Supplementary for Multi-Agent Graph-Attention Communication and Teaming

Yaru Niu*
Georgia Institute of Technology
Atlanta, Georgia
yaruniu@gatech.edu

Rohan Paleja*
Georgia Institute of Technology
Atlanta, Georgia
rpaleja3@gatech.edu

Matthew Gombolay
Georgia Institute of Technology
Atlanta, Georgia
matthew.gombolay@cc.gatech.edu

ACM Reference Format:
Yaru Niu*, Rohan Paleja*, and Matthew Gombolay. 2021. Supplementary for Multi-Agent Graph-Attention Communication and Teaming. In *Proc. of the 20th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2021), Online, May 3–7, 2021*, IFAAMAS, 2 pages.

1 ADDITIONAL ENVIRONMENT INFORMATION

Here, we present additional information about each domain used to benchmark MAGIC against baseline algorithms.

1.1 Predator-Prey

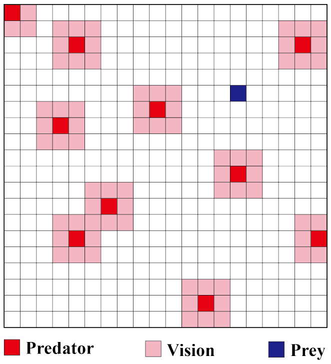

Figure 1: The visualization of the 10-agent Predator-Prey task. The predators (in red) with limited visions (light red region) of size 1 are searching for a randomly initialized fixed prey (in blue).

We utilize the predator-prey environment from Singh et al. [2]. Here, there are N predators with limited visions searching for a stationary prey. A predator or a prey occupies a single cell within the grid world at any time, and its location is initialized randomly at the start of each episode. The state at each point in the grid is the concatenation of a one-hot vector which represents its own location and binary values indicating the presence of predator and prey at this point. The observation of each agent is a concatenated array of the states of all points within the agent's vision. The predators can take actions up, $down$, $left$, $right$ or $stay$. We utilize the 'mixed' mode of Predator-Prey in which the predator incurs a reward -0.05 for each time step until the prey is found. An episode is defined as successful if all the predators find the prey before a predefined maximum time limit. We test two levels of difficulty in this environment. The difficulty varies as the grid size, and the number of predators increases, as more coordination is required to achieve success. The corresponding grid sizes and the number of predators are set to 10×10 with 5 predators and 20×20 with 10 predators. The 10-agent task is shown in Figure 1. We set the maximum steps for an episode (i.e., termination condition) to be 40 and 80, respectively. The vision is set to a unit length. We define a higher-performing algorithm in this domain as one that minimizes the average steps to complete an episode.

1.2 Traffic Junction

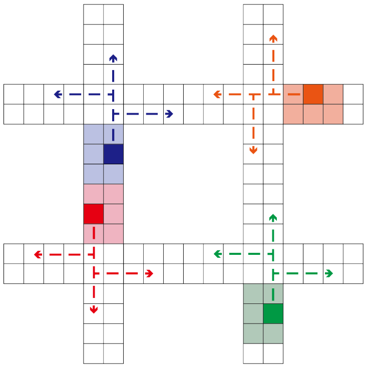

Figure 2: The visualization of the hard level Traffic Junction task. This task consists of four, two-way roads on a 18×18 grid with eight arrival points, each with seven different routes. Each agent is with a limited vision of size 1.

The second domain we utilize is the Traffic Junction environment. This environment, composed of intersecting routes and cars (agents) with limited vision, requires communication to avoid collisions. Cars enter the traffic junction from all entry points at each time step with a probability p_{arrive}, and are randomly assigned a route at the start. The maximum number of cars in the environment at a specific time is denoted by N_{max}, which varies across difficulty levels. A car occupies one cell at a time step and can take action "gas" or "brake" on its route. The state of each cell is the concatenation of a one-hot vector representing its location, and a value indicating the number of cars in this cell. The observation of each car is the concatenation of its previous action, route identifier, and all states of the cells within its vision. Two cars collide if they are in the same location, resulting in a reward of -10 for each car. The simulation terminates once all agents reach the end of its route or if the time

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

Figure 3: The visualization of 3 vs. 2 in Google Research Football. The five people shown in this figure are three offending players, one defending player and the goalie (left to right).

surpasses the predefined timeout parameter. Collisions will not incur "death" of agents or terminate the simulation. The agents will only be "dead" when it reaches the end of its route. There is a time penalty -0.01τ at each time step, where τ is the number of time steps that have passed since the agent's entry. An episode is considered successful if there are no collisions within the episode.

We validate our algorithm on three difficulty levels. The easy level consists of two, one-way roads on a 7×7 grid. There are two arrival points and two possible routes for each arrival point, and there are at most five agents ($N_{max} = 5$, $p_{arrive} = 0.3$). For the medium level, the junction consists of two, two-way roads on a 14×14 grid with four arrival points, each with three different routes. Here, there are at most ten agents ($N_{max} = 10$, $p_{arrive} = 0.2$). The hard level, as shown in Figure 2, consists of four, two-way roads on a 18×18 grid with eight arrival points, each with seven different routes, and there are at most twenty agents ($N_{max} = 20$, $p_{arrive} = 0.05$). The goal is to maximize the average success rate (i.e., no collisions within an episode). We set the limited vision parameter to 1 for both levels. Similar to [2], in Traffic Junction, we fix the gating action to be 1 for IC3Net and TarMAC-IC3Net, set all the hard attention outputs in GA-Comm to be 1, and set all the graphs used by the Message Processor in our method to be complete.

1.3 Google Research Football

Our final domain of Google Research Football [1] presents a challenging, mixed cooperative-competitive, multi-agent scenario with high stochasticity and sparse rewards. Google Research Football (GRF) is a physics-based 3D soccer simulator for reinforcement learning. This last domain presents an additional challenge as there are opponent artificial agents (AIs), significantly increasing the complexity of the state-action space. We present a depiction of this environment in Figure 3. To align with the partially observable setting, we extract the local observations from the provided global observations. The local observations include the relative positions of the players on both teams, the relative position of the ball, and one-hot encoding vectors which represent the ball-owned team and the game mode. GRF provides 19 actions including moving actions, kicking actions, and other actions such as dribbling, sliding and sprint. GRF provides several pre-defined reward signals, consisting of a scoring and a penalty box proximity reward. The penalty box proximity reward is shaped to push attackers to move forward towards certain locations. Many MARL frameworks have required these highly shaped rewards functions to perform well [1]. However, we choose to use only the scoring reward to verify the ability of our algorithm and baselines to function in a high-complexity stochastic domain with sparse rewards. Accordingly, the only reward all agents will receive in our evaluation is +1 when scoring a goal. The termination criterion is the team scoring, ball out of bounds, or possession change. We evaluate algorithms in the football academy scenario 3 vs. 2, as shown in Figure 3, where we have 3 attackers vs. 1 defender, and 1 goalie. The three offending agents are controlled by the MARL algorithm, and the two defending agents are controlled by a built-in AI. We find that utilizing a 3 vs. 2 scenario challenges the robustness of MARL algorithms to stochasticity and sparse rewards. In this domain, we seek to maximize the average success rate (i.e., a goal is scored) and minimize the average steps taken to complete an episode, thereby scoring a goal in the shortest amount of time.

2 ADDITIONAL TRAINING DETAILS

We distribute the training over 16 threads and each thread runs batch learning with a batch size of 500. The threads share the parameters of the policy network and update synchronously. There are 10 updates in one epoch. We use RMSProp with a learning rate of 0.001 in all the domains except Predator-Prey ten-agent scenario where we use 0.0003. The value coefficient β and discount factor λ are set to 0.01 and 1 respectively. The size of each agent's hidden state for LSTM is 128. The sizes of original encoded messages and the final messages for decision making are 128. 2/3 layers of GNNs have been used in practice and shown to work well [3]. Empirically, we find that two rounds of communication achieve the best performance with comparable training speeds to simpler methods such as CommNet and IC3Net. As such, we use two rounds of communication to test the performance of our method in all domains, and the number of heads for the first GAT layer (sub-processor 1) is set to be 4, 4, 1 in Predator-Prey, Traffic Junction and GRF respectively, and the number of heads for the output GAT layer (sub-processor 2) is set to be 1. We use one-round communication for efficiency evaluation for fair comparison, and the number of heads for the GAT layer is 1. The output size of the GAT encoder in the Scheduler is set to 32. We implement our method and baselines on each task over 5 random seeds and average the results.

REFERENCES

[1] Karol Kurach, Anton Raichuk, Piotr Stanczyk, Michal Zajac, Olivier Bachem, Lasse Espeholt, Carlos Riquelme, Damien Vincent, Marcin Michalski, Olivier Bousquet, and Sylvain Gelly. 2020. Google Research Football: A Novel Reinforcement Learning Environment. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020.* AAAI Press, 4501–4510. https://aaai.org/ojs/index.php/AAAI/article/view/5878

[2] Amanpreet Singh, Tushar Jain, and Sainbayar Sukhbaatar. 2019. Learning when to Communicate at Scale in Multiagent Cooperative and Competitive Tasks. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019.* OpenReview.net. https://openreview.net/forum?id=rye7knCqK7

[3] Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph Attention Networks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings.* OpenReview.net. https://openreview.net/forum?id=rJXMpikCZ