Boosting Studies of Multi-Agent Reinforcement Learning on Google Research Football Environment: the Past, Present, and Future

Yan Song[*]
Institute of Automation, Chinese Academy of Science
Beijing, China
yan.song@ia.ac.cn

He Jiang[*]
Digital Brain Lab
Shanghai, China
srevir@foxmail.com

Haifeng Zhang[†]
Institute of Automation, CAS
School of Artificial Intelligence, UCAS
Nanjing Artificial Intelligence Research of IA
Beijing, China
haifeng.zhang@ia.ac.cn

Zheng Tian
Shanghai Tech University
Shanghai, China
tianzheng@shanghaitech.edu.cn

Weinan Zhang
Shanghai Jiao Tong University
Shanghai, China
wnzhang@sjtu.edu.cn

Jun Wang
University College London
London, UK
jun.wang@cs.ucl.ac.uk

ABSTRACT

Even though Google Research Football (GRF) was initially benchmarked and studied as a single-agent environment in its original paper [19], recent years have witnessed an increasing focus on its multi-agent nature by researchers utilizing it as a testbed for Multi-Agent Reinforcement Learning (MARL), especially in the cooperative scenarios. However, the absence of standardized environment settings and unified evaluation metrics for multi-agent scenarios hampers the consistent understanding of various studies. Furthermore, the challenging *5 vs 5* and *11 vs 11* full-game scenarios have received limited thorough examination due to their substantial training complexities. To address these gaps, this paper extends the original environment by not only standardizing the environment settings and benchmarking cooperative learning algorithms across different scenarios, including the most challenging full-game scenarios, but also by discussing approaches to enhance football AI from diverse perspectives and introducing related research tools for learning beyond multi-agent cooperation. Specifically, we provide a distributed and asynchronous population-based self-play framework with diverse pre-trained policies for faster training, two football-specific analytical tools for deeper investigation, and an online leaderboard for broader evaluation. The overall expectation of this work is to advance the study of Multi-Agent Reinforcement Learning both on and with Google Research Football environment, with the ultimate goal of deploying these technologies to real-world applications, such as sports analysis.

KEYWORDS

Multi-agent Reinforcement Learning; Agent Coordination

ACM Reference Format:
Yan Song, He Jiang, Haifeng Zhang, Zheng Tian, Weinan Zhang, and Jun Wang. 2024. Boosting Studies of Multi-Agent Reinforcement Learning on Google Research Football Environment: the Past, Present, and Future. In *Proc. of the 23rd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2024), Auckland, New Zealand, May 6 – 10, 2024*, IFAAMAS, 22 pages.

1 INTRODUCTION

Soccer is universally enjoyed, and so are soccer games. They have been proven to be valuable for exploring multi-agent reinforcement learning through research conducted in various environments, including *Markov Soccer Game*[24], the *RoboCup Soccer Simulator* [17], *Google Research Football (GRF)* [19], *rSoccer* [29], *DeepMind MuJoCo Multi-Agent Soccer Environment* [26]. Among them, GRF stands out due to its ability to emulate realistic scenarios (Figure 1) like FIFA and Real Football with not only cooperation between teammates but also competition against opponent teams. As for the interface, it allows algorithms to control all players on the field with high-level actions rather than low-level dynamics. Consequently, GRF presents an appealing platform for studying both cooperative and competitive multi-agent reinforcement learning at a strategic level. The abundant real-world and virtual-game match data also offer a great opportunity to study how to learn from multi-agent demonstrations. Meanwhile, modern sports have an emerging need for smarter analyses, which suggests a natural application for related studies. Indeed, similar strategic reasoning with multi-agent systems is ubiquitous. Therefore, we believe that the GRF environment could be yet another catalyst to accelerate the study and deployment of MARL.

However, even though the GRF simulator provides support for both single-agent and multi-agent settings, its original paper merely benchmarks single-agent scenarios [19]. In this setting, only one player is controlled at a time, and the player to be controlled is determined by an underlying heuristic. The highlight moment for

[*] Equal Contribution
[†] Corresponding Author

Proc. of the 23rd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2024), N. Alechina, V. Dignum, M. Dastani, J.S. Sichman (eds.), May 6 – 10, 2024, Auckland, New Zealand. © 2024 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org).

(a) 3 vs 1 (b) counterattack (c) 11 vs 11

Figure 1: Snapshots of different Google Research Football scenarios

the single-agent setting studies was the 2020 Kaggle competition, *Google Research Football with Manchester City F.C.* [3]. This competition attracted more than 1,000 teams to compete online. Top-performing teams such as *Wekick*, *Saltyfish* and *liveinparis* made a substantial impact on subsequent studies of multi-agent scenarios through their ideas, code, and datasets.

In recent years, more researchers have turned their attention towards the multi-agent nature of GRF, considering it a testbed for their MARL algorithms [22, 54, 56, 59]. Inspired by the Kaggle Competition, *IEEE Conference on Games Football AI Competition* [32] was held in 2022, which focused on multi-agent scenarios. Such growing popularity of GRF may also be attributed to the extensive exploration and saturation of many other multi-agent environments, which are no longer suitable for further academic research [6]. For example, the well-known MARL benchmark, StarCraft Multi-Agent Challenge (SMAC), has been extensively investigated and reported to achieve near-optimal performance in numerous studies [35, 54, 56]. Some criticism has also been directed towards SMAC due to its lack of stochasticity [6]. On the contrary, GRF has strong stochasticity, as discussed in its original work [19]. Consequently, the MARL community has expressed a need for more challenging testbeds, and Google Research Football emerges as a desirable choice.

However, most previous studies evaluate their algorithms on various sets of simplified cooperative soccer tasks known as academy scenarios with only 1 to 5 agents [8, 10, 22, 31, 37, 40, 41, 56, 60]. Some researchers even develop customized simple tasks for evaluation purposes [22, 25, 53]. Due to the considerable variations in scenario settings and evaluation metrics employed by different studies, it becomes challenging to comprehensively understand and compare performance across works.

In addition, the more challenging full-game scenarios with each team of either 5 or 11 players are less touched because of their difficulties originating from more players, larger spaces, and longer horizons. only very recently, a few studies have ventured into tackling these much more difficult cooperative settings [13, 23, 54]. Yet, they rely on either complex training or sophisticated reward designs with no one presenting a simple and succinct enough solution for easy follow-up or comparison.

Therefore, this paper aims to standardize the scenario settings and provide benchmark results for cooperative scenarios with representative algorithms but without any advanced techniques or tricks. Furthermore, as pointed out in [46], over-fitting to a fixed opponent does not give a generally strong football AI. Therefore, to step towards future studies of cooperative and competitive MARL, we introduce several ways of building strong Football AI and related research tools. We also discuss the limitations of current studies with corresponding potential research directions.

In summary, our major contributions can be outlined as follows:

(1) Standard Settings: We establish standardized settings for both academy and full-game scenarios in the multi-agent context of Google Research Football, following some best practices.
(2) Benchmark and Analysis: We conduct an empirical comparison of representative MARL algorithms on the standardized tasks, accompanied by a comprehensive analysis of various algorithmic designs.
(3) Research Tools: To our knowledge, we are the first to release a comprehensive set of research tools for GRF studies[1]. These tools encompass an efficient distributed population-based self-play training framework, diverse pre-trained models, two football analytical tools, and an online leaderboard.

Overall, we expect to boost studies of MARL on and with Google Research Football environment. Hopefully, one day, related research could go beyond virtual games and bring benefits to real-world sports analysis and strategy reasoning.

The overall structure of our paper is as follows:

(1) The Past: Early works on GRF mainly study the single-agent setting and the later research on the multi-agent scenarios uses inconsistent environmental settings. In Section 1 and 2, we discuss previous studies on GRF and argue the need for a unified evaluation.
(2) The Present: More recent studies have started to use GRF as a testbed for cooperative MARL but a benchmark is still lacking. In Section 3 and 4, we provide a fully reproducible MARL benchmark for standardized cooperative tasks on GRF with detailed experiments and analysis.
(3) The Future: We aim to go beyond cooperative tasks to competitive tasks and learn sophisticated human-like strategies. In Section 5 and 6, we provide related tools for future studies and discuss limitations with potential research directions.

2 RELATED WORK

The primary work related to our research is the original GRF paper [19], which introduces the environment but only studies single-agent scenarios, where only a single active player of a team is controlled. Some works focus on solving competitive tasks in this single-agent setting. For example, Liu et al. [27] beat the strongest built-in AI with a novel PSRO [20] algorithm that combines both behavior diversity and response diversity. Though such a competitive task in a single-agent scenario is also multi-agent by definition, our work mainly focuses on multi-agent scenarios, where multiple players of a single team are controlled by agents, which renders the cooperation between teammates important. Our research is motivated by three key factors.

Firstly, we recognize the necessity of benchmarking multi-agent cooperative learning due to its growing popularity, coupled with the absence of standardized benchmarks. Several recently proposed multi-agent reinforcement learning algorithms, such as Multi-Agent PPO (MAPPO)[59], Multi-Agent Transformer (MAT) [56] and A2PO

[1] All our code, including experiment settings and research tools, are available at https://github.com/jidiai/GRF_MARL.git

[54], all choose to validate their superior performance on GRF multi-agent tasks. The works on MARL communications [31, 41] and behavioral diversity [22, 53] also utilize GRF multi-agent tasks as a testbed. However, these studies often adopt different experimental settings, including varying sets of scenarios, environment configurations, and evaluation metrics, making it challenging to compare and interpret results consistently. Our work addresses these challenges by first standardizing the test suite and subsequently benchmarking representative algorithms on it.

Furthermore, we find the need for more attention to the full-game scenarios. Currently, only a few studies have explored methods in these challenging settings. TiKick [13] claims to be the first work tackling the *11 vs 11* scenario, but it allows agents to use a default built-in action, which delegates the control of players to the underlying heuristic. Recently, TiZero [23] and Fictitious Cross-Play (FXP) [57] both beat the strongest built-in AI in the *11 vs 11* setting by respectively learning from a curriculum of self-play and setting a counter-policy population. Meanwhile, Wang et al. [54] also outperformed the toughest built-in AI in full games learning training from scratch but with sophisticated reward designs and asynchronous training. In addition, these papers focus mainly on their new algorithms, providing limited analysis of GRF. In contrast, our work extensively benchmarks and analyzes full-game scenarios using only the official SCORING and CHECKPOINT rewards provided by the game, without employing any additional tricks.

Lastly, we realize the importance of sharing research tools to facilitate further studies in cooperative and competitive learning on GRF. Most related studies, such as [8, 22, 31, 41, 59], only release codes limited to simple academy scenarios. Other papers like [13, 23] have only released evaluation scripts with trained models. In contrast, we provide a comprehensive training, evaluation, and analysis toolkit.

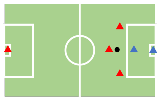
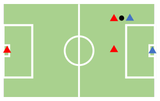
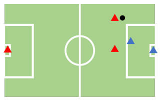
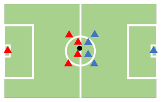
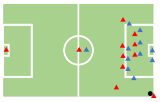
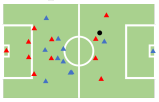
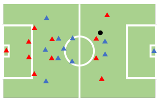
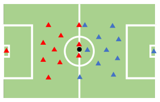
3 SCENARIOS

We carefully select scenarios that are suitable for multi-agent cooperation from the original set of GRF scenarios to form our benchmark suite (Table 1). We strive to make minimal modifications to these scenarios while adhering to best practices. The selected are those commonly used in previous MARL research work to examine the performance of their proposed algorithm [13, 19, 27]. Meanwhile, they also cover different levels of cooperation. For example, *3 vs. 1 with keeper*, *pass and shoot with keeper*, and *run pass and shoot with keeper* are three small-scale tasks focusing on short-horizon offensive strategy learning and only involve a small group (3-4) of players on the pitch. Whereas scenarios like *corner*, *counterattack easy*, and *counterattack hard*, require controlling all players in a team, though only a fraction of them may play a critical role. On the other hand, full-game settings, such as *5 vs. 5* and *11 vs. 11*, emphasize more on long-horizon planning, and agents are required to consider both offense and defense strategies. Academy scenarios involving a single player are excluded. Additionally, the guidelines governing our modification to these tasks can be summarized as follows:

(1) To save computational resources, we choose one difficulty level for each scenario, as it mainly affects the reaction time of the built-in AI rather than its strategy [19].
(2) Given the distinct action set of the goalkeeper (GK) compared to other non-GK players, we delegate GK control to the built-in AI in all scenarios. We utilize the default action set with 19 actions for all other players. We do not recommend using alternative official action sets, such as action set v2 and the full action set adopted in the work [13] and [8] respectively. These alternative sets include a built-in default action that delegates action selection to the underlying heuristic, which contradicts the purpose of using reinforcement learning to learn complex controls.
(3) In academy scenarios primarily designed for studying attacks, we end the episode immediately upon possession exchange. This approach alleviates the need for algorithms to consider defensive strategies. For full-game scenarios, we ensure fairness by forcing the two teams to exchange sides at halftime.

During the evaluation phase, we use winning rate as the main criterion, supplemented by football-specific statistics as auxiliary metrics. The winning rate, being the essential objective of any game, serves as a well-normalized and commonly utilized measure, consistent with prior research findings. Moreover, football-specific metrics such as ball possession and passing offer preliminary insights into team dynamics, facilitating more profound behavioral analysis (as demonstrated in Section 5.1.3).

Table 1: Scenario name and line-up illustration for each benchmark scenario (red triangle: left-team player; blue triangle: right-team player; ●: ball).

			
3 vs 1 with keeper (3v1)	*pass and shoot* (pass)	*run pass and shoot* (run)	*5 vs 5*
			
corner	*counterattack-easy* (ct-e)	*counterattack-hard* (ct-h)	*11 vs 11*

4 MULTI-AGENT COOPERATIVE LEARNING BENCHMARK

4.1 Problem Formulation

The multi-agent Google Research Football scenarios can be formulated as a Markov game G with N agents. The game is defined by a set of elements $\langle N, \mathcal{S}, \{\mathcal{A}^i\}_{i\in\{1,\dots,N\}}, P, \{R^i\}_{i\in\{1,\dots,N\}}, \gamma\rangle$. $\mathcal{S}$ is the set of game states shared by all agents, $\mathcal{A}^i$ is the set of actions of agent i and we denote $\mathbb{A} := \mathcal{A}^1 \times \dots \times \mathcal{A}^N$ as the joint action. $P : \mathcal{S} \times \mathbb{A} \rightarrow \mathcal{S}$ is the transition probability function. $R^i : \mathcal{S} \times \mathbb{A} \times \mathcal{S} \rightarrow \mathbb{R}$ is the reward function for agent i. $\gamma \in [0, 1]$ is the discount factor that represents the decaying rate. The game also can be viewed from two perspectives:

- Single-agent setting: a naive solution to game G is to transform the problem into a Markov Decision Process (MDP) defined by a tuple of elements $\langle \mathbb{S}, \mathbb{A}, P, R, \gamma \rangle$. $\mathbb{S} = \{\mathcal{S}\}$, $\mathbb{A} = \{\mathcal{A}^1 \times ... \times \mathcal{A}^N\}$ are joint state and action of N agents and can be seen as the state and action of a single integrated agent. The goal of the integrated agent is to solve the MDP, that is to find the optimal joint policy function $\boldsymbol{\pi} : \mathbb{S} \rightarrow \mathbb{A}$ such that the discounted cumulative reward is maximized:

$$\mathbb{E}_{s_{t+1}\sim P(\cdot|s_t,\boldsymbol{a}_t);\boldsymbol{a}_t\sim\boldsymbol{\pi}(\cdot|s_t)}\left(\sum_{t\geq 0}\gamma^t R(s_t,\boldsymbol{a}_t,s_{t+1})\middle|s_0\right)$$

$$s_t, s_{t+1} \in \mathbb{S}, \boldsymbol{a}_t \in \mathbb{A}$$

- Fully cooperative setting: this setting can be regarded as a multi-player extension to the MDP where agents are assumed to be homogeneous and interchangeable. Agents also share the same reward function: $R = R^i = R^1 = ... = R^N$. The game proceeds as follows: at each time step t, the environment has a state s_t, each agent executes its action a_t^i simultaneously with all other agents, giving the joint action $A_t = a_t^1, ..., a_t^N$. The environment transit to the next state $s_{t+1} \sim P(\cdot|s_t, A_t)$. Then, the environment determines an immediate reward $R^i(s_t, A_t, s_{t+1})$ for each agent. The goal of each agent i is to solve the game by finding an individual optimal policy $\pi^i \in \Pi^i : \mathcal{S} \rightarrow \mathcal{A}^i$ such that the discounted cumulative reward is maximised:

$$\mathbb{E}_{s_{t+1}\sim P(\cdot|s_t,a_t^{1:N});a_t^i\sim\pi^i(\cdot|s_t)}\left(\sum_{t\geq 0}\gamma^t R_t^i(s_t,a_t^{1:N},s_{t+1})\middle|s_0\right)$$

$$s_t, s_{t+1} \in \mathbb{S}, a_t^i \in \mathcal{A}^i, i = 1, ..., N$$

In this setting, the optimal policy of each agent is influenced by not only its own policy but also the policies of the other agents in the game. This is one of the fundamental differences between single-agent RL and multi-agent RL [58].

4.2 Algorithms

We benchmark a variety of representative MARL algorithms across our chosen scenarios, comprising both classic and cutting-edge approaches within their respective categories. Our categorization and selection process align with established MARL benchmarks [34, 35], with further details provided in Table 2 and Appendix B.4.1. Since the GRF environment requires discrete action controls, hence algorithms tailored for continuous-action space such as MADDPG [28] and FACMAC [36] are not part of our benchmark.

4.3 Experiment Settings

4.3.1 Feature Engineering. We compare two feature encoders in our experiments: the *Simple* and the *Complex*. The *Simple* features refer to the *simple-115* features provided by the original GRF paper [19], which encodes the location and motion information of all players and the ball. To enrich the feature representation, we also design *Complex* features which include additional information such as the relative position, the closest teammate, and the closest opponent. The detailed design can be found in Table 5.

4.3.2 Reward Shaping. We study two reward functions in our experiments: the *Sparse* and the *Dense*, which are based on the official SCORING and CHECKPOINT rewards introduced in the original GRF paper [19]. Both of them are simple and have been used in previous works. The SCORING rewards with +1 when we score and penalizes with −1 when we lose a score, while the CHECKPOINT gives positive feedback whenever our player moves the ball to a checkpoint that is closer to the opponent's goal. SCORING reward can be hard to obtain but CHECKPOINTS reward is easily attainable. Our *Sparse* reward refers to using only the SCORING and *Dense* reward refers to the sum of both SCORING and CHECKPOINT.

Table 2: Overview of algorithms selected for our benchmark. Agent Update Scheme [54] refers to simultaneous and sequential update of agents within a single optimization step. Cen and Dec refer to centralized and decentralized operation modes respectively.

Algorithm	Value/Policy Based	Update Scheme	Training Mode	Execution Mode
QMIX [39]	Value	Simultaneous	Cen	Dec
QPLEX [52]	Value	Simultaneous	Cen	Dec
IPPO [5]	Policy	Simultaneous	Dec	Dec
MAPPO [59]	Policy	Simultaneous	Cen	Dec
HAPPO [18]	Policy	Sequential	Cen	Dec
A2PO [54]	Policy	Sequential	Cen	Dec
MAT [56]	Policy	Simultaneous	Cen	Cen

4.3.3 Parameter Sharing. Parameter sharing is also considered in our experiments. With parameter-sharing, all agents share a single copy of network parameters. Without parameter sharing, each agent needs to maintain its own network parameters. Parameter sharing has been shown to provide more efficient learning [35] but may cause high resemblance between behaviors of individual agents [22]. To alleviate such an issue, we also include in the *Complex* features a one-hot vector demonstrating the agent's identity.

For each scenario, we simulate the same number of environment steps and compare the win rates of different algorithms. Additional experiment settings can be found in Appendix B.

4.4 Academy Scenarios

We first benchmark performance on relatively simpler academy scenarios and study different settings, including reward shaping, feature engineering, and parameter-sharing. Then we tackle the full-game scenarios in section 4.5 following experience drawn from the study of academy scenarios.

4.4.1 Policy-Based vs Value-Based Algorithms. The final performance of all algorithms on academy scenarios is presented in Table 3 and the training curves are illustrated in Figure 2. In these scenarios, policy-based methods tend to exhibit overall better performance compared to value-based methods, particularly in *run & pass*, *corner*, and *counterattack*. This discrepancy might be attributed to the curse of dimensionality in state and action spaces, due to the combinatorial nature of multi-agent systems which has been shown to be challenging for value-based algorithms [58].

In Figure 2, we also find that different policy-based methods could achieve similar performance in small-scale scenarios. For example, in *3 versus 1 with keeper* and *pass and shoot with keeper*, all policy-based algorithms achieve the 85% test win rate within 10M environment steps. In *run pass and shoot with keeper* scenario, the performance becomes slightly worse possibly due to higher defensive pressure exerted by the closer opponent (see the line-up in Table 1). In scenarios with a larger number of players (*corner*, *counterattack-easy & hard*), most policy-based algorithms still manage to achieve a win rate of over 75% in counterattack tasks within 13M environment steps but struggle to beat the built-in AI in corner tasks. By looking at the line-up in Table 1, we canfi nd that the counterattack tasks allocate only four players to the front-court for participation in offensive maneuvers, with the remaining players situated in the back-court, distanced from the ball. This configuration simplifies the task complexity, as the MARL learning algorithm can center its attention on the offensive players. Whereas in corner tasks, all players are grouped together, requiring the policy to coordinate their actions, which could pose a relatively greater challenge in learning. Overall, the Multi-Agent Transformer (MAT) performs the best in complex scenarios with a larger number of players. This is probably due to the stronger ability of transformers to learn contextual relationships between input data.

Table 3: Final performances of all algorithms on academy scenarios. Thefi nal performance is recorded as the maximum mean win rate(standard deviation) scaled by 100. Thefirst column lists all the scenarios with their abbreviations as described in Table 1.

SCEN	IPPO	MAPPO	HAPPO	A2PO	MAT	QMIX	QPLEX
pass	93.7(0.9)	92.9(2.6)	94.0(3.6)	93.2(1.2)	**96.6**(0.8)	95.6(5.8)	88.1(8.2)
run	73.7(13)	66.0(6.3)	70.4(7.2)	79.9(6.0)	**81.1**(5.7)	58.1(24)	68.8(14)
3v1	**91.7**(3.1)	90.0(3.2)	91.4(3.9)	87.6(1.4)	88.5(2.0)	86.9(8.2)	81.9(6.7)
corner	50.4(10)	50.5(7.2)	47.9(9.9)	59.7(6.2)	**71.0**(8.1)	20.0(19)	28.8(17)
ct-e	85.7(6.6)	88.9(6.6)	78.0(14.8)	80.9(7.3)	**89.3**(5.8)	57.5(19)	43.3(27)
ct-h	71.6(4.9)	81.3(9.6)	75.2(11)	80.7(4.2)	**87.0**(6.0)	56.3(18)	33.8(26)
5v5	99.1(0.6)	96.0(1.8)	98.0(1.3)	95.0(2.9)	**99.3**(1.2)	0.0(0.0)	0.0(0.0)
11v11	52.7(2.4)	45.4(2.7)	52.1(4.8)	50.1(3.6)	**59.7**(3.6)	0.0(0.0)	0.0(0.0)

4.4.2 An Analysis on Different Algorithm Settings: Rewards, Feature Encoders, and Parameter Sharing. To confirm the efficacy of our approaches in tackling more demanding full-game scenarios, we conducted additional ablation studies focusing on academy scenarios, the results of which are depicted in Figure 3. Notably, the presented results are averaged across all policy-based algorithms, excluding the two value-based algorithms due to their current inability to effectively handle challenging scenarios, as validated in Section 4.4.1. Further details on individual algorithm results and their consistency analysis are provided in the Appendix B.5.

Across all six academy scenarios, dense reward (both SCORING and CHECKPOINTS) shows better and more stable performance than sparse reward (only SCORING). A possible reason might be that the additional CHECKPOINTS reward, which effectively guides the player toward the opponent's goal, substantially reduces exploratory actions. In terms of selecting appropriate feature encoders, complex features offer more efficient learning than simple ones. This has demonstrated the importance of domain knowledge integration when solving a complex task. We also observe from Figure 3 that training without parameter-sharing gives slightly better results in some scenarios but worse performance on counterattack tasks. However, the time consumption of non-parameter-sharing is considerably higher, making it less suitable for large-scale training.

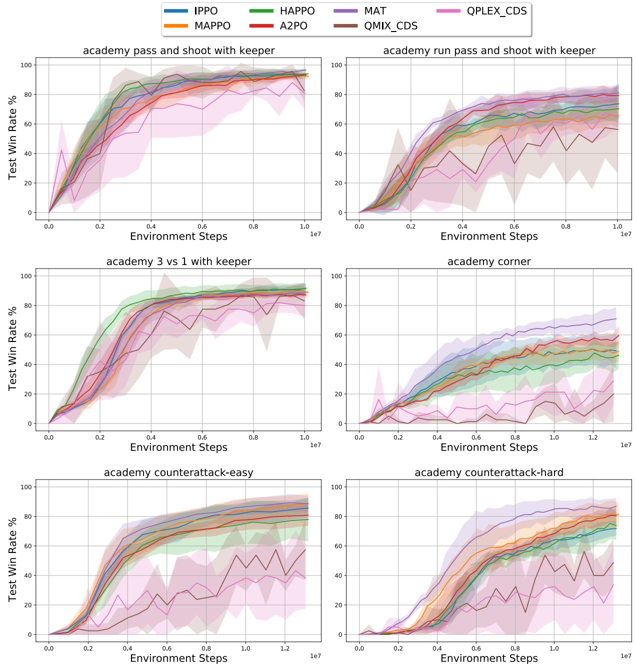
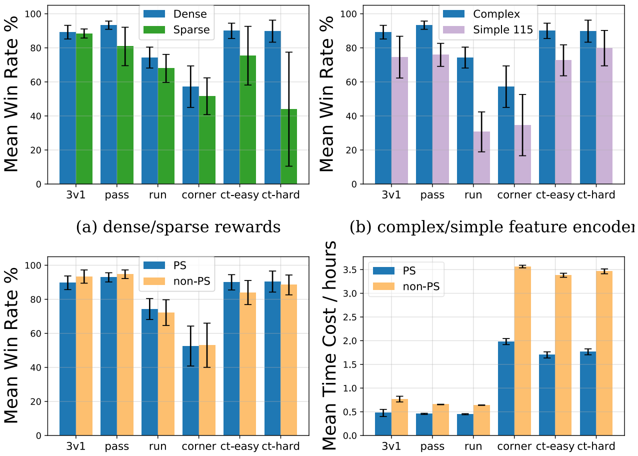


Figure 2: Average win rates on six academy scenarios: *3 vs 1 with keeper, pass and shot with keeper, run pass and shoot with keeper, corner, counterattack easy* and *counterattack hard.* Results are averaged overfi ve random seeds and the shaded area represents the standard deviation of the testing win rate.

4.5 Full-Game Scenarios

We leverage the insights gained from the benchmark in academy scenarios to address the challenges posed by full-game scenarios. Our approach involves utilizing dense rewards, the default feature encoder, and parameter-sharing. The corresponding performances are documented in Table 3, while the sample efficiency curves are illustrated in Figure 4. In particular, we observe that value-based methods fail to learn meaningful behaviors within a reasonable number of environmental steps. This observation aligns with our previousfi ndings on academy scenarios, empirically indicating that these value-based methods encounter difficulties in complex multi-agent environments without specialized treatment. Various policy-based algorithms still exhibit similar performance after careful hyper-parameter tuning, with MAT slightly outperforming the others, potentially attributed to its fully centralized advantage. Although the *5 vs 5* scenario has been largely addressed, the *11 vs 11* scenario remains challenging. Aside from the win rate, given the substantial training time (12 hours for an experiment on the *11 vs 11* scenario with a 128 CPU and 2 A100 GPU server), it becomes crucial to develop algorithms with improved sample efficiency, as



(a) dense/sparse rewards

(b) complex/simple feature encoder

(c) with/without Parameter-Sharing (win rate)

(d) with/without Parameter-Sharing (time)

Figure 3: Comparisons under different experiment settings. (a), (b) & (c): win rate in each scenario averaged over all policy-based algorithms with dense/sparse rewards; with complex/simple feature encoders, and with/without parameter-sharing. (d): Time cost in each scenario averaged over all policy-based algorithms with/without parameter-sharing. Error bars show the standard deviation. 3v1: *3 vs 1 with keeper*; pass: *pass and shoot with keeper*; run: *run pass and shoot with keeper*; ct-easy: *counterattack-easy*; ct-hard: *counterattack-hard.*

we need to repeatedly compute the best response to the current opponent mixture in self-play.

Additionally, we delve into the behaviors learned by these algorithms in full-game scenarios and discover that all algorithms learn similar strategies. In both 5-vs-5 and 11-vs-11 tasks, these policies effectively exploit the weaknesses of the built-in AI and display sophisticated dribbling and shooting skills. During attacks, teammates usually serve as distractions for opponent defenders, while a single key player exploits the weak point of the defense and performs the shot. In defense, all our defenders frequently coordinate their movement forward to force opponent attackers into an offside position. A visualization of these behaviors on the *11 vs 11* scenario is given in Figure 5. **Despite beating the strongest built-in AI, this strategy is far from being robust.** Simply training policies against fixed opponents only leads to overfitting to a specific playing style, which limits the policies' adaptability and versatility. This problem inspires us to explore the methods in the following section of building stronger football AI.

5 BUILDING STRONGER FOOTBALL AI

In Section 4.5, it is evident that the policy trained solely on the benchmark scenarios lacks sufficient intelligence and robustness. In the following, we will focus mainly on our ready-to-use training framework and online ranking system. The former could help with building a stronger AI with faster speed and less effort, while the latter could help with building a more robust one by organizing matches against other unknown strategies from the community.

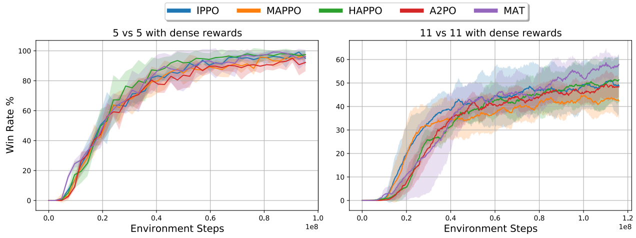
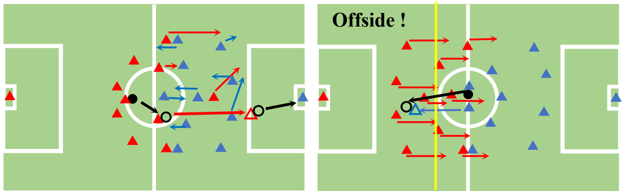


Figure 4: Performance of baseline algorithm in *5 vs 5* (Left) and *11 vs 11* (Right) full-game scenarios. Results are averaged over five and three random seeds. Each *5 vs 5* experiment (1e8 steps) takes approximately 8 hours to complete and an experiment of *11 vs 11* (1e8 steps) takes around 12 hours on a 200 CPU and 2×A100 GPU server).



(a) Offensive strategy **(b) Defensive strategy**

Figure 5: Visualisation for team strategy in 11-vs-11 full-game. (red triangle: left-team player; blue triangle: right-team player; ●: ball; →: left-team player's movement; →: right-team player's movement; →: ball's movement; ‖: off-side line.) More visualization of behavior analysis can be found in Appendix B.8.

We will also briefly introduce several accompanying tools, such as diverse pre-trained policies (Appendix C.2 and C.3), a match-decomposition data structure (Appendix C.4) for better analysis and a single-step visual debugger (Appendix C.5) for easier investigation.

5.1 Population-based Self-Play Framework, Pre-Trained Policies and Analytical Tools

One approach to achieving a stronger and less exploitable policy is to utilize more advanced self-play algorithms. To ease the efforts of other researchers, we release our distributed and asynchronous population-based training framework, which implements Policy Space Response Oracle (PSRO) [20] and League Training [15]. So far as we know, it is the first publicly available training framework that not only beats the hardest built-in AI on full-game scenarios with only one round of best-response computation but also continues to improve policies with population-based self-play. In this section, we only include key results and leave the information regarding the architecture, tutorial, and training procedure details in Appendix C.1 and C.2.

5.1.1 Asynchronous Implementation. Traditionally, most works benchmark algorithms in a synchronous mode to evaluate sample complexity. Namely, the algorithm updates the model after a certain amount of rollout steps. However, in practical large-scale environments such as those in [15, 46], data collection can be very time-consuming. This leads to two bottlenecks: rollouts within a single batch must wait for one another, and trainers must wait for the entire batch of rollouts to complete. To address these challenges, we first implement an asynchronous approach that offers improved efficiency during run-time. In the asynchronous mode, rollout and training processes run in parallel, nearly independently, with simple coordination facilitated by a producer-consumer queue. This allows for more efficient utilization of computational resources. Importantly, in the asynchronous mode, samples are allowed to be reused. This means that not only does the waiting time for data decrease, but sample efficiency also increases. By reusing samples, we minimize redundant computations and maximize the utilization of collected data. Figure 6 demonstrates the superiority of the asynchronous implementation in *11 vs 11* regarding both learning speed and sample efficiency. Such a speed-up is essential in self-play pipelines, where repeated best-response computations are required.

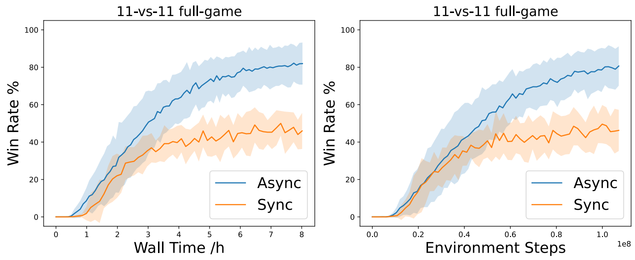
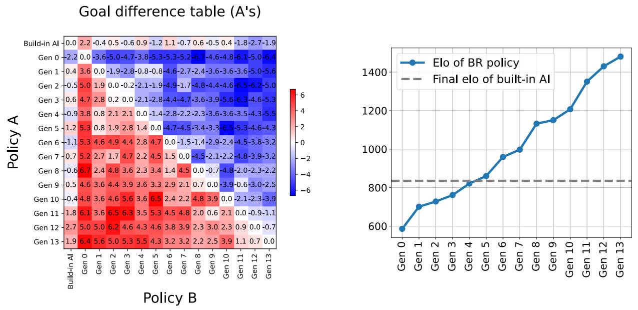


Figure 6: Average win rate with respect to time cost (left) and environment steps (right) under sync/async settings in the *11 vs 11* full-game (with standard deviation). The results are averaged over policy-based algorithms.

5.1.2 Population-based Self-Play Training. Next, we outline the application of population-based self-play training to enhance the performance of football AI, focusing on the specific example of the Policy-Space Response Oracle (PSRO) algorithm [20] (Detailed algorithm can be found in Algorithm 1). Figure 7a and Figure 7b provide visual representations of the policy evolution in a *5 vs 5* PSRO trial, showcasing the progression of payoffs and Elo scores. The experiment begins with a simple population consisting solely of the built-in AI. At each generation, a new best response policy is trained against a policy combination and added to the population. We can observe the non-transitivity of the game by analyzing the payoff tables in Figure 7a. For example, in `Gen 1`, the best-response policy beats the built-in AI, and in `Gen 2`, the new best-response policy surpasses `Gen 1` but is subsequently defeated by the built-in AI. This non-transitivity implies the complex nature of the game and the difficulty of achieving consistent superiority. However, as the training progresses, the algorithm gradually overcomes the non-transitivity and achieves dominance over every policy in the population after `Gen 11`. The resulting strategy from this PSRO experiment exhibits a similar gameplay style to one of our released pre-trained policies, `Group Defense` (Details found in Appendix C.3).

Furthermore, it's worth noting that our codebase also supports other population-based self-play pipelines, such as the *League Training* algorithm (Details can be found in Appendix C.1.6). These population-based self-play pipelines allow for experimentation with different training methodologies and the exploration of diverse AI strategies. For more details on the framework and related self-play procedure, please refer to Appendix C.1.



(a) The goal difference table on *5 vs 5* full-game

(b) The Elo scores of all policies in the final population

Figure 7: (a) The goal difference table between built-in AI and best response policies trained at each generation in a *5 vs 5* PSRO trial. "Gen" is the abbreviation for "generation". The goal difference is calculated by subtracting the score of the column policy from the score of the row policy. A positive goal difference often represents a higher possibility of winning the game than losing. The training starts with only built-in AI in the population. (b) the Elo scores of all policies in the final population. The policies become stronger as the generation increases in terms of Elo scores.

5.1.3 Diverse Pre-Trained Policies. Utilizing the population-based training framework discussed above, we have obtained policies with diverse playing styles. We simulate matches between them, count the match statistics, and evaluate their performances in various dimensions, shown as the radar plots in Figure 8. We release these pre-trained models, which can be potentially used for imitation learning, policy initialization, or offline evaluation in future research. For more details on the pre-trained models and related training procedures, please refer to Appendix C.2 and C.3.

5.1.4 Analytical Tools. Given the complexity of the GRF environment, we have also designed two analytic tools to boost training and help better understand policies beyond win rates. There have been previous attempts at automatic event detection [51] and behavior analysis software [30] in real football matches. However, to the best of our knowledge, no such attempts have been made publicly to the game of GRF and the default game replay tools are inconvenient to use for in-depth analysis. In particular, wefirst design a data structure for better match decomposition and event detection (Figure 20 in Appendix C.4). This data structure helps with computing complex statistics like assists, which is essential for

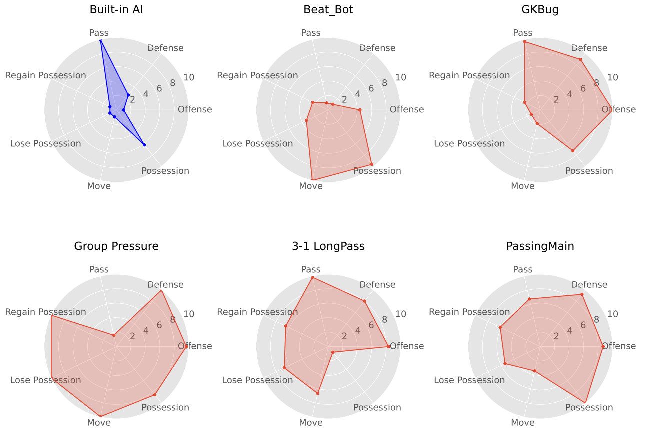


Figure 8: We use radar plots to depict diverse styles of pre-trained policies in *5 vs 5* scenarios. The performance is evaluated by simulating matches and measured in football-specific metrics with normalization.

credit assignments during training [2]. Then, we provide a visual debugger that could replay a match frame by frame, which associates the 3D view with a 2D minimap indicating both locations and speeds (Figure 21 in Appendix C.5). The current-step environmental information and statistics are also presented for convenience. More details are covered in Appendix C.4 and C.5.

5.2 Online Ranking

A publicly accessible online ranking is widely recognized as having significant benefits for related research, such as various competitions held on Kaggle [14, 47]. In light of this, we introduce the *Google Research Football Online Ranking* targeting specifically at GRF *5 vs 5* and *11 vs 11* multi-agent full-game scenarios [16] on an online evaluation platform called *JIDI* [48].

As illustrated in Figure 9, JIDI automatically simulates football matches between agents submitted by users and continuously updates the live ranking based on match results. This online ranking system allows agents to compete against unseen opponents, who are not available during training. This is crucial for studying generalization ability, which is an important aspect of algorithms emphasized by many works [1, 6, 9, 21]. Importantly, users are allowed to download the replay of these simulated matches so as to analyze the weakness of their strategies and thus, algorithms. *Tikick* [13] incorporate the online ranking as a vital component of their research.

Moreover, this ranking system serves as the official platform for the *IEEE Conference on Games Football AI Competition* in both 2022 [32] and 2023 [33]. Within the competition period, we often hold multiple competition rounds and apply the *Swiss-system tournament* evaluation mechanism [4] to obtain the Elo rate of each submission. The scores for each competition round will be accumulated by weights and ranked when the competition ends. Table 4 presents the current statistics on the number of users, agents, and matches. More details regarding the online ranking can be found in Appendix C.6.**We strongly encourage interested researchers to actively participate in this ranking system and contribute to its growth**.

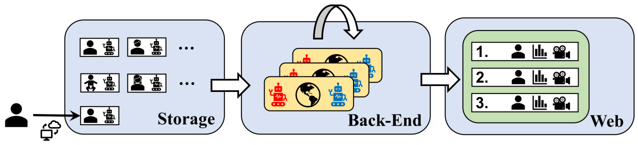


Figure 9: JIDI online Ranking System. When a user submits his customized decision-making agent to the agent storage, the back-end evaluation process executes parallel evaluation tasks distributedly on computing nodes. The evaluation results are updated to the user's score and the online ranking is updated accordingly. Users can view replays of their matches.

Table 4: The numbers of users, agents, and matches on JIDI Online Ranking System so far.

Scenarios	Number of Users	Number of Agents	Number of Matches
5 vs 5	109	372	24520
11 vs 11	119	486	23562

6 LIMITATIONS AND FUTURE DIRECTIONS

This paper provides a cooperative MARL benchmark and a set of useful research tools for future studies on the Google Research Football Environment. Nonetheless, several limitations inherent in this work reveal potential future direction:

(1) This work adheres to the basic settings in the original GRF paper [19], such as the setup of scenarios and reward functions, with few extensions. It would be interesting to study beyond these settings. For example, the existing paradigm allows players to observe nearly all the pitch. Instead, we can study the more partially observable but realistic cases by limiting players' vision.

(2) While our work establishes a benchmark for cooperative learning, an equivalent benchmark for competitive scenarios is absent. In the future, we need tofirst address the reproducibility of complex self-play training pipelines. This often involves the identification of randomness in each phase and the optimization for efficiency as these pipelines typically require a tremendous amount of computational resources.

(3) Compared to real-life football tactics, our trained policies are still immature and the connections between virtual games and real football matches are still lacking. Bridging the gap is an interesting future direction, including learning from real-world football data to make our agent akin to human[49] and evaluating real players' actions in specific situations [38].

ACKNOWLEDGMENTS

Haifeng Zhang thanks the support of the National Natural Science Foundation of China, Grant No. 62206289.

REFERENCES

[1] John P Agapiou, Alexander Sasha Vezhnevets, Edgar A Duéñez-Guzmán, Jayd Matyas, Yiran Mao, Peter Sunehag, Raphael Köster, Udari Madhushani, Kavya Kopparapu, Ramona Comanescu, et al. 2022. Melting Pot 2.0. *arXiv preprint arXiv:2211.13746* (2022).

[2] Vyacheslav Alipov, Riley Simmons-Edler, Nikita Putintsev, Pavel Kalinin, and Dmitry Vetrov. 2021. Towards practical credit assignment for deep reinforcement learning. *arXiv preprint arXiv:2106.04499* (2021).

[3] Kaggle Competition. 2020. *Google Research Football with Manchester City F.C.* https://www.kaggle.com/c/google-football

[4] László Csató. 2013. Ranking by pairwise comparisons for Swiss-system tournaments. *Central European Journal of Operations Research* 21 (2013), 783–803.

[5] Christian Schroeder de Witt, Tarun Gupta, Denys Makoviichuk, Viktor Makoviychuk, Philip HS Torr, Mingfei Sun, and Shimon Whiteson. 2020. Is independent learning all you need in the starcraft multi-agent challenge? *arXiv preprint arXiv:2011.09533* (2020).

[6] Benjamin Ellis, Skander Moalla, Mikayel Samvelyan, Mingfei Sun, Anuj Mahajan, Jakob N Foerster, and Shimon Whiteson. 2022. SMACv2: An Improved Benchmark for Cooperative Multi-Agent Reinforcement Learning. *arXiv preprint arXiv:2212.07489* (2022).

[7] Robert Nishihara Philipp Moritz Roy Fox Ken Goldberg Joseph E. Gonzalez Michael I. Jordan Ion Stoica Eric Liang, Richard Liaw. 2018. RLlib: Abstractions for Distributed Reinforcement Learning. In *Proceedings of the 35th International Conference on Machine Learning*.

[8] Wei Fu, Chao Yu, Zelai Xu, Jiaqi Yang, and Yi Wu. 2022. Revisiting Some Common Practices in Cooperative Multi-Agent Reinforcement Learning. In *Proceedings of the 39th International Conference on Machine Learning*, Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (Eds.).

[9] R. Gorsane, Omayma Mahjoub, Ruan de Kock, Roland Dubb, Siddarth S. Singh, and Arnu Pretorius. 2022. Towards a Standardised Performance Evaluation Protocol for Cooperative MARL. *ArXiv* abs/2209.10485 (2022).

[10] Jianye HAO, Xiaotian Hao, Hangyu Mao, Weixun Wang, Yaodong Yang, Dong Li, YAN ZHENG, and Zhen Wang. 2023. Boosting Multiagent Reinforcement Learning via Permutation Invariant and Permutation Equivariant Networks. In *The 11th International Conference on Learning Representations*. https://openreview.net/forum?id=OxNQXyZK-K8

[11] Johannes Heinrich, Marc Lanctot, and David Silver. 2015. Fictitious Self-Play in Extensive-Form Games. In *Proceedings of the 32nd International Conference on Machine Learning*.

[12] Jonathan Ho and Stefano Ermon. 2016. Generative Adversarial Imitation Learning. In *NIPS*.

[13] Shiyu Huang, Wenze Chen, Longfei Zhang, Shizhen Xu, Ziyang Li, Fengming Zhu, Deheng Ye, Ting Chen, and Jun Zhu. 2021. TiKick: towards playing multi-agent football full games from single-agent demonstrations. *arXiv preprint arXiv:2110.04507* (2021).

[14] Vladimir I. Iglovikov, Sergey Mushinskiy, and Vladimir Osin. 2017. Satellite Imagery Feature Detection using Deep Convolutional Neural Network: A Kaggle Competition. *ArXiv* abs/1706.06169 (2017). https://api.semanticscholar.org/CorpusID:7710301

[15] Max Jaderberg, Alexander Vezhnevets, Rémi Leblond, Tobias Pohlen, Valentin Dalibard, David Budden, Yury Sulsky, James Molloy, Tom Le Paine, Caglar Gulcehre, Ziyu Wang, Tobias Pfaff, Yuhuai Wu, Roman Ring, Dani Yogatama, Dario Wünsch, Katrina McKinney, Oliver Smith, Tom Schaul, Timothy P. Lillicrap, Koray Kavukcuoglu, Oriol Vinyals, Igor Babuschkin, Wojciech Marian Czarnecki, Michael Mathieu, Andrew Dudzik, Junyoung Chung, David H. Choi, Richard E. Powell, Timo Ewalds, Petko Georgiev, Junhyuk Oh, Dan Horgan, Manuel Kroiss, Ivo Danihelka, Aja Huang, Laurent Sifre, Trevor Cai, John P. Agapiou, Demis Hassabis, Chris Apps, and David Silver. 2019. Grandmaster level in StarCraft II using multi-agent reinforcement learning. *Nature* (2019).

[16] JIDI. 2021. *GRF Online Ranking, url=http://www.jidiai.cn/env_detail?envid=34*. http://www.jidiai.cn/env_detail?envid=34

[17] Shivaram Kalyanakrishnan, Yaxin Liu, and Peter Stone. 2006. Half Field Offense in RoboCup Soccer: A Multiagent Reinforcement Learning Case Study. In *Robot Soccer World Cup*.

[18] Jakub Grudzien Kuba, Ruiqing Chen, Muning Wen, Ying Wen, Fanglei Sun, Jun Wang, and Yaodong Yang. 2022. Trust Region Policy Optimisation in Multi-Agent Reinforcement Learning. In *International Conference on Learning Representations*. https://openreview.net/forum?id=EcGGFkNTxdJ

[19] Karol Kurach, Anton Raichuk, Piotr Stańczyk, MichałZają c, Olivier Bachem, Lasse Espeholt, Carlos Riquelme, Damien Vincent, Marcin Michalski, Olivier Bousquet, et al. 2020. Google research football: A novel reinforcement learning environment. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

[20] Marc Lanctot, Vinicius Zambaldi, Audrunas Gruslys, Angeliki Lazaridou, Karl Tuyls, Julien Perolat, David Silver, and Thore Graepel. 2017. A Unified Game-Theoretic Approach to Multiagent Reinforcement Learning. In *Advances in Neural Information Processing Systems*, I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (Eds.).

[21] Joel Z Leibo, Edgar A Dueñez-Guzman, Alexander Vezhnevets, John P Agapiou, Peter Sunehag, Raphael Koster, Jayd Matyas, Charlie Beattie, Igor Mordatch, and Thore Graepel. 2021. Scalable evaluation of multi-agent reinforcement learning with melting pot. In *International conference on machine learning*. PMLR, 6187–6199.

[22] Chenghao Li, Tonghan Wang, Chengjie Wu, Qianchuan Zhao, Jun Yang, and Chongjie Zhang. 2021. Celebrating diversity in shared multi-agent reinforcement learning. *Advances in Neural Information Processing Systems* (2021).

[23] Fanqi Lin, Shiyu Huang, Tim Pearce, Wenze Chen, and Wei-Wei Tu. 2023. TiZero: Mastering Multi-Agent Football with Curriculum Learning and Self-Play. *arXiv preprint arXiv:2302.07515* (2023).

[24] Michael L Littman. 1994. Markov games as a framework for multi-agent reinforcement learning. In *Machine learning proceedings*.

[25] Iou-Jen Liu, Zhongzheng Ren, Raymond A Yeh, and Alexander G Schwing. 2021. Semantic tracklets: An object-centric representation for visual multi-agent reinforcement learning. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*.

[26] Siqi Liu, Guy Lever, Zhe Wang, Josh Merel, S. M. Ali Eslami, Daniel Hennes, Wojciech M. Czarnecki, Yuval Tassa, Shayegan Omidshafiei, Abbas Abdolmaleki, Noah Siegel, Leonard Hasenclever, Luke Marris, Saran Tunyasuvunakool, H. Francis Song, Markus Wulfmeier, Paul Muller, Tuomas Haarnoja, Brendan D. Tracey, Karl Tuyls, Thore Graepel, and Nicolas Manfred Otto Heess. 2021. From Motor Control to Team Play in Simulated Humanoid Football. *Science Robotics* (2021).

[27] Xiangyu Liu, Hangtian Jia, Ying Wen, Yaodong Yang, Yujing Hu, Yingfeng Chen, Changjie Fan, and Zhipeng Hu. 2021. Unifying behavioral and response diversity for open-ended learning in zero-sum games. *arXiv preprint arXiv:2106.04958* (2021).

[28] Ryan Lowe, Yi I Wu, Aviv Tamar, Jean Harb, OpenAI Pieter Abbeel, and Igor Mordatch. 2017. Multi-agent actor-critic for mixed cooperative-competitive environments. *Advances in neural information processing systems* 30 (2017).

[29] Felipe Bitencourt Martins, Mateus G. Machado, Hansenclever de F. Bassani, Pedro H. M. Braga, and Edna N. S. Barros. 2021. rSoccer: A Framework for Studying Reinforcement Learning in Small and Very Small Size Robot Soccer. In *Robot Soccer World Cup*.

[30] Ben McGuckin, Johnny Bradley, Mike Hughes, Peter G. O'Donoghue, and Denise Martin. 2020. Determinants of successful possession in elite Gaelic football. *International Journal of Performance Analysis in Sport* 20 (2020), 420 – 431. https://api.semanticscholar.org/CorpusID:219002442

[31] Yaru Niu, Rohan R Paleja, and Matthew C Gombolay. 2021. Multi-Agent Graph-Attention Communication and Teaming.. In *Proceedings of the 20th International Conference on Autonomous Agents and Multiagent Systems*.

[32] IEEE Conference on Game. 2022. *Football AI Competition, url=http://www.jidiai.cn/cog_2022/*. http://www.jidiai.cn/cog_2022/

[33] IEEE Conference on Game. 2023. *Football AI Competition, url=http://www.jidiai.cn/cog_2023/*. http://www.jidiai.cn/cog_2023/

[34] Xuehai Pan, Mickel Liu, Fangwei Zhong, Yaodong Yang, Song-Chun Zhu, and Yizhou Wang. 2022. Mate: Benchmarking multi-agent reinforcement learning in distributed target coverage control. *Advances in Neural Information Processing Systems* 35 (2022), 27862–27879.

[35] Georgios Papoudakis, Filippos Christianos, Lukas Schäfer, and Stefano V. Albrecht. 2020. Benchmarking Multi-Agent Deep Reinforcement Learning Algorithms in Cooperative Tasks. In *NeurIPS Datasets and Benchmarks*.

[36] Bei Peng, Tabish Rashid, Christian Schroeder de Witt, Pierre-Alexandre Kamienny, Philip Torr, Wendelin Böhmer, and Shimon Whiteson. 2021. Facmac: Factored multi-agent centralised policy gradients. *Advances in Neural Information Processing Systems* 34 (2021), 12208–12221.

[37] Zhiqiang Pu, Huimu Wang, Boyin Liu, and Jianqiang Yi. 2022. Cognition-Driven Multi-Agent Policy Learning Framework for Promoting Cooperation. *IEEE Transactions on Games* (2022).

[38] Michael Pulis and Josef Bajada. 2022. Reinforcement Learning for Football Player Decision Making Analysis. (2022).

[39] 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*.

[40] Julien Roy, Paul Barde, Félix Harvey, Derek Nowrouzezahrai, and Chris Pal. 2020. Promoting coordination through policy regularization in multi-agent deep reinforcement learning. *Advances in Neural Information Processing Systems* (2020).

[41] Jingqing Ruan, Yali Du, Xuantang Xiong, Dengpeng Xing, Xiyun Li, Linghui Meng, Haifeng Zhang, Jun Wang, and Bo Xu. 2022. GCS: Graph-Based Coordination Strategy for Multi-Agent Reinforcement Learning. In *Proceedings of the 21st International Conference on Autonomous Agents and Multiagent Systems*.

[42] John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. 2015. High-dimensional continuous control using generalized advantage estimation. *arXiv preprint arXiv:1506.02438* (2015).

[43] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347* (2017).

[44] David Silver, Aja Huang, Christopher J. Maddison, Arthur Guez, Laurent Sifre, George van den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, Sander Dieleman, Dominik Grewe, John Nham, Nal Kalchbrenner, Ilya Sutskever, Timothy Lillicrap, Madeleine Leach, Koray Kavukcuoglu, Thore Graepel, and Demis Hassabis. 2016. Mastering the game of Go with deep neural networks and tree search. *Nature* (2016).

[45] Kyunghwan Son, Daewoo Kim, Wan Ju Kang, David Earl Hostallero, and Yung Yi. 2019. QTRAN: Learning to Factorize with Transformation for Cooperative Multi-Agent Reinforcement Learning. In *Proceedings of the 36th International Conference on Machine Learning*.

[46] Yan Song, He Jiang, Zheng Tian, Haifeng Zhang, Yingping Zhang, Jiangcheng Zhu, Zonghong Dai, Weinan Zhang, and Jun Wang. 2023. An Empirical Study on Google Research Football Multi-agent Scenarios. *arXiv preprint arXiv:2305.09458* (2023).

[47] Souhaib Ben Taieb and Rob J Hyndman. 2014. A gradient boosting approach to the Kaggle load forecasting competition. *International Journal of Forecasting* 30 (2014), 382–394. https://api.semanticscholar.org/CorpusID:154496412

[48] JIDI team. 2021. *JIDI Open-Source Evaluation Platform.* http://www.jidiai.cn/homepage

[49] Karl Tuyls, Shayegan Omidshafiei, Paul Muller, Zhe Wang, Jerome Connor, Daniel Hennes, Ian Graham, William Spearman, Tim Waskett, Dafydd Steel, et al. 2021. Game Plan: What AI can do for Football, and What Football can do for AI. *Journal of Artificial Intelligence Research* 71 (2021), 41–88.

[50] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems*, I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (Eds.).

[51] Ferran Vidal-Codina, Nicolas Evans, Bahaeddine El Fakir, and Johsan Billingham. 2022. Automatic event detection in football using tracking data. *Sports Engineering* 25 (2022), 1–15. https://api.semanticscholar.org/CorpusID:246473152

[52] Jianhao Wang, Zhizhou Ren, Terry Liu, Yang Yu, and Chongjie Zhang. 2021. QPLEX: Duplex Dueling Multi-Agent Q-Learning. In *International Conference on Learning Representations*.

[53] Li Wang, Yupeng Zhang, Yujing Hu, Weixun Wang, Chongjie Zhang, Yang Gao, Jianye Hao, Tangjie Lv, and Changjie Fan. 2022. Individual Reward Assisted Multi-Agent Reinforcement Learning. In *Proceedings of the 39th International Conference on Machine Learning*.

[54] Xihuai Wang, Zheng Tian, Ziyu Wan, Ying Wen, Jun Wang, and Weinan Zhang. 2023. Order Matters: Agent-by-agent Policy Optimization. In *The Eleventh International Conference on Learning Representations*.

[55] Ziyu Wang, Tom Schaul, Matteo Hessel, Hado Hasselt, Marc Lanctot, and Nando Freitas. 2016. Dueling Network Architectures for Deep Reinforcement Learning. In *Proceedings of The 33rd International Conference on Machine Learning*.

[56] Muning Wen, Jakub Kuba, Runji Lin, Weinan Zhang, Ying Wen, Jun Wang, and Yaodong Yang. 2022. Multi-agent reinforcement learning is a sequence modeling problem. *Advances in Neural Information Processing Systems* (2022).

[57] Zelai Xu, Yancheng Liang, Chao Yu, Yu Wang, and Yi Wu. 2023. Fictitious Cross-Play: Learning Global Nash Equilibrium in Mixed Cooperative-Competitive Games. In *Proceedings of the 2023 International Conference on Autonomous Agents and Multiagent Systems*. 1053–1061.

[58] Yaodong Yang and Jun Wang. 2020. An overview of multi-agent reinforcement learning from game theoretical perspective. *arXiv preprint arXiv:2011.00583* (2020).

[59] Chao Yu, Akash Velu, Eugene Vinitsky, Jiaxuan Gao, Yu Wang, Alexandre Bayen, and Yi Wu. 2022. The surprising effectiveness of ppo in cooperative multi-agent games. *Advances in Neural Information Processing Systems* 35 (2022), 24611–24624.

[60] Bin Zhang, Hangyu Mao, Lijuan Li, Zhiwei Xu, Dapeng Li, Rui Zhao, and Guoliang Fan. 2023. Stackelberg Decision Transformer for Asynchronous Action Coordination in Multi-Agent Systems. *arXiv preprint arXiv:2305.07856* (2023).

[61] Ming Zhou, Ziyu Wan, Hanjing Wang, Muning Wen, Runzhe Wu, Ying Wen, Yaodong Yang, Yong Yu, Jun Wang, and Weinan Zhang. 2023. MALib: A Parallel Framework for Population-based Multi-agent Reinforcement Learning. *Journal of Machine Learning Research* 24, 150 (2023), 1–12. http://jmlr.org/papers/v24/22-0169.html