The Representational Capacity of Action-Value Networks for Multi-Agent Reinforcement Learning

Extended Abstract

Jacopo Castellini
Dept. of Computer Science
University of Liverpool
Liverpool, United Kingdom
J.Castellini@liverpool.ac.uk

Frans A. Oliehoek
Interactive Intelligence Group
Delft University of Technology
Delft, The Netherlands
F.A.Oliehoek@tudelft.nl

Rahul Savani
Dept. of Computer Science
University of Liverpool
Liverpool, United Kingdom
rahul.savani@liverpool.ac.uk

Shimon Whiteson
Dept. of Computer Science
University of Oxford
Oxford, United Kingdom
shimon.whiteson@cs.ox.ac.uk

ABSTRACT

Recent years have seen the application of deep reinforcement learning techniques to cooperative multi-agent systems, with great empirical success. In this work, we empirically investigate the representational power of various network architectures on a series of one-shot games. Despite their simplicity, these games capture many of the crucial problems that arise in the multi-agent setting, such as an exponential number of joint actions or the lack of an explicit coordination mechanism. Our results quantify how well various approaches can represent the requisite value functions, and help us identify issues that can impede good performance.

KEYWORDS

multi-agent systems; neural networks; decision-making; action-value representation; one-shot games

ACM Reference Format:
Jacopo Castellini, Frans A. Oliehoek, Rahul Savani, and Shimon Whiteson. 2019. The Representational Capacity of Action-Value Networks for Multi-Agent Reinforcement Learning. In *Proc. of the 18th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2019), Montreal, Canada, May 13–17, 2019,* IFAAMAS, 3 pages.

1 INTRODUCTION

In this paper, we focus on value-based multi-agent reinforcement learning (MARL) [2, 5, 6, 14, 20, 24] approaches for cooperative multi-agent systems (MASs) [10, 12, 22, 26, 29]. Value-based single-agent RL methods use (deep) neural networks to represent the action-value function $Q(s, a; \theta)$ to select actions directly [17] or as a 'critic' in an actor-critic scheme [13, 16]. Current deep MARL approaches are either based on the assumption that the joint-action value function $Q(s, a)$ can be represented efficiently by neural networks (when, in fact, the exponential number of joint actions usually makes a good approximation hard to learn and scales poorly in the number of agents [4]), or that it suffices to represent individual action values $Q_i(s_i, a_i)$ [15, 25], that are known to be hard to learn because of non-stationarities from the perspective of a single agent due to the simultaneous learning of the others [4, 25, 28].

To overcome these difficulties and be able to learn useful representations while not incurring excessive costs, a middle ground is to learn *factored Q-value functions* [7, 8], which represent the joint value but decompose it as the sum of a number of local components, each involving only a subset of the agents.

This paper examines the representational capacity of these approaches by studying the accuracy of the learned Q-function approximations $\hat{Q}$, as recently factored approaches have shown some success in deep MARL [20, 23]. We consider the optimality of the greedy joint action, which is important when using $\hat{Q}$ to select actions, and the distance to optimal value $\Delta Q = |Q - \hat{Q}|$. Minimising ΔQ is important for deriving good policy gradients in actor-critic architectures and for sequential value estimation in any approach (such as Q-learning) that relies on bootstrapping.

To minimise confounding factors, we focus on one-shot (i.e., non-sequential) problems [19] that require a high level of coordination. Despite their simplicity, these one-shot games capture many of the crucial problems that arise in the multi-agent setting, such as an exponential number of joint actions. Thus, assessing the accuracy of various representations in these games is key step towards understanding and improving deep MARL techniques.

2 ACTION-VALUE FUNCTIONS FOR MARL

In many problems, the decision of an agent is influenced by those of only a small subset of other agents [8, 9] and thus the joint action-value function $Q(a)$ can be represented as a *factorization*, i.e. a sum of smaller action-value functions $Q_e(a_e)$ defined over a *coordination graph* [7, 11, 21] describing these influences. However, there are many cases in which the problem itself is not perfectly factored according to such a graph, or the underlying factorization may be unknown beforehand. In these cases, however, it can still be useful to resort to an *approximate factorization* [8]:

$$Q(a) \approx \hat{Q}(a) = \sum_e \hat{Q}_e(a_e), \quad (1)$$

Proc. of the 18th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2019), N. Agmon, M. E. Taylor, E. Elkind, M. Veloso (eds.), May 13–17, 2019, Montreal, Canada.

obtained by considering a decomposition of the original function into a desired number of local approximate terms $\hat{Q}_e(a_e)$, thus forming an approximation $\hat{Q}$ of the original action-value function.

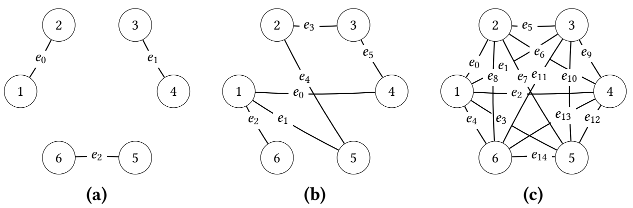

Figure 1: Example coordination graphs for: (a) random partition, (b) overlapping factors, (c) complete factorization.

We investigate four different coordination graph structures used to approximate the action-value function:

Single agent decomposition [9]: each agent i is represented by an individual neural network and computes its own individual action-values $\hat{Q}_i(a_i)$, one for each output unit, based on its local action a_i. This corresponds to the value decomposition networks from [23] [1]

Random partition: agents are randomly partitioned to form factors of size f, with each agent i in the team D involved in only one factor, resulting in $\frac{|D|}{f}$ factors.

Overlapping factors: a fixed number of factors is picked at random from the set of all possible factors of size f.

Complete factorization: each agent i is grouped with every possible combination of the other agents in the team $D \setminus i$ to form factors of size f, resulting in $\frac{|D|!}{f!(|D|-f)!}$ factors.

In each of the investigated factorizations, each factor is represented by an individual neural network that represents local action-values $\hat{Q}_e(a_e)$, using an output unit for each on them, for a certain factor e, where a_e is the *local joint action* of agents in factor e. We consider factors of size $f \in \{2, 3\}$. Also, a *joint learner* with an exponential number of output units is used as a baseline.

3 EXPERIMENTS

We investigate the representations obtained with the proposed factorizations on a series of challenging one-shot coordination games with $n = 6$ agents that do not present an explicit decomposition of the reward function $Q(a)$ (non-factored games), and then on two factored games. Complete results and plots for all the proposed games are presented in the full-length paper available online [3].

Here, we are reporting results for only two of them: the non-factored *Climb Game* [27], known to enforce a phenomenon called *relative overgeneralization* that pushes the agents to underestimate a certain action even if it is optimal when perfectly coordinating on it, and a one-shot version of the factored game *Aloha* [18], in which neighbouring agents in the coordination graph can interfere with agents' actions.

Table 1 presents the accuracy using various measures of the investigated representations on these two problems in terms of reconstruction error, action ranking and action selection. We report mean values and standard errors across 10 runs. Some of our main findings are:

Model	Mean square error	MSE on optimal actions	Optimal actions found	Value loss	Boltzmann value loss	Correctly ranked	Kendall τ
Climb game (728 joint actions, 1 optimal)							
Joint	0.17 ± 0.1	18.45 ± 4.9	0 ± 0	2.70 ± 0.9	1.52 ± 0.3	727 ± 1	1.00 ± 0.0
F1	0.58 ± 0.0	52.29 ± 0.1	0 ± 0	3.00 ± 0.0	2.16 ± 0.0	726 ± 0	0.98 ± 0.0
F2R	0.52 ± 0.0	40.95 ± 0.0	0 ± 0	3.00 ± 0.0	2.06 ± 0.0	726 ± 0	0.98 ± 0.0
F3R	0.44 ± 0.0	36.51 ± 0.2	0 ± 0	3.00 ± 0.0	1.92 ± 0.0	726 ± 0	0.98 ± 0.0
F2C	0.25 ± 0.0	7.86 ± 0.1	1 ± 0	0.00 ± 0.0	1.40 ± 0.0	729 ± 0	1.00 ± 0.0
F3C	0.17 ± 0.0	70.77 ± 0.7	0 ± 0	3.00 ± 0.0	0.96 ± 0.0	726 ± 0	0.98 ± 0.0
F2O	0.45 ± 0.0	30.83 ± 0.1	0 ± 0	3.00 ± 0.0	1.94 ± 0.0	726 ± 0	0.98 ± 0.0
F3O	0.30 ± 0.0	28.89 ± 1.9	0 ± 0	3.00 ± 0.0	1.54 ± 0.0	726 ± 0	0.98 ± 0.0
Aloha (64 joint actions, 2 optimal)							
Joint	1.13 ± 0.0	0.00 ± 0.0	2 ± 0	0.00 ± 0.0	0.08 ± 0.0	51 ± 1	0.88 ± 0.0
F1	4.78 ± 0.0	50.93 ± 0.1	0 ± 0	6.00 ± 0.0	4.04 ± 0.0	27 ± 1	0.67 ± 0.0
F2R	4.05 ± 0.4	35.00 ± 7.0	0 ± 0	5.00 ± 1.3	3.69 ± 0.4	22 ± 4	0.70 ± 0.0
F3R	3.16 ± 0.5	20.64 ± 4.6	0 ± 0	4.20 ± 1.4	3.23 ± 0.9	26 ± 4	0.74 ± 0.0
F2C	0.91 ± 0.0	0.14 ± 0.0	2 ± 0	0.00 ± 0.0	-0.04 ± 0.0	42 ± 0	0.89 ± 0.0
F3C	0.07 ± 0.0	0.14 ± 0.0	2 ± 0	0.00 ± 0.0	0.22 ± 0.0	64 ± 0	1.00 ± 0.0
F2O	3.27 ± 0.3	20.63 ± 3.0	0 ± 0	4.40 ± 1.2	3.24 ± 0.5	23 ± 4	0.74 ± 0.0
F3O	1.46 ± 0.3	3.55 ± 1.3	1 ± 1	0.80 ± 1.3	1.19 ± 0.4	29 ± 5	0.83 ± 0.0

Table 1: Accuracy results for two of the investigated games.

- There are pathological examples where all types of factorization result in selecting the worst possible joint action. Given that only joint learners seem to be able to address such problems, currently no scalable deep RL methods for dealing with those seem to exist.
- Beyond those pathological examples, 'complete factorizations' of modest factor size yield near perfect reconstructions and rankings of the actions, also for non-factored action-value functions, while exhibiting better scaling behaviour.
- For these more benign problems, random overlapping factors also achieve excellent performance.

4 CONCLUSIONS

In this work, we investigated how well neural networks can represent action-value functions arising from multi-agent systems. This is an important question since accurate representations can enable taking (near-) optimal actions in value-based approaches, and computing good gradient estimates in actor-critic methods. In this paper, we focused on one-shot games as the simplest setting that captures the exponentially large joint action space of MASs. We compared a number of existing and new action-value network factorizations and learning approaches.

Our results highlight the difficulty of compactly representing action values in problems that require tight coordination, but indicate that using higher-order factorizations with multiple agents in each factor can improve the accuracy of these representations substantially. We also demonstrate that there are non-trivial coordination problems - some without a factored structure - that can be tackled quite well with simpler factorizations. Intriguingly, incomplete, overlapping factors perform very well.

ACKNOWLEDGEMENTS

This research made use of a GPU donated by NVIDIA. F.A.O. is funded by EPSRC First Grant EP/R001227/1. This project received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No. 758824 —INFLUENCE).

[1]In the full length version of this paper [3], we call this approach the *factored Q-function* approach. There, we also investigate another approach called the *mixture of experts* approach [1], which, in the case of one agent per factor, corresponds to independent learners [25].

REFERENCES

[1] Christopher Amato and Frans A. Oliehoek. 2015. Scalable Planning and Learning for Multiagent POMDPs. In *Proceedings of the 29th AAAI Conference on Artificial Intelligence (AAAI'15)*. American Association for Artificial Intelligence, 1995–2002.

[2] Lucian Busoniu, Robert Babuska, and Bart De Schutter. 2008. A Comprehensive Survey of Multiagent Reinforcement Learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 38 (2008), 156–172.

[3] Jacopo Castellini, Frans A. Oliehoek, Rahul Savani, and Shimon Whiteson. 2019. The Representational Capacity of Action-Value Networks for Multi-Agent Reinforcement Learning. *CoRR* abs/1902.07497 (2019), 19.

[4] Caroline Claus and Craig Boutilier. 1998. The Dynamics of Reinforcement Learning in Cooperative Multiagent Systems. In *Proceedings of the 15th/10th AAAI Conference on Artificial Intelligence/Innovative Applications of Artificial Intelligence (AAAI'98/IAAI'98)*. American Association for Artificial Intelligence, 746–752.

[5] Jakob Foerster, Ioannis Alexandros Assael, Nando de Freitas, and Shimon Whiteson. 2016. Learning to Communicate with Deep Multi-Agent Reinforcement Learning. In *Advances in Neural Information Processing Systems 29 (NIPS'16)*. Curran Associates, Inc., 2137–2145.

[6] Jakob N. Foerster, Gregory Farquhar, Triantafyllos Afouras, Nantas Nardelli, and Shimon Whiteson. 2018. Counterfactual Multi-Agent Policy Gradients. In *Proceedings of the 32th AAAI Conference on Artificial Intelligence (AAAI'18)*. American Association for Artificial Intelligence, 2974–2982.

[7] Carlos Guestrin, Daphne Koller, and Ronald Parr. 2002. Multiagent Planning with Factored MDPs. In *Advances in Neural Information Processing Systems 14 (NIPS'02)*. Morgan Kaufmann Publishers Inc., 1523–1530.

[8] Carlos Guestrin, Daphne Koller, Ronald Parr, and Shobha Venkataraman. 2003. Efficient Solution Algorithms for Factored MDPs. *Journal of Artificial Intelligence Research* 19, 1 (2003), 399–468.

[9] Carlos Guestrin, Michail G. Lagoudakis, and Ronald Parr. 2002. Coordinated Reinforcement Learning. In *Proceedings of the 19th International Conference on Machine Learning (ICML'02)*. Morgan Kaufmann Publishers Inc., 227–234.

[10] Jayesh K. Gupta, Maxim Egorov, and Mykel Kochenderfer. 2017. Cooperative Multi-agent Control Using Deep Reinforcement Learning. (2017), 66–83.

[11] Jelle R. Kok and Nikos Vlassis. 2006. Collaborative Multiagent Reinforcement Learning by Payoff Propagation. *Journal of Machine Learning Research* 7 (2006), 1789–1828.

[12] Joel Z. Leibo, Vinicius Zambaldi, Marc Lanctot, Janusz Marecki, and Thore Graepel. 2017. Multi-Agent Reinforcement Learning in Sequential Social Dilemmas. In *Proceedings of the 16th International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS'17)*. International Foundation for Autonomous Agents and Multiagent Systems, 464–473.

[13] Timothy P. Lillicrap, Jonathan J. Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. 2015. Continuous Control with Deep Reinforcement Learning. *CoRR* abs/1509.02971 (2015), 14.

[14] Ryan Lowe, Yi Wu, Aviv Tamar, Jean Harb, Pieter Abbeel, and Igor Mordatch. 2017. Multi-Agent Actor-Critic for Mixed Cooperative-Competitive Environments. In *Advances in Neural Information Processing Systems 30 (NIPS'17)*. Curran Associates, Inc., 6379–6390.

[15] Laëtitia Matignon, Guillaume J. Laurent, and Nadine Le Fort-Piat. 2012. Independent Reinforcement Learners in Cooperative Markov games: a Survey Regarding Coordination Problems. *Knowledge Engineering Review* 27, 1 (2012), 1–31.

[16] Volodymyr Mnih, Adrià Puigdomènech Badia, Mehdi Mirza, Alex Graves, Tim Harley, Timothy P. Lillicrap, David Silver, and Koray Kavukcuoglu. 2016. Asynchronous Methods for Deep Reinforcement Learning. In *Proceedings of the 33rd International Conference on Machine Learning (ICML'16)*, Vol. 48. PMLR, 1928–1937.

[17] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A. Rusu, Joel Veness, Marc G. Bellemare, Alex Graves, Martin Riedmiller, Andreas K. Fidjeland, Georg Ostrovski, Stig Petersen, Charles Beattie, Amir Sadik, Ioannis Antonoglou, Helen King, Dharshan Kumaran, Daan Wierstra, Shane Legg, and Demis Hassabis. 2015. Human-Level Control Through Deep Reinforcement Learning. *Nature* 518, 7540 (2015), 529–533.

[18] Frans A. Oliehoek. 2010. *Value-Based Planning for Teams of Agents in Stochastic Partially Observable Environments*. Ph.D. Dissertation. Informatics Institute, University of Amsterdam.

[19] Frans A. Oliehoek, Shimon Whiteson, and Matthijs T. J. Spaan. 2011. Exploiting Agent and Type Independence in Collaborative Graphical Bayesian Games. (2011).

[20] Tabish Rashid, Mikayel Samvelyan, Christian Schröder de Witt, Gregory Farquhar, Jakob N. Foerster, and Shimon Whiteson. 2018. QMIX: Monotonic Value Function Factorisation for Deep Multi-Agent Reinforcement Learning. In *Proceedings of the 35th International Conference on Machine Learning (ICML'18)*. JMLR.org, 4292–4301.

[21] A. Rogers, A. Farinelli, R. Stranders, and N. R. Jennings. 2011. Bounded Approximate Decentralised Coordination via the Max-sum Algorithm. *Artificial Intelligence* 175, 2 (2011), 730–759.

[22] Sainbayar Sukhbaatar, Arthur Szlam, and Rob Fergus. 2016. Learning Multiagent Communication with Backpropagation. In *Proceedings of the 30th International Conference on Neural Information Processing Systems (NIPS'16)*. Curran Associates, Inc., 2252–2260.

[23] Peter Sunehag, Guy Lever, Audrunas Gruslys, Wojciech Marian Czarnecki, Vinicius Zambaldi, Max Jaderberg, Marc Lanctot, Nicolas Sonnerat, Joel Z. Leibo, Karl Tuyls, and Thore Graepel. 2018. Value-Decomposition Networks For Cooperative Multi-Agent Learning Based On Team Reward. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS'18)*. International Foundation for Autonomous Agents and Multiagent Systems, 2085–2087.

[24] Ardi Tampuu, Tambet Matiisen, Dorian Kodelja, Ilya Kuzovkin, Kristjan Korjus, Juhan Aru, Jaan Aru, and Raul Vicente. 2017. Multiagent Cooperation and Competition with Deep Reinforcement Learning. *PLoS ONE* 12, 4 (2017), 1–15.

[25] Ming Tan. 1993. Multi-Agent Reinforcement Learning: Independent vs. Cooperative Agents. In *Proceedings of the 10th International Conference on Machine Learning (ICML'93)*. Morgan Kaufmann Publishers Inc., 330–337.

[26] Elise Van der Pol and Frans A. Oliehoek. 2016. Coordinated Deep Reinforcement Learners for Traffic Light Control. In *NIPS'16 Workshop on Learning, Inference and Control of Multi-Agent Systems*. 8.

[27] Ermo Wei and Sean Luke. 2016. Lenient Learning in Independent-Learner Stochastic Cooperative Games. *Journal of Machine Learning Research* 17, 84 (2016), 1–42.

[28] Michael Wunder, Michael L. Littman, and Monica Babes. 2010. Classes of Multiagent Q-learning Dynamics with Epsilon-Greedy Exploration. In *Proceedings of the 27th International Conference on Machine Learning (ICML'10)*. Omnipress, 1167–1174.

[29] Dayong Ye, Minjie Zhang, and Yun Yang. 2015. A Multi-Agent Framework for Packet Routing in Wireless Sensor Networks. *Sensors* 15, 5 (2015), 10026–10047.