Towards More Effective Multi-agent Coordination via Alignment

Zixian Ma

Stanford University
September 2022

An honors thesis submitted to the department of Computer Science
in partial fulfillment of the requirements for the undergraduate honors program

Advisor: Fei-Fei Li

_____ Date: _____

Fei-Fei Li (Thesis Advisor)
Sequoia Professor of Computer Science
School of Engineering

_____ Date: _____

Michael Bernstein (Co-Advisor)
Associate Professor of Computer Science
School of Engineering

Abstract

Multi-agent systems in the real world, such as autonomous vehicles and robot swarms, often rely on robust reinforcement learning algorithms to achieve effective coordination. However, existing multi-agent reinforcement learning frameworks struggle to scale to new tasks and new agents without access to shaped rewards, centralized training, or full observability. By contrast, animals learn to collectively collaborate on tasks without any centralized training by *aligning* their behaviors within a local context. This is known as the self-organization principle in Zoology. Inspired by this principle, I introduce a simple and task-agnostic alignment-driven intrinsic reward in my thesis. This intrinsic reward encourages *aligning* dynamics: individual agents learn behaviors that match their neighbors' expectations. Compared to alternative intrinsic rewards based on curiosity, alignment as an intrinsic reward improves decentralized coordination across cooperative and competitive tasks. Alignment also enables agents to successfully coordinate under partially observable settings and scales well as the number of agents grows. In investigating how alignment benefits multi-agent training, I find that alignment helps break coordination symmetries. These results suggest that agents learn to divide tasks amongst themselves better as a result of alignment, which may be a more useful strategy than curiosity-driven exploration for multi-agent coordination.

Acknowledgements

I would like to take this opportunity to sincerely thank all the people, including my thesis advisors, mentors, labmates, friends and parents, who have helped and supported me in research. I could have not made this far without you. You encouraged and empowered me to accomplish this honors thesis and gave me the courage to further pursue a research career.

To Professors Fei-Fei Li and Michael Bernstein: I appreciate greatly your generous support and help. To Professor Li: I remember presenting this project to you at its very early stage at a Visual Genome meeting, and your interest in this project has been one of my greatest sources of motivation. I have always looked up to you and yearn to become a phenomenal researcher like you. To Professor Bernstein, you showed me the charm of computer science research through CS197: Undergraduate CS Research. Your research methodology, including crucial components like vectoring, has been guiding and will continue to guide me through my adventurous research journey. You recommended us seeking inspirations from literature on animals' group behaviors, such as Iain Couzin's work, for our work on multi-agent coordination; and animals' collective intelligence has become our most important inspiration.

To Ranjay Krishna and Rose Wang: I cannot express my boundless gratitude for your mentorship. To Ranjay: You are the best (and arguably the funniest) mentor I have ever met. This research project would not have been possible at all without your support and help. You taught me not only how to conduct research but also how to become a better researcher in all ways. I have learned a lot but still have so much more to learn from you. You encouraged me to achieve beyond what I thought I was capable of in research. You showed me the light and guided me through challenging times. To Rose: I have had so much respect for you and learned a ton from you. You think clearly, execute quickly, and write sharply. You showed me what qualities a great researcher should have. I aspire to become a researcher like you.

To my labmates Austin Narcomey, Helena Vasconcelos, Madeleine Grunde-McLaughlin, Jerry Hong and Omer Gul: You made me feel that I could always talk to you about research and life. I am so grateful for your presence and support along my research journey and so delighted that we have developed friendship through weekly stand-ups and meetings.

To my dearest friends Katherine Wu, Irena Gao, Edward Vendrow and Ethan Schonfeld: You have been the most supportive people in my life. Thank you for letting me share the ups and downs in research with you.

To my parents: You are the BEST parents in the world, and you are the reason for everything I have accomplished today.

Contents

Chapter 1

Introduction

Many real world AI applications can be formulated as multi-agent systems, including autonomous vehicles (Cao, Yu, Ren, & Chen, 2012), resource management (Ying & Dayong, 2005), traffic control (Sunehag et al., 2017), robot swarms (Swamy, Reddy, Levine, & Dragan, 2020) and multi-player video games (Silver et al., 2016). Such applications require agents to adapt their behaviors with respect to one another in order to successfully coordinate with each other. Unfortunately, adaptive coordination algorithms are challenging to develop because they must account for other agents' behaviors which may change over the course of training. This non-stationarity makes learning difficult and unstable, particularly when agents cannot fully observe each other (J. N. Foerster et al., 2017).

Prior work has explored the use of centralized training (J. Foerster, Farquhar, Afouras, Nardelli, & Whiteson, 2018; Rashid et al., 2018; Sunehag et al., 2017; Lowe et al., 2017) and intrinsic rewards (Iqbal & Sha, 2020) to overcome these challenges. Centralized training assumes access to all agents' observations and actions to improve joint state-action estimates. However, it neither scales with the number of agents (Iqbal & Sha, 2019a; Liu, Yeh, & Schwing, 2020) nor handles scenarios in which agents can not communicate easily, such as in human-robot collaborations. Other

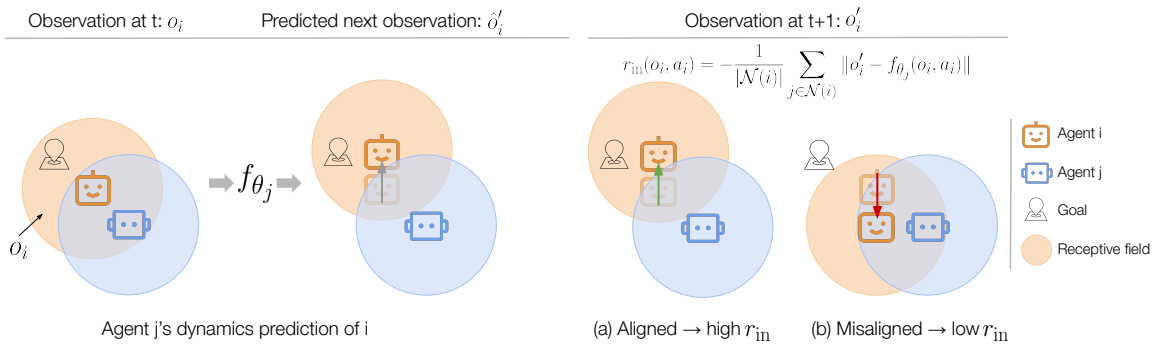

Figure 1.1: We introduce alignment, a task-agnostic intrinsic reward to improve multi-agent systems. Intuitively, alignment encourages agents to become more predictable to their neighbors. An agent (e.g. agent i here) learns to behave in ways that match its neighbors' (e.g agent j's) predictions of its next observation. Here, agent j expects agent i to move up instead of down, moving closer to a point of interest above it. Agent i attains (a) a higher reward when its action (e.g. upward) aligns with this prediction or (b) a lower reward when its action (e.g. downward) is misaligned.

works rely on task-specific rewards (Jain et al., 2020; Lowe et al., 2017); although these rewards alleviate the dependency on having complete knowledge of other agents, they are meticulous and expensive to generate because they require domain knowledge. Single-agent reinforcement learning avoids hand-designing rewards by introducing intrinsic rewards that incentivize an agent to explore novel states (Pathak, Agrawal, Efros, & Darrell, 2017; Stadie, Levine, & Abbeel, 2015).

In this work, we propose *alignment* as a multi-agent intrinsic reward to overcome these challenges. Intuitively, alignment encourages agents to elicit behaviors that decrease future uncertainty for their team: it encourages each agent to choose actions that match its teammates' expectations. Consider a collaborative navigation task where N agents aim to simultaneously occupy N goal locations. Alignment encourages each agent to move to goals others expect it to occupy, like goals that are either closest to the agent or goals that other agents aren't moving towards (Figure 1.1).

Our formulation of alignment is inspired by the self-organization principle in Zoology (I. Couzin, 2007). This principle hypothesizes that collective animal intelligence emerges because groups synchronize their behaviors using only their local environment (Figure 1.2 and 1.3); they do not rely on

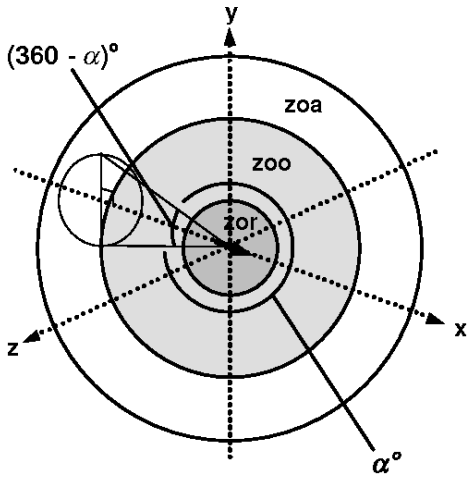

Figure 1.2: Representation of an individual animal in the self-organizing model centred at the origin: zor - zone of repulsion, zoo - zone of orientation (or alignment), zoa - zone of attraction (I. D. Couzin et al., 2002).

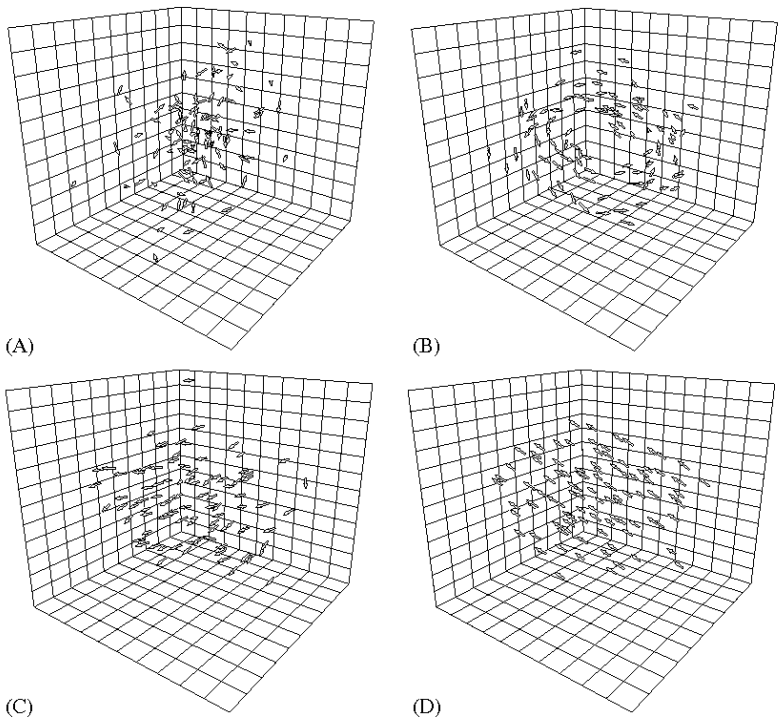

Figure 1.3: Different collective behaviours exhibited by the self-organizing model: (A) swarm, (B) torus, (C) dynamic parallel group, (D) highly parallel group (I. D. Couzin et al., 2002).

complete information about other agents and can coordinate successfully by predicting the dynamics of agents within their field-of-view (Collett, Despland, Simpson, & Krakauer, 1998; Theraulaz & Bonabeau, 1995; Ben-Jacob et al., 1994; Buhl et al., 2006). Similarly, alignment as an intrinsic reward is calculated based on the agent's local observations; it does not require a centralized controller nor full observability; it uses neighboring agents' expectations to choose actions that reinforce those expectations. Alignment is task-agnostic and can be applied to both collaborative and competitive multi-agent tasks.

We demonstrate the utility of alignment as an intrinsic reward by showing that it improves multi-agent performance across collaborative and competitive tasks on the multi-agent particle environment, a popular environment for multi-agent reinforcement learning (Lowe et al., 2017). Empirically, we show that alignment improves the performance of decentralized algorithms across different multi-agent tasks, and overcomes the learning challenges in partially observable environments with sparse task rewards. It outperforms curiosity-based multi-agent intrinsic reward baselines (Ndousse, Eck, Levine, & Jaques, 2021; Stadie et al., 2015; Iqbal & Sha, 2020) and scales to more agents. Investigating why alignment improves performance, we find that aligning behaviors lead to better sub-task division amongst collaborators (Hu, Lerer, Peysakhovich, & Foerster, 2020). It also enables zero-shot coordination with new agents that weren't trained together, implying that alignment can support not just decentralized but disjoint multi-agent training. Finally, we find that performance improvements are correlated with the accuracy of the learned dynamics model. Taken together, our experimental results provide nuanced empirical evidence that alignment improves multi-agent collaboration, especially in ecologically valid conditions where the self-organization principle has been observed: in cooperative tasks with decentralized training in partially observable environments.

Chapter 2

Related work

Our formulation of alignment, a task-agnostic intrinsic reward for multi-agent training, draws inspiration from the self-organization principle in Zoology, which posits that synchronized group behavior is mediated by local behavioral rules (I. Couzin, 2007) and not by a centralized controller (Camazine et al., 2020). Group cohesion emerges by predicting and adjusting one's behavior to that of near neighbors (Buhl et al., 2006). This principle underlies the coordination found in multi-cellular organisms (Camazine et al., 2020), the migration of wingless locusts (Collett et al., 1998), the collective swarms of bacteria (Ben-Jacob et al., 1994), the construction of bridge structures by ants (Theraulaz & Bonabeau, 1995), and some human navigation behaviors (I. Couzin, 2007).

2.1 Intrinsic motivation for single agents

Although we draw inspiration from Zoology for formalizing alignment as an intrinsic reward, there is a rich body of work on intrinsic rewards within the single-agent reinforcement learning community. Sparse rewards make it difficult for agents to explore and discover optimal policies. To incentivize continued exploration, even when non-optimal successful trajectories are uncovered first,

scholars have argued for the use of intrinsic motivation (Schmidhuber, 1991). Single-agent intrinsic motivation has focused on exploring previously unencountered states (Pathak et al., 2017; Burda, Edwards, Pathak, et al., 2018), which works particularly well in discrete domains. In continuous domains, identifying unseen states requires keeping track of an intractable number of visited states; instead, literature has recommended learning a forward dynamics model to predict future states and identify novel states using the uncertainty of this model (Achiam & Sastry, 2017). Other formulations encourage re-visiting states where the dynamics model's prediction of future states errs (Stadie et al., 2015; Pathak et al., 2017). Follow up papers have improved how uncertainty (Kim, Sano, De Freitas, Haber, & Yamins, 2020) and model errors (Burda, Edwards, Storkey, & Klimov, 2018; Sekar et al., 2020) are calculated.

2.2 Intrinsic motivation for multiple agents

Most intrinsic rewards used for multi-agent systems have been adapted from single-agent exploration incentives (Iqbal & Sha, 2019b; Böhmer, Rashid, & Whiteson, 2019; Schafer, 2019) and have primarily focused on cooperative tasks. Recent works propose task-specific intrinsic rewards to improve either coordination, collaboration, or deception: These rewards either maximize information conveyed by an agent's actions (Iqbal & Sha, 2019b; Chitnis, Tulsiani, Gupta, & Gupta, 2020; T. Wang, Wang, Wu, & Zhang, 2019), shape the influence of an agent (Jaques et al., 2019; J. N. Foerster et al., 2017), incentivize agents to hide intentions (Strouse, Kleiman-Weiner, Tenenbaum, Botvinick, & Schwab, 2018), build accurate models of other agents' policies (Hernandez-Leal, Kartal, & Taylor, 2019; Jaques et al., 2019), or break extrinsic rewards to do better credit assignment (Du et al., 2019).

Several multi-agent intrinsic rewards (Hernandez-Leal et al., 2019; Jaques et al., 2019), including ours, rely on the ability to model others' dynamics in a shared environment. This ability is a key

component to coordination, closely related to Theory of Mind (Tomasello, Carpenter, Call, Behne, & Moll, 2005). Our work can be interpreted as using a Theory of Mind model of others' behaviors to calculate an intrinsic motivation loss. Our proposal is related to model-based reinforcement learning (Jaderberg et al., 2016; R. E. Wang, Kew, et al., 2020); however, instead of learning a dynamics model for control, we learn a dynamics model as a source of reward. Our work is closely related to a recently proposed auxiliary loss on predicting an agent's own future states (Ndousse et al., 2021). However, there are three key differences. First, their work predicts ego-agent observations, whereas our work additionally predicts future observations from the other agents' point of view. Second, their loss optimizes state embeddings while ours optimizes agents' policies. Third, their work focuses on cooperative tasks whereas ours applies to both cooperative and competitive domains.

2.3 Multi-agent reinforcement learning algorithms

Today, the predominant training framework for deep multi-agent reinforcement learning follows a paradigm of centralized training and decentralized execution (Lowe et al., 2017; J. Foerster et al., 2018; Iqbal & Sha, 2019a; Liu et al., 2020; Rashid et al., 2018). This framework allows a critic to access the observations and actions of all agents to ease training. However, there are several situations where centralized training may not be desirable or possible. Examples include low bandwidth communication restrictions or human-robot tasks where observations cannot be easily shared between agents (Ying & Dayong, 2005; Cao et al., 2012; Huang, Cakmak, & Mutlu, 2015). Decentralized training is therefore the most practical training paradigm but it suffers from unstable training: the environment is nonstationary from a single-agent's perspective (Lowe et al., 2017). Our work uses a decentralized training framework and tackles the nonstationarity challenge with an intrinsic reward designed to improve an agent's ability to model others. We also apply alignment to centralized training and observe that it still aids cooperative and some competitive tasks.

Chapter 3

Methods

3.1 Background

We formulate our setting as a partially observable Markov game $(\mathcal{S}, \mathcal{O}, \mathcal{A}, \mathcal{T}, r_{\text{ex}}, N)$ (Littman, 1994). A Markov game for N agents is defined by a state space $\mathcal{S}$ describing the possible configurations of the environment. The observation space for agents is $\mathcal{O} = (\mathcal{O}_1, \ldots, \mathcal{O}_N)$ and the action space is $\mathcal{A} = (\mathcal{A}_1, \ldots, \mathcal{A}_N)$. Each agent i observes $\mathbf{o}_i \in \mathcal{O}_i$, a private partial view of the state, and performs actions $a_i \in \mathcal{A}_i$. Using the observation, each agent uses a stochastic policy $\pi_{\theta_i} : \mathcal{O}_i \times \mathcal{A}_i \to [0, 1]$, where θ_i parameterizes the policy. The environment changes according to the state transition function which transitions to the next state using the current state and each agent's actions, $\mathcal{T} : \mathcal{S} \times \mathcal{A} \to \mathcal{S}$. The team of agents obtains a shared extrinsic reward as a function of the environment state, $r_{\text{ex}} : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$. The team's goal is to maximize the total expected return: $R = \sum_{t=0}^{T} \gamma^t r_{\text{ex}}^t$ where $0 \leq \gamma \leq 1$ is the discount factor, t is the time step, and T is the time horizon. The environment may also contain adversarial agents who have their own reward structure.

3.2 Alignment

To understand alignment intuitively, let's revisit the cooperative navigation task, where N agents are rewarded for simultaneously occupying as many goal locations as possible. In Figure 1.1, agent i has a dynamics model trained on its past experiences. It predicts how future states will evolve from the point of view of agent j, who is within i's view. In this example, j will expect i to move towards the goal since i is closer to it. Alignment encourages i to pursue the action that j expects (Figure 1.1(a)). In turn, j can now assume that the observed goal location will eventually be occupied by i and should therefore explore to find another goal. By aligning shared expectations, agent behaviors become more predictable. Conversely, when neighbors behave opposite to an agent's predictions, the agent can infer about the environment outside of its own receptive field (Krause, Ruxton, Ruxton, Ruxton, et al., 2002). For example, in Figure 1.1 (b), if agent j observes i running away from a goal, this surprising behavior might indicate the existence of an adversary outside j's receptive field.

Our training algorithm consists of three interwoven phases of learning a dynamics model, calculating the alignment reward, and training the agent policies. The following details the phrases and is summarized in Algorithm 1.

3.3 Training the dynamics model

Similar to prior work (X. Wang, Xiong, Wang, & Wang, 2018; Kidambi, Rajeswaran, Netrapalli, & Joachims, 2020), each agent i learns a dynamics model f_{θ_i} to predict the next observation $\hat{o}'_i$ given its current observation and action o_i, a_i,

$$\hat{o}'_i = f_{\theta_i}(o_i, a_i).$$

Algorithm 1 Alignment

Initialize replay buffer D and D'
Initialize N agents with random θ_i: $i \in [1, N]$
while not converged **do**
  for $b = 1 \dots B$ **do**
    Populate buffer D with episode using policies $(\pi_{\theta_1}, \dots, \pi_{\theta_N})$
  end for
  // Train dynamics model
  for agent $i = 1 \dots N$ **do**
    Sample transitions: $\{(o_i, a_i, r_{\text{ex}}, o'_i)\} \sim D_i$
    Predict $\hat{o}'_i = f_{\theta_i}(o_i, a_i)$
    Update dynamics θ_i using o'_i.
  end for
  // Calculate alignment reward
  for agent $i = 1 \dots N$ **do**
    Sample B transitions: $\{(o_i, a_i, r_{\text{ex}}, o'_i)\} \sim D_i$
    Compute intrinsic rewards $r_{\text{in}}(o_i, a_i)$
    Add $\{(o_i, a_i, r_{\text{ex}} + \beta r_{\text{in}}, o'_i)\}$ to D'_i
  end for
  // Policy learning
  Update all θ_is using transitions from D'
end while

We use a three-layer Multi-Layer Perceptron (MLP) with ReLU non-linearities as the dynamics model. We minimize the mean squared error between its prediction and ground truth next observation o'_i.

3.4 Calculating intrinsic reward

The intrinsic reward captures how well agent i aligns to its neighbors' (agent j's) expectations on its next state. Calculating this reward requires j to accurately predict i's behavior, simulating a Theory of Mind (Tomasello et al., 2005). As suggested by the self-organization principle, i must learn to align to j's predictions. Ideally, the alignment intrinsic reward is calculated as:

$$r_{\text{in}}(o_i, a_i) = -\frac{1}{|\mathcal{N}(i)|} \sum_{j \in \mathcal{N}(i)} \|o'_i - f_{\theta_j}(o_i, a_i)\|$$

where $\mathcal{N}(i)$ is the set of neighbors within i's receptive field, including i itself. The alignment reward is high when the average L_2 loss is small, when i's actual next observation is close to agent j's predicted observation of i for all j in its neighbors. In that case, i has chosen an action that aligns with j's expectations of how i should act.

In a decentralized training setup, however, i doesn't have access to j's dynamics model f_{θ_j}, so i approximates j's dynamic model with a proxy: its own dynamics model f_{θ_i}. Such an approximation is ecologically valid since we often approximate others' behaviors using a second-order cognitive Theory of Mind (Morin, 2006). Additionally, i doesn't have access to j's entire observation; so, we restrict the future prediction from j's point of view by using the portion of j's observation i can see: $o_{i\cap j} = o_i \odot o_j$.

Agent i's decentralized intrinsic reward then becomes:

$$r_{\text{in}}(o_i, a_i) = -\frac{1}{|\mathcal{N}(i)|} \sum_{j \in \mathcal{N}(i)} \|o'_{i\cap j} - f_{\theta_i}(o_{i\cap j}, a_i)\|$$

We found that the approximation of f_{θ_j} using f_{θ_i} works well emprically. Dynamics model losses for all agents quickly decrease within 5-10 training epochs. We validate its applicability in heterogeneous multi-agent tasks where agents have variable capabilities.

3.5 Policy learning

Once the alignment rewards are calculated, the total rewards at each step for each agent i is: $r_i = r_{\text{ex}} + \beta r_{\text{in}}(o_i, a_i)$ where r_{ex} is the extrinsic reward provided by the environment and β is a hyperparameter for weighing the intrinsic reward in the agent's overall reward calculation. In practice, we set β to be $\frac{1}{|\mathcal{O}_i|}$ where $|\mathcal{O}_i|$ is the observation dimension; we find this scale generalizes well across tasks. Since our contribution is agnostic to any particular multi-agent training algorithm,

the team of agents can now be trained using any multi-agent training algorithm to maximize returns $R = \sum_{t=0}^{T} \gamma^t r$. Both centralized and decentralized training algorithms can make use of these rewards. In our experiments, we use the multi-agent variant of the soft-actor critic algorithm with both decentralized as well as centralized critics (Haarnoja, Zhou, Abbeel, & Levine, 2018; Iqbal & Sha, 2019a).

3.6 Extending alignment to competitive tasks

We extend the alignment formulation to competitive tasks where a teaam of agents compete against adversaries. In this case, agents are encouraged to *misalign* with their adversaries' expectations, agents are incentivized to be unpredictable to their adversaries:

$$r_{\text{in}} = \frac{1}{|\mathcal{N}_{\text{adv}}(i)|} \sum_{k \in \mathcal{N}_{\text{adv}}(i)} \|o'_{i \cap k} - f_{\theta_i}(o_{i \cap k}, a_i))\|$$

where $\mathcal{N}_{\text{adv}}(i)$ are its adversaries within its receptive field.

Chapter 4

Experiments

Our experiments explore the utility of using alignment as an intrinsic reward along several axes of multi-agent algorithms: **how does alignment interact with different training paradigms (centralized vs. decentralized), reward types (sparse vs. curiosity-based intrinsic rewards), task dynamics (cooperative vs. competitive), observability (partial vs. full), and number of agents in the environment?**

We end by investigating how and why alignment improves coordination performance in three evaluation conditions: **how does alignment help with symmetry-breaking (Hu et al., 2020; R. E. Wang, Wu, et al., 2020), enable zero-shot generalization to new partners, and interact with a noisy dynamics model?**

4.1 Environment

Our experiments use the multi-agent particle (Mordatch & Abbeel, 2017; Lowe et al., 2017) and Google Research football (Kurach et al., 2019) environments for evaluation across both multi-agent collaborative and competitive tasks.

4.1.1 Basics

The multi-agent particle environment is a two-dimensional world with continuous space and discrete time steps. Agents can "stay" or apply a fixed force (to increase or decrease its velocity) towards one of the four cardinal directions: "up", "down", "left", "right". The Google Research football environment is a three-dimensional world with continuous space and discrete time steps. Each agent controls one player, who observes the ball and all the other players' positions and directions. Agents can apply one of ten actions from "top_left", "top", "top_right", "right", "bottom_right", "bottom", "bottom_left", "sprint", and "dribble".

4.1.2 Observability

Both environments originally assume full observability where each agent can observe the position $p = (x, y)$ and velocity $v = (\Delta x, \Delta y)$ of all agents, yielding:

$$\mathbf{o}_{i,\text{full}} = [p_1, \ldots, p_N, v_1, \ldots, v_N].$$

We extend these environments to a partially observable experimental condition, where agent i observes only the portion within its receptive field; like prior work with partial observability (Corder, Vindiola, & Decker, 2019), we hide the position and velocity information of any agent j outside of agent i's receptive field; if the Euclidean distance between agent i and j surpasses a vicinity threshold τ, then p_j and v_j are 0 in $\mathbf{o}_{i,\text{partial}}$. We set $\tau = 0.5$ for partially observable and ∞ in the original fully observable case, where the world's width and height are 2.0 in the multi-agent particle environment and $0.84 : 2.0$ in the Google Research football environment. The multi-agent particle environment also contains inanimate objects, some of which act as goal locations; they are similarly represented with position and velocity. Partial observability is a more ecologically valid training condition since most agents in real-world tasks can only observe a small portion of their

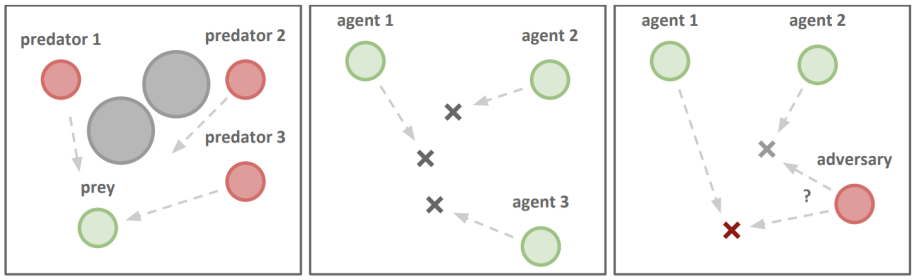

Figure 4.1: Illustrations of some tasks from the multi-agent particle environment: a) Predator-Prey b) Cooperative Navigation c) Physical Deception.

environment at a given time; it also poses the additional challenge of estimating an accurate state value from only the observations.

4.2 Tasks

4.2.1 Multi-agent particle environment

We use the following tasks from the multi-agent particle environment (Lowe et al., 2017; Liu et al., 2020) (See Figure 4.1 for example illustrations). We choose N based on prior work. We test scalability by doubling the number of agents and adversaries.

Cooperative navigation: N agents must cooperate to reach a set of N goal locations. Agents are collectively rewarded based on the occupancy of any agent on any goal location.

Heterogeneous navigation: N agents must reach N goals but they differ in speeds and sizes. $\frac{N}{2}$ agents are slow and big, and the other $\frac{N}{2}$ agents are fast and small.

Keep-away: There are N landmarks, one of which is the goal location. N agents know which landmark is the goal location. They are rewarded for occupying it and for preventing M adversaries from occupying it. Adversaries are rewarded for pushing the agents away from the goal but they do not know which landmark is the goal. This must be inferred from the agents' behavior.

Physical deception: N agents must cooperate to reach a single goal location and are rewarded if any

one of them occupies the goal. However, $\frac{N}{2}$ adversaries are also rewarded for occupying the goal; if this happens, the agents are penalized. Similar to keep-away, the adversaries do not know which landmark is the goal and this information must be inferred from the agents' behavior. The agents must learn deceive the adversaries by covering all the landmarks.

Predator-prey: N slow adversaries chase and capture N fast cooperating agents around a randomly generated obstacle-filled environment. Each time an adversary catches an agent, the agent is penalized and the adversary is rewarded.

Symmetry-breaking initializations

We create a symmetry-breaking version of each task for evaluation by initializing the environment in the following ways:

Cooperative Navigation and Heterogenous Navigation: All agents are initialized at the origin (i.e. center of the world), and target landmarks are placed randomly on a circle perimeter with the maximum radius (i.e. world radius - the greatest landmark size) so that each agent is equidistant from each target landmark.

Physical Deception: Both agents and adversaries start at the origin. All the landmarks, including the goal, are randomly initialized on a circle perimeter.

Predator-prey: The collaborative agents are initialized at the center while the adversaries are placed randomly on a circle perimeter. All the landmarks are randomly initialized in the world.

Keep-away: All the cooperative agents are placed at the origin. Adversaries and landmarks, including the goal, are randomly initialized on a circle perimeter. In this task setup, we do not initialize the adversaries at the center because they are awarded for colliding with the cooperative agents.

Figure 4.2: An illustration of the Academy 3 vs 1 with Keeper scenario from the Google research football environment.

4.2.2 Google Research football

We use the *Academy 3 vs 1 with Keeper* task from the Google Research football environment (Kurach et al., 2019) (See Figure 4.2 for an illustration). In this task, three of our players try to score from the edge of the box, one on each side, and the other at the center. Initially, the player at the center has the ball and is facing the defender. There is an opponent keeper.

4.3 Training and evaluation

We detail the training and evaluation setups in this section. All the hyperparameters used in the training can be found in the appendix.

4.3.1 Multi-agent particle environment

We train all algorithms with 5 random state initializations. Each experiment uses one Tesla K40 GPU to train for 200 epochs or until convergence, i.e. the best evaluation episode reward hasn't changed for 100 epochs. Each epoch equates to 200k policy update steps, or 800k episodes of 25 timesteps.

We evaluate the algorithms by running $1,000$ test episodes of 25 timesteps and mainly report the mean average test episode reward and standard error across the 5 random seeds. We also evaluate on task-specific metrics, including agent-goal occupancy/agent-adversary collision count, and agent-goal/agent-adversary distance.

4.3.2 Google Research football

We train all algorithms on the football environment with 3 random seeds. Each experiment uses one Tesla TITANX GPU to train for 50K iterations, or 5M timesteps. We evaluate the algorithms on the average episode rewards across the last 100K timesteps and report the mean average episode reward and the standard errors across the seeds.

4.4 Baselines

All algorithms are trained using the same agent architectures and optimization algorithm, but with different task-specific extrinsic rewards. We optimize using two variations of the soft actor-critic algorithm (Haarnoja et al., 2018): a decentralized one that trains each agent individually without access to other agents' observations and actions(ie. the original soft-actor critic algorithm from haarnoja2017soft) and a centralized one with a critic that has access to other agents' observations and actions (Iqbal & Sha, 2019a). Centralized training is typically assumed to lead to better multi-agent performance since the critic has access to more information when estimating the state value (Rashid et al., 2018; Lowe et al., 2017); however, it does not scale with an increasing number of agents. Our intrinsic reward can also be added to other optimization methods such as COMA (J. Foerster et al., 2018) and VDN (Sunehag et al., 2017). We leave this to future work to avoid conflating the effects of alignment as an intrinsic reward with COMA's counterfactual and VDN's value decomposition.

For rewards, we use SPARSE (Lowe et al., 2017; Kurach et al., 2019), CURIO$_{\text{self}}$ (Stadie et al.,

2015), CURIO$_{\text{team}}$ (Iqbal & Sha, 2020), and variations of our ALIGN rewards. SPARSE awards agents with extrinsic rewards only when they reach a goal state. SPARSE refers to the SCORING reward in the Google Research football environment. CURIO$_{\text{team}}$ is a curiosity-based multi-agent intrinsic reward which maximizes the average L_2 loss (instead of minimizing it in ALIGN) to reward agents for exploring more novel states. CURIO$_{\text{self}}$ also maximizes the L_2 loss but only using agent i's own observation and self-prediction. To ensure that our results are comparable between baselines, only collaborative agents receive intrinsic rewards; adversaries are trained with SAC *without* alignment.

We experiment with three variants of our alignment reward: ALIGN$_{\text{self}}$ only incentivizes self-alignment (similar to the auxiliary loss in ndousse2021emergent but we treat it as an intrinsic reward for policy optimization); ALIGN$_{\text{team}}$ encourages agents to align to their team; and ALIGN$_{\text{adv}}$ encourages misalignment to adversaries.

Chapter 5

Results and analysis

5.1 Multi-agent particle environment

Alignment with decentralized training outperforms sparse and curiosity rewards for *cooperative* tasks. Table 5.1 (top row) reports the task performance under partial observability. ALIGN$_{\text{self/team/adv}}$ consistently improves multi-agent performance when compared against SPARSE and CURIO$_{\text{self/team}}$ under partial observability. This provides empirical evidence that the self-organizing principle improves coordination under partial information, a setting that is most realistic to real world multiagent systems. Additionally, alignment performs better than CURIO$_{\text{self/team}}$ intrinsic rewards, suggesting that intrinsic rewards that align behaviors may be more useful than curiosity-driven exploration for coordination.

Although curiosity has proven useful for exploration in single-agent tasks, we find that alignment—which mathematically encourages agents to be more predictable instead of finding novelty—outperforms curiosity in multi-agent tasks. One example to illustrate this is the cooperative navigation task: rather than having all agents explore and find each individual goal (incentivized by

Table 5.1: We report the mean test episode extrinsic rewards and standard errors of *decentralized* methods with different intrinsic rewards under partial and full observability. Under partial observability, ALIGN$_{\text{self/team/adv}}$ outperforms SPARSE and both curiosity-based intrinsic rewards CURIO$_{\text{self,team}}$ on all tasks. Under full observability, ALIGN$_{\text{team}}$ mostly surpasses SPARSE and CURIO$_{\text{self}}$ except for *Keep-away (2v2)* and *Physical deception (2v1)* respectively. These results demonstrate the benefit of using alignment as intrinsic reward to train better decentralized policies, especially under partial observability.

		Cooperative		**Competitive**		
Task (Agt # vs. Adv #)		**Coop nav. (3v0)**	**Hetero nav. (4v0)**	**Phy decep. (2v1)**	**Pred-prey (2v2)**	**Keep-away (2v2)**
Partial observability	SPARSE[1]	139.07 ± 13.63	284.42 ± 12.83	93.60 ± 8.61	-4.72 ± 2.4	4.58 ± 3.27
	CURIO$^{2}_{\text{self}}$	133.93 ± 7.66	286.22 ± 9.97	68.80 ± 7.93	-6.50 ± 2.18	11.88 ± 2.88
	CURIO$^{3}_{\text{team}}$	125.42 ± 11.95	262.28 ± 22.59	85.31 ± 11.93	-3.57 ± 1.75	9.54 ± 5.04
	ALIGN$_{\text{self}}$	$\mathbf{155.88 \pm 5.11}$	292.34 ± 9.24	69.91 ± 4.51	-7.58 ± 2.55	12.84 ± 4.29
	ALIGN$_{\text{team}}$	141.04 ± 8.04	$\mathbf{311.67 \pm 10.88}$	$\mathbf{101.72 \pm 6.31}$	-7.69 ± 2.69	2.96 ± 4.03
	ALIGN$_{\text{adv}}$	—	—	92.20 ± 4.23	$\mathbf{-2.51 \pm 1.70}$	$\mathbf{19.46 \pm 5.05}$
Full observability	SPARSE[1]	154.00 ± 10.51	274.75 ± 19.74	82.97 ± 12.23	-10.48 ± 4.20	$\mathbf{4.95 \pm 2.96}$
	CURIO$_{\text{self}}$	154.71 ± 8.00	268.85 ± 15.61	$\mathbf{100.66 \pm 15.14}$	-8.74 ± 4.62	-2.00 ± 1.24
	ALIGN$_{\text{self}}$	$\mathbf{161.70 \pm 4.52}$	$\mathbf{280.16 \pm 17.12}$	87.50 ± 15.40	$\mathbf{-5.60 \pm 2.60}$	0.40 ± 1.92

[1] (Lowe et al., 2017),[2] (Stadie et al., 2015),[3] (Iqbal & Sha, 2020)

curiosity-driven intrinsic rewards), a more effective multi-agent strategy is for each agent to move to one goal and expect its teammates to explore other goals (incentivized by alignment). We hypothesize that our results arise because today's multi-agent task state space requires significantly less exploration than those used for single-agent (e.g. Atari games).

Alignment with decentralized training outperforms sparse and curiosity rewards for *competitive* tasks. In competitive tasks under partial observability, at least one of the ALIGN variants outperforms SPARSE and CURIO. These results suggest different tasks might require different alignment strategies. For example, in *Keep-away* under partial observability, the ALIGN$_{\text{adv}}$ strategy incentivizes agents to act unpredictably to adversaries; they learn to move away from the adversaries' receptive fields and then quickly change course to reach the goal undetected by adversaries.

Decentralized alignment improves *cooperative* tasks under full observability. The results for full observation are reported in Table 5.1 (bottom panel). The agents observe the entire state

Table 5.2: We report the mean test episode extrinsic rewards and standard errors of *decentralized* methods with different intrinsic rewards in *scaled* environments. Under partial observability, both ALIGN$_{\text{self,team}}$ outperform SPARSE in cooperative tasks. One of ALIGN$_{\text{self,team,adv}}$ always achieves the best performance in competitive tasks. With full observability, ALIGN$_{\text{self}}$ beats SPARSE across all tasks.

		Cooperative		**Competitive**		
Task (Agt # vs. Adv #)		**Coop nav. (5v0)**	**Hetero nav. (6v0)**	**Phy decep. (4v2)**	**Pred-prey (4v4)**	**Keep-away (4v4)**
Partial observability	SPARSE[1]	459.92 ± 22.44	616.62 ± 25.30	166.89 ± 27.72	-28.75 ± 7.3	0.75 ± 1.82
	ALIGN$_{\text{self}}$	$\mathbf{498.24 \pm 9.77}$	$\mathbf{646.70 \pm 23.25}$	137.38 ± 30.00	$\mathbf{-9.14 \pm 5.57}$	9.83 ± 11.22
	ALIGN$_{\text{team}}$	488.83 ± 20.82	638.74 ± 28.93	$\mathbf{186.83 \pm 21.92}$	-20.4 ± 5.93	2.07 ± 4.55
	ALIGN$_{\text{adv}}$	—	—	182.61 ± 17.63	-21.37 ± 7.02	$\mathbf{11.29 \pm 9.02}$
Full observability	SPARSE[1]	523.71 ± 34.56	533.67 ± 23.28	262.93 ± 31.88	-64.35 ± 8.71	-2.31 ± 2.45
	ALIGN$_{\text{self}}$	$\mathbf{545.35 \pm 19.28}$	$\mathbf{547.27 \pm 18.73}$	$\mathbf{285.64 \pm 16.08}$	$\mathbf{-45.02 \pm 7.46}$	$\mathbf{-2.01 \pm 3.53}$

[1] (Lowe et al., 2017)

Table 5.3: We report the mean test episode extrinsic rewards and standard errors of *centralized* algorithms trained with different intrinsic rewards under partial observability. In cooperative tasks, either ALIGN$_{\text{self}}$ or ALIGN$_{\text{team}}$ outperforms all other intrinsic reward baselines. Among competitive tasks, ALIGN$_{\text{self,team,adv}}$ all beat the other baselines in *Predator-prey (2v2)* but not *Physical deception (2v2)* or *Keep-away (2v2)*, suggesting that alignment might not provide additional useful signals to centralized algorithms in competitive settings.

		Cooperative		**Competitive**		
Task (Agt # vs. Adv #)		**Coop nav. (3v0)**	**Hetero nav. (4v0)**	**Phy decep. (2v1)**	**Pred-prey (2v2)**	**Keep-away (2v2)**
Partial observability	SPARSE[1]	113.25 ± 8.10	178.62 ± 9.62	$\mathbf{117.45 \pm 10.63}$	-1.96 ± 1.45	$\mathbf{35.79 \pm 14.93}$
	CURIO$^{2}_{\text{self}}$	128.77 ± 7.70	190.30 ± 7.73	111.08 ± 10.09	-1.63 ± 1.27	13.94 ± 12.56
	CURIO$^{3}_{\text{team}}$	114.13 ± 11.84	189.80 ± 11.81	114.32 ± 5.46	-3.04 ± 1.09	6.01 ± 3.36
	ALIGN$_{\text{self}}$	$\mathbf{137.14 \pm 3.63}$	169.58 ± 14.99	93.27 ± 3.70	-0.41 ± 0.28	22.77 ± 9.91
	ALIGN$_{\text{team}}$	119.10 ± 10.89	$\mathbf{210.81 \pm 9.70}$	96.49 ± 6.46	-0.92 ± 0.72	24.94 ± 12.58
	ALIGN$_{\text{adv}}$	—	—	102.37 ± 6.98	$\mathbf{-0.13 \pm 0.03}$	8.70 ± 4.44

[1] (Lowe et al., 2017),[2] (Stadie et al., 2015),[3] (Iqbal & Sha, 2020)

space rather than their partial field of view, which means that ALIGN$_{\text{self}}$ is equivalent to ALIGN$_{\text{team}}$, as all inputs to the dynamics model are the same. The results indicate that alignment is more useful than SPARSE and CURIO$_{\text{self}}$ on collaborative tasks; this too suggests that being predictable is more beneficial than being more exploratory.

Decentralized alignment doesn't always improve *competitive* tasks under full observability. By contrast, we observe little to no benefit with alignment on competitive tasks. This

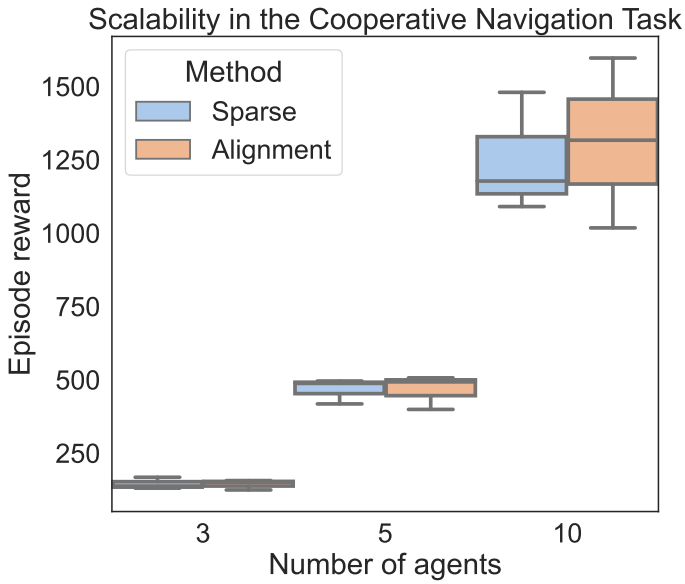

Figure 5.1: Decentralized alignment achieves consistent gains compared against SPARSE when the number of agents increases in the Cooperative Navigation task.

intuitively makes sense: if an agent is trained to be more predictable and the adversaries can observe the entire state space, the adversaries benefit. They can better predict where the agent will move towards and outsmart it.

Alignment scales to more agents. Our experiments demonstrate consistent gains from decentralized alignment compared against SPARSE when the number of agents increases (Figure 5.1). This is true across all cooperative and competitive tasks, showing that our decentralized alignment scales well, mitigating the limitations of centralized training when the number of agents increases (Liu et al., 2020).

Alignment doesn't always improve *centralized* training. In Table 5.3, we observe that $\text{ALIGN}_{\text{self}}$'s and $\text{ALIGN}_{\text{team}}$'s performance is higher for both cooperative tasks but for only one of the competitive tasks: *Predator-prey*. We hypothesize these results arise because centralized training, with access to all agents' states and actions, can incentivize an agent for actions that are counter to

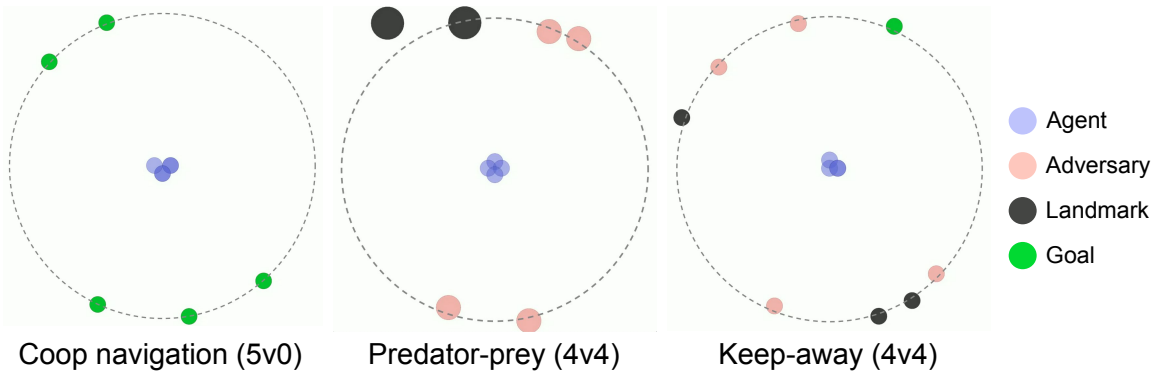

Figure 5.2: We visualize the symmetry-breaking setups in three example tasks. In *Coop navigation (5v0)*, 5 agents are initialized at the center and equidistant to the 5 goals. In *Predator-prey (4v4)*, 4 agents are initialized in the middle and equidistant to the 4 adversaries; the landmarks are placed randomly. In *Keep-away*, 4 agents are placed at the center, and the same distance from all 4 adversaries and 4 landmarks, where one of them is the goal.

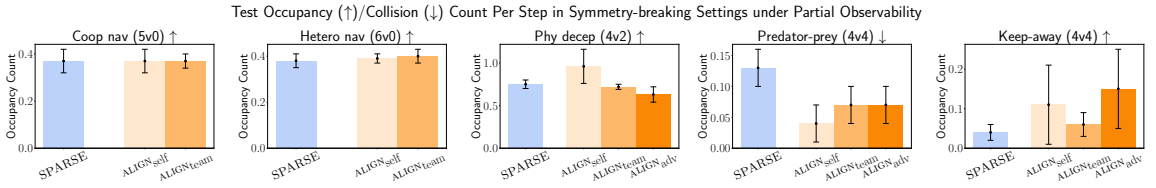

Figure 5.3: We plot the average test occupancy/collision count per step of decentralized algorithms in symmetry-breaking settings under partial observability. We find that our ALIGN intrinsic reward consistently beats the SPARSE baseline in cooperative tasks with symmetry-breaking initializations. In competitive tasks, one but not all of our ALIGN variants always surpasses SPARSE, suggesting the need for different alignment strategies when adversaries play different roles in different tasks.

alignment's local context.

5.1.1 Investigating how alignment reward helps

We further investigate how alignment improves multi-agent training through three additional evaluation setups.

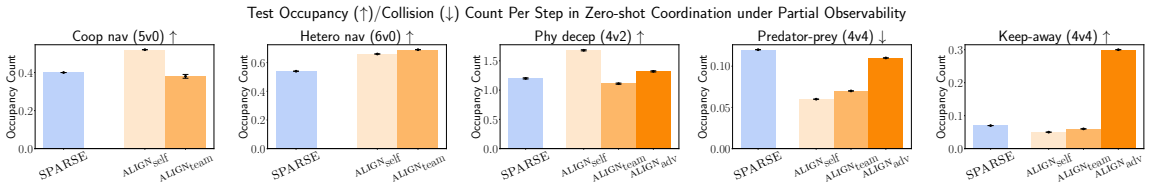

Figure 5.4: We sample agents from different decentralized training runs and evaluate their zero-shot performance under partial observability. We report the average test occupancy/collision count per step. We find that $\text{ALIGN}_{\text{self,team}}$ both outperform SPARSE in *Heterogenous navigation (6v0)*, and that $\text{ALIGN}_{\text{self,team,adv}}$ all improve on *Predator-prey (4v4)*. In *Cooperative navigation (5v0)*, *Physical deception (4v2)*, and *Keep-away (4v4)*, one of $\text{ALIGN}_{\text{self,team,adv}}$ achieves the best performance.

Alignment helps agents divide sub-tasks. A core challenge in multi-agent collaboration is efficient task division wang2020cooks. Here, we test whether alignment improves sub-task allocation; we initialize agents in states without an optimal sub-task allocation, necessitating symmetry-breaking (Hu et al., 2020). Figure 5.2 illustrates symmetry-breaking setups: In cooperative navigation, when agents are initialized equidistant to all the goal locations, there isn't an optimal allocation of agents to goals.

We find that both $\text{ALIGN}_{\text{self}}$ and $\text{ALIGN}_{\text{team}}$ achieve better performance than SPARSE on collaboration tasks, and at least one ALIGN variant surparsses SPARSE on competitive ones (Figure 5.3). Upon a qualitative evaluation of cooperative navigation, we observe that, with alignment, agents are able to predict which goals will be covered by their collaborators and move towards their allocated one. Without alignment, agents often move towards the same goal.

Alignment helps agents generalize to new partners. Another core challenge of multi-agent training algorithms is disjoint training, where agents are tested to collaborate with new partners they haven't been trained with. Disjoint training holds the promise of enabling multi-agent collaboration with humans partners. We study whether alignment enables better zero-shot coordination. New partners are sampled from other training runs with different seeds and the team is evaluated using the same metrics as before. We conduct this investigation on decentralized agents under partial

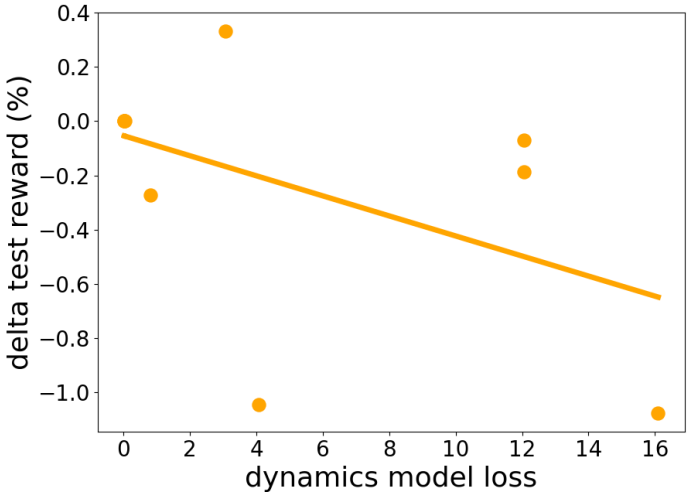

Figure 5.5: Test performance decreases with dynamics model loss ($R^2 = 0.242$), implying that alignment requires an accurate dynamics model.

observability and report results in Figure 5.4. We observe that all ALIGN strategies enable better performance than SPARSE when evaluated with new agents in *Heterogenous navigation* and *Predator-prey*, and at least one of $\text{ALIGN}_{\text{self,team,adv}}$ performs the best in the other tasks. These results suggest that alignment results in better zero-shot coordination with new partners sampled from separate training runs.

Accuracy of the dynamics model affects alignment. Third, we investigate the role of the dynamics model in calculating the intrinsic rewards. Since alignment uses a dynamics model to calculate rewards, we test whether an inaccurate model misleads agents towards unaligned behaviors. We trained agents on noisy dynamics models by adding Gaussian noise $\epsilon \sim \mathcal{N}(0, \sigma)$. We picked $\sigma \in [0.5, 1.0, 2.0]$ and ran experiments on one cooperative and one competitive task. Figure 5.5 plots the *final* dynamics loss against the reward change from the noiseless run with 8 data points (4 from each task). As the dynamics model degrades, we observe that the task performance also drops. This study identifies the importance of an accurate dynamics model, suggesting that alignment should only be used in environments where an accurate dynamics model can be learned.

Table 5.4: We report the mean average episode rewards and standard errors of the algorithms in the *Academy 3vs1 with keeper* task.

Method	Mean avg. episode reward
SPARSE[1]	0.020 ± 0.004
$\text{ALIGN}_{\text{team}}$	$\mathbf{0.026 \pm 0.001}$
$\text{ALIGN}_{\text{adv}}$	0.023 ± 0.001

[1] (Kurach et al., 2019)

5.2 Google Research football

The experiments on Google Research football are still going on. Preliminary results in Table 5.4 show that $\text{ALIGN}_{\text{team}}$ with the soft actor-critic algorithm achieves a higher mean average episode reward than SPARSE. This suggests that alignment is helpful even in the Google Research football environment, which has a more complex action space. My next step is to collect and analyze the results of the other ALIGN variants and all the CURIO baselines.

Chapter 6

Limitations and future directions

Our results indicate that alignment is a more useful strategy than curiosity across collaborative and competitive multi-agent tasks, and that aligned agents generalize better to new partners. However, these findings are limited to the multi-agent particle environment and the simpler tasks in the Google Research football environment, which have a smaller action space than most ecologically valid scenarios. Language, motion, and human gesture are all combinatorially vast; in such action spaces, alignment might develop social dynamics that hinder non-optimal multi-agent behaviors. Similarly, photorealistic environments have a larger state space, where teams perform common household activities or drive together in crowded cities (Srivastava et al., 2021). To better understand the utility of alignment versus curiosity, future work should develop new multi-agent environments that demand exploration complexity and where both curiosity and alignment would be necessary for collaboration. For example, in a search and rescue task where a single agent is unable to carry the injured, curiosity would encourage "search" while alignment would speed up "rescue".

Enabling multi-agent training without centralized training could open up future opportunities to train and evaluate multi-agent algorithms in existing human environments. Agents with interpretable

actions can induce more faithful human mental models, improving human-AI interaction; however, predictability does not imply legibility (Dragan, Lee, & Srinivasa, 2013). Future work could explore the role of legibility in designing intrinsic rewards.

Alignment, generating predictable and consistent behaviors, can be viewed as a self-supervised loss similar to the ones recently used in computer vision (Chen, Kornblith, Norouzi, & Hinton, 2020). Future work could study the role of self-supervised multi-agent objectives, which might similarly lead to emergent visual, linguistic, and social features.

Chapter 7

Conclusion

In this thesis, I introduce alignment, a simple, task-agnostic intrinsic reward for multi-agent systems inspired from the self-organizing principle in Zoology (Chapters 1 and 2). My formulation of alignment rewards agents when they act predictably to their teammates and unpredictably to their adversaries (Chapter 3). Through extensive experiments in the multi-agent particle and Google Research football environments (Chapter 4), I found that decentralized alignment improves multi-agent performance across cooperative and competitive tasks, partial and full observability, and different team sizes (Chapter 5).

I also sought to understand why and how alignment helps across various multi-agent tasks with additional experiments. The results show that alignment helps agents break symmetries in collaborative tasks, thus enabling more effective coordination. Further, the zero-shot experiments demonstrate that alignment also helps agents generalize to new partners. Although these findings are limited to the simple multi-agent environment, they are exciting to me and motivate me to keep exploring alignment-driven strategies for multi-agent coordination.

While I have reached the end of this thesis and submitted it to the 39th International

Conference on Machine Learning (ICML 2022), it is only the start of my research journey. I believe effective strategies for multi-agent coordination are a stepping stone for Human-AI collaboration - where my heart belongs. I am beyond grateful and hopeful to bring this research experience with me and begin my deeper exploration on ways to improve on Human-AI collaboration in the near future.

Appendix A

Appendix

A.1 Emergent behavior visualization

We upload video examples of agents' emergent behaviors in both cooperative and competitive tasks to a Google drive accessible via this link: `https://drive.google.com/drive/folders/1gJvc0-6HXvl_vS43ZfMA1RbYcbwubQXM?usp=sharing`.

A.2 Hyperparameters

Table A.1 presents the hyperparameters used to train the algorithms in the multi-agent particle environment.

A.3 Additional results

We include 18 tables of additional results that quantify the agents' performance beyond extrinsic reward.

Table A.1: Model and training hyperparameters

Parameter	Value
SAC actor model architecture	FC layers [128,128]
SAC critic model architecture	FC layers [128,128]
World model architecture	FC layers [128,128]
Replay buffer size	1,000,000
Batch size	1,024
Actor learning rate	0.001
Critic learning rate	0.001
Discount factor gamma	0.95
SAC soft update coefficient	0.01
SAC policy entropy regularization coefficient	0.1

Table A.2 and A.3 report two sets of metrics of *decentralized* methods trained with different intrinsic rewards in both partially and fully observable settings. Table A.2 reports the average number of agent-target occupancies per step (or, we can understand it as: on average, the total number of goals occupied by the agents at any given timestep throughout an episode) and agent-adversary collisions in *Predator-prey*. Higher scores are better for the occupancy metric, and lower scores are better for collision. Table A.3 reports the average minimum agent-to-target distance and agent-to-adversary distance. Agent-to-target distances measure the closest distance an agent achieves to the target location; lower scores are better on this metric. Agent-to-adversary distances measure the closest distance an adversary gets to a good agent; higher scores are better on this metric. Note that these distance-based metrics are not included in the reward functions, and should mainly be used to make comparisons in the case where primary metrics (reward and occupancy/collision count) have the same values.

Table A.4 and A.5 report the same metrics as A.2 and A.3 respectively, but in *scaled* environments with more agents.

Table A.6 reports the test mean episode rewards of *centralized* algorithms with different intrinsic rewards under full observability. Table A.7 and A.8 show the other two sets of metrics (occupancy/collision count and agent-target/agent-adversary distance) of *centralized* algorithms.

Table A.2: The average test occupancy/collision count per step and standard errors of *decentralized* methods with different intrinsic rewards under partial and full observability. Higher scores are better for the occupancy metric (↑), and lower scores are better for the collision metric (↓).

		Cooperative		**Competitive**		
Task (Agt # vs. Adv #)		**Coop nav. (3v0) ↑**	**Hetero nav. (4v0) ↑**	**Phy decep. (2v1) ↑**	**Pred-prey (2v2) ↓**	**Keep-away (2v2) ↑**
Partial observability	SPARSE	0.46 ± 0.05	0.57 ± 0.01	0.98 ± 0.07	0.02 ± 0.01	0.07 ± 0.02
	CURIO$_{\text{self}}$	0.43 ± 0.03	0.60 ± 0.01	0.99 ± 0.03	0.02 ± 0.01	0.14 ± 0.02
	CURIO$_{\text{team}}$	0.42 ± 0.05	0.59 ± 0.01	0.95 ± 0.01	0.02 ± 0.01	0.10 ± 0.03
	ALIGN$_{\text{self}}$	$\mathbf{0.52 \pm 0.03}$	$\mathbf{0.61 \pm 0.01}$	0.95 ± 0.02	0.03 ± 0.01	0.10 ± 0.02
	ALIGN$_{\text{team}}$	0.44 ± 0.04	0.58 ± 0.02	0.99 ± 0.07	0.03 ± 0.01	0.07 ± 0.02
	ALIGN$_{\text{adv}}$	—	—	$\mathbf{1.00 \pm 0.06}$	$\mathbf{0.01 \pm 0.01}$	$\mathbf{0.15 \pm 0.03}$
Full observability	SPARSE	0.46 ± 0.11	0.57 ± 0.01	0.88 ± 0.09	0.03 ± 0.01	$\mathbf{0.06 \pm 0.02}$
	CURIO$_{\text{self}}$	$\mathbf{0.50 \pm 0.07}$	$\mathbf{0.59 \pm 0.02}$	$\mathbf{1.09 \pm 0.13}$	0.03 ± 0.01	0.02 ± 0.00
	ALIGN$_{\text{self}}$	0.48 ± 0.11	0.58 ± 0.02	0.83 ± 0.10	$\mathbf{0.02 \pm 0.01}$	0.04 ± 0.01

Table A.9, A.10, and A.11 contain the same metrics as A.6, A.7 and A.8 respectively, but in *scaled* environments.

Tables A.12, A.13 and A.14 report the test episode reward and additional metrics of *decentralized* algorithms in the *symmetry-breaking* experiments conducted under "Investigating how alignment reward helps". Table A.15, A.16 and A.17 report the same set of metrics but from experiments conducted in *scaled* and *symmetry-breaking* environments.

Finally, Tables A.18 and A.19 report the test episode reward values and secondary distance-based metrics for the zero-shot generalization experiments conducted under "Investigating how alignment reward helps". These experiments measure how well agents trained on different seeds generalized to new partners trained on other seeds.

Table A.3: The average test agent-to-target (agt-target) and agent-to-adversary (agt-adv) distances and standard errors of *decentralized* methods with different intrinsic rewards under partial and full observability. Lower scores are better for agt-target (↓), and higher scores are better for agt-adv (↑).

		Cooperative		**Competitive**		
Task (Agt # vs. Adv #)		**Coop nav. (3v0) ↓**	**Hetero nav. (4v0) ↓**	**Phy decep. (2v1) ↓**	**Pred-prey (2v2) ↑**	**Keep-away (2v2) ↓**
Partial observability	SPARSE	$\mathbf{0.30 \pm 0.02}$	0.23 ± 0.00	0.26 ± 0.01	1.45 ± 0.11	1.41 ± 0.07
	$\text{CURIO}_{\text{self}}$	0.32 ± 0.02	0.25 ± 0.01	0.25 ± 0.00	1.36 ± 0.06	1.14 ± 0.09
	$\text{CURIO}_{\text{team}}$	0.31 ± 0.01	0.25 ± 0.01	0.26 ± 0.00	1.48 ± 0.13	1.31 ± 0.10
	$\text{ALIGN}_{\text{self}}$	0.33 ± 0.03	0.25 ± 0.01	0.26 ± 0.00	1.39 ± 0.12	1.26 ± 0.09
	$\text{ALIGN}_{\text{team}}$	0.33 ± 0.02	0.23 ± 0.01	0.25 ± 0.01	1.38 ± 0.13	1.38 ± 0.09
	$\text{ALIGN}_{\text{adv}}$	—	—	0.25 ± 0.01	1.54 ± 0.08	1.14 ± 0.09
Full observability	SPARSE	0.32 ± 0.09	0.23 ± 0.00	0.26 ± 0.01	1.23 ± 0.12	1.27 ± 0.09
	$\text{CURIO}_{\text{self}}$	$\mathbf{0.28 \pm 0.04}$	0.22 ± 0.01	0.23 ± 0.01	1.37 ± 0.15	1.53 ± 0.03
	$\text{ALIGN}_{\text{self}}$	0.30 ± 0.07	0.23 ± 0.01	0.27 ± 0.01	1.40 ± 0.13	1.41 ± 0.10

Table A.4: The average test occupancy/collision count per step and standard errors of *decentralized* methods with different intrinsic rewards in *scaled* environments under partial and full observability. Higher scores are better for the occupancy metric (↑), and lower scores are better for the collision metric (↓).

		Cooperative		**Competitive**		
Task (Agt # vs. Adv #)		**Coop nav. (5v0) ↑**	**Hetero nav. (6v0) ↑**	**Phy decep. (4v2) ↑**	**Pred-prey (4v4) ↓**	**Keep-away (4v4) ↑**
Partial observability	SPARSE	0.50 ± 0.04	0.46 ± 0.08	1.20 ± 0.10	0.11 ± 0.02	0.08 ± 0.02
	$\text{ALIGN}_{\text{self}}$	0.49 ± 0.03	$\mathbf{0.55 \pm 0.11}$	$\mathbf{1.30 \pm 0.23}$	0.04 ± 0.02	0.14 ± 0.08
	$\text{ALIGN}_{\text{team}}$	$\mathbf{0.56 \pm 0.04}$	0.56 ± 0.00	1.21 ± 0.09	$\mathbf{0.08 \pm 0.02}$	0.10 ± 0.02
	$\text{ALIGN}_{\text{adv}}$	—	—	1.23 ± 0.10	$\mathbf{0.08 \pm 0.02}$	$\mathbf{0.16 \pm 0.07}$
Full observability	SPARSE	0.52 ± 0.11	0.46 ± 0.08	0.99 ± 0.09	0.21 ± 0.01	0.03 ± 0.00
	$\text{ALIGN}_{\text{self}}$	$\mathbf{0.55 \pm 0.11}$	$\mathbf{0.56 \pm 0.00}$	$\mathbf{1.04 \pm 0.07}$	$\mathbf{0.15 \pm 0.03}$	$\mathbf{0.06 \pm 0.02}$

Table A.5: The average test agent-to-target (agt-target) and agent-to-adversary (agt-adv) distances and standard errors of *decentralized* methods with different intrinsic rewards in *scaled* environments under partial and full observability. Lower scores are better for agt-target (↓), and higher scores are better for agt-adv (↑).

		Cooperative		**Competitive**		
Task (Agt # vs. Adv #)		**Coop nav. (5v0) ↓**	**Hetero nav. (6v0) ↓**	**Phy decep. (4v2) ↓**	**Pred-prey (4v4) ↑**	**Keep-away (4v4) ↓**
Partial observability	SPARSE	0.22 ± 0.01	0.27 ± 0.05	0.23 ± 0.02	2.03 ± 0.15	2.97 ± 0.17
	$\text{ALIGN}_{\text{self}}$	0.29 ± 0.03	0.19 ± 0.00	0.24 ± 0.02	2.39 ± 0.11	2.97 ± 0.30
	$\text{ALIGN}_{\text{team}}$	0.23 ± 0.04	0.21 ± 0.00	0.23 ± 0.01	2.16 ± 0.12	2.88 ± 0.19
	$\text{ALIGN}_{\text{adv}}$	—	—	0.22 ± 0.01	2.12 ± 0.16	2.66 ± 0.23
Full observability	SPARSE	0.23 ± 0.06	0.27 ± 0.05	0.21 ± 0.02	1.64 ± 0.02	3.28 ± 0.02
	$\text{ALIGN}_{\text{self}}$	0.20 ± 0.04	0.21 ± 0.00	0.21 ± 0.01	1.82 ± 0.10	2.97 ± 0.17

Table A.6: We report the mean test episode extrinsic rewards and standard errors of *centralized* methods with different intrinsic rewards under full observability.

		Cooperative		**Competitive**		
Task (Agt # vs. Adv #)		**Coop nav. (3v0)**	**Hetero nav. (4v0)**	**Phy decep. (2v1)**	**Pred-prey (2v2)**	**Keep-away (2v2)**
Full observability	SPARSE	106.02 ± 20.95	123.17 ± 18.77	$\mathbf{130.90 \pm 6.59}$	-1.90 ± 1.61	12.49 ± 9.83
	$\text{CURIO}_{\text{self}}$	86.52 ± 16.02	108.84 ± 6.89	107.84 ± 13.67	$\mathbf{-1.69 \pm 0.60}$	$\mathbf{23.70 \pm 12.95}$
	$\text{ALIGN}_{\text{self}}$	$\mathbf{120.47 \pm 12.26}$	$\mathbf{134.30 \pm 5.84}$	105.74 ± 9.72	-2.37 ± 1.39	22.92 ± 7.00

Table A.7: The average test occupancy/collision count per step and standard errors of *centralized* methods with different intrinsic rewards under partial and full observability. Higher scores are better for the occupancy metric (↑), and lower scores are better for the collision metric (↓).

		Cooperative		**Competitive**		
Task (Agt # vs. Adv #)		**Coop nav. (3v0)** ↑	**Hetero nav. (4v0)** ↑	**Phy decep. (2v1)** ↑	**Pred-prey (2v2)** ↓	**Keep-away (2v2)** ↑
Partial observability	SPARSE	0.29 ± 0.10	0.50 ± 0.03	$\mathbf{0.94 \pm 0.06}$	$\mathbf{0.00 \pm 0.00}$	$\mathbf{0.36 \pm 0.12}$
	$\text{CURIO}_{\text{self}}$	0.28 ± 0.09	0.47 ± 0.03	$\mathbf{0.94 \pm 0.03}$	0.01 ± 0.00	0.17 ± 0.10
	$\text{CURIO}_{\text{team}}$	$\mathbf{0.33 \pm 0.10}$	0.47 ± 0.04	0.92 ± 0.01	0.01 ± 0.00	0.08 ± 0.03
	$\text{ALIGN}_{\text{self}}$	0.21 ± 0.10	0.50 ± 0.01	0.92 ± 0.02	$\mathbf{0.00 \pm 0.00}$	0.25 ± 0.08
	$\text{ALIGN}_{\text{team}}$	0.23 ± 0.09	$\mathbf{0.55 \pm 0.02}$	0.90 ± 0.07	0.01 ± 0.00	0.24 ± 0.11
	$\text{ALIGN}_{\text{adv}}$	—	—	$\mathbf{0.94 \pm 0.04}$	$\mathbf{0.00 \pm 0.00}$	0.10 ± 0.05
Full observability	SPARSE	$\mathbf{0.34 \pm 0.10}$	0.33 ± 0.07	$\mathbf{0.88 \pm 0.04}$	$\mathbf{0.01 \pm 0.00}$	0.26 ± 0.11
	$\text{CURIO}_{\text{self}}$	0.30 ± 0.07	0.32 ± 0.05	0.82 ± 0.02	$\mathbf{0.01 \pm 0.01}$	$\mathbf{0.33 \pm 0.16}$
	$\text{ALIGN}_{\text{self}}$	0.30 ± 0.11	$\mathbf{0.40 \pm 0.04}$	$\mathbf{0.88 \pm 0.05}$	$\mathbf{0.01 \pm 0.01}$	0.30 ± 0.07

Table A.8: The average test agent-to-target (agt-target) and agent-to-adversary (agt-adv) distances and standard errors of *centralized* methods with different intrinsic rewards under partial and full observability. Lower scores are better for agt-target (↓), and higher scores are better for agt-adv (↑).

		Cooperative		**Competitive**		
Task (Agt # vs. Adv #)		**Coop nav. (3v0)** ↓	**Hetero nav. (4v0)** ↓	**Phy decep. (2v1)** ↓	**Pred-prey (2v2)** ↑	**Keep-away (2v2)** ↓
Partial observability	SPARSE	0.42 ± 0.05	0.29 ± 0.02	0.27 ± 0.01	1.54 ± 0.02	1.38 ± 0.13
	$\text{CURIO}_{\text{self}}$	0.42 ± 0.05	0.29 ± 0.01	0.27 ± 0.01	1.46 ± 0.05	1.40 ± 0.13
	$\text{CURIO}_{\text{team}}$	0.41 ± 0.06	0.29 ± 0.02	0.28 ± 0.01	1.49 ± 0.04	1.43 ± 0.14
	$\text{ALIGN}_{\text{self}}$	0.50 ± 0.07	0.29 ± 0.01	0.27 ± 0.01	1.60 ± 0.04	1.26 ± 0.12
	$\text{ALIGN}_{\text{team}}$	0.45 ± 0.05	0.27 ± 0.01	0.28 ± 0.01	1.52 ± 0.04	1.35 ± 0.14
	$\text{ALIGN}_{\text{adv}}$	—	—	0.28 ± 0.01	1.55 ± 0.03	1.45 ± 0.10
Full observability	SPARSE	0.38 ± 0.07	0.34 ± 0.04	0.25 ± 0.00	1.59 ± 0.06	1.43 ± 0.09
	$\text{CURIO}_{\text{self}}$	0.36 ± 0.05	0.32 ± 0.02	0.25 ± 0.01	1.53 ± 0.09	1.08 ± 0.15
	$\text{ALIGN}_{\text{self}}$	0.43 ± 0.08	0.30 ± 0.02	0.25 ± 0.00	1.51 ± 0.08	1.18 ± 0.15

Table A.9: We report the mean test episode extrinsic rewards and standard errors of *centralized* methods with different intrinsic rewards in *scaled* environments.

		Cooperative		**Competitive**		
Task (Agt # vs. Adv #)		**Coop nav. (5v0)**	**Hetero nav. (6v0)**	**Phy decep. (4v2)**	**Pred-prey (4v4)**	**Keep-away (4v4)**
Partial observability	SPARSE	100.63 ± 19.36	346.16 ± 18.95	-38.99 ± 16.18	$\mathbf{-17.33 \pm 4.29}$	-2.50 ± 2.64
	ALIGN$_{\text{self}}$	$\mathbf{112.15 \pm 19.69}$	$\mathbf{375.21 \pm 26.10}$	13.71 ± 29.53	-20.12 ± 1.42	-4.68 ± 1.21
	ALIGN$_{\text{team}}$	97.93 ± 25.23	372.41 ± 44.28	$\mathbf{60.07 \pm 13.26}$	-27.87 ± 0.99	$\mathbf{1.72 \pm 3.79}$
	ALIGN$_{\text{adv}}$	—	—	21.67 ± 48.17	-17.68 ± 5.59	-4.92 ± 1.81
Full observability	SPARSE	50.60 ± 13.10	$\mathbf{153.76 \pm 19.81}$	97.32 ± 17.95	-38.25 ± 5.06	$\mathbf{-3.39 \pm 2.77}$
	ALIGN$_{\text{self}}$	$\mathbf{186.55 \pm 53.15}$	127.97 ± 13.02	$\mathbf{103.46 \pm 28.91}$	$\mathbf{-23.29 \pm 5.00}$	-4.90 ± 0.67

Table A.10: The average test occupancy/collision count per step and standard errors of *centralized* methods with different intrinsic rewards in *scaled* environments under partial and full observability. Higher scores are better for the occupancy metric (↑), and lower scores are better for the collision metric (↓).

		Cooperative		**Competitive**		
Task (Agt # vs. Adv #)		**Coop nav. (5v0)** ↑	**Hetero nav. (6v0)** ↑	**Phy decep. (4v2)** ↑	**Pred-prey (4v4)** ↓	**Keep-away (4v4)** ↑
Partial observability	SPARSE	0.11 ± 0.02	0.29 ± 0.06	0.56 ± 0.05	0.06 ± 0.02	0.07 ± 0.02
	ALIGN$_{\text{self}}$	0.23 ± 0.09	$\mathbf{0.33 \pm 0.04}$	0.56 ± 0.06	0.08 ± 0.00	0.05 ± 0.00
	ALIGN$_{\text{team}}$	$\mathbf{0.27 \pm 0.10}$	$\mathbf{0.33 \pm 0.05}$	0.50 ± 0.08	0.09 ± 0.00	$\mathbf{0.09 \pm 0.03}$
	ALIGN$_{\text{adv}}$	—	—	$\mathbf{0.60 \pm 0.09}$	$\mathbf{0.05 \pm 0.02}$	0.05 ± 0.01
Full observability	SPARSE	0.10 ± 0.04	$\mathbf{0.16 \pm 0.04}$	0.50 ± 0.03	0.12 ± 0.01	$\mathbf{0.06 \pm 0.01}$
	ALIGN$_{\text{self}}$	$\mathbf{0.16 \pm 0.09}$	0.11 ± 0.00	$\mathbf{0.55 \pm 0.02}$	$\mathbf{0.10 \pm 0.02}$	0.04 ± 0.01

Table A.11: The average test agent-to-target (agt-target) and agent-to-adversary (agt-adv) distances and standard errors of *centralized* methods with different intrinsic rewards in *scaled* environments under partial and full observability. Lower scores are better for agt-target (↓), and higher scores are better for agt-adv (↑).

		Cooperative		**Competitive**		
Task (Agt # vs. Adv #)		**Coop nav. (5v0)** ↓	**Hetero nav. (6v0)** ↓	**Phy decep. (4v2)** ↓	**Pred-prey (4v4)** ↑	**Keep-away (4v4)** ↓
Partial observability	SPARSE	0.33 ± 0.01	0.29 ± 0.02	0.34 ± 0.02	2.27 ± 0.08	3.12 ± 0.18
	ALIGN$_{\text{self}}$	0.32 ± 0.04	0.28 ± 0.01	0.36 ± 0.02	2.32 ± 0.09	3.25 ± 0.05
	ALIGN$_{\text{team}}$	0.30 ± 0.04	0.28 ± 0.01	0.37 ± 0.03	2.29 ± 0.08	3.01 ± 0.20
	ALIGN$_{\text{adv}}$	—	—	0.33 ± 0.04	2.44 ± 0.08	3.23 ± 0.13
Full observability	SPARSE	0.37 ± 0.03	0.37 ± 0.02	0.29 ± 0.02	1.98 ± 0.07	3.13 ± 0.17
	ALIGN$_{\text{self}}$	0.36 ± 0.04	0.39 ± 0.00	0.27 ± 0.01	1.98 ± 0.08	3.25 ± 0.12

Table A.12: We report the mean test episode extrinsic rewards and standard errors of *decentralized* methods with different intrinsic rewards in *symmetry-breaking* settings under partial and full observability.

		Cooperative		**Competitive**		
Task (Agt # vs. Adv #)		**Coop nav. (3v0)**	**Hetero nav. (4v0)**	**Phy decep. (2v1)**	**Pred-prey (2v2)**	**Keep-away (2v2)**
Partial observability	SPARSE	97.45 ± 10.49	184.18 ± 7.63	59.39 ± 21.10	-1.89 ± 1.69	3.85 ± 4.25
	CURIO$_{\text{self}}$	85.23 ± 10.88	184.07 ± 9.99	54.17 ± 27.40	-2.86 ± 1.19	$\mathbf{19.57 \pm 4.92}$
	CURIO$_{\text{team}}$	81.50 ± 15.78	141.78 ± 20.04	41.12 ± 13.37	-2.80 ± 1.91	10.21 ± 6.34
	ALIGN$_{\text{self}}$	$\mathbf{110.29 \pm 9.67}$	176.98 ± 6.38	$\mathbf{98.90 \pm 17.71}$	-4.00 ± 2.14	9.47 ± 3.99
	ALIGN$_{\text{team}}$	92.41 ± 10.70	$\mathbf{187.42 \pm 11.29}$	74.06 ± 21.58	-2.00 ± 1.39	3.32 ± 3.04
	ALIGN$_{\text{adv}}$	—	—	87.55 ± 15.35	$\mathbf{-1.40 \pm 1.25}$	13.77 ± 3.58
Full observability	SPARSE	150.42 ± 15.18	250.41 ± 14.23	69.06 ± 14.06	-7.62 ± 3.50	$\mathbf{3.50 \pm 4.00}$
	CURIO$_{\text{self}}$	149.48 ± 9.42	241.69 ± 19.58	52.69 ± 17.97	-10.40 ± 6.33	-1.10 ± 0.59
	ALIGN$_{\text{self}}$	$\mathbf{152.08 \pm 6.68}$	$\mathbf{275.69 \pm 7.49}$	$\mathbf{75.79 \pm 24.54}$	$\mathbf{-4.44 \pm 2.05}$	0.96 ± 3.14

Table A.13: The average test occupancy/collision count per step and standard errors of *decentralized* methods with different intrinsic rewards in *symmetry-breaking* settings under partial and full observability. Higher scores are better for the occupancy metric (↑), and lower scores are better for the collision metric (↓).

		Cooperative		**Competitive**		
Task (Agt # vs. Adv #)		**Coop nav. (3v0)** ↑	**Hetero nav. (4v0)** ↑	**Phy decep. (2v1)** ↑	**Pred-prey (2v2)** ↓	**Keep-away (2v2)** ↑
Partial observability	SPARSE	0.26 ± 0.04	0.27 ± 0.02	0.67 ± 0.09	0.02 ± 0.01	0.04 ± 0.03
	CURIO$_{\text{self}}$	0.22 ± 0.01	0.28 ± 0.04	0.61 ± 0.06	0.02 ± 0.01	$\mathbf{0.15 \pm 0.04}$
	CURIO$_{\text{team}}$	0.26 ± 0.06	0.29 ± 0.02	0.65 ± 0.05	0.01 ± 0.01	0.08 ± 0.04
	ALIGN$_{\text{self}}$	$\mathbf{0.29 \pm 0.05}$	$\mathbf{0.32 \pm 0.03}$	0.62 ± 0.02	0.02 ± 0.01	0.08 ± 0.03
	ALIGN$_{\text{team}}$	0.27 ± 0.04	0.27 ± 0.02	$\mathbf{0.72 \pm 0.10}$	0.02 ± 0.01	0.05 ± 0.04
	ALIGN$_{\text{adv}}$	—	—	0.68 ± 0.07	$\mathbf{0.00 \pm 0.00}$	0.14 ± 0.04
Full observability	SPARSE	0.45 ± 0.12	0.54 ± 0.01	0.89 ± 0.11	0.03 ± 0.01	$\mathbf{0.05 \pm 0.02}$
	CURIO$_{\text{self}}$	$\mathbf{0.48 \pm 0.08}$	0.54 ± 0.01	$\mathbf{1.13 \pm 0.14}$	0.04 ± 0.02	0.00 ± 0.00
	ALIGN$_{\text{self}}$	0.46 ± 0.11	0.54 ± 0.01	0.86 ± 0.12	$\mathbf{0.02 \pm 0.01}$	0.02 ± 0.02

Table A.14: The average test agent-to-target (agt-target) and agent-to-adversary (agt-adv) distances and standard errors of *decentralized* methods with different intrinsic rewards in *symmetry-breaking* settings under partial and full observability. Lower scores are better for agt-target (↓), and higher scores are better for agt-adv (↑).

		Cooperative		**Competitive**		
Task (Agt # vs. Adv #)		**Coop nav. (3v0)** ↓	**Hetero nav. (4v0)** ↓	**Phy decep. (2v1)** ↓	**Pred-prey (2v2)** ↑	**Keep-away (2v2)** ↓
Partial observability	SPARSE	0.53 ± 0.02	0.57 ± 0.02	0.35 ± 0.03	1.49 ± 0.14	1.57 ± 0.15
	CURIO$_{\text{self}}$	0.57 ± 0.04	0.55 ± 0.04	0.37 ± 0.02	1.29 ± 0.06	1.07 ± 0.17
	CURIO$_{\text{team}}$	0.53 ± 0.03	0.55 ± 0.03	0.35 ± 0.02	1.49 ± 0.13	1.37 ± 0.18
	ALIGN$_{\text{self}}$	0.68 ± 0.05	0.52 ± 0.03	0.34 ± 0.01	1.39 ± 0.13	1.26 ± 0.18
	ALIGN$_{\text{team}}$	0.55 ± 0.05	0.56 ± 0.02	0.31 ± 0.02	1.41 ± 0.10	1.53 ± 0.17
	ALIGN$_{\text{adv}}$	—	—	0.33 ± 0.04	1.61 ± 0.08	1.08 ± 0.15
Full observability	SPARSE	0.45 ± 0.12	0.29 ± 0.00	0.25 ± 0.01	1.28 ± 0.15	1.30 ± 0.15
	CURIO$_{\text{self}}$	0.33 ± 0.06	0.30 ± 0.01	0.22 ± 0.01	1.47 ± 0.17	1.71 ± 0.05
	ALIGN$_{\text{self}}$	0.46 ± 0.11	0.30 ± 0.00	0.25 ± 0.02	1.49 ± 0.16	1.53 ± 0.16

Table A.15: We report the mean test episode extrinsic rewards and standard errors of *decentralized* methods with different intrinsic rewards in *scaled* and *symmetry-breaking* settings.

		Cooperative		Competitive		
Task (Agt # vs. Adv #)		**Coop nav. (5v0)**	**Hetero nav. (6v0)**	**Phy decep. (4v2)**	**Pred-prey (4v4)**	**Keep-away (4v4)**
Partial observability	SPARSE	328.24 ± 24.17	405.08 ± 21.53	172.87 ± 32.43	-35.40 ± 8.63	1.37 ± 3.48
	ALIGN$_{\text{self}}$	328.24 ± 24.17	412.39 ± 12.63	129.07 ± 51.08	$\mathbf{-7.34 \pm 5.12}$	11.97 ± 13.30
	ALIGN$_{\text{team}}$	$\mathbf{354.14 \pm 19.53}$	$\mathbf{417.94 \pm 22.29}$	$\mathbf{184.21 \pm 23.16}$	-19.37 ± 6.44	4.05 ± 5.78
	ALIGN$_{\text{adv}}$	—	—	148.69 ± 31.79	-23.42 ± 8.32	$\mathbf{18.71 \pm 14.78}$
Full observability	SPARSE	466.17 ± 28.16	471.19 ± 16.23	$\mathbf{233.61 \pm 25.44}$	-39.24 ± 6.63	-5.10 ± 0.26
	ALIGN$_{\text{self}}$	$\mathbf{520.25 \pm 9.68}$	$\mathbf{510.18 \pm 25.71}$	222.31 ± 15.39	$\mathbf{-30.56 \pm 9.87}$	$\mathbf{-4.27 \pm 2.53}$

Table A.16: The average test occupancy/collision count per step and standard errors of *decentralized* methods with different intrinsic rewards in *scaled* and *symmetry-breaking* settings. under partial and full observability. Higher scores are better for the occupancy metric (↑), and lower scores are better for the collision metric (↓).

		Cooperative		Competitive		
Task (Agt # vs. Adv #)		**Coop nav. (5v0)** ↑	**Hetero nav. (6v0)** ↑	**Phy decep. (4v2)** ↑	**Pred-prey (4v4)** ↓	**Keep-away (4v4)** ↑
Partial observability	SPARSE	$\mathbf{0.37 \pm 0.05}$	0.38 ± 0.03	0.75 ± 0.04	0.13 ± 0.03	0.04 ± 0.02
	ALIGN$_{\text{self}}$	$\mathbf{0.37 \pm 0.05}$	0.39 ± 0.02	$\mathbf{0.96 \pm 0.20}$	$\mathbf{0.04 \pm 0.03}$	0.11 ± 0.10
	ALIGN$_{\text{team}}$	$\mathbf{0.37 \pm 0.03}$	$\mathbf{0.40 \pm 0.03}$	0.72 ± 0.03	0.07 ± 0.03	0.06 ± 0.03
	ALIGN$_{\text{adv}}$	—	—	0.63 ± 0.09	0.07 ± 0.03	$\mathbf{0.15 \pm 0.10}$
Full observability	SPARSE	0.52 ± 0.11	0.43 ± 0.09	0.86 ± 0.07	0.19 ± 0.02	0.00 ± 0.00
	ALIGN$_{\text{self}}$	$\mathbf{0.55 \pm 0.11}$	$\mathbf{0.55 \pm 0.00}$	$\mathbf{0.94 \pm 0.08}$	$\mathbf{0.12 \pm 0.03}$	$\mathbf{0.02 \pm 0.01}$

Table A.17: The average test agent-to-target (agt-target) and agent-to-adversary (agt-adv) distances and standard errors of *decentralized* methods with different intrinsic rewards in *scaled* and *symmetry-breaking* settings under partial and full observability. Lower scores are better for agt-target (↓), and higher scores are better for agt-adv (↑).

		Cooperative		Competitive		
Task (Agt # vs. Adv #)		**Coop nav. (5v0)** ↓	**Hetero nav. (6v0)** ↓	**Phy decep. (4v2)** ↓	**Pred-prey (4v4)** ↑	**Keep-away (4v4)** ↓
Partial observability	SPARSE	0.36 ± 0.01	0.42 ± 0.02	0.35 ± 0.01	2.04 ± 0.18	3.10 ± 0.29
	ALIGN$_{\text{self}}$	0.36 ± 0.01	0.41 ± 0.02	0.37 ± 0.04	2.52 ± 0.15	3.22 ± 0.42
	ALIGN$_{\text{team}}$	0.42 ± 0.03	0.41 ± 0.02	0.37 ± 0.01	2.25 ± 0.15	3.00 ± 0.33
	ALIGN$_{\text{adv}}$	—	—	0.41 ± 0.05	2.27 ± 0.16	2.62 ± 0.30
Full observability	SPARSE	0.29 ± 0.07	0.37 ± 0.06	0.26 ± 0.01	1.81 ± 0.04	3.69 ± 0.05
	ALIGN$_{\text{self}}$	0.26 ± 0.05	0.29 ± 0.00	0.27 ± 0.01	2.10 ± 0.12	3.24 ± 0.26

Table A.18: We sample agents from different *decentralized* training runs and evaluate their zero-shot performance in *scaled* environments under partial observability. We report the mean test episode extrinsic rewards and standard errors of decentralized methods with different intrinsic rewards.

		Cooperative		**Competitive**		
Task (Agt # vs. Adv #)		**Coop nav. (5v0)**	**Hetero nav. (6v0)**	**Phy decep. (4v2)**	**Pred-prey (4v4)**	**Keep-away (4v4)**
Partial observability	SPARSE	434.68 ± 6.42	561.16 ± 31.63	128.64 ± 17.31	-32.12 ± 3.63	-2.80 ± 2.91
	ALIGN$_{\text{self}}$	471.07 ± 5.00	676.01 ± 16.53	$\mathbf{248.16 \pm 6.62}$	$\mathbf{-16.77 \pm 2.25}$	-5.03 ± 1.06
	ALIGN$_{\text{team}}$	$\mathbf{511.97 \pm 6.95}$	$\mathbf{699.56 \pm 11.64}$	190.06 ± 29.10	-19.40 ± 2.86	-3.10 ± 3.09
	ALIGN$_{\text{adv}}$	—	—	228.53 ± 25.03	-31.03 ± 3.13	$\mathbf{27.24 \pm 4.48}$

Table A.19: We sample agents from different *decentralized* training runs and evaluate their zero-shot performance in *scaled* environments under partial observability. We report the average test agent-to-target (agt-target) and agent-to-adversary (agt-adv) distances and standard errors of *decentralized* methods with different intrinsic rewards. Lower scores are better for agt-target (↓), and higher scores are better for agt-adv (↑).

		Cooperative		**Competitive**		
Task (Agt # vs. Adv #)		**Coop nav. (5v0)** ↓	**Hetero nav. (6v0)** ↓	**Phy decep. (4v2)** ↓	**Pred-prey (4v4)** ↑	**Keep-away (4v4)** ↓
Partial observability	SPARSE	0.22 ± 0.00	0.23 ± 0.00	0.23 ± 0.00	1.93 ± 0.00	3.16 ± 0.01
	ALIGN$_{\text{self}}$	0.19 ± 0.00	0.20 ± 0.00	0.17 ± 0.00	2.33 ± 0.00	3.31 ± 0.01
	ALIGN$_{\text{team}}$	0.43 ± 0.01	0.19 ± 0.00	0.24 ± 0.00	2.04 ± 0.01	3.15 ± 0.01
	ALIGN$_{\text{adv}}$	—	—	0.21 ± 0.00	2.11 ± 0.01	2.31 ± 0.01

Bibliography

Achiam, J., & Sastry, S. (2017). Surprise-based intrinsic motivation for deep reinforcement learning. *arXiv preprint arXiv:1703.01732*.

Ben-Jacob, E., Schochet, O., Tenenbaum, A., Cohen, I., Czirok, A., & Vicsek, T. (1994). Generic modelling of cooperative growth patterns in bacterial colonies. *Nature*, *368*(6466), 46–49.

Böhmer, W., Rashid, T., & Whiteson, S. (2019). Exploration with unreliable intrinsic reward in multi-agent reinforcement learning. *arXiv preprint arXiv:1906.02138*.

Buhl, J., Sumpter, D. J., Couzin, I. D., Hale, J. J., Despland, E., Miller, E. R., & Simpson, S. J. (2006). From disorder to order in marching locusts. *Science*, *312*(5778), 1402–1406.

Burda, Y., Edwards, H., Pathak, D., Storkey, A., Darrell, T., & Efros, A. A. (2018). Large-scale study of curiosity-driven learning. *arXiv preprint arXiv:1808.04355*.

Burda, Y., Edwards, H., Storkey, A., & Klimov, O. (2018). Exploration by random network distillation. *arXiv preprint arXiv:1810.12894*.

Camazine, S., Deneubourg, J.-L., Franks, N. R., Sneyd, J., Theraula, G., & Bonabeau, E. (2020). *Self-organization in biological systems*. Princeton university press.

Cao, Y., Yu, W., Ren, W., & Chen, G. (2012). An overview of recent progress in the study of distributed multi-agent coordination. *IEEE Transactions on Industrial informatics*, *9*(1), 427–438.

Chen, T., Kornblith, S., Norouzi, M., & Hinton, G. (2020). A simple framework for contrastive learning of visual representations. In *International conference on machine learning* (pp. 1597–1607).

Chitnis, R., Tulsiani, S., Gupta, S., & Gupta, A. (2020). Intrinsic motivation for encouraging synergistic behavior. *arXiv preprint arXiv:2002.05189*.

Collett, M., Despland, E., Simpson, S. J., & Krakauer, D. C. (1998). Spatial scales of desert locust gregarization. *Proceedings of the National Academy of Sciences*, *95*(22), 13052–13055.

Corder, K., Vindiola, M. M., & Decker, K. (2019). Decentralized multi-agent actor-critic with generative inference. *arXiv preprint arXiv:1910.03058*.

Couzin, I. (2007). Collective minds. *Nature*, *445*(7129), 715–715.

Couzin, I. D., Krause, J., James, R., Ruxton, G. D., & Franks, N. R. (2002). Collective memory and spatial sorting in animal groups. *Journal of theoretical biology*, *218 1*, 1-11.

Dragan, A. D., Lee, K. C., & Srinivasa, S. S. (2013). Legibility and predictability of robot motion. In *2013 8th acm/ieee international conference on human-robot interaction (hri)* (pp. 301–308).

Du, Y., Han, L., Fang, M., Dai, T., Liu, J., & Tao, D. (2019). Liir: learning individual intrinsic reward in multi-agent reinforcement learning. In *Proceedings of the 33rd international conference on neural information processing systems* (pp. 4403–4414).

Foerster, J., Farquhar, G., Afouras, T., Nardelli, N., & Whiteson, S. (2018). Counterfactual multi-agent policy gradients. In *Proceedings of the aaai conference on artificial intelligence* (Vol. 32).

Foerster, J. N., Chen, R. Y., Al-Shedivat, M., Whiteson, S., Abbeel, P., & Mordatch, I. (2017). Learning with opponent-learning awareness. *arXiv preprint arXiv:1709.04326*.

Haarnoja, T., Zhou, A., Abbeel, P., & Levine, S. (2018). Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *Icml*.

Hernandez-Leal, P., Kartal, B., & Taylor, M. E. (2019). Agent modeling as auxiliary task for deep

reinforcement learning. In *Proceedings of the aaai conference on artificial intelligence and interactive digital entertainment* (Vol. 15, pp. 31–37).

Hu, H., Lerer, A., Peysakhovich, A., & Foerster, J. (2020). "other-play" for zero-shot coordination. In *International conference on machine learning* (pp. 4399–4410).

Huang, C.-M., Cakmak, M., & Mutlu, B. (2015). Adaptive coordination strategies for human-robot handovers. In *Robotics: science and systems* (Vol. 11).

Iqbal, S., & Sha, F. (2019a). Actor-attention-critic for multi-agent reinforcement learning. In *International conference on machine learning* (pp. 2961–2970).

Iqbal, S., & Sha, F. (2019b). Coordinated exploration via intrinsic rewards for multi-agent reinforcement learning. *arXiv preprint arXiv:1905.12127*.

Iqbal, S., & Sha, F. (2020). *Coordinated exploration via intrinsic rewards for multi-agent reinforcement learning.* Retrieved from `https://openreview.net/forum?id=rkltE0VKwH`

Jaderberg, M., Mnih, V., Czarnecki, W. M., Schaul, T., Leibo, J. Z., Silver, D., & Kavukcuoglu, K. (2016). Reinforcement learning with unsupervised auxiliary tasks. *arXiv preprint arXiv:1611.05397*.

Jain, U., Weihs, L., Kolve, E., Farhadi, A., Lazebnik, S., Kembhavi, A., & Schwing, A. (2020). A cordial sync: Going beyond marginal policies for multi-agent embodied tasks. In *European conference on computer vision* (pp. 471–490).

Jaques, N., Lazaridou, A., Hughes, E., Gulcehre, C., Ortega, P., Strouse, D., ... De Freitas, N. (2019). Social influence as intrinsic motivation for multi-agent deep reinforcement learning. In *International conference on machine learning* (pp. 3040–3049).

Kidambi, R., Rajeswaran, A., Netrapalli, P., & Joachims, T. (2020). Morel: Model-based offline reinforcement learning. *arXiv preprint arXiv:2005.05951*.

Kim, K., Sano, M., De Freitas, J., Haber, N., & Yamins, D. (2020). Active world model learning

with progress curiosity. In *International conference on machine learning* (pp. 5306–5315).

Krause, J., Ruxton, G. D., Ruxton, G., Ruxton, I. G., et al. (2002). *Living in groups*. Oxford University Press.

Kurach, K., Raichuk, A., Stanczyk, P., Zajac, M., Bachem, O., Espeholt, L., ... Gelly, S. (2019). Google research football: A novel reinforcement learning environment. *CoRR*, *abs/1907.11180*. Retrieved from `http://arxiv.org/abs/1907.11180`

Littman, M. L. (1994). Markov games as a framework for multi-agent reinforcement learning. In *Machine learning proceedings 1994* (pp. 157–163). Elsevier.

Liu, I.-J., Yeh, R. A., & Schwing, A. G. (2020). Pic: permutation invariant critic for multi-agent deep reinforcement learning. In *Conference on robot learning* (pp. 590–602).

Lowe, R., Wu, Y., Tamar, A., Harb, J., Abbeel, P., & Mordatch, I. (2017). Multi-agent actor-critic for mixed cooperative-competitive environments. *Advances in Neural Information Processing Systems*.

Mordatch, I., & Abbeel, P. (2017). Emergence of grounded compositional language in multi-agent populations. *arXiv preprint arXiv:1703.04908*.

Morin, A. (2006). Levels of consciousness and self-awareness: A comparison and integration of various neurocognitive views. *Consciousness and cognition*, *15*(2), 358–371.

Ndousse, K., Eck, D., Levine, S., & Jaques, N. (2021). *Emergent social learning via multi-agent reinforcement learning.*

Pathak, D., Agrawal, P., Efros, A. A., & Darrell, T. (2017). Curiosity-driven exploration by self-supervised prediction. In *International conference on machine learning* (pp. 2778–2787).

Rashid, T., Samvelyan, M., de Witt, C. S., Farquhar, G., Foerster, J., & Whiteson, S. (2018). *Qmix: Monotonic value function factorisation for deep multi-agent reinforcement learning.*

Schafer, L. (2019). *Curiosity in multi-agent reinforcement learning* (Unpublished doctoral dissertation). Master's thesis, The University of Edinburgh.

Schmidhuber, J. (1991). A possibility for implementing curiosity and boredom in model-building neural controllers. In *Proc. of the international conference on simulation of adaptive behavior: From animals to animats* (pp. 222–227).

Sekar, R., Rybkin, O., Daniilidis, K., Abbeel, P., Hafner, D., & Pathak, D. (2020). Planning to explore via self-supervised world models. In *International conference on machine learning* (pp. 8583–8592).

Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., Van Den Driessche, G., ... others (2016). Mastering the game of go with deep neural networks and tree search. *nature*, *529*(7587), 484–489.

Srivastava, S., Li, C., Lingelbach, M., Martín-Martín, R., Xia, F., Vainio, K., ... others (2021). Behavior: Benchmark for everyday household activities in virtual, interactive, and ecological environments. *arXiv preprint arXiv:2108.03332*.

Stadie, B. C., Levine, S., & Abbeel, P. (2015). Incentivizing exploration in reinforcement learning with deep predictive models. *arXiv preprint arXiv:1507.00814*.

Strouse, D., Kleiman-Weiner, M., Tenenbaum, J., Botvinick, M., & Schwab, D. J. (2018). Learning to share and hide intentions using information regularization. *Advances in Neural Information Processing Systems*, *31*, 10249–10259.

Sunehag, P., Lever, G., Gruslys, A., Czarnecki, W. M., Zambaldi, V., Jaderberg, M., ... others (2017). Value-decomposition networks for cooperative multi-agent learning. *arXiv preprint arXiv:1706.05296*.

Swamy, G., Reddy, S., Levine, S., & Dragan, A. D. (2020). Scaled autonomy: Enabling human operators to control robot fleets. In *2020 ieee international conference on robotics and automation*

(icra) (pp. 5942–5948).

Theraulaz, G., & Bonabeau, E. (1995). Coordination in distributed building. *Science*, *269*(5224), 686–688.

Tomasello, M., Carpenter, M., Call, J., Behne, T., & Moll, H. (2005). Understanding and sharing intentions: The origins of cultural cognition. *Behavioral and brain sciences*, *28*(5), 675–691.

Wang, R. E., Kew, J. C., Lee, D., Lee, T.-W. E., Zhang, T., Ichter, B., ... Faust, A. (2020). *Model-based reinforcement learning for decentralized multiagent rendezvous.*

Wang, R. E., Wu, S. A., Evans, J. A., Tenenbaum, J. B., Parkes, D. C., & Kleiman-Weiner, M. (2020). *Too many cooks: Bayesian inference for coordinating multi-agent collaboration.*

Wang, T., Wang, J., Wu, Y., & Zhang, C. (2019). Influence-based multi-agent exploration. *arXiv preprint arXiv:1910.05512*.

Wang, X., Xiong, W., Wang, H., & Wang, W. Y. (2018). Look before you leap: Bridging model-free and model-based reinforcement learning for planned-ahead vision-and-language navigation. In *Proceedings of the european conference on computer vision (eccv)* (pp. 37–53).

Ying, W., & Dayong, S. (2005). Multi-agent framework for third party logistics in e-commerce. *Expert Systems with Applications*, *29*(2), 431–436.