SAEIR: Sequentially Accumulated Entropy Intrinsic Reward for Cooperative Multi-Agent Reinforcement Learning with Sparse Reward

Xin He , **Hongwei Ge*** , **Yaqing Hou** and **Jincheng Yu**
School of Computer Science and Technology, Dalian University of Technology
hx_dlut@mail.dlut.edu.cn, hwge@dlut.edu.cn

A Additional Environment Information

Here, we present additional information about two sparse-reward environments used to benchmark SAEIR against baseline methods.

A.1 Cooperative Navigation

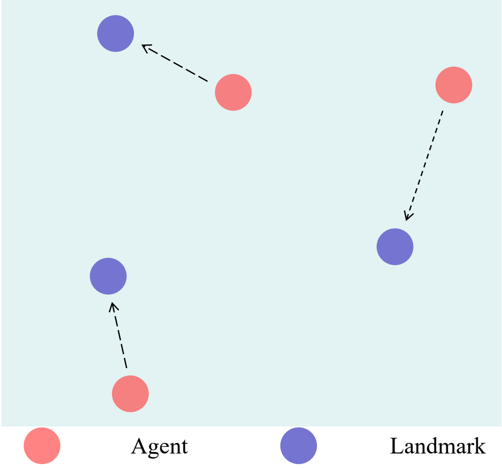

Figure 1: The visualization of the cooperative navigation (N=3)

We utilize the cooperative navigation environment from [Lowe *et al.*, 2017]. Here, there are N agents co-operative reaching the same number of the landmarks while avoiding collision. The locations of the agents and the landmarks are initialized randomly at the start of each episode. The agents can take actions in action set $\{up, down, lef, right, stay\}$. The agents get $+l$ reward when l landmarks are reached and get a reward of -1 when any agents collide with each other. The agents are only penalized by collision once at every time step. We test our method on two sparse reward scenario with different number of the agents in this environment. The scenario with $N = 3$ agents and $L = 3$ landmarks is shown in Appendix Fig. 1. The higher-performing method addressing the challenge of sparse reward in this domain is define as one that maximizes the coverage rate for the landmarks reached by agents.

A.2 Google Football Research

Figure 2: The visualization of the academy 3 vs.1 with keeper

The second domain we utilize is the google football research environment [Kurach *et al.*, 2020], which is a physics-based 3D soccer simulator for multi-agent reinforcement learning. It provides several challenging, mixed cooperative-competitive, multi-agent scenarios with high stochasticity and sparse rewards. The players can take 19 actions including moving actions, kicking actions, and other actions such as dribbling, sliding and sprint. There are two types of reward that can be used out-of-the-box: SCORING and CHECKPOINT. Under SCORING reward function, the players only get a $+1$ reward for scoring a goal and a -1 reward when conceding one to the opposing team. CHECKPOINT reward function provides a dense reward to address the sparsity of SCORING reward function by encoding the domain knowledge. To fit more realistic, we test our method under SCORING reward function. Appendix Fig 2 shows one of scenarios called academy 3 vs.1 with keeper. Another scenario we test our method is academy counterattack easy. Both two scenarios are widely utilized to evaluate the robustness of MARL algorithms to stochasticity and sparse rewards [Ma *et al.*, 2022; Xu *et al.*, 2023a; Xu *et al.*, 2023b]. In this domain, we seek to maximize the winning rate (i.e., the agent scores a goal in a match)

B The Final Convergence of Intrinsic Reward

During the training procedure, the disorder of the system state gradually decreases and intrinsic rewards finally converge to a constant close to 0. Fig. 3 shows each intrinsic reward in the final training episode. We can see that each intrinsic reward is very small compared to the external reward, indicating that intrinsic rewards finally converge to a constant close to 0.

C Hyperparameters

All algorithms are trained on a Ubuntu 20.04 operation system with CPU Intel Xeon E5-2630 v3 and GPU Tesla P40

Table 1: Hyper-parameters of implementation and configuration for SAEIR and MAPPO, SN-MAPPO, RND, Elign.

Hyperparameter	Cooperative Navigation	Google Football Research
Network initialization	Orthogonal	Orthogonal
Optimizer	Adam	Adam
Activation	Tanh	Relu
Optimizer Epsilon	1e-5	1e-5
GAE λ	0.95	0.95
Discount Factor γ	0.99	0.99
Learning Rate	7e-4	5e-4
Clip ϵ	0.2	0.2
Max Training Steps	2e7	2.5e7
PPO Epochs	10	15
Buffer Length	25	200
Recurrent data chunk length	10	10
Hidden Layer Size	[64,64,64,64,64]	[64,64,128,128,128]

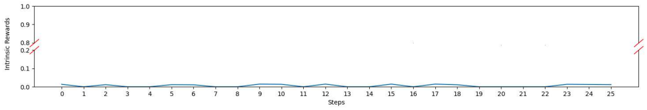

Figure 3: Each intrinsic reward in the final training episode.

and the neural network framework is built on Pytorch [Paszke *et al.*, 2019]. Table 1 shows the hyper-parameters of implementation and configuration for SAEIR and MAPPO, SN-MAPPO, RND, Elign. We provide our code at here.

D Training Details

The pseudo code of the SAEIR applied to the agent is given in the submission of Sec.4.3 Algorithm 1. SAETRA trains four separate neural networks: an plain actor network with parameters π, and three value function networks (referred to as three scales of hypergraph critics) with parameters $\theta^K, K \in \{individual, group, team\}$. The parameters π and θ are initialized with Orthogonal initialization technique [Huang *et al.*, 2021]. Firstly, the plain actor network maps agent observations to a distribution used for sampling an action and the multi-scale hypergraph critic networks perform the following mapping: $S^K \to \mathbb{R}$. Next, the needed information is stored in the replay buffer and the intrinsic reward is calculated. Then, the advantages estimate and the the discounted reward-to-go are calculated by PopArt technique [van Hasselt *et al.*, 2016]. Finally, the parameters of neural networks are updated and the policy is evaluated after training.

E Exact Results of Ablation Study

Appendix Table 2 shows the exact results in academy 3 vs.1 with keeper scenario.

Table 2: Exact results of ablation methods in academy 3 vs.1 with keeper scenario.

Methods	Success Rate(%)	Episode Rewards
MAPPO	53.77 ± 8.53	9.86 ± 2.50
w. MSHG-Critic	69.31 ± 13.95	12.41 ± 3.18
w. Entropy Bouns	69.08 ± 15.55	13.27 ± 2.81
Our	$\mathbf{84.95 \pm 4.10}$	$\mathbf{15.06 \pm 1.50}$

References

[Huang *et al.*, 2021] Wei Huang, Weitao Du, and Richard Yi Da Xu. On the neural tangent kernel of deep networks with orthogonal initialization. In *Proc. 30th Int. Joint Conf. Artif. Intell.*, pages 2577–2583, 2021.

[Kurach *et al.*, 2020] Karol Kurach, Anton Raichuk, Piotr Stanczyk, Michal Zajac, Olivier Bachem, Lasse Espeholt, Carlos Riquelme, Damien Vincent, Marcin Michalski, Olivier Bousquet, and Sylvain Gelly. Google research football: A novel reinforcement learning environment. In *Proc. 34th AAAI Conf. Artif. Intell.*, pages 4501–4510, 2020.

[Lowe *et al.*, 2017] Ryan Lowe, Yi Wu, Aviv Tamar, Jean Harb, Pieter Abbeel, and Igor Mordatch. Multi-agent actor-critic for mixed cooperative-competitive environments. In *Proc. Int. Conf. Neural Inf. Process. Syst. 30*, pages 6379–6390, 2017.

[Ma *et al.*, 2022] Zixian Ma, Rose Wang, Fei-Fei Li, Michael S. Bernstein, and Ranjay Krishna. ELIGN: expectation alignment as a multi-agent intrinsic reward. In *Proc. Int. Conf. Neural Inf. Process. Syst. 35*, 2022.

[Paszke *et al.*, 2019] Adam Paszke, Sam Gross, Francisco Massa, and et al. Pytorch: An imperative style, high-performance deep learning library. In *Proc. Int. Conf. Neural Inf. Process. Syst. 32*, pages 8024–8035, 2019.

[van Hasselt *et al.*, 2016] Hado van Hasselt, Arthur Guez, Matteo Hessel, Volodymyr Mnih, and David Silver. Learning values across many orders of magnitude. In *Proc. Int. Conf. Neural Inf. Process. Syst. 29*, pages 4287–4295, 2016.

[Xu *et al.*, 2023a] Pei Xu, Junge Zhang, and Kaiqi Huang. Exploration via joint policy diversity for sparse-reward multi-agent tasks. In *Proc. 32th Int. Joint Conf. Artif. Intell.*, pages 326–334, 2023.

[Xu *et al.*, 2023b] Pei Xu, Junge Zhang, Qiyue Yin, Chao Yu, Yaodong Yang, and Kaiqi Huang. Subspace-aware exploration for sparse-reward multi-agent tasks. In *Proc. 37th AAAI Conf. Artif. Intell.*, pages 11717–11725, 2023.