Supplementary Material: Multi-Agent Coordination in Adversarial Environments through Signal Mediated Strategies

Anonymous Author(s)
Submission Id: 158

KEYWORDS

Team Games, Multi-Agent Reinforcement Learning, Coordination

ACM Reference Format:

Anonymous Author(s). 2021. Supplementary Material: Multi-Agent Coordination in Adversarial Environments through Signal Mediated Strategies. In *Proc. of the 20th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2021), London, UK, May 3–7, 2021*, IFAAMAS, 2 pages.

A ALGORITHMS FOR TRAJECTORY SAMPLING

Trajectory sampling plays an important role for the convergence of the players' strategies to the equilibrium of the game. We give a brief description of two offline RL algorithms that can be used in our framework to collect trajectories on the relaxed game.

Neural Fictitious Self-Play (NFSP). Fictitious Play [1] is a game-theoretic self-play algorithm in which players iteratively play their best responses against their opponents' average strategies. In certain classes of games, e.g. two-player zero-sum [9] and many-player potential games [7], the average strategies are guaranteed to converge to Nash Equilibria (NE), or to approximate ϵ-NE [5], under approximate best responses and perturbed average strategy updates.

The original formulation of Fictitious Play is defined for normal-form games, resulting in exponential complexity for extensive-form games. Heinrich et al. [3] introduced exact (XFP) and machine learning-based (FPS) versions of the model that are implemented in behavioural strategies. The algorithms are realization equivalent to their normal-form counterpart, inheriting its convergence guarantees to an exact and ϵ-NE, respectively while reducing the complexity from exponential to linear in time and space.

A third variant, used in this work, *Neural Fictitious Self-Play* (NFSP) [4], combines FSP with neural function approximators. In NFSP, agents interact with each others generating datasets of experience in self-play. Each agent collects and stores its experienced transition tuples, $(s_t, a_t, r_{t+1}, s_{t+1})$, in a memory, $\mathcal{M}_{RL}$, while its own best response behaviour, (s_t, a_t), is stored in a separate memory, $\mathcal{M}_{SL}$. Each agent uses off-policy reinforcement learning by training a DQN [6], $Q(s, a|\theta^Q)$, to predict action values, from the memory $\mathcal{M}_{RL}$ of its opponents' behaviour. The agent's approximate best response strategy is defined as $\beta = \epsilon\text{-greedy}(Q)$, which selects a random action with probability ϵ and chooses the action that maximizes the predicted action values otherwise. A separate neural network, $\Pi(s, a|\theta^\Pi)$, is trained to imitate the agent's own past best response behaviour using supervised classification on the data in $\mathcal{M}_{SL}$. This network maps states to action probabilities and defines the agent's average strategy, the one that is guaranteed to converge to the equilibrium strategy.

QMIX. QMIX [8] is an offline value-based method that can train decentralised policies in a centralised end-to-end fashion inducing agents' coordination. Learning how to coordinate multiple agents toward cooperative behaviours is hard due to numerous challenges. One can train fully decentralized policies disregarding agents' interactions as in *Independent Q-Learning (IQL)* [11], but this may not converge because of non-stationary caused by others agents learning and exploration. On the other hand, centralised learning of joint actions can naturally handle coordination problems and avoids non-stationarity, but is hard to scale, as the joint action space grows exponentially in the number of agents. Similarly to *Value Decomposition Networks (VDNs)* [10], QMIX lies between IQL and centralised Q-Learning. By estimating a joint action-values Q_{tot} as a non-linear combination of per-agent values Q_a, conditioned only on local observations, QMIX can represent complex centralised action-value functions with a factored representation that scales well in the number of agents. QMIX enforces that a global arg max performed on Q_{tot} yields the same result as a set of individual *argmax* operations performed on each Q_a. This allows each agent to compute decentralized policies by choosing greedy actions with respect to its Q_a. Agents can also be conditioned on their action-observation history to deal with partial observability in the environment by using recurrent neural networks to model their value functions [2]. For more detailed descriptions of the presented algorithms please refer to the original papers.

REFERENCES

[1] George W Brown. 1951. Iterative solution of games by fictitious play. *Activity analysis of production and allocation* 13, 1 (1951), 374–376.

[2] Matthew Hausknecht and Peter Stone. 2015. Deep Recurrent Q-Learning for Partially Observable MDPs. In *AAAI Fall Symposium on Sequential Decision Making for Intelligent Agents (AAAI-SDMIA15)* (Arlington, Virginia, USA).

[3] Johannes Heinrich, Marc Lanctot, and David Silver. 2015. Fictitious self-play in extensive-form games. In *International Conference on Machine Learning*. 805–813.

[4] Johannes Heinrich and David Silver. 2016. Deep reinforcement learning from self-play in imperfect-information games. *arXiv preprint arXiv:1603.01121* (2016).

[5] David S Leslie and Edmund J Collins. 2006. Generalised weakened fictitious play. *Games and Economic Behavior* 56, 2 (2006), 285–298.

[6] 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.

[7] Dov Monderer and Lloyd S Shapley. 1996. Fictitious play property for games with identical interests. *Journal of economic theory* 68, 1 (1996), 258–265.

[8] Tabish Rashid, Mikayel Samvelyan, Christian Schroeder, Gregory Farquhar, Jakob 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 (Proceedings of Machine Learning Research, Vol. 80)*, Jennifer Dy and Andreas Krause (Eds.). PMLR, Stockholmsmässan, Stockholm Sweden, 4295–4304. http://proceedings.mlr.press/v80/rashid18a.html

[9] Julia Robinson. 1951. An iterative method of solving a game. *Annals of mathematics* (1951), 296–301.

Proc. of the 20th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2021), U. Endriss, A. Nowé, F. Dignum, A. Lomuscio (eds.), May 3–7, 2021, London, UK.

[10] Peter Sunehag, Guy Lever, Audrunas Gruslys, Wojciech Marian Czarnecki, Vinícius Flores Zambaldi, Max Jaderberg, Marc Lanctot, Nicolas Sonnerat, Joel Z. Leibo, Karl Tuyls, and Thore Graepel. 2017. Value-Decomposition Networks For Cooperative Multi-Agent Learning. *CoRR* abs/1706.05296 (2017). arXiv:1706.05296 http://arxiv.org/abs/1706.05296

[11] Ming Tan. 1993. Multi-agent reinforcement learning: Independent vs. cooperative agents. In *Proceedings of the tenth international conference on machine learning*. 330–337.