the network.

6. Conclusion

Based on the observation of the above simulation, it would be self-evident to reach a conclusion that cognitive algorithms have much more advantageous performance in competitive circumstances over conventional strategies. Its primary merits include: smart transmission to evade adversary's jamming and channel collision, smart jamming adaptively adjust channel access according to adversarial deployment. These factors combine enable the network to be efficient, intelligent and robust while the designation of the network can be succinct.

References

- [1] Wang, B., & Liu, K. R., "Advances in cognitive radio networks: A survey," *IEEE Journal of selected topics in signal processing*, 5(1), 5-23(2010).
- [2] Zhang, K., Yang, Z., & Başar, T., "Multi-agent reinforcement learning: A selective overview of theories and algorithms," *Handbook of Reinforcement Learning and Control*, 321-384(2021).
- [3] Abramson, N., "THE ALOHA SYSTEM: Another Alternative for Computer Communications," *Proceedings of the ACM Fall Joint Computer Conference*, 281-285(1970).
- [4] Dastangoo, S., Fossa, C. and Gwon, Y., "Competing Cognitive Tactical Networks," *Lincoln Laboratory Journal*, 16-35(2014).
- [5] Gwon, Y., Dastangoo, S., and Kung, H.T., "Competing Mobile Network Game: Embracing Antijamming and Jamming Strategies with Reinforcement Learning," *IEEE Conference on Communications and Network Security*, 2013.
- [6] Watkins, C. J., & Dayan, P., "Q-learning," *Machine learning*, 8(3), 279-292(1992).
- [7] Bellman, R., "Dynamic Programming," *Princeton University Press*, 1957.
- [8] Hu, J., & Wellman, M. P., "Nash Q-learning for general-sum stochastic games," *Journal of machine learning research*, 4(Nov), 1039-1069(2003).
- [9] Littman, M. L., "Friend-or-foe Q-learning in general-sum games," *International Conference on Machine Learning* , 322-328(2001).
- [10] Carta, S., Ferreira, A., Podda, A. S., Recupero, D. R., & Sanna, A., "Multi-DQN: An ensemble of Deep Q-learning agents for stock market forecasting," *Expert systems with applications*, 164, 113820(2021).
- [11] Konda, V. and Tsitsikls, J., "Actor-Critic Algorithms," *Advances in Neural Information Processing Systems* 12(1999).