Multi-Agent Path Finding via Reinforcement Learning with Hybrid Reward

Extended Abstract

Cheng Zhao
Univ. of Sci. & Tech. of China
Hefei, China
zc16@mail.ustc.edu.cn

Liansheng Zhuang
Univ. of Sci. & Tech. of China
Hefei, China
lszhuang@ustc.edu.cn

Haonan Liu
Univ. of Sci. & Tech. of China
Hefei, China
phoenix_@mail.ustc.edu.cn

Yihong Huang
Univ. of Sci. & Tech. of China
Hefei, China
hyh1109@mail.ustc.edu.cn

Jian Yang
Beijing Institute of Technology
Beijing, China
yuhengzi_8205@163.com

ABSTRACT

Multi-agent path finding (MAPF) aims to find a set of conflict-free paths for multiple agents so that each agent can reach its destination while optimizing a global cost. Recently, learning-based methods gain much attention due to their better real-time performance and scalability. However, most existing learning-based methods suffer from poor cooperation among agents since only local observations are used to make decisions. Meanwhile, methods that are bent on team benefits perform poorly due to a lack of individual exploration. To address this problem, this paper proposes a novel Hybrid Reward Path Finding (HRPF), which employs the global information to learn a cooperation mechanism for agents during the training, and embeds it in distributed networks to generate strategies during the execution. HRPF enforces agents to learn strategies from a new type of reward function that decomposes a complex MAPF task into a team task and individual tasks. Experiments on random obstacle grid worlds show that, HRPF performs significantly better in success rate and collision rate than state-of-the-art learning-based methods.

KEYWORDS

Multi-agent path finding; Multi-agent reinforcement learning; Hybrid reward function

ACM Reference Format:
Cheng Zhao, Liansheng Zhuang, Haonan Liu, Yihong Huang, and Jian Yang. 2023. Multi-Agent Path Finding via Reinforcement Learning with Hybrid Reward: Extended Abstract. In *Proc. of the 22nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2023), London, United Kingdom, May 29 – June 2, 2023*, IFAAMAS, 3 pages.

1 INTRODUCTION

The task of multi-agent path finding (MAPF) aims at planning a group of conflict-free and shortest paths for multiple agents [13]. MAPF arises in many real world applications of multi-agent systems such as warehouse robots [4], office robots [15], aircraft-towing vehicles [8]. A key challenge of MAPF is to avoid frequent collisions and blockages, as multiple agents interact with each other. Many search-based methods [2, 3, 11, 12] perform well in finding collision-free solutions using global information in simple environments, but perform poorly in terms of real-time performance and scalability.

Recently, a great amount of work focus on learning-based methods [9] for better real-time performance and scalability. Learning-based methods typically generate one-step policies online for agents based on localised observations. In order to implement distributed planning with effective collaboration, existing learning-based methods usually use multi-agent reinforcement learning (MARL) algorithms in a centralised training with decentralised execution (CTDE) paradigm [1, 6, 10, 16]. The popular MARL methods use team rewards to encourage agents to complete team tasks. However, a MAPF task is complex, with different sub-tasks to be completed by each agent, and the team reward does not guide each agent through the sub-task in detail. As a result, typical MARL methods that do not make full use of individual rewards are less efficient in training and perform poorly in practice.

This paper proposes a novel Hybrid Reward Path Finding (HRPF) for MAPF, which considers both the global team task and individual sub-tasks. HRPF builds on the CTDE paradigm, which inherits the benefits of distributed execution, while enabling agents to use global information to acquire collaborative skills. During the training, HRPF uses a novel mechanism of reward function, hybrid reward, to guide the agents' behaviour. Agents learn blueprint strategies for completing team tasks from team rewards and learn more refined strategies from individual rewards. During the execution, each agent independently makes decisions, only according to its individual observations, which guarantees the real-time performance and scalability of the method.

2 METHODOLOGY

HRPF defines hybrid reward for MAPF tasks, consisting of the team reward and the individual reward. The individual reward R^{in} is only visible to each agent itself and is only related to that agent's observation o and action a. It is defined as follows:

$$R^{\text{in}}(o,a) = \epsilon_g^{\text{in}}(o,a)R_g^{\text{in}} + \epsilon_c^{\text{in}}(o,a)R_c^{\text{in}} + \epsilon_p^{\text{in}}(o,a)R_p^{\text{in}}. \quad (1)$$

ϵ_g^{in}, ϵ_g^{in}, and ϵ_g^{in} are indicator functions for whether the agent first reaches the target, collides with other agents, and move or stay off the goal, respectively. R_g^{in}, R_c^{in}, R_p^{in} are reward values for the agent performing these actions. The team reward R^{te} is shared by the

Proc. of the 22nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2023), A. Ricci, W. Yeoh, N. Agmon, B. An (eds.), May 29 – June 2, 2023, London, United Kingdom. © 2023 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org). All rights reserved.

Table 1: The comparison of algorithms in different environments

Methods	8 Agent (0%, 10%, 20%, 30% Obstacle Densities)																							
	CA↓				CO↓				SR↑				MS↓				CR↓				TM↓			
HRPF	**0.4**	**0.4**	**0.6**	3.1	**0.0**	**0.0**	**0.0**	**0.0**	**100**	**99**	**91**	**50**	**26**	**29**	**52**	**152**	**0.01**	**0.01**	**0.01**	0.02	**110**	**115**	**144**	**314**
PRIMAL	1.9	3.0	3.0	6.0	**0.0**	**0.0**	**0.0**	**0.0**	93	90	48	15	35	63	149	234	0.06	0.05	0.02	0.03	221	233	345	565
PICO	0.6	0.6	1.3	**2.3**	**0.0**	**0.0**	**0.0**	**0.0**	100	96	55	25	27	42	135	205	0.02	**0.01**	**0.01**	**0.01**	124	143	290	463
Methods	**16 Agent (0%, 10%, 20%, 30% Obstacle Densities)**																							
	CA↓				CO↓				SR↑				MS↓				CR↓				TM↓			
HRPF	**1.1**	**2.1**	5.1	11.6	**0.0**	**0.0**	**0.0**	**0.0**	**100**	**96**	**65**	**13**	**28**	**39**	**127**	**232**	**0.04**	**0.05**	0.04	0.05	**219**	**244**	**380**	**686**
PRIMAL	6.6	8.3	11.6	17.6	**0.0**	**0.0**	0.1	0.1	92	88	50	3	57	72	176	249	0.11	0.12	0.07	0.07	482	510	766	1396
PICO	3.0	3.9	**5.0**	**8.0**	**0.0**	**0.0**	**0.0**	**0.0**	**100**	95	57	7	31	49	145	240	0.10	0.08	**0.03**	**0.03**	251	299	526	1292
Methods	**32 Agent (0%, 10%, 20%, 30% Obstacle Densities)**																							
	CA↓				CO↓				SR↑				MS↓				CR↓				TM↓			
HRPF	**5.3**	**10.5**	**24.3**	**42.0**	**0.0**	**0.0**	**0.0**	**0.0**	95	**78**	**21**	**1**	45	**88**	**218**	**255**	**0.12**	**0.12**	**0.11**	**0.16**	**471**	**571**	**1162**	**1620**
PRIMAL	26.2	30.5	47.3	98.3	**0.0**	0.4	1.6	2.1	92	72	9	0	54	108	245	256	0.49	0.28	0.19	0.38	958	1094	2227	3431
PICO	14.8	20.6	36.3	83.4	**0.0**	0.2	1.3	1.6	**100**	75	19	0	**38**	97	225	256	0.39	0.21	0.16	0.33	551	774	1713	3176

whole team and is related to the joint observation $\mathbf{o}$ and the joint action $\mathbf{a}$. It is defined as follows:

$$R^{\text{te}}(\mathbf{o},\mathbf{a}) = \epsilon_w^{\text{te}}(\mathbf{o},\mathbf{a})R_w^{\text{te}} + n_g^{\text{te}}(\mathbf{o},\mathbf{a})R_g^{\text{te}} + n_c^{\text{te}}(\mathbf{o},\mathbf{a})R_c^{\text{te}}. \tag{2}$$

The first term of RHS indicates the final reward when the task is completed, where ϵ_w^{te} is the indicator function for whether the task is completed. n_g^{te}, n_c^{te} denote the number of agents that first reach the destination and the number of agents that collide at the current time step, respectively. R_w^{te}, R_g^{te}, R_c^{te} are reward values.

HRPF trains two different Q-value networks for each agent, individual Q-network and team Q-network, based on DQN [7]. The input embedding and network structure of the Q-network is similar to that in PRIMAL [10], as shown in Fig. 1.

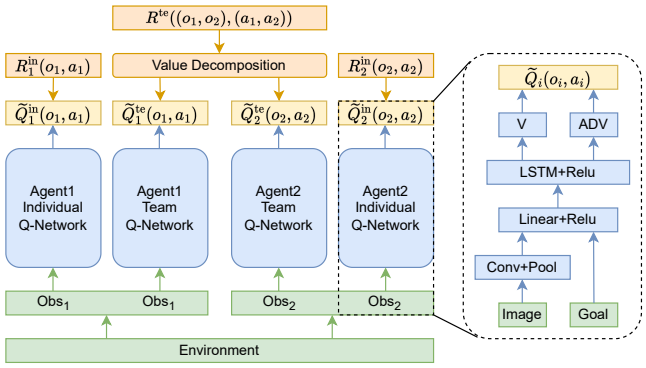

Figure 1: Network structure of HRPF.

During the training, each agent's networks output an estimated individual Q-value and a part of the team Q-value, denoted $\tilde{Q}_i^{\text{in}}$ and $\tilde{Q}_i^{\text{te}}$ respectively, where the subscript i denotes the agent index. The individual loss function L^{in} is the TD error between the individual reward and the network output:

$$L^{\text{in}} = [(R_i^{\text{in}} + \gamma \max_{a_i^{t+1}} \tilde{Q}_i^{\text{in}}(o_i^{t+1}, a_i^{t+1})) - \tilde{Q}_i^{\text{in}}(o_i^t, a_i^t)]^2, \tag{3}$$

where γ is the discount of the model, and the superscript t (or $t+1$) denotes the time step. For the team Q-network, all agents aim to approximate a total team Q-value $\tilde{Q}^{\text{te}}$ that is the sum of the output of each team Q-network $\tilde{Q}_i^{\text{te}}$. The monotonicity allows each agent to maximise the team Q^{te} and participate in distributed execution by selecting greedy actions only for its Q_i^{te} [14]. With the value decomposition technique, the team loss function L^{te} is:

$$L^{\text{te}} = [(R^{\text{te}} + \gamma \max_{a_i^{t+1}} \sum_i \tilde{Q}_i^{\text{te}}(o_i^{t+1}, a_i^{t+1})) - \sum_i \tilde{Q}_i^{\text{te}}(o_i^t, a_i^t)]^2. \tag{4}$$

During the execution, each agent combines the Q-values of the two networks to get a new type of value, denoted Q_i^{comb}. We balance the weight of the individual Q-value and the team Q-value by a parameter β. The formula for Q_i^{comb} is:

$$Q_i^{\text{comb}}(o_i^t, a_i^t) = \frac{1}{1+\beta}\tilde{Q}_i^{\text{in}}(o_i^t, a_i^t) + \frac{\beta}{1+\beta}\tilde{Q}_i^{\text{te}}(o_i^t, a_i^t). \tag{5}$$

Typically, the individual Q-value and the team Q-value should be weighted approximately equally, so in practice β is usually on the same scale as the number of agents k, as the total team Q-value is decomposed into k components for each agent. Next, each agent generates a ϵ-greedy policy based on Q_i^{comb}.

3 EXPERIMENT

We conduct experiments on random obstacle grid worlds with reference to the setup in PICO [5]. To test the performance under different settings, the number of agents varies in three cases: 8, 16, 32, and the density of obstacles varies in four cases: 0%, 10%, 20%, and 30%. The performance is evaluated on 6 different measurements compared with two baselines, PRIMAL [10] and PICO [5].

To further analyze the effectiveness of hybrid reward, we conduct ablation studies on the reward function. We propose two methods, IRPF and TRPF, with the team reward and the individual reward are removed respectively. Fig. 2 shows the result on grid worlds with an obstacle density of 20% and a number of agents of 8.

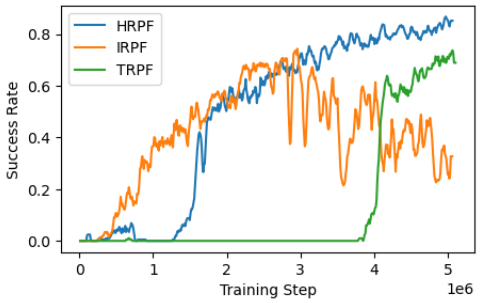

Figure 2: Ablation experiment result.

REFERENCES

[1] Vasilii Davydov, Alexey Skrynnik, Konstantin Yakovlev, and Aleksandr Panov. 2021. Q-Mixing Network for Multi-agent Pathfinding in Partially Observable Grid Environments. In *Russian Conference on Artificial Intelligence*. Springer, 169–179.

[2] Ariel Felner, Roni Stern, Solomon Shimony, Eli Boyarski, Meir Goldenberg, Guni Sharon, Nathan Sturtevant, Glenn Wagner, and Pavel Surynek. 2017. Search-based optimal solvers for the multi-agent pathfinding problem: Summary and challenges. In *International Symposium on Combinatorial Search*, Vol. 8.

[3] Cornelia Ferner, Glenn Wagner, and Howie Choset. 2013. ODrM* optimal multi-robot path planning in low dimensional search spaces. In *2013 IEEE International Conference on Robotics and Automation*. IEEE, 3854–3859.

[4] Wolfgang Hönig, Scott Kiesel, Andrew Tinka, Joseph W Durham, and Nora Ayanian. 2019. Persistent and robust execution of mapf schedules in warehouses. *IEEE Robotics and Automation Letters* 4, 2 (2019), 1125–1131.

[5] Wenhao Li, Hongjun Chen, Bo Jin, Wenzhe Tan, Hongyuan Zha, and Xiangfeng Wang. 2022. Multi-Agent Path Finding with Prioritized Communication Learning. *arXiv preprint arXiv:2202.03634* (2022).

[6] Zuxin Liu, Baiming Chen, Hongyi Zhou, Guru Koushik, Martial Hebert, and Ding Zhao. 2020. Mapper: Multi-agent path planning with evolutionary reinforcement learning in mixed dynamic environments. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 11748–11754.

[7] 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), 529–533.

[8] Robert Morris, Corina S Pasareanu, Kasper Luckow, Waqar Malik, Hang Ma, TK Satish Kumar, and Sven Koenig. 2016. Planning, scheduling and monitoring for airport surface operations. In *Workshops at the Thirtieth AAAI Conference on Artificial Intelligence*.

[9] Walaa Othman and Nikolay Shilov. 2021. Deep Reinforcement Learning for Path Planning by Cooperative Robots: Existing Approaches and Challenges. In *2021 28th Conference of Open Innovations Association (FRUCT)*. IEEE, 349–357.

[10] Guillaume Sartoretti, Justin Kerr, Yunfei Shi, Glenn Wagner, TK Satish Kumar, Sven Koenig, and Howie Choset. 2019. Primal: Pathfinding via reinforcement and imitation multi-agent learning. *IEEE Robotics and Automation Letters* 4, 3 (2019), 2378–2385.

[11] Guni Sharon, Roni Stern, Ariel Felner, and Nathan R Sturtevant. 2015. Conflict-based search for optimal multi-agent pathfinding. *Artificial Intelligence* 219 (2015), 40–66.

[12] David Silver. 2005. Cooperative pathfinding. In *Proceedings of the aaai conference on artificial intelligence and interactive digital entertainment*, Vol. 1. 117–122.

[13] Roni Stern, Nathan Sturtevant, Ariel Felner, Sven Koenig, Hang Ma, Thayne Walker, Jiaoyang Li, Dor Atzmon, Liron Cohen, TK Kumar, et al. 2019. Multi-agent pathfinding: Definitions, variants, and benchmarks. In *Proceedings of the International Symposium on Combinatorial Search*, Vol. 10. 151–158.

[14] Peter Sunehag, Guy Lever, Audrunas Gruslys, Wojciech Marian Czarnecki, Vinicius Zambaldi, Max Jaderberg, Marc Lanctot, Nicolas Sonnerat, Joel Z Leibo, Karl Tuyls, et al. 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*. 2085–2087.

[15] Manuela Veloso, Joydeep Biswas, Brian Coltin, and Stephanie Rosenthal. 2015. Cobots: Robust symbiotic autonomous mobile service robots. In *Twenty-Fourth International Joint Conference on Artificial Intelligence*.

[16] Kaifang Wan, Dingwei Wu, Bo Li, Xiaoguang Gao, Zijian Hu, and Daqing Chen. 2022. ME-MADDPG: An efficient learning-based motion planning method for multiple agents in complex environments. *International Journal of Intelligent Systems* 37, 3 (2022), 2393–2427.