Uncertainty-Based Transformation for Monotonic Value Function Factorization in Multi-Agent Reinforcement Learning

Anonymous Authors[1]

1. Related work

This section briefly introduces recent related work on cooperative multi-agent reinforcement learning (MARL) in the paradigm of centralized training with decentralized execution (CTDE). One of the most significant challenges in CTDE is to ensure the correspondence between the individual Q-value functions and the joint Q-value function Q_{tot}, i.e., the Individual-Global Max (IGM) principle (Son et al., 2019). VDN (Sunehag et al., 2018) and QMIX (Rashid et al., 2018) learn the joint Q-values and factorize them into individual Q-value functions in an additive and a monotonic fashion, respectively. Qatten (Yang et al., 2020b) is a variant of QMIX, which applies a multi-head attention structure to the mixing network. QPD (Yang et al., 2020a) utilizes integrated gradients to decompose Q_{tot} along trajectory paths. SMIX(λ) (Yao et al., 2021) changes the one-step Q-learning target with a SARSA(λ) target for QMIX. RODE (Wang et al., 2020b) decomposes joint action spaces into restricted role action spaces to boost learning efficiency and policy generalization. These methods apply the same monotonic mixing network and thus can only represent the same class as QMIX. However, as many previous studies pointed out, monotonic value function factorization limits the representational capacity of Q_{tot}, and fails to learn the optimal policy when the target Q-value functions are non-monotonic (Mahajan et al., 2019; Son et al., 2019; Rashid et al., 2020).

Some recent works try to achieve the full representational capacity of Q_{tot} to solve this problem. QPLEX (Wang et al., 2020a) proposes a duelling mixing network, in which the weights are produced through joint actions. Deep coordination graph (Böhmer et al., 2020) decomposes the Q_{tot} into individual utilities and payoff contributions based on the actions of the agents connected by the (hyper-)edges. Tesseract (Mahajan et al., 2021) decomposes the Q-tensor across agents and utilises low-rank tensor approximations to model agent interactions relevant to the task, and thus learns a compact approximation of the target Q-value function. However, since the dimension of the state-action space increases exponentially as the number of agents grows, it is not easy to achieve the full representational capacity in complex MARL tasks.

Another solution is to learn a biased Q_{tot} by prioritizing the optimal joint action. QTRAN (Son et al., 2019) uses two soft regularisations to align the greedy action selections between the joint Q-value and the individual values. WQMIX (Rashid et al., 2020) introduces a weighting mechanism to place more importance on better joint actions. QTRAN can be viewed as a variant of WQMIX, which additionally uses the weight 0 for the joint Q-value whose target is smaller than the current estimate and the chosen action is not greedy. QTRAN approximates $\hat{Q}^*$ as the target instead of the original target y and thus can deal with stochasticity. However, due to the 0 weight for overestimated Q-values, QTRAN is empirically hard to scale to more complex tasks (Samvelyan et al., 2019). We also prove that there may not exist an appropriate weight when the targets are non-monotonic and stochastic.

In addition, there have been many developments in policy-based methods under CTDE settings. MAPPO (Yu et al., 2021) applies PPO (Schulman et al., 2017) into MARL and shows strong empirical performance. However, Kuba et al. (2021) points out MAPPO suffers from instability arising from the non-stationarity induced by simultaneously learning and exploring agents. Therefore, they introduce the sequential policy update scheme to achieve monotonic improvement on the joint policy. However, this method can only guarantee convergence to one of the Nash Equilibriums and falls into the suboptimal if this solution cannot be improved by coordinate descent (Bertsekas, 2019). This problem can be interpreted as the non-monotonic target problem in value function factorization methods.

Relationship To Reward Shaping. Reward shaping is a common technique for improving single-agent learners' performance by integrating expert knowledge into MDP (Gullapalli & Barto, 1992). Since it is not always possible to find an expert with complete domain knowledge, setting rewards for complex MARL tasks in advance is difficult. Therefore, shaping the reward adaptively is a more attractive way.

[1] Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

Some reward randomisation methods (Gupta et al., 2021; Tang et al., 2021) are proposed to convert the original MDP to a new MDP with random rewards through universal successor features. However, these methods require a massive number of randomisation to find a desirable reward function. GVR (Wan et al., 2022) proposes inferior target reshaping and superior experience replay to eliminate the non-optimal self-transition nodes, which is similar to WQMIX and ignores the stochasticity of the transition. Compared with these methods, UTRAN is more efficient because it can perfectly represent the expected value for stochastic targets and guarantee policy invariance during target shaping.

Relationship To Independent Learning Algorithms. Some independent learning algorithms have proven robust to solve relative overgeneralization in the low-dimensional setting. Distributed Q-learning (Lauer, 2000) and Hysteretic Q-learning (Matignon et al., 2007) place more importance on positive updates that increase a Q-value estimate, which is similar to the weighting function in WQMIX. However, Wei & Luke (2016) prove that these methods are vulnerable towards misleading stochasticity and propose LMRL2 (Wei & Luke, 2016), where agents forgive the other's miscoordination in the initial exploration phase but become less lenient when the visitation of state-action pair increases. However, it requires carefully tuned hyperparameters that rarely translate across domains. Best possible Q-learning (Jiang & Lu, 2023) computes the expected values of all possible transition probabilities and updates the state-action value to be the maximal one, which is similar to the best per-agent value function we used for transformation. However, this method achieves lower performance than UTRAN in the team-reward task because it cannot guarantee consistency between decentralized policies and ignores credit assignment, which is verified in our ablation study (UBEST in Sec. 2.1).

2. Experiments

2.1. Ablation Study

In this section, we conduct an ablation study to demonstrate the contribution of each component in UTRAN. We compare UTRAN with its variants: 1) UTRAN without target transformation (UBEST), where agents choose actions based on the best per-agent value function, 2) UTRAN without the transition model (UTRAN-wo-T), 3) UTRAN without the reward model (UTRAN-wo-R), and 4) QMIX, the natural ablation of UTRAN.

Fig. 1 shows that UTRAN-wo-R cannot solve MMDP and PSH because these tasks involve stochastic rewards. The negative results in MMDP for UTRAN-wo-S demonstrate that it cannot identify stochastic state transitions. However, UTRAN-wo-S converge to the optimal quickly on PSH because this task only involves miscoordination penalty and stochastic rewards. Despite learning steadily on MMDP and PSH, UBEST fails to achieve good performance on SMAC and takes far longer to reach UTRAN and other variants. In contrast, UTRAN outperforms its variants across all tasks, demonstrating the effectiveness of the uncertainty-based target transformation for monotonic value factorization.

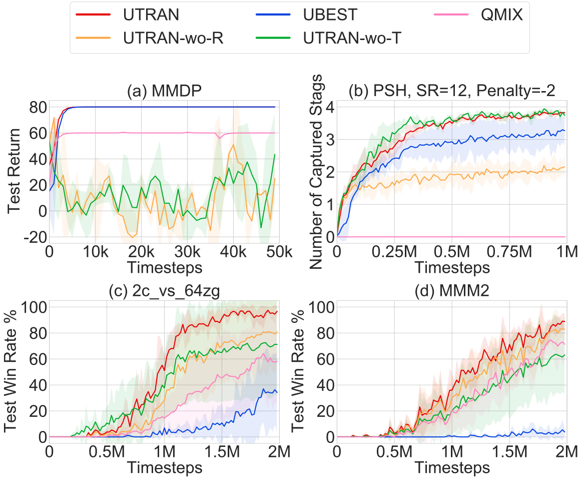

Figure 1. Ablation results on MMDP, PSH, and SMAC.

Here we emphasize the contribution of target transformation by analyzing why UTRAN outperforms UBEST. The first reason is that UTRAN realizes full representational capacity for the transformed targets, while UBEST does not have this advantage. The second one is that UBEST does not apply monotonic value function factorization in policy learning. Value factorization methods guarantee consistency between decentralized policies and implicitly achieve credit assignment using counterfactual baselines. In contrast, UBEST introduces additional estimation errors that may cause significant variances, leading to slow convergence when the most and the second greatest target value is close.

To verify our analysis, we consider a one-step 10×10 matrix game, where the suboptimal is filled with random numbers generated uniformly between -20 and 19, and the unique optimal value is +20. Fig. 2 shows that UTRAN has lower standard deviations than UBEST and QMIX, implying that monotonic value function factorization with transformed targets can reduce variances.

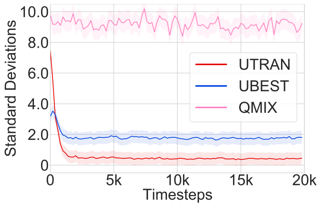

Figure 2. Standard deviations of per-agent action values.

References

Bertsekas, D. Multiagent rollout algorithms and reinforcement learning. *arXiv preprint arXiv:1910.00120*, 2019.

Böhmer, W., Kurin, V., and Whiteson, S. Deep coordination graphs. In *International Conference on Machine Learning*, pp. 980–991. PMLR, 2020.

Gullapalli, V. and Barto, A. G. Shaping as a method for accelerating reinforcement learning. In *Proceedings of the 1992 IEEE international symposium on intelligent control*, pp. 554–559. IEEE, 1992.

Gupta, T., Mahajan, A., Peng, B., Böhmer, W., and Whiteson, S. Uneven: Universal value exploration for multi-agent reinforcement learning. In *International Conference on Machine Learning*, pp. 3930–3941. PMLR, 2021.

Jiang, J. and Lu, Z. Best possible q-learning, 2023.

Kuba, J. G., Chen, R., Wen, M., Wen, Y., Sun, F., Wang, J., and Yang, Y. Trust region policy optimisation in multi-agent reinforcement learning. *arXiv preprint arXiv:2109.11251*, 2021.

Lauer, M. An algorithm for distributed reinforcement learning in cooperative multiagent systems. In *Proc. 17th International Conf. on Machine Learning*, 2000.

Mahajan, A., Rashid, T., Samvelyan, M., and Whiteson, S. Maven: Multi-agent variational exploration. *Advances in Neural Information Processing Systems*, 32, 2019.

Mahajan, A., Samvelyan, M., Mao, L., Makoviychuk, V., Garg, A., Kossaifi, J., Whiteson, S., Zhu, Y., and Anandkumar, A. Tesseract: Tensorised actors for multi-agent reinforcement learning. In *International Conference on Machine Learning*, pp. 7301–7312. PMLR, 2021.

Matignon, L., Laurent, G. J., and Le Fort-Piat, N. Hysteretic q-learning: an algorithm for decentralized reinforcement learning in cooperative multi-agent teams. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 64–69. IEEE, 2007.

Rashid, T., Samvelyan, M., Schroeder, C., Farquhar, G., Foerster, J., and Whiteson, S. Qmix: Monotonic value function factorisation for deep multi-agent reinforcement learning. In *International Conference on Machine Learning*, pp. 4295–4304. PMLR, 2018.

Rashid, T., Farquhar, G., Peng, B., and Whiteson, S. Weighted qmix: Expanding monotonic value function factorisation for deep multi-agent reinforcement learning. *Advances in neural information processing systems*, 33: 10199–10210, 2020.

Samvelyan, M., Rashid, T., De Witt, C. S., Farquhar, G., Nardelli, N., Rudner, T. G., Hung, C.-M., Torr, P. H., Foerster, J., and Whiteson, S. The starcraft multi-agent challenge. *arXiv preprint arXiv:1902.04043*, 2019.

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

Son, K., Kim, D., Kang, W. J., Hostallero, D. E., and Yi, Y. Qtran: Learning to factorize with transformation for cooperative multi-agent reinforcement learning. In *International Conference on Machine Learning*, pp. 5887–5896. PMLR, 2019.

Sunehag, P., Lever, G., Gruslys, A., Czarnecki, W. M., Zambaldi, V., Jaderberg, M., Lanctot, M., Sonnerat, N., Leibo, J. Z., Tuyls, K., et al. 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*, pp. 2085–2087, 2018.

Tang, Z., Yu, C., Chen, B., Xu, H., Wang, X., Fang, F., Du, S. S., Wang, Y., and Wu, Y. Discovering diverse multi-agent strategic behavior via reward randomization. In *International Conference on Learning Representations*, 2021.

Wan, L., Liu, Z., Chen, X., Lan, X., and Zheng, N. Greedy based value representation for optimal coordination in multi-agent reinforcement learning. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162, pp. 22512–22535. PMLR, 17–23 Jul 2022.

Wang, J., Ren, Z., Liu, T., Yu, Y., and Zhang, C. Qplex: Duplex dueling multi-agent q-learning. In *International Conference on Learning Representations*, 2020a.

Wang, T., Gupta, T., Mahajan, A., Peng, B., Whiteson, S., and Zhang, C. Rode: Learning roles to decompose multi-agent tasks. *arXiv preprint arXiv:2010.01523*, 2020b.

Wei, E. and Luke, S. Lenient learning in independent-learner stochastic cooperative games. *The Journal of Machine Learning Research*, 17(1):2914–2955, 2016.

Yang, Y., Hao, J., Chen, G., Tang, H., Chen, Y., Hu, Y., Fan, C., and Wei, Z. Q-value path decomposition for deep multiagent reinforcement learning. In *International Conference on Machine Learning*, pp. 10706–10715. PMLR, 2020a.

Yang, Y., Hao, J., Liao, B., Shao, K., Chen, G., Liu, W., and Tang, H. Qatten: A general framework for cooperative multiagent reinforcement learning. *arXiv preprint arXiv:2002.03939*, 2020b.

Yao, X., Wen, C., Wang, Y., and Tan, X. Smix (λ): Enhancing centralized value functions for cooperative multiagent reinforcement learning. *IEEE Transactions on Neural Networks and Learning Systems*, 2021.

Yu, C., Velu, A., Vinitsky, E., Wang, Y., Bayen, A., and Wu, Y. The surprising effectiveness of ppo in cooperative, multi-agent games. *arXiv preprint arXiv:2103.01955*, 2021.