Research of Multi-agent Deep Reinforcement Learning based on Value Factorization

Shiyi Liu *

Jinling High School Hexi Campus, Nanjing, China

* Corresponding author email: mvnogmq7362@163.com

Abstract. One of the numerous multi-agents' deep reinforcements learning methods and a hotspot for research in the field is multi-agent deep reinforcement learning based on value factorization. In order to effectively address the issues of environmental instability and the exponential expansion of action space in multi-agent systems, it uses some constraints to break down the joint action value function of the multi-agent system into a specific combination of individual action value functions. Firstly, in this paper, the reason for the factorization of value function is explained. The fundamentals of multi-agent deep reinforcement learning are then introduced. The multi-agent deep reinforcement learning algorithms based on value factorization may then be separated into simple factorization and attention-mechanism based algorithms depending on whether other mechanisms are incorporated and which various mechanisms are introduced. Then several typical algorithms are introduced and their advantages and disadvantages are compared and analyzed. Finally, the content of reinforcement learning elaborated in this paper is summarized.

Keywords: Multi-agent; Deep Reinforcement Learning; Value Factorization.

1. Introduction

Reinforcement learning is a method of achieving a given goal through trial and error [1]. Its core is to make the agent learn the best strategy to maximize the return in the process of interacting. Deep reinforcement learning integrates the decision-making and perceptual capabilities of reinforcement learning. Deep neural networks' potent representational capabilities are employed to tackle challenging decision-making issues. With the development of deep reinforcement learning technology, humans gradually introduce deep reinforcement learning into multi-agent field, Multi-Agent Deep Reinforcement Learning (MADRL) become a hot research area [2].

Deep reinforcement learning faces significant challenges in the multi-agent field:1). The increase in the number of agents leads to an exponential increase in the open AI gym. It causes great difficulties in the calculation of Q value. 2). The different between goals and tasks of agents, as well as the mutual influence between them, lead to the difficulty in determining the goal reward, the convergence of the algorithm is seriously affected. 3). The exploration of each agent will cause changes in the environment, and also affect the strategy selection of other agents, resulting in slow learning speed and difficult stability of the algorithm. To solve the above problems, researchers have proposed a large number of multi-agent deep reinforcement learning algorithms. These algorithms can be roughly divided into three categories: association-free, communication rule, and value function decomposition.

Because of the significant advantages of value function decomposition, it has gradually become a research hotspot in the field of reinforcement learning. This paper will introduce MADRL method based on value factorization, including the basic theory of MADRL, the classification method of MADRL algorithm based on value factorization, and the analysis of advantages and disadvantages.

2. Basic Theory of Multi-agent Deep Reinforcement Learning

2.1 Deep Reinforcement Learning based on Value Function

The fundamentals of multi-agent deep reinforcement learning are then introduced. The multi-agent deep reinforcement learning algorithms based on value factorization may then be separated into

simple factorization and attention-mechanism based algorithms depending on whether other mechanisms are incorporated and which various mechanisms are introduced. The corresponding classical algorithms are temporal differences learning and Q learning respectively [3].

It is well known that the data obtained by reinforcement learning is Markov property (that is the data are highly correlated with each other). The inputs to the neural network should be independent. As a result, the reinforcement learning process will be unstable or even divergent if the deep neural network is employed to approximate the action value function. Mnih et al. suggested DQN to address this issue, which employed parameterized deep neural networks to approximate the ideal action value function Q and employed an empirical replay mechanism to remove data correlation [4]. Following that, researchers advanced a number of DQN-based algorithms. Hasselt et al. presented the Double-DQN method, which distinguished between the greedy selection of actions and the assessment of those activities [5]. It employed an online network to pick actions and a target network to evaluate those actions. Lipton et al. proposed a backpropagation Bayesian Q network to quantify the uncertainty of value function estimation when selecting actions. This method can learn effective strategies in the environment with sparse reward values and large action space. Anschel et al. proposed the average DQN, which averages the previous estimated Q values and uses the target approximation error to further improve the stability of the algorithm. Wang et al. proposed A new neural network structure -- dueling DQN, which converts Q value into the sum of scalar state value function V(s) and action dominance function A (s, u).

To make the arithmetic better than DQN and DDQN in Atari games. Hausknecht et al. proposed deep cyclic Q network, which combined Long Short-Term Memory (LSTM) and DQN to improve the learning ability of the algorithm in partially observable environment [6]. Nair et al. proposed a distributed algorithm -- Gorila DQN, which combined the distributed Architecture Gori-la (General Reinforcement Learning Architecture) with DQN. The algorithm can use a large number of resources and increase the operation efficiency of the algorithm. Sorokin et al. proposed the Attention mechanism Deep Cyclic Q network (DAR-QN), which combined the attention mechanism with DRQN to enable the algorithm to select and centrally process local information and reduce the computational cost of the algorithm. The above DRL algorithm based on value function can extract data features from high-dimensional sensory input data and learn them, which has achieved great success in the fields of social science and engineering. However, introducing DRL method based on value function in multi-agent field also faces a new problem: how to realize the cooperative control of multi-agent? If the method of centralized learning is adopted, the dynamic space is too large, and the power of the hard part should be too high. If the independent learning method is adopted, the convergence of the algorithm is not guaranteed by any method. The method of value function factorization can not only avoid too large action space, but also ensure the convergence of the algorithm. Therefore, the method of value function factorization can provide an effective solution to the multi-agent cooperative control problem.

2.2 Multi-agent Reinforcement Learning based on Value Function Factorization

The fundamentals of multi-agent deep reinforcement learning are then introduced. The multi-agent deep reinforcement learning algorithms based on value factorization may then be separated into simple factorization and attention-mechanism based algorithms depending on whether other mechanisms are incorporated and which various mechanisms are introduced. The individual action value function is the Q value obtained from the local observation of the intellectual body, which is used to guide a single agent to select actions. Since the joint action value function is shared by all agents, this method is only suitable for multi-agent reinforcement learning tasks in cooperative environments.

Cooperative multi-agent deep reinforcement learning trains the network through a single joint reward signal, which is difficult to learn. Because the environment is partially observable, it is impossible to effectively solve the problem of coordination among agents by means of independent Q learning or complete Chinese learning.

Independent Q learning by training independent Q-learner to optimize the number of award letters. In this method, each agent regards other agents as part of the environment, so the strategy and return of each agent are not only affected by the environment, but also influenced by the actions of other agents, resulting in the problem of false rewards. By connecting the action space and state space of a single agent into a joint action space and a joint state space, the problem is transformed into a single intelligent body question. Although this method solves the problem of consistency and convergence to a certain extent, with the increase of the number of agents, there will be an exponential explosion of motion space. Moreover, when an agent explores a better strategy, if the exploration of other agents does not have a positive impact on it, other agents will not learn it, resulting in the phenomenon of lazy agents.

Therefore, the researchers balance between these two extreme approaches and propose the concepts of in-set training and decentralized execution to solve the problem of multi-agent tasks. Set in training, that is, during the training period, the communication between the intelligent bodies is not restricted, and the joint action value function Q_π (s_t, u_t) is used to learn the algorithm. During the training process, the algorithm can access the motion observation history h and the whole local states of all the agents. Distributed execution, that is, during the execution period, the communication between agents is limited, and they can only choose actions by calculating individual action value function based on their own observation history ha, without considering the actions of other agents.

3. Deep Reinforcement Learning for Multi-agent based on Value Factorization

The value function factorization method has great advantages in multi-agent reinforcement learning in cooperative environment, and can solve the problems in multi-agent reinforcement learning, such as partial observability of the environment, exponential explosion of motion space, poor algorithm stability and multi-agent reliability partitioning. For this reason, in recent years, people have devoted themselves to the study of the method of value function segmentation, combining other mechanisms with MADRL, and putting forward many MADRL algorithms based on value segmentation. Different adding mechanisms may lead to different side points of algorithm performance improvement. In this paper, MADRL algorithm based on value factorization is divided into three categories according to whether principle/machine system is introduced and the class type of original principle/machine system is introduced. The simple factor factorization, and the attention-mechanism-based are classified into body types as listed in Table 1.

Table 1. Classification of MADRL algorithms based on value factorization

Model	Characteristic	Algorithm
Simple factorization	Rely on simple structural constraint	VDN [7] QMIX [8] WQMIX CoRe
Based on the attention mechanism type	Dependent attention mechanism	Qatten [9] REFIL AVD-net value factorization in conjunction with communication ARE

3.1 Simple Factorization

The simple factorization algorithm directly decomposed the joint action value function into one body action value function (such as simple phase addition, single tunable reduced bundle, etc.), and did not introduce some other mechanisms into the network. This kind of algorithm is relatively easy to implement, but it is due to the strong bundle reduction and the limited representation ability of the

algorithm, so it is more suitable for the environment where the relationship between agents is simple. The main table calculation methods are: Value Decomposition Networks (VDN), Monotonic Value Function Factorization for Deep Multi-Agent Reinforcement Learning (QMIX) and Weighted QMIX.

VDN and QMIX are the first proposed multi-agent depth based on value factorization degree reinforcement learning algorithm is used as a benchmark in the field of value factorization. On the basis of QMIX, the optimal joint action value is simply weighted to make the calculation method converge to the optimal policy.

VDN algorithm is an addition able factorization method of individual learning proposed by Sunehag et al., aiming to solve the false reward problem in independent Q learning and the lazy agent problem in fully centralized learning. The core of VDN algorithm is to combine the action value function $Q_{tot}(h,u)$ .The simple sum of the action value function decomposed into n agents, that is:

$$\begin{aligned} Q_{tot}(h,u) &= Q_{tot}((h^1,h^2,\dots,h^n)(u^1,u^2,\dots,u^n)) \\ &= \sum_{i=0}^{n} Q^i(h^i,u^i;\theta^i) \end{aligned} \tag{1}$$

The function of the value factorization network is to learn the optimal linear value factorization from associative rewards by backpropagating the ladder of the joint action value function Qtot (h, u) of the deep degree neural network representing the individual action value function. During the training process, each intelligence body obtains based on its own local observations and chooses actions according to the greedy strategy to produce a split strategy, without considering the influence of other agents. Later, however, the new rules of using DQN will make Q (s, u; θ) is replaced by Q tot (h, u) to obtain a new loss function Ltot (θ), and update the network parameters by minimizing the loss function.

VDN calculation method is the first work of value function factorization method, which has important implications for the development of MADRL. Although great achievements have been made, there are still many shortcomings: 1) the constraint on value function is strong, the strictness limit limits the complexity of the joint action value function class that can be shown, and the range of applicable learning scenarios is small; 2) Completely ignoring any additional information available during practice and learning the best strategy only in a few simple Settings.

Since QMIX cannot represent the joint action value function with non-monotonic characteristics, when faced with this type of task, QMIX may be unable to reach the optimal strategy and fall into the suboptimal strategy. Rashid et al. suggested the weighted QMIX algorithm (WQMIX), which added weight to weight the square error of the joint action value when updating the network in order to acquire the optimal joint strategy, in order to promote algorithm convergence to the ideal strategy.

8	- 12	- 12
- 12	0	0
- 12	0	0

(a) Q_{tot} generated by the payment matrix

- 12	- 12	- 12
- 12	0	0
- 12	0	0

(b) The Q_{tot} returned from Qmix

Fig 1. Comparison of Q_{tot} in QMIX and payment matrix

Q_{tot} in Fig. 1(b) is a close similarity to Q_{tot} in Fig. 1(a). According to the data in the figure, QMIX algorithm cannot produce correct argmax operation. The QMIX algorithm may also underestimate the value of the best joint action due to incorrect argmax operation, as shown in Figure 3(b), where the value is -12 instead of 8.

The incorrect argmax operation may be due to the same weight being given to all joint actions when minimizing loss-and-loss functions. If the problem is considered at the fruit end, only the loss of the best joint action is optimized, then the value of the best joint action can be obtained.

Because of this, WQMIX offers a weighting function w (s, u) based on the QMIX algorithm and provides two distinct weighting techniques: thinking center weighting (CWQMIX) and Music perspective weighting (OWQMIX).

WQMIX is a modified version of QMIX algorithm, which highlights the importance of the best joint action by weighting the loss function, so as to promote the algorithm convergence to the best joint strategy. WQMIX has two weighting methods, namely ideal center weighting and optimistic weighting. The experimental results show that both weighted formulas can effectively avoid the convergence of the algorithm to the suboptimal strategy. Although WQMIX has a slight improvement in performance, the numerical determination of weighted parameters is more arbitrary and there is no theoretical basis for filling.

At present, most research works tend to introduce other mechanisms to improve the performance of the algorithm, but there are still a small number of research works on the simple factorization method of MADRL. Shao et al. propose a multi-agent reinforcement learning algorithm with Counterfactual Reward mechanism CoRe (Counterfactual Reward), which obtains the counterfactual reward when the agent's behavior changes by calculating the local reward of each agent real value to determine each agent's contribution to the global reward.

3.2 Based on the Attention Mechanism

The attention-mechanism-based algorithm can calculate the degree of influence of the mental body on the system according to the global information, which provides a theoretical basis for the factorization of value function and makes the value function factorization more reasonable, thus improving the efficiency of the algorithm. The disadvantage of the algorithm is that the calculation amount is large and the calculation cost is high. Because of this, this kind of algorithm is suitable for MARL environment with complex inter-agent system, but it is also expected to be extended to large-scale multi-agent environment. The main representative algorithms are Qatten (Deep Q learning based on multi-head attention) and REFIL (Random Entity Factorization for Imaginary Learning).

3.2.1 Mechanism Attention

In recent years, attention mechanism has been widely used in various research fields, especially the MARL domain. For this reason, introducing intension mechanism into multi-agent reinforcement learning has become a research hotspot.

The attention mechanism is implemented through the attention function. Note that function is a kind of projection from query Q and a set of key-value pairs (K-V) to output, in which query, key, value and output are vectors, the main path of the projection is: Firstly, the weight W corresponding to the value is generated by the query and key, and then the weighted sum of the value is obtained (Fig.2).

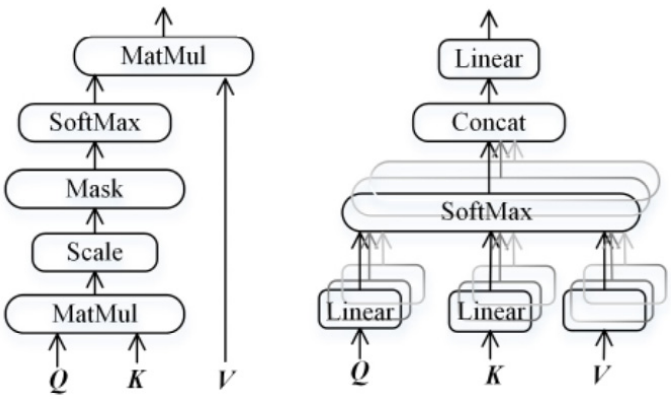

Fig 2. Attention mechanism

Multi-head attention allows the model to pay attention to the information from different eigenspaces at different locations. The main process is: the input is divided into multiple layers, and the input is calculated through multiple injection-force machines. Then the output is connected by Concat to form the same value as the single-head output dimension.

3.2.2 Random Entity Factorization for Imaginary Learning

Value factorization method uses some independence of intellectual body to improve learning efficiency. However, most of this method uses domain knowledge to select fixed sub-sets of certain features, which has poor expansibility. To solve the question, Iqbal et al. put forward a Randomized Entity factorization method for imaginary learning – REFIL (Randomized Entity-wise Factorization for Multi Agent) Reinforcement Learning, which is used to select subsets of observed entities and simulate the predictive utility of each agent in these subsets.

REFIL algorithm is a random entity factorization method for imaginary learning, which uses attentional mechanism masking process to separate value prediction from unrelated multi-agent task entities. In this method, the individual action value function is divided into two types: in-group effect function and out-group effect function, and the loss of factorization is used as the auxiliary target for training. Therefore, the number of parameters in the algorithm is larger and the calculation cost is higher. Because the algorithm needs to distinguish different entities in the environment, it can realize the cooperative control of multi-agents in the scene with different types and quantities of entities.

Although there are many methods of value function factorization based on attention mechanism, it is still in the initial stage, and the researchers are still studying deeply. Zhang et al. proposed a new attention-based method -- AVD-net, which introduced the attentional mechanism into VDN and QMIX to learn the correlation between agents, so as to effectively decompose the joint action value function. So that it can adapt to learn the coordination of the mental body. Wu et al. combined attention mechanism with communication to learn a more general value function factorization form, so as to achieve multi-intelligence cooperation in complex scenarios. Liu et al. proposed an Attention Relational Encoder (ARE) for the representation of attentional relation states in decentralized multi-agent reinforcement learning. The encoder uses the attention mechanism to gather the information of the neighboring agents to expand the observation space of the agents. It has the characteristics of invariability of arrangement, high calculation efficiency, and flexibility to the interactive multi-agent system, and shows strong cooperation in the micromanagement task of the StarCraft 2.

4. The Advantages and Disadvantages of Value Function Factorization Algorithm

This paper mainly introduces two kinds of value function factorization algorithms, including simple factor factorization (VDN, QMIX, WQMIX) and attention-mechanism based (Qatten, REFIL). The comparative analysis results of each calculation point are listed in Table 2.

Table 2. Comparative analysis of algorithm characteristics

Type of algorithm	Name of algorithm	Whether to introduce other mechanisms	Complexity of computation	Representation of ability
Simple factorization	VDN	no	easy	weak
	QMIX	no	be easier	the weaker
	WQMIX	no	be easier	general
Based on the attention mechanism type	Qatten	Yes, mechanism of attention	General	General
	REFIL	Yes, mechanism of attention	difficult	strong

Simple factorization algorithm does not introduce other mechanisms, in general, the value. The computational complexity is low and the characterization ability is relatively weak. VDN is the easiest

to calculate, but the binding force of additivity is strong, and its characterization ability is also the weakest. QMIX and WQMIX calculate weights according to the global states, which is slightly more difficult to calculate than VDN. Based on IGM principal algorithm, the definition of IGM principle is introduced and the consistency constraint of action value function is realized based on this [10]. The representable function class of this algorithm is more comprehensive, and the characterization ability is strong. But the structure design on the whole body is more complicated and difficult to realize. By introducing the attention mechanism into the hybrid network, the influence degree (i.e., weight) of a single agent on the global can be obtained, and the eigenenergy of the algorithm can be improved.

5. Conclusion

In this paper, the MADRL algorithm based on value factorization is introduced and analyzed theoretically, and the application and development prospect of the algorithm are briefly described. It can be seen from this paper that: MADRL algorithm based on value factorization has great advantages over other types of algorithms in solving MADRL problems, such as small computation amount, fast learning speed, algorithm construction and calculation simplification, etc., and has a large development space and potential in the aspects of heterogeneous multi-intelligence cooperative cooperation and multi-intelligence cooperative control under sparse rewards. However, the MADRL algorithm based on value segmentation also has some shortcomings, such as the stability of the algorithm is worse than that of the strategy iterative algorithm, can only deal with discrete action problems, and is only suitable for cooperative environment. It is believed that the MADRL algorithm based on value factorization can overcome these limitations, effectively solve complex practical problems, and enter people's life with the unbroken depth of research.

References

[1] Sutton R S, Barto A G, Introduction to reinforcement learning. Cambridge: MIT press, 1998.

[2] Nasir Y S, Guo D. Multi-Agent Deep Reinforcement Learning for Dynamic Power Allocation in Wireless Networks. IEEE Transactions on Wireless Communications, 2018, 26(99):2788-2799.

[3] Sutton R S. Learning to predict by the methods of temporal differences. Machine Learning, 1988, 3(1):9-44.

[4] Mnih V, Kavuk K, Silver D, et al. Human-level control through deep reinforcement learning. Nature, 2015, 518(7540):529-533.

[5] Hasselt H, Guez A, Silver D. Deep reinforcement learning with double Q-learning, Proceedings of the AAAI Conference on Artificial Intelligence. 2016, 30(1):2094-2100.

[6] Hausknecht M, Stone P. Deep recurrent Q-learning for partially observable MDPs, 2015 AAAI Fall Symposium Series. 2015:29-37.

[7] Sun P, Lever G, Gruslys A, et al. Value decomposition networks for cooperative multi-agent learning based on team reward, Proceedings of AAMAS. 2018:2085-2087.

[8] Lin Dai and Khaled Khechen and Sara Khan, The Effect of QMix, an Experimental Antibacterial Root Canal Irrigant, on Removal of Canal Wall Smear Layer and Debris. Journal of Endodontics, 2011(2) 433-442.

[9] Yang Y, Hao J, Liao B, et al. Qatten: A General Framework for Cooperative Multiagent Reinforcement Learning. IEEE Transactions on Wireless Communications 2020:482-491.

[10] Farias D, Roy B V. On the Existence of Fixed Points for Approximate Value Iteration and Temporal-Difference Learning. Journal of Optimization Theory & Applications, 2000, 105(3):589-608.