Reward-Sharing Relational Networks in Multi-Agent Reinforcement Learning as a Framework for Emergent Behavior

Doctoral Consortium

Hossein Haeri
Department of Mechanical Engineering
University of Massachusetts Lowell
hossein_haeri@uml.edu

ABSTRACT

Most prior works on MARL seek to implement intra-agent complex interactions by explicitly communicating agent actions. However, there have only been a few efforts that examine emergence as arising from complex 'social' interactions or relations based on individual objectives and reward functions. This study is about integrating a user-defined relational network into the MARL setup and evaluating the effects of agent-agent relations on the generation of emergent behaviors. Specifically, we propose a framework that uses the notion of Reward-Sharing Relational Networks (RSRN) to determine the relationship between agents where edge weights determine how much one agent is invested in the success of (or 'cares about') another. The preliminary results indicate that reward-sharing relational networks can effectively influence the learned behaviors towards the imposed relational network.

KEYWORDS

Multi-agent Systems; Reinforcement Learning; Social Simulation

ACM Reference Format:
Hossein Haeri. 2021. Reward-Sharing Relational Networks in Multi-Agent Reinforcement Learning as a Framework for Emergent Behavior: Doctoral Consortium. In *Proc. of the 20th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2021), Online, May 3–7, 2021*, IFAAMAS, 3 pages.

1 INTRODUCTION

Recent works in the field of Multi-Agent Reinforcement Learning (MARL) are taking first steps towards developing a better understanding of interactions between artificially intelligent (AI) agents and their resulting emergent behaviors. While several interesting results have been generated thus far [5–7, 10], we still lack a broader framework that formulates and solves the MARL problem and generates a subsequent theory of emergent behaviors for a network of interacting AI agents.

We formulate the problem as a Dec-POMDP where agents are trained using a decentralized method [2, 8] which allows agents to co-evolve in a shared environment; hence learning becomes a *collective* process. Moreover, while many state-of-the-art approaches primarily focus on either cooperative or competitive behaviors, the designed MARL framework must go beyond and allow the learned behaviors to span the entire spectrum of social behaviors. In many cases, such behavior may only be possible via the implementation of distinct reward structures for disparate agents.

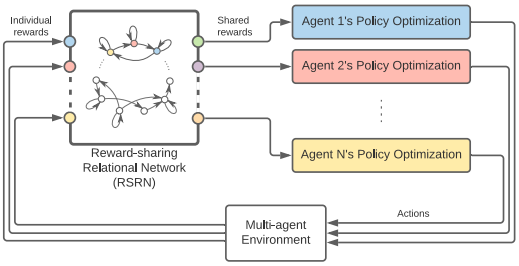

Figure 1: Different network structures generate different shared rewards, which are then used by the policy optimizers and produce distinct emergent behaviors.

To form a framework for emergent behavior, we need to better understand how relational networks should be structured. We begin with two insights into the nature of learning in social systems comprised of primates (including humans) or other mammals [3]. The first insight is that individual learning occurs in social settings [1]. Specifically, it has been shown that learning amongst primates and other mammals is governed by whom the individuals can relate to. Individual learning is driven by actions ranging from simple mimicking to complex evaluative behaviors, all within the presence of other *socially-related* individuals [9]. The presented work on reward-sharing relational networks formalizes this notion within the context of MARL.

The second insight borrows from neuroscience and indicates that learning individuals can connect or relate to a *limited* number of their counterparts [4]. This finding originates from studies of the size of the neocortex in primate brains and its relationship to the group size of the social systems these primates inhabit. Building on these two insights, we propose that reward-sharing and learning in a multi-agent system should: (a) be governed by whom an agent is related to, and (b) be limited to a small number of relations to obtain different types of emergent behaviors. The relational networks thus generated are used for training the agents, and may potentially generate distinct emergent behaviors depending on the structure of the relational network.

2 METHODOLOGY

We define a Relationally Networked Decentralized Partially Observable Markov Decision Process (RN-Dec-POMDP) as a tuple $(\mathcal{S}, \mathcal{A}, O, \mathcal{T}, \mathcal{R}, \mathcal{G})$, where $\mathcal{S} = \{S_1, ..., S_N\}$, $\mathcal{A} = \{A_1, ..., A_N\}$, $O =$

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.

$\{O_1, ..., O_N\}$, and $\mathcal{R} = \{R_1, ..., R_N\}$ are the joint set of individual states, actions, observations, and rewards, respectively. Agents take their actions based on their policies (in this case, deterministic) $\pi_i : O_i \rightarrow A_i$ and transition to the next state according to the joint probabilistic transition function $\mathcal{T} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \rightarrow [0, 1]$. Once an agent reaches its next state, it receives a reward r_i according to its personal reward function $R_i : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$. The tuple element $\mathcal{G} = (V_\mathcal{G}, E_\mathcal{G}, W_\mathcal{G})$ represents the ordered collection of all agents as vertices in the set $V_\mathcal{G}$, all binary agent relations as directed edges in the set $E_\mathcal{G}$, and edge weights as $W_\mathcal{G}$. The presence of an edge between two agents represents that they are related. The direction of the edge from a first agent to a second agent represents that the actions of the first agent are driven by the rewards obtained by the second. This may be understood as the second agent *sharing rewards* with the first agent, or that the first agent 'cares about' the success of the related second agent. Consequently, the first agent is likely to learn policies that benefit the other agent. Of course, the relational network also allows for self-directed edges, so an agent's actions can also be driven by its own rewards. The nature of these relationships is encapsulated in the matrix $W_\mathcal{G}$ which denotes the weights associated with the edges. The relational network weights can be expressed as the matrix $W_\mathcal{G}$ where its elements $w_{i,j} \in \mathbb{R}$ indicate how much agent i 'cares about' the success of agent j, based on the individual reward obtained by agent j. Thus, actions of agent i may be driven in part by the individual rewards of agent j, if these two agents are related, i.e. if $w_{i,j} \neq 0$.

In our RSRN-MARL framework, agents try to maximize their long-term shared return $\bar{R}_i$ by accumulating the discounted shared relational rewards $\bar{r}_i$ across a finite horizon T as $\bar{R}_i = \mathbb{E}\left(\sum_{k=0}^{T} \gamma^k \bar{r}_{i,k}\right)$ where $\bar{r}_i$ represents the shared relational reward of agent i, which incorporates the individual rewards of all related agents (and itself) and can be calculated using the scalarization function $f(\mathbf{r}_k, \mathbf{w}_i)$ that maps all individual rewards $\mathbf{r}_k = [r_1, r_2, ..., r_N]_k^\top$ at time step k to a single shared relational reward value $\bar{r}_{i,k}$ for agent i according to the agent-specific relational weight vector $\mathbf{w}_i = [w_{i,1}, w_{i,2}, ..., w_{i,N}]$.

It is possible to choose the scalarization function from several different alternatives, with the most commonly used choice often being a simple weighted sum across all the individual rewards, given by $\bar{r}_i = f_\text{s}(\mathbf{r}, \mathbf{w}_i) = \sum_{j=1}^{N} w_{i,j} r_j$. However, we found that the Weighted Product Model (WPM), given by $\bar{r}_i = f_\text{p}(\mathbf{r}, \mathbf{w}_i) = \prod_{j=1}^{N} r_j^{w_{i,j}}$, performed much better in our experiments. The weighted product model dramatically lowers the shared reward when even one of the individual rewards is close to the zero. Therefore, trainers are strictly promoted to evenly care about the related agents.

3 SIMULATION AND RESULTS

We consider the scenario where three agents try to reach three unlabeled landmarks, i.e. they get rewarded if they reach any landmark. Both state and action spaces are continuous and agents have been trained using our framework by integrating the Relational Network and Multi-Agent Deep Deterministic Policy Gradient (MADDPG) policy optimization algorithm [7]. To make the multi-agent environment more complex and create opportunities to observe emergent behaviors arise, we limit the mobility of one of the agents. Specifically, Agent 3 is hindered systematically so it cannot move as fast as other agents, and may be unable to reach a landmark within a single episode. To evaluate the collective behavior of the agents, we examine 6 different relational network configurations as shown in Figure 2. The individual performance of each agent i is then measured through its individual reward r_i, which is determined based on its distance to the closest landmark. We the run test episodes, as shown in Figure 3, to evaluate both their individual and relational, i.e. social performances.

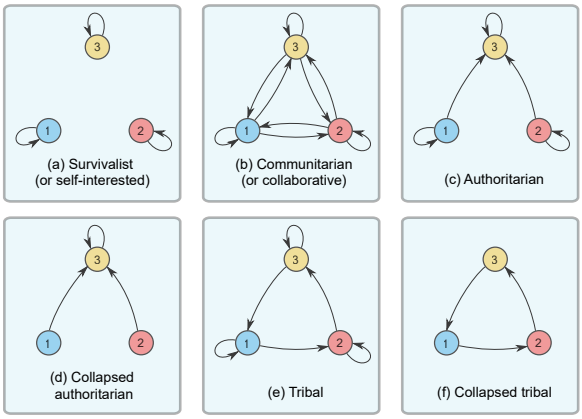

Figure 2: Examples of sociology-inspired relational network structures. An arrow directed from a first agent to a second represents that the first agent's actions are governed by the rewards obtained by the second agent.

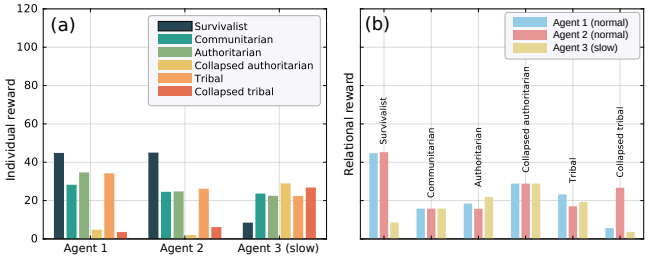

Figure 3: (a) Individual and (b) relational rewards averaged across 5000 test episodes, after training the agents over 500,000 episodes.

The learned policies often include emergent behaviors not explicitly defined in the problem formulation. E.g., agents in a *survivalist* relational networks quickly learned to go to a landmark, even pushing slower agents out of the way. On the other hand, agents trained with a *communitarian* relational network adapt and learn to distribute their task (capturing all landmarks) according to their initial positions and capabilities. The system exhibits emergent behavior as fast-moving agents learn to assist the slow-moving agent towards the landmark, before finding a landmark of their own. Similar emergent behaviors manifest in *authoritarian* and *tribal* relational networks as well.

These figures and associated videos[1] help reveal insights into how different relational networks produce distinct emergent behaviors, as well as the performance of individual agents.

[1]Videos are available at `sites.google.com/view/marl-rsrn`

REFERENCES

[1] Nadav Aharony, Wei Pan, Cory Ip, Inas Khayal, and Alex Pentland. 2011. Social fMRI: Investigating and shaping social mechanisms in the real world. *Pervasive and Mobile Computing* 7, 6 (2011), 643–659.

[2] Bowen Baker, Ingmar Kanitscheider, Todor Markov, Yi Wu, Glenn Powell, Bob McGrew, and Igor Mordatch. 2019. Emergent tool use from multi-agent autocurricula. arXiv. *arXiv preprint arXiv:1909.07528* (2019).

[3] Frans De Waal. 2010. *The age of empathy: Nature's lessons for a kinder society*. Broadway Books.

[4] R. I.M. Dunbar. 1992. Neocortex size as a constraint on group size in primates. *Journal of Human Evolution* 22, 6 (jun 1992), 469–493. https://doi.org/10.1016/0047-2484(92)90081-J

[5] Jakob Foerster, Ioannis Alexandros Assael, Nando De Freitas, and Shimon Whiteson. 2016. Learning to communicate with deep multi-agent reinforcement learning. In *Advances in neural information processing systems*. 2137–2145.

[6] Joel Z Leibo, Edward Hughes, Marc Lanctot, and Thore Graepel. 2019. Autocurricula and the emergence of innovation from social interaction: A manifesto for multi-agent intelligence research. *arXiv preprint arXiv:1903.00742* (2019).

[7] 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. In *Advances in neural information processing systems*. 6379–6390.

[8] Shayegan Omidshafiei, Jason Pazis, Christopher Amato, Jonathan P How, and John Vian. 2017. Deep decentralized multi-task multi-agent reinforcement learning under partial observability. *arXiv preprint arXiv:1703.06182* (2017).

[9] Charles Pezeshki and Ryan Kelley. 2014. It's all about relationships: how human relational development and social structures dictate product design in the socio-sphere. In *Product Development in the Socio-sphere*. Springer, 143–168.

[10] Oriol Vinyals, Igor Babuschkin, Wojciech M Czarnecki, Michaël Mathieu, Andrew Dudzik, Junyoung Chung, David H Choi, Richard Powell, Timo Ewalds, Petko Georgiev, et al. 2019. Grandmaster level in StarCraft II using multi-agent reinforcement learning. *Nature* 575, 7782 (2019), 350–354.