

System Neural Diversity: Measuring Behavioral Heterogeneity in Multi-Agent Learning

Matteo Bettini

MB2389@CL.CAM.AC.UK

*Department of Computer Science and Technology,
University of Cambridge, UK*

Ajay Shankar

AS3233@CL.CAM.AC.UK

*Department of Computer Science and Technology,
University of Cambridge, UK*

Amanda Prorok

ASP45@CL.CAM.AC.UK

*Department of Computer Science and Technology,
University of Cambridge, UK*

Editor: Martha White

Abstract

Evolutionary science provides evidence that diversity confers resilience in natural systems. Yet, traditional multi-agent reinforcement learning techniques commonly enforce homogeneity to increase training sample efficiency. When a system of learning agents is not constrained to homogeneous policies, individuals may develop diverse behaviors, resulting in emergent complementarity that benefits the system. Despite this, there is a surprising lack of tools that *quantify* behavioral diversity. Such techniques would pave the way towards understanding the impact of diversity in collective artificial intelligence and enabling its control. In this paper, we introduce System Neural Diversity (SND): a measure of behavioral heterogeneity in multi-agent systems. We discuss and prove its theoretical properties, and compare it with alternate, state-of-the-art behavioral diversity metrics used in the robotics domain. Through simulations of a variety of cooperative multi-robot tasks, we show how our metric constitutes an important tool that enables measurement and control of behavioral heterogeneity. In dynamic tasks, where the problem is affected by repeated disturbances during training, we show that SND allows us to measure latent resilience skills acquired by the agents, while other proxies, such as task performance (reward), fail to. Finally, we show how the metric can be employed to control diversity, allowing us to enforce a desired heterogeneity set-point or range. We demonstrate how this paradigm can be used to bootstrap the exploration phase, finding optimal policies faster, thus enabling novel and more efficient MARL paradigms.

Keywords: Diversity, Multi-Agent Reinforcement Learning, Heterogeneity Measure

1. Introduction

Diversity is key to collective intelligence (Woolley et al., 2015) and commonplace in natural systems (Kellert, 1997). Just as biologists and ecologists have demonstrated the role of functional diversity in ecosystem survival (Cadotte et al., 2011), it has also been shown to provide resilience and performance benefits in Multi-Agent Reinforcement Learning (MARL) (Bettini et al., 2023). While a variety of measures of diversity exist for natural

systems (Tucker et al., 2017), there is a lack of work studying *behavioral* diversity¹ in engineered systems. Developing a principled diversity measure would allow us to directly quantify previously unobservable properties of the system (such as resilience) as well as enable its control (*e.g.*, in a closed-loop fashion). In this research, we are interested in measuring this diversity and its impact in multi-robot tasks, where collective intelligence and cooperation are critical to success.

In order to boost sample efficiency, it is common practice in MARL to share policy parameters for agents with the same objective (Gupta et al., 2017; Rashid et al., 2018; Sukhbaatar et al., 2016). This causes the agents to become behaviorally homogeneous. On the other hand, when policies can diverge, and thus produce different action distributions for the same observation, we refer to the team as ‘heterogeneous’. Recent MARL paradigms for training heterogeneous policies have been shown to provide intrinsic resilience benefits (Bettini et al., 2023). While several recent methods promote behavioral diversity in MARL (Jaderberg et al., 2019; Wang et al., 2021; Christianos et al., 2021; Chenghao et al., 2021; Wang et al., 2020), they lack a principled measure for it. Because of this, they are limited to analyzing the system from a performance (i.e., reward) perspective, which is frequently a suboptimal proxy for the underlying diversity. The development of a principled behavioral heterogeneity metric would enable an analysis of the impact of diversity, and subsequently, its control (Bettini et al., 2024a).

In this paper, we introduce System Neural Diversity (SND), a measure of behavioral heterogeneity in multi-agent learning. In addition to satisfying formal properties of a *metric*, SND follows two key properties that are desirable in such a measure: (1) given a fixed inter-agent pairwise behavioral distance, the metric should not depend on the number of agents, and, (2) the metric should decrease as more agents assume the same behaviors. Property (1) allows us to compare heterogeneity across different team sizes, while Property (2) allows the metric to measure behavioral redundancy. To compute SND, we first define a pairwise inter-agent behavioral distance. Pairwise distances are then aggregated into a system-wide diversity value. We employ the Wasserstein metric (Vaserstein, 1969) to measure pairwise distances, allowing us to compute the behavioral diversity of multivariate normal action distributions in closed-form. Furthermore, we juxtapose SND with Hierarchic Social Entropy (HSE) (Balch, 2000), a state-of-the-art behavioral heterogeneity metric used in robotics, and show that HSE does not follow the properties defined, thus providing an alternate (and yet, complementary) view of behavioral diversity.

Using the proposed metric, we perform a set of evaluations to study the following research question: “*What is the impact of neural diversity on multi-agent learning?*” Our studies in static (i.e., fixed during training) and dynamic (i.e., changing during training) cooperative multi-robot problems show that SND is a powerful tool to assess previously unobservable performance and resilience properties of multi-robot systems. In particular, in cases where the task undergoes repeated dynamic disruptions, we show that measuring SND reveals *latent* diversity in skills that the system has developed to cope with the disruptions. We further show that SND can be used to explicitly control for a target diversity during training, thus bootstrapping the search for the optimal policy and enabling the emergence of novel strategies.

1. also referred to as ‘neural’ diversity

Table 1: Comparison of SND with the related works.

Paper	System aggregation ^a	Behavioral distance ^b	Action space ^c	Probability distributions ^d	Population-based ^e
SND (Ours)	Mean	Wasserstein	Continuous	Yes (closed-form)	No
McKee et al. (2022)	Mean	Total variation distance	Discrete	Yes (empirically estimated)	Yes
Yu et al. (2021)	NA	KL	Discrete	Yes (empirically estimated)	Yes
Yu et al. (2021)	NA	Wasserstein	Continuous	Yes (empirically estimated)	Yes
Hu et al. (2022)	Mean	Symmetric KL	Discrete	Yes (closed-form)	No
Balch (2000)	Clustering	Difference	Continuous	No (deterministic)	No
Liu et al. (2021)	None	f -divergence	Discrete	Yes (closed-form)	Yes
Parker-Holder et al. (2020)	Determinant	Kernel	Continuous	No (deterministic)	Yes (single-agent)
Lupu et al. (2021)	None	Jensen-Shannon divergence	Discrete	Yes	Yes

^a Aggregation function to compute the system-level value

^b Pairwise distance function used to compare agent pairs

^c Action space type

^d Whether the behavioral distance is computed between probability distributions

^e If the approach is population-based

A summary of our contributions in this work is as follows:

- We introduce SND, a novel metric to quantify behavioral heterogeneity in multi-agent policies;
- We present and prove SND’s theoretical properties, juxtaposing it to previous diversity measures;
- Through a variety of collective multi-robot tasks, we study the impact and insights gathered from measuring and controlling diversity in MARL, showing how SND provides a novel tool, enabling us to analyze as well as act upon previously unobservable latent properties in multi-agent learning.

2. Related Work

In this section, we give an overview of existing diversity measures used in the multi-agent and multi-robot domain. In Tab. 1 we provide a detailed table comparing related works to ours, highlighting how SND is the only statistical metric that can be computed in closed-form between the stochastic continuous action distributions provided as output by the policies. Sec. A contains further discussion and comparisons on diversity indices in the biological domain.

2.1 Diversity Measures in MARL

In this work, we are interested in *measuring* diversity, and thus, we only consider MARL policies that are already heterogeneous, i.e., those that do not employ parameter sharing. This notion of diversity has gained increasing attention in recent years, and several works have outlined the caveats and implications of parameter sharing (Christianos et al., 2021; Fu et al., 2022; Bettini et al., 2023). These have shown that even if homogeneous policies can emulate heterogeneous behavior in certain cases (using, for instance, some input context), they are often brittle in the presence of real-world noise, and sometimes, completely prevent

task success. We refer to Bettini et al. (2023) for a classification of existing heterogeneous MARL solutions. While some solutions are able to promote diversity to boost training performance (Wang et al., 2021; Christianos et al., 2021; Chenghao et al., 2021; Wang et al., 2020; Mahajan et al., 2019; Jiang and Lu, 2021; Wang et al., 2019), they do so without being able to measure the resulting heterogeneity – the problem of developing a reliable diversity metric is frequently overlooked.

Nevertheless, some prior work has considered approximate diversity metrics. In McKee et al. (2022) the authors introduce a diversity metric that uses sampled discrete actions from agents’ policies to approximate their distributions, and then total variation distance to compute the divergence among them. Similarly, in Yu et al. (2021), behavioral distances are obtained using sampled action datapoints. These methods use approximated distributions, leading to a decrease in the metric’s accuracy. In contrast, SND measures the distance between agent action distributions in closed-form, avoiding approximations. Another action-based diversity metric (Hu et al., 2022) uses symmetric Kullback–Leibler (KL) divergence to measure behavioral distances in discrete action spaces. However, symmetric KL does not satisfy the triangle inequality property, and hence, it is not a statistical metric. In Liu et al. (2021) the authors propose to use f-divergence between occupancy measures to quantify behavioral diversity in zero-sum games. The proposed solution, however, becomes computationally intractable for more general Markov games.

Population-based RL employs methods from Quality-Diversity optimization (Chatzilygeroudis et al., 2021; Stanley et al., 2019), in which populations are ‘bred’ to maximize both performance and diversity, with some techniques to analyze diversity in this domain (Doncieux and Mouret, 2010; Masood and Doshi-Velez, 2019). In Parker-Holder et al. (2020), authors propose a diversity measure that computes the population diversity as the determinant of the agents’ behavioral distance matrix. The agent distances, however, are bounded and are computed between deterministic actions over randomly sampled states, which can lead to policies being evaluated in states they have not been trained on. TrajeDi (Lupu et al., 2021) measures the Jensen-Shannon divergence between the trajectory distributions of agents within a population to improve zero-shot coordination. In contrast, our work does not employ population-based training and discrete action spaces, and considers the problem of measuring the behavioral diversity of continuous distributions for concurrently acting agents within a team.

2.2 Diversity Measures in Robotics

When agents are embodied as robots, physical differences can emerge, leading to different capabilities. Prorok et al. (2017) characterize such differences in a diversity matrix representing the species and traits of a robot population. The rank of such matrices can be related to behavioral redundancy in the population and *eigenspecies* can then be identified.

Behavioral differences, however, are not always correlated with physical ones and need a dedicated measure in policy space. Li et al. (2004) propose a behavioral specialization metric that is, however, obtained through a correlation with team performance. Twu et al. (2014) propose a computation-light heterogeneity metric that assumes a fixed number of diversity classes. Closest to our work, Balch’s Hierarchic Social Entropy (HSE) (Balch, 2000) computes the behavioral distance between two agents as the difference of their deterministic

actions over all states. It then uses this distance to compute a team level metric. In particular, agents are hierarchically divided into behavioral clusters. For each hierarchical level, the Shannon entropy is computed on the distribution of agents in behavioral clusters. At hierarchical level l , for example, agents i, j with behavioral distance $d_{\text{HSE}}(i, j) \leq l$ will belong to the same cluster and $E(l)$ will denote the Shannon entropy of the clusters. HSE is then computed as

$$\text{HSE} = \int_0^\infty E(l) dl. \quad (1)$$

Since Shannon entropy does not take into account the distance between classes, hierarchical clustering is used in HSE to make the metric depend on such distances. We will analyze and compare HSE with SND in the following sections, showing how these metrics provide complementary tools to assess diversity.

3. Problem Formulation

We now formulate the MARL problem analyzed in this work. To do so, we first introduce the multi-agent extension of a Partially Observable Markov Decision Process (POMDP) (Kaelbling et al., 1998).

3.1 Partially Observable Markov Games

A Partially Observable Markov Game (POMG) is defined as a tuple

$$\langle \mathcal{N}, \mathcal{S}, \{\mathcal{O}_i\}_{i \in \mathcal{N}}, \{\sigma_i\}_{i \in \mathcal{N}}, \{\mathcal{A}_i\}_{i \in \mathcal{N}}, \{\mathcal{R}_i\}_{i \in \mathcal{N}}, \mathcal{T}, \gamma \rangle,$$

where $\mathcal{N} = \{1, \dots, n\}$ denotes the set of agents, \mathcal{S} is the state space, shared by all agents, and, $\{\mathcal{O}_i\}_{i \in \mathcal{N}}$ and $\{\mathcal{A}_i\}_{i \in \mathcal{N}}$ are the observation and action spaces, with $\mathcal{O}_i \subseteq \mathcal{S}$, $\forall i \in \mathcal{N}$. Further, $\{\sigma_i\}_{i \in \mathcal{N}}$ and $\{\mathcal{R}_i\}_{i \in \mathcal{N}}$ are the agent observation and reward functions (potentially identical for all agents), such that $\sigma_i : \mathcal{S} \mapsto \mathcal{O}_i$, and, $\mathcal{R}_i : \mathcal{S} \times \{\mathcal{A}_i\}_{i \in \mathcal{N}} \times \mathcal{S} \mapsto \mathbb{R}$. \mathcal{T} is the stochastic state transition model, defined as $\mathcal{T} : \mathcal{S} \times \{\mathcal{A}_i\}_{i \in \mathcal{N}} \times \mathcal{S} \mapsto [0, 1]$, which outputs the probability $\mathcal{T}(s^t, \{a_i^t\}_{i \in \mathcal{N}}, s^{t+1})$ of transitioning to state s^{t+1} given the current state s^t and actions $\{a_i^t\}_{i \in \mathcal{N}}$. γ is the discount factor.

Agents have a stochastic policy $\pi_{\theta_i}(a_i|o_i)$, which maps observations to action distributions that are sampled in the POMG to maximize the sum of discounted rewards received from the environment. The policy of agent i is conditioned on neural network parameters θ_i . To train policies π_{θ_i} we adopt the (Het)GPPO models presented in (Bettini et al., 2023). Using GPPO, we perform homogeneous training via parameter sharing, thus $\theta = \theta_1 = \dots = \theta_n$. Homogeneous agents are constrained to use the *same* policy π_θ but benefit from the higher sample efficiency resulting from sharing parameters. On the other hand, when using HetGPPO, agents are able to learn independent heterogeneous policies π_{θ_i} and thus can develop behavioral differences. Note that, as mentioned before, homogeneous policies can sometimes *emulate* diverse behavior by exploiting contextual information from their input (Bettini et al., 2023), and, thus, our comparisons here will employ the same inputs for both homogeneous and heterogeneous models.

3.2 Objective

The goal of this work is to measure the behavioral heterogeneity in a system of learning agents with heterogeneous policies π_{θ_i} . Our objective is to develop a metric,

$$\text{SND} : \{\pi_{\theta_i}\}_{i \in \mathcal{N}} \mapsto \mathbb{R}_{\geq 0},$$

that takes as input the agents' policies and outputs a scalar value representing the behavioral diversity of the system. This metric will depend only on the learned agent policies and, thus, does not take the POMG components directly as input. Since we are comparing policies of different agents, we assume that all agents have the same observation space $\mathcal{O} = \mathcal{O}_1 = \dots = \mathcal{O}_n$ and action space $\mathcal{A} = \mathcal{A}_1 = \dots = \mathcal{A}_n$. Importantly, differences in parameter space do not necessarily map to differences in behavioral space (Stanley et al., 2019; Kaplanis et al., 2019), thus we cannot measure diversity directly through differences between parameters θ_i and θ_j , but we need to measure it from policy outputs $\pi_{\theta_i}(o)$ and $\pi_{\theta_j}(o)$.

4. Method: A Metric for System Neural Diversity

Developing a metric to measure neural diversity in a multi-agent system is a challenging task. This is due to the fact that reducing a behavioral property dependent on potentially millions of neural connections to a single scalar value inevitably leads to high information loss. Our goal is to minimize this information loss while retaining informative properties in the resulting metric.

In order to measure system diversity we first need a way to compare individuals. We thus tackle the following two tasks. First, we define a measure of inter-agent pairwise **behavioral distance**. We structure pairwise distances obtained with this measure in a behavioral distance matrix, which represents the distribution of agents' policies in behavioral space. Second, we aggregate the behavioral distance matrix into a single diversity value, which represents the **neural diversity metric** of the entire system. In the following, we discuss the choice of the functions used for these two tasks.

4.1 Behavioral Distance

Heterogeneity is a collective concept. That is, it cannot be measured as an absolute property pertaining to a single agent in the system, but it has to be expressed as a relative measure among agents. We thus need to develop a mathematical metric that measures the behavioral distance between two agents $i, j \in \mathcal{N}$ as $d(i, j)$, where $d : \mathcal{N} \times \mathcal{N} \mapsto \mathbb{R}_{\geq 0}$. We want $d(i, j)$ to follow the properties of a mathematical metric (Menger, 1942).

Definition 1 (Properties of the behavioral distance metric (Menger, 1942)) *For the distance d to be a metric, it has to satisfy the following properties $\forall i, j, k \in \mathcal{N}$:*

1. *Non-negativity:* $d(i, j) \geq 0$
2. *Identity of indiscernibles:* $d(i, j) = 0$ iff $\pi_{\theta_i} = \pi_{\theta_j}$
3. *Symmetry:* $d(i, j) = d(j, i)$
4. *Triangle inequality:* $d(i, j) \leq d(i, k) + d(k, j)$

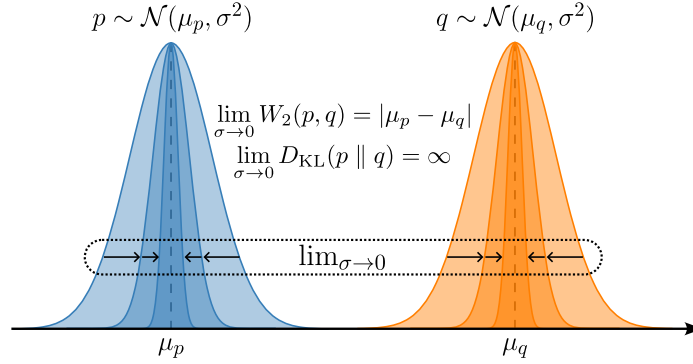


Figure 1: Wasserstein metric $W_2(p, q)$ and KL divergence $D_{\text{KL}}(p \parallel q)$ of two univariate distributions p, q as their standard deviation approaches 0. The value of W_2 in this scenario approaches the absolute difference of their means while KL divergence approaches infinity.

This set of properties aligns naturally with behavioral distance. We can think of two agents i, j to be homogeneous when $d(i, j) = 0$ and increasingly diverse as the metric grows.

Our goal is to develop a distance

$$d(i, j) = \int_{\mathcal{O}} f(\pi_{\theta_i}(o), \pi_{\theta_j}(o)) do$$

where a function f , providing a distance between two policies for a given observation, is evaluated over all possible observations. Two key challenges follow: (1) f has to be a measure between the two probability distributions outputted by the policies and (2) evaluating distance over all possible continuous observations is intractable and, thus, we need a clever sampling strategy to create a subset of observations for evaluating the distance. In the following two subsections, we discuss how our proposed distance addresses these issues.

4.1.1 DISTANCE FOR STOCHASTIC POLICIES

Stochastic policies map a given observation to an action distribution. Thus, a comparison of two agent behaviors can be made by measuring the statistical distance between their distributions. There exist a variety of statistical distances. Distances that follow properties 1-4 of Def. 1 are referred to as *metrics*, while statistical distances that only satisfy 1-2 are called *divergences*. Divergences (such as the widespread Kullback–Leibler, KL) do not follow triangle inequality (property-4), which is useful when determining an upper bound of the behavioral distance between two agents whose respective distances to a third agent are known. Among metrics, on the other hand, we focus our attention on the Wasserstein metric (W_2) (Vaserstein, 1969). The W_2 distance between multivariate Gaussian action distributions can be computed in closed-form using the distributions’ parameters. It represents the minimum cost required to move all the probability mass from one distribution to the other in an optimal transport problem formulation.

Fig. 1 depicts an illustrative example to further elucidate the differences between the Wasserstein metric and KL divergence. In this example, $p \sim \mathcal{N}(\mu_p, \sigma^2)$ and $q \sim \mathcal{N}(\mu_q, \sigma^2)$ are univariate Gaussian distributions. $W_2(p, q)$ represents the Wasserstein metric, $D_{\text{KL}}(p \parallel q)$ the KL divergence. We observe that, as σ approaches 0, meaning that the probability mass

of both distributions converges to their mean (as in Dirac delta distributions), $D_{\text{KL}}(p \parallel q) \rightarrow \infty$ while $W_2(p, q) \rightarrow |\mu_p - \mu_q|$. This shows that, in the case the distributions assign increasing probability mass to their mean, a common scenario in MARL policies, Wasserstein outputs a bounded value proportional to the distance between the means, while KL does not. A similar argument was used to motivate Wasserstein generative adversarial networks (Arjovsky et al., 2017) where the model architecture significantly benefited from the use of W_2 over KL.

Following the reasons above, we use the Wasserstein metric to measure the probability distance between two policies². The resulting behavioral distance takes the following form:

$$d(i, j) = \int_{\mathcal{O}} W_2(\pi_{\theta_i}(o), \pi_{\theta_j}(o)) \, do. \quad (2)$$

We parameterize action distributions as multivariate normals, independent in each action dimension. This allows to compute the Wasserstein distance W_2 between agent pairs in a closed-form.

Definition 1 (Wasserstein metric for multivariate normal distributions) *Let $\pi_1 = \mathcal{N}(\mu_1, \Sigma_1)$ and $\pi_2 = \mathcal{N}(\mu_2, \Sigma_2)$ be two multivariate normal distributions on \mathbb{R}^m . Then, the 2-Wasserstein distance between π_1 and π_2 is computed as:*

$$W_2(\pi_1, \pi_2) = \sqrt{\|\mu_1 - \mu_2\|_2^2 + \text{trace}(\Sigma_1 + \Sigma_2 - 2(\Sigma_2^{\frac{1}{2}} \Sigma_1 \Sigma_2^{\frac{1}{2}})^{\frac{1}{2}})}.$$

We consider policies with the form $\pi_{\theta_i}(o) = \mathcal{N}(\mu_{\theta_i}(o), \sigma_{\theta_i}(o))$, with $\mu_{\theta_i}(o), \sigma_{\theta_i}(o) \in \mathbb{R}^m$, where σ_{θ_i} is a standard deviation vector which uniquely defines a diagonal covariance matrix $\Sigma_{\theta_i}(o) \in \mathbb{R}^{m \times m}$, $\Sigma_{\theta_i}(o) = \text{diag}(\sigma_{\theta_i}(o)^2)$.

4.1.2 OBSERVATION SAMPLING FOR DISTANCE EVALUATION

Since the integral in Eq. 2 is theoretically computed over the entire continuous space of observations, we require a tractable sampling technique to generate a finite subset $\mathcal{B} \subseteq \mathcal{O}$ over which to evaluate the metric. We choose to create \mathcal{B} via rollouts (i.e., executions of the policy over time). In particular, every time we evaluate the behavioral distance, a set of rollouts is collected from the environment. A rollout is a collection of environment interactions stored in tuples of the form $\langle \{o_i^t\}_{i \in \mathcal{N}}, \{a_i^t\}_{i \in \mathcal{N}}, \{o_i^{t+1}\}_{i \in \mathcal{N}} \rangle$ with time $t \in [0, T]$. We denote the sets of agents' observations and actions at time t as $\mathbf{o}^t = \{o_i^t\}_{i \in \mathcal{N}}$ and $\mathbf{a}^t = \{a_i^t\}_{i \in \mathcal{N}}$, respectively. Thus, we construct $\mathcal{B} = \{\mathbf{o}^t\}_{t \in [0, T]}$.

When building \mathcal{B} , we want to avoid evaluating diversity on observations that were unseen by the agents during training. This is because their policies might present undefined behavior in such cases. Our process of sampling via environment rollouts in a Monte-Carlo fashion (Sutton and Barto, 2018) provides a high likelihood that these observations were seen previously during training. While it has zero sampling bias, it is known to have a high variance in the states visited. We mitigate this by performing multiple rollouts. The number of rollouts performed has to be chosen based on the state distribution in the POMDP under evaluation. Increasing the number of rollouts will decrease variance but increase the

2. In our code we provide implementations of several metrics and distances, including the Hellinger distance (Hellinger, 1909), which can alternatively be used.

computational cost of evaluating the distance. Now, we can write the final formulation of the behavioral distance d as:

$$d(i, j) = \frac{1}{|\mathcal{B}||\mathcal{N}|} \sum_{\mathbf{o}^t \in \mathcal{B}} \sum_{k \in \mathcal{N}} W_2(\pi_{\theta_i}(\mathbf{o}_k^t), \pi_{\theta_j}(\mathbf{o}_k^t)). \quad (3)$$

This formulation states that the behavioral distance between agents i and j is the average Wasserstein metric computed between the distributions outputted by their policies over the observations of the agents collected over policy rollouts.

Computing the behavioral distance relies on the set of observations \mathcal{B} to be evaluated on, but is agnostic to how these observations are collected. They can be obtained in an asynchronous, distributed, or decentralized manner. When computing $d(i, j)$, one only needs to access this set and the pair of policies being evaluated. As such, this computation is possible also in the case asynchronously executing agents, and it is entirely up to the user to decide how to collect the observation set and when to perform the computation.

4.1.3 BEHAVIORAL DISTANCE MATRIX

We can now structure these inter-agent distances in a behavioral distance matrix. Let

$$\mathbf{d}(i) = [d(i, 1), \dots, d(i, n)] \quad n = |\mathcal{N}|$$

define the distances between i and all other agents. We can then define the behavioral distance matrix as

$$\mathbf{D} = [\mathbf{d}(1)^\top, \dots, \mathbf{d}(n)^\top].$$

Looking at the way this matrix is constructed, we can note some of its proprieties. Firstly, being constructed from a metric distance, it inherits the properties from Def. 1. In particular, this matrix is *non-negative* (Property 1), *hollow* (Property 2), and *symmetric* (Property 3). Furthermore, by computing the sum of each row $\mathbf{d}(i)$, we can obtain a per-agent contribution to the system diversity. For example, in a system with n agents where $d(1, j) = x$, for all $j \in \mathcal{N} \setminus \{1\}$ and $d(i, j) = 0$, for all $i, j \in \mathcal{N} \setminus \{1\}$, we get a contribution of $\sum_{j \in \mathcal{N}} d(1, j) = \frac{x(n-1)}{n}$ for agent 1 and a contribution of $\sum_{j \in \mathcal{N}} d(i, j) = \frac{x}{n}$ for all other agents $i \in \mathcal{N} \setminus \{1\}$. By then calculating the relative fraction over the resulting values, we can express the percentage agent contributions to the team heterogeneity.

4.1.4 EXAMPLE: MULTI-AGENT GOAL NAVIGATION

Let us now look at an experimental case study to complement the theoretical discussion on behavioral distance. In the *Multi-Agent Goal Navigation* example, depicted in Fig. 2, n agents are spawned at random positions in a 2D workspace. Each agent is assigned a goal, also spawned at random. Agents observe the relative position to all goals and output the mean and standard deviation of a 2D action distribution representing their desired velocity. This distribution is represented by two univariate Gaussians, which are sampled to get the action for each dimension. The reward for each agent is the difference in the relative distance to its goal over two consecutive timesteps, incentivizing agents to move towards their goals.

We run four training experiments with $n = 4$ and the following setups:

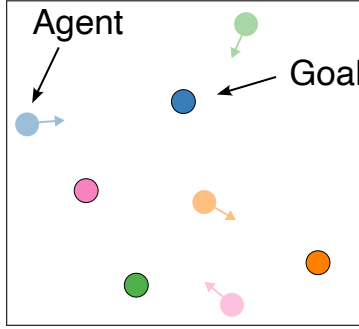


Figure 2: *Multi-Agent Goal Navigation* example. Agents are spawned at random positions in a 2D workspace and take velocity actions (colored arrows) to reach their assigned goal, also spawned at random positions.



Figure 3: Behavioral distance matrices for the four experiments run on *Multi-Agent Goal Navigation*. We can observe how, when agents are assigned the same goal, they become behaviorally homogeneous and thus reduce their behavioral distance. We report mean and standard deviation for $d(i, j)$ over 5 random seeds for each experiment. The values are collected after 300 training iterations each performed over 600 episodes of experience.

- *4 goals*: all agents are assigned a different goal
- *3 goals*: agents 1, 2 are assigned the same goal and the rest have different goals
- *2 goals*: agents 1, 2, 3 are assigned the same goal and the remaining agent has a different goal
- *1 goal*: all agents are assigned the same goal

In Fig. 3 we report the behavioral distance matrices for the four experiments. We can observe how, when agents are assigned the same goal, they learn the same policy, thus decreasing their behavioral distance.

4.2 System Neural Diversity

In the following, we introduce System Neural Diversity (SND), a diversity metric that maps a behavioral distance matrix to a scalar diversity value. SND takes inspiration from the Gini coefficient (Gini, 1912) used in the field of economics. This coefficient provides a measure of statistical dispersion and was created to represent income inequality over a population. In a similar way, we represent diversity as behavioral dispersion using distances $d(i, j)$, allowing

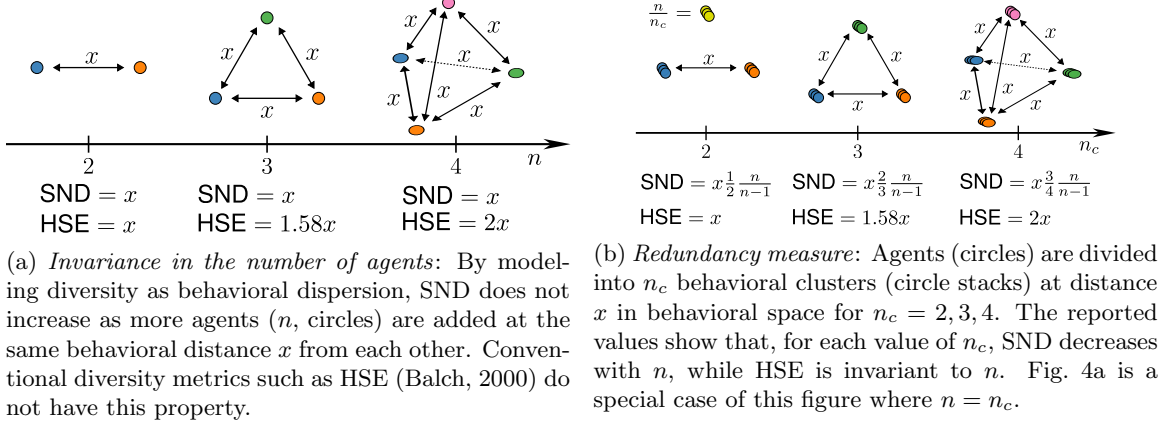


Figure 4: An illustration of the properties of the proposed System Neural Diversity (SND) metric, contrasted against Heirarchical Social Entropy (HSE).

SND values to range from zero to infinity. The proposed SND takes the form:

$$\text{SND}(\mathbf{D}) = \frac{2 \sum_{i=1}^n \sum_{j=i+1}^n d(i, j)}{n(n-1)}, \quad (4)$$

where, due to the symmetry of \mathbf{D} and the fact that its diagonal is zero, we can consider only the upper right triangle of the behavioral distance matrix during computation. SND can be interpreted as the mean behavioral distance over unique pairs of agents in the system.

We now introduce two key properties of SND which highlight its complementary nature with respect to HSE.

4.2.1 INVARIANCE IN THE NUMBER OF EQUIDISTANT AGENTS

When measuring heterogeneity, an important question is: “If in a system with two agents at behavioral distance x we add a third agent, also at distance x from the other two, does the system’s heterogeneity increase?”. While HSE answers positively, SND provides a negative answer. This is because, aligning with the economic interpretation of the Gini coefficient, SND considers diversity as the behavioral dispersion of the system. Maximum dispersion is independent from the number of individuals considered. We refer to this property as *invariance in the number of equidistant agents*.

Property 1 (Invariance in the number of equidistant agents Fig. 4a) *Given a behavioral distance matrix \mathbf{D} , where $d(i, j) = x, \forall i, j \in \mathcal{N}$ with $i \neq j$, representing a system with all agents at behavioral distance x from each other, $\text{SND}(\mathbf{D})$ is invariant with respect to the number of agents n in the system.*

Proof Given that $d(i, j) = x, \forall i, j \in \mathcal{N}$ with $i \neq j$, we can write $\sum_{i=1}^n \sum_{j=i+1}^n d(i, j) = x \frac{n(n-1)}{2}$. Substituting in SND we get $\text{SND}(\mathbf{D}) = \frac{2n(n-1)x}{2n(n-1)} = x$, which is not dependent on n . ■

Fig. 4a depicts this property by showing the SND and HSE values for $n = 2, 3, 4$. The desirability of this property may depend on the use case. Nevertheless, it highlights how SND provides a complementary and additional tool to HSE when measuring diversity.

4.2.2 A MEASURE OF BEHAVIORAL REDUNDANCY

In heterogeneous systems, multiple agents might specialize in the same core skill in order to provide redundancy. For example, if we consider a population survival scenario, some agents may be required to forage food, while others may need to stay static and monitor an area. In games like football, agents may distribute between defenders and attackers. While it seems intuitive that behavioral redundancy should lead to a decrease in system diversity, this aspect is not captured in HSE. HSE clusters behaviorally identical agents together and then computes the Shannon entropy over the population distribution in the behavioral clusters. Thus, if n_c behavioral clusters are present, with $\frac{n}{n_c}$ agents at distance 0 from each other in each of them, HSE will not depend on n , and thus not measure behavioral redundancy. On the other hand, SND is able to measure redundancy.

Property 2 (Redundancy measure Fig. 4b) *Given a behavioral distance matrix \mathbf{D} , where n agents are divided equally in n_c behavioral clusters $\mathcal{C} = \{\mathcal{C}_1, \dots, \mathcal{C}_{n_c}\}$ with $|\mathcal{C}_h| = \frac{n}{n_c} \in \mathbb{N}_{>0}$, $\forall \mathcal{C}_h \in \mathcal{C}$, $c(i) : \mathcal{N} \mapsto \mathcal{C}$ is a function mapping each agent to its cluster, and*

$$d(i, j) = \begin{cases} 0 & \text{if } c(i) = c(j) \\ x & \text{otherwise} \end{cases},$$

SND is a monotonically decreasing function of n and a monotonically increasing function of n_c , and it takes the form

$$\text{SND}(\mathbf{D}) = x \frac{n(n_c - 1)}{n_c(n - 1)}.$$

Proof Given that n agents are equally distributed in n_c behaviorally equidistant clusters, $\sum_{i=1}^n \sum_{j=i+1}^n d(i, j)$ can be rewritten as $x \frac{n^2}{n_c^2} \frac{n_c(n_c-1)}{2}$. Meaning that each pair of agents from two different clusters ($\frac{n^2}{n_c^2}$) is at distance x for each unique pair of clusters ($\frac{n_c(n_c-1)}{2}$). Simplifying, we get $\sum_{i=1}^n \sum_{j=i+1}^n d(i, j) = \frac{xn^2(n_c-1)}{2n_c}$. Substituting in SND we get $\text{SND}(\mathbf{D}) = \frac{2xn^2(n_c-1)}{2n_cn(n-1)} = x \frac{n(n_c-1)}{n_c(n-1)}$, where $\frac{n}{n-1}$ is monotonically decreasing function of n and $\frac{n_c-1}{n_c}$ is a monotonically increasing function of n_c . ■

In other words, SND increases with the number of behavioral clusters, and decreases with the number of agents per cluster. Fig. 4b depicts this property by showing the SND and HSE as a function of n for $n_c = 2, 3, 4$.

4.2.3 EXAMPLE: MULTI-AGENT GOAL NAVIGATION

To exemplify empirically the system-level view provided by SND, we consider again the *Multi-Agent Goal Navigation* scenario from Sec. 4.1.4. We utilise the same setup with $n = 4$ agents and an increasing number of goals.

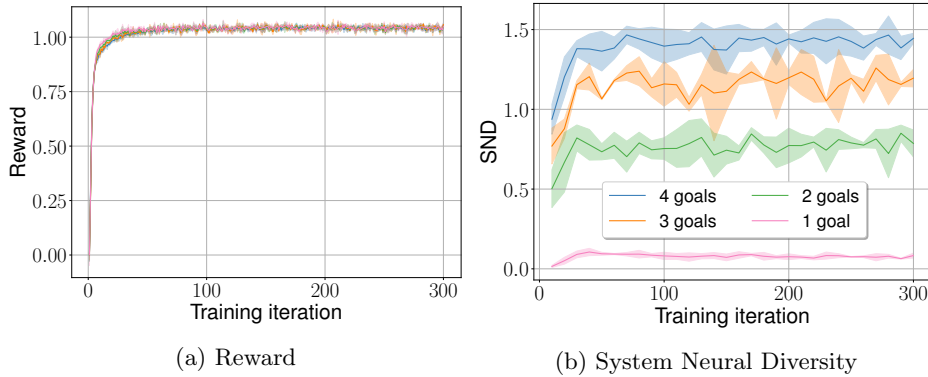


Figure 5: SND in the *Multi-Agent Goal Navigation* scenario. We can observe that, while all setups reach the same reward, SND decreases as the agents share more goals, until the system becomes homogeneous when all agents are sharing the same goal. We report mean and standard deviation over 5 random seeds for each experiment. The values are measured over 300 training iterations each performed over 600 episodes of experience.

The SND measured throughout training (Fig. 5b) decreases with the number of goals. This means that, as the number of goals decreases, more agents will be assigned the same goal and will thus develop more homogeneous navigation policies. In particular, when the agents all share one goal, SND approaches 0. An SND value of 0 indicates that the agents are behaving homogeneously. In such a case, the metric acts an important diagnostic tool, suggesting that a homogeneous training strategy should be preferred in subsequent experiment iterations in order to benefit from parameter sharing and increased sample efficiency. More details on the sample efficiency benefits of homogeneous training are reported in Sec. B.

5. Results

We are now interested in using SND as a heterogeneity metric to measure, control, and develop insights on team diversity during a MARL training phase. For this, we run a series of experiments on various multi-robot tasks. These generally represent multi-robot coordination problems with POMGs that require inter-agent communication to be solved. We analyze two types of tasks: (1) *static tasks* where the agents have to solve a problem that does not change throughout training, and (2) *dynamic tasks* where the problem can change throughout training due to unmodeled disruptions (i.e. noise, external forces, adversaries, etc.). Agents are trained using HetGPPO (Bettini et al., 2023), HetIPPO (HetGPPO without communication) and their homogeneous counterparts: IPPO and GPPO. The simulation environments representing these tasks are partly new implementations and partly adapted from existing environments in the VMAS benchmark set (Bettini et al., 2022).

5.1 Comparison to HSE

We begin by contrasting SND and HSE in the *Multi-Agent Goal Navigation* task (Fig. 2). These provide empirical demonstrations and motivation for the theoretical properties of SND (Sec. 4.2).

5.1.1 INVARIANCE IN THE NUMBER OF EQUIDISTANT AGENTS

To showcase *Invariance in the number of equidistant agents* experimentally, we implement HSE (using our W_2 as the behavioral distance to account for stochastic policies) and run 7 experiments in the *Multi-Agent Goal Navigation* (Sec. 4.1.4) with an increasing number of agents n , each with one goal. Tab. 2 shows that SND value remains invariant to the increasing addition of new agents with new goals, while HSE grows.

5.1.2 A MEASURE OF BEHAVIORAL REDUNDANCY

To showcase how SND measures behavioral redundancy experimentally, we modify the *Multi-Agent Goal Navigation* scenario (Sec. 4.1.4) by fixing the number of goals to 2, corresponding to two behavioral clusters ($n_c = 2$). We then run experiments with $n = \{2, 4, 6, 8\}$ in which the first half of the team is assigned to the first goal and the second half to the other. Tab. 3 shows that SND decreases with the number of agents, thus capturing a redundancy in behavioral diversity as more agents behave similarly. On the other hand, HSE not only does not decrease, but slightly increases, as agents with the same goal will be at behavioral distance ≈ 0 .

5.2 Static Tasks

Next, we train agents on a set of static tasks that benefit from behavioral diversity, and use SND to analyze the impact of such diversity in the learning process. We use the term *static tasks* to refer to multi-agent problems modeled by a fixed³ POMG.

5.2.1 DIFFERENT SIZE JOINT PASSAGE

This task, shown in Fig. 6a, involves two robots of different sizes (blue circles), connected by a rigid linkage through two revolute joints. The team needs to cross a passage while keeping the linkage parallel to it and then match the desired goal position (green circles) on the other side. The passage is comprised of a bigger and a smaller gap, which are spawned in a random position and order on the wall, but always at the same distance between each other. The team is spawned in a random order and position on the lower side with the linkage always perpendicular to the passage. The goal is spawned horizontally in a random position on the upper side. Each agent observes and communicates its velocity, relative position to

3. A POMG that does not vary throughout the learning process.

Table 2: Invariance in the number of equidistant agents in the *Multi-Agent Goal Navigation* scenario. We experimentally show that SND is invariant to the increasing addition of new agents to the system, while HSE grows. We report mean and standard deviation over 4 random seeds for each n . The values are collected after 300 training iterations each performed over 600 episodes of experience.

n	2	3	4	5	6	7	8
SND	1.51 ± 0.14	1.46 ± 0.11	1.43 ± 0.07	1.45 ± 0.06	1.42 ± 0.06	1.44 ± 0.06	1.43 ± 0.05
HSE	1.51 ± 0.14	2.50 ± 0.18	3.17 ± 0.23	3.86 ± 0.21	4.32 ± 0.24	4.77 ± 0.31	5.21 ± 0.29

Table 3: Redundancy measure in the *Multi-Agent Goal Navigation* scenario with $n_c = 2$. We experimentally show that while SND decreases with the redundancy of agents in clusters, HSE slightly increases. We report mean and standard deviation over 6 random seeds for each n . The values are collected after 300 training iterations each performed over 600 episodes of experience.

n	2	4	6	8
SND	1.49 ± 0.12	0.98 ± 0.09	0.85 ± 0.07	0.81 ± 0.06
HSE	1.49 ± 0.12	1.65 ± 0.17	1.76 ± 0.16	1.91 ± 0.16

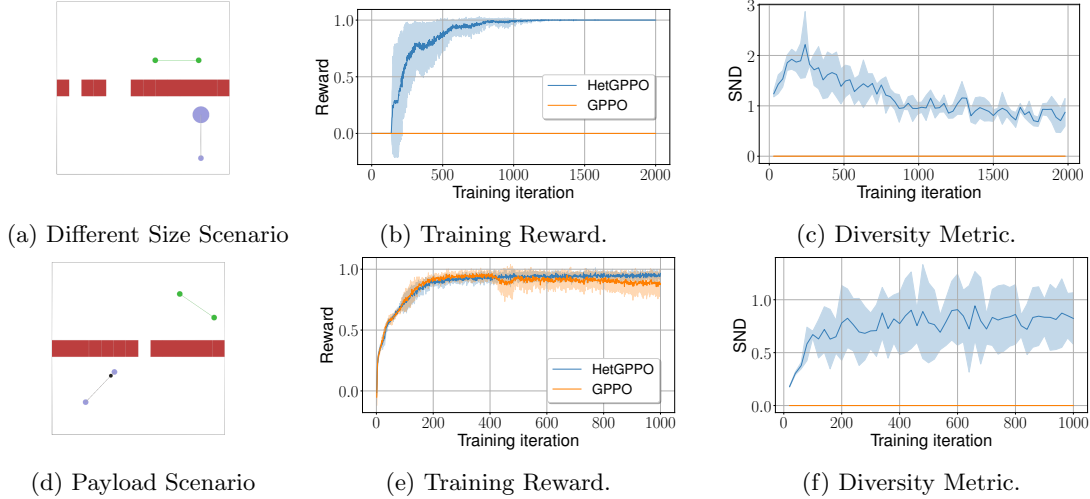


Figure 6: Evaluations on static tasks. **Top row (left to right):** *Different Size Joint Passage* scenario with reward and SND. **Bottom row (left to right):** *Asymmetric Payload Joint Passage* scenario with reward and SND. These evaluations in static scenarios show that heterogeneity can grant performance improvements (top) and resilience improvements (bottom paired with Fig. 7). We report mean and standard deviation over 3 random seeds for each experiment. Each training iteration is performed over 600 episodes of experience.

each gap, and relative position to the goal center. The relative positions and velocities to the other agents are obtained through communication. The sizes of the agents or of the gaps are not part of the observations. The reward function is global and shared by the team. It is composed of two convex terms: before the passage, the robots are rewarded to keep the linkage parallel to the goal and to carry its center to the center of the passage; after the passage, the robots are rewarded for carrying it to the goal at the desired orientation. Collisions are also penalized.

In Fig. 6b we show the training reward (proportional to the percentage of episodes in each batch that complete the task). The plotted reward shows that this task requires heterogeneous behavior to be solved. In fact, the homogeneous agents, unable to observe their physical differences, cannot learn specialized roles, which would allow them to assign the smaller agent to the smaller gap. Agents with homogeneous policies never manage to cross the passage, being deterred by unavoidable collisions. With the heterogeneous model, on the other hand, each agent is able to learn a specialized role and tackle the respective gap. This is confirmed by observing SND in Fig. 6c, which indicates that the agents behave heterogeneously, but learn to smooth out their diversity through training. This is due to

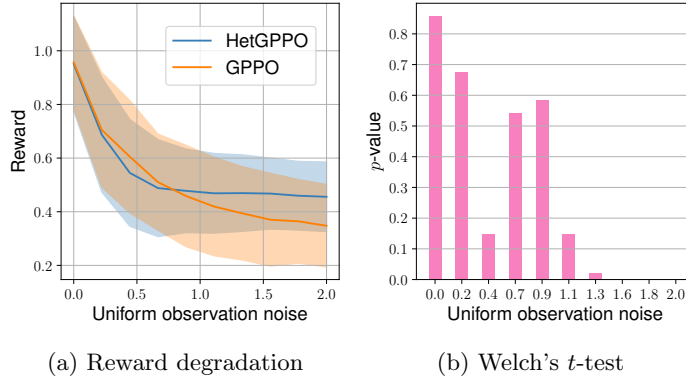


Figure 7: Performance degradation in the *Asymmetric Payload Joint Passage* scenario in Fig. 6d in the presence of deployment noise. We apply uniform observation noise $\mathcal{U}(-\delta, \delta)$ in the same units as the observations for 10 values of δ in the range $[0, 2]$. For each δ we report the mean and standard deviation of the reward for 50 episodes (left) and perform a Welch’s unequal variances t -test (Welch, 1947) for the means of the samples collected with the two models (right).

the fact that diversity is needed only during the initial ‘assignment’ action sequence (to position themselves with respect to the correct gap) and the rest of the navigation task can be solved homogeneously.

5.2.2 ASYMMETRIC PAYLOAD JOINT PASSAGE

In the previous task, agents were physically different. In this task, we run an evaluation in a scenario where agents are physically identical but are impacted in diverse ways by the environment. We consider the task depicted in Fig. 6d. The setup of this task is similar to the *Different Size Joint Passage* scenario, with the difference that the robots are now physically identical, but the linkage has an asymmetrically positioned payload (black circle). The passage now is a single gap, located randomly in the wall. The agents need to cross it while keeping the linkage perpendicular to the wall and avoiding collisions. The team and the goal are spawned in a random position, order, and rotation on opposite sides of the passage. Each robot observes and communicates its velocity, relative position to the gap, relative position to the goal center and goal orientation. The relative positions and velocities to the other agents are obtained through communication. The reward is shared and global and composed of two convex terms: before the passage, the team is rewarded for keeping the linkage perpendicular to the wall and moving towards the center of the gap. After the passage, the team is rewarded for aligning with the goal position and orientation.

By looking at the reward curve for this scenario in Fig. 6e, we can observe that both heterogeneous and homogeneous agents are able to solve the task and obtain the maximum reward. This is because the homogeneous model is able to infer the agent differences from physical observations through a process called *inferred behavioral typing* (Bettini et al., 2023) and thus learn a single multi-behavioral policy conditioned on these differences. In the heterogeneous model, on the other hand, two diverse policies are learned, leading to a significant difference in SND (Fig. 6f). This leads to the following question: “*Given that the two models achieve the same performance with different diversity scores, what advantages, if any, does heterogeneity offer in this scenario?*”. To answer this question, we take the frozen

learned policies for both models and evaluate them under increasing deployment noise. In other words, we inject uniform observation noise at test time. The result, reported in Fig. 7, shows that the heterogeneous model proves more resilient to increasing noise⁴. From the p -values of a Welch’s unequal variances t -test (Welch, 1947) we can observe that the performance curves are statistically different. The low p -values for high noise injections suggest that we fail to accept the null hypothesis of the samples having the same mean.

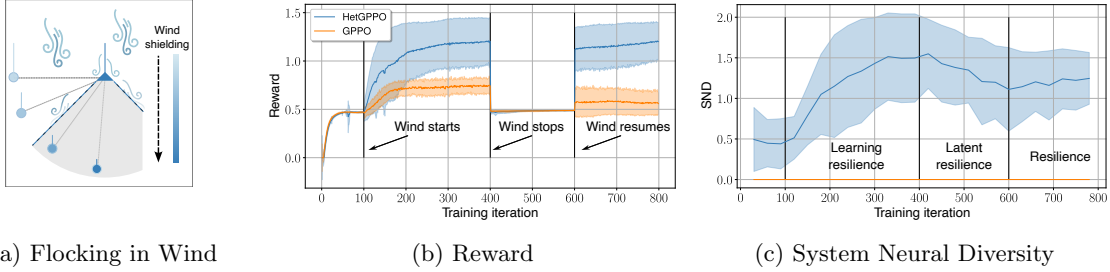


Figure 8: Evaluations on the *Flocking in Wind* dynamic task. These plots show the emergence of *latent resilience* for heterogeneous learning. Heterogeneous agents are able to acquire resilience skills when facing a disturbance and utilize those skills in case the disturbance reappears. We report mean and standard deviation over 11 random seeds for each experiment. Each training iteration is performed over 300 (wind) episodes of experience.

5.3 Dynamic Tasks: Flocking in Wind

We use the term *dynamic tasks* to refer to multi-agent problems modeled by a dynamic POMG (i.e., a POMG that can vary throughout execution). Modifications to the POMG could occur due to unmodeled external disturbances such as noise, faults, or adversaries. These modifications could disrupt the multi-agent system, which would then need to undergo an adaptation process (e.g., changing formation, communication topology, etc.) to regain performance. We refer to the property of adaptation in the face of a disruption as *resilience* (Prorok et al., 2021). Heterogeneous training in MARL has previously been shown to have resilience properties in numerous simulated and real-world scenarios (Bettini et al., 2023). In this section, we are interested in analyzing the adaptation of a multi-agent system to a disruption during the training process. In particular, we consider scenarios where such adaptation requires heterogeneous behaviors, and study how resilience relates to SND.

In the *Flocking in Wind* scenario, shown in Fig. 8a, we consider a flock of $n = 2$ agents (blue circle and blue triangle) in 2D space tasked with tracking a desired velocity vector of 0.5 m/s directed North (top of the figure) while keeping a desired distance of 1 m (dotted line between the agents). The agents receive a shared global reward proportional to the reduction in the errors from the reference velocity and team distance every consecutive timestep. They are initialized 1 m from each other in a random order and at a random angle between $[-\frac{\pi}{8}, \frac{\pi}{8}]$, with zero aligned to West/East. They take 2D velocity actions which result in the control forces shown as blue lines. Each agent observes its velocity and obtains the relative position and velocity to the other agent through communication.

4. Similar resilience results have been observed in (Bettini et al., 2023) in various scenarios.

This scenario is designed to undergo a disruption, which manifests itself as an external southbound wind (a force acting in the direction opposite to the agents’ desired velocity). This results in the agents having to exert more work to track a desired velocity. However, the agents are physically different (see Fig. 8a): the triangular agent has an aerodynamically shielding property, which means that, if it flocks in front of the circular agent, it is able to deflect wind and thus reduce the team energy expenditure. We model the effect of this shielding on the circular agent as proportional to its angular displacement ‘behind’ the triangular agent (in the direction of wind), with its perceived wind force dipping to 0% in case of full alignment. Conversely, an alignment that keeps them horizontal, or the triangular agent behind, causes both of them to be effected by the same maximum wind. The agents receive a reward inversely proportional to the total perceived wind, rewarding the team to minimize the total energy needed for the task. This additional term is simply 0 in the base (windless) version of the task.

In Fig. 8b and Fig. 8c we report the reward and SND, respectively, both for heterogeneous and homogeneous models. From iteration 0 to 100 no wind is present in the environment. Both models learn the optimal solution, which is to flock northwards in the formation they are spawned in. This is because, at this stage, the team has no reason to prefer one formation over the other, and changing formation would require sacrifices in velocity-tracking performance. Furthermore, without wind, the robots cannot benefit from diversity, and thus we observe the heterogeneous model behaving almost homogeneously ($\text{SND} \leq 0.5$).

When wind is added to the environment (iteration 100), the team can now collect additional reward by reducing the perceived wind. The homogeneous model, unable to observe the physical difference of the agents, fails to perform wind shielding and has to employ the same policy for both agents. On the other hand, the heterogeneous agents, being able to learn diverse policies, learn that they can decrease the wind impacting the team by diverging in behavior and performing wind shielding. We can see that, during the time window when wind is introduced (100-400), the heterogeneous agents gradually adapt and increase their reward as well as diversity. We then remove the wind (between iterations 400-600), reverting the POMG to its initial form. The performance of both paradigms is now the same as before the wind was introduced, with one fundamental difference. The difference, imperceptible in the reward, is visible in the SND. The heterogeneous agents have learned to keep the wind shielding formation, and, since this does not have a significant impact on the reward, they retain this skill even during times where it is not needed. We refer to this as *latent resilience*.

The main advantage of this latent resilience is observed when the wind is reintroduced (iteration 600). The heterogeneous agents are able to *immediately* obtain the maximum reward thanks to the latent shielding skill learned from the previous appearance of this disturbance. Furthermore, they are able to do so with less diversity than before. This is because the agents have learned to act homogeneously in the parts of the task that benefit from it, and thus find the optimal trade-off between homogeneity and heterogeneity through time (as already observed in Fig. 6c).

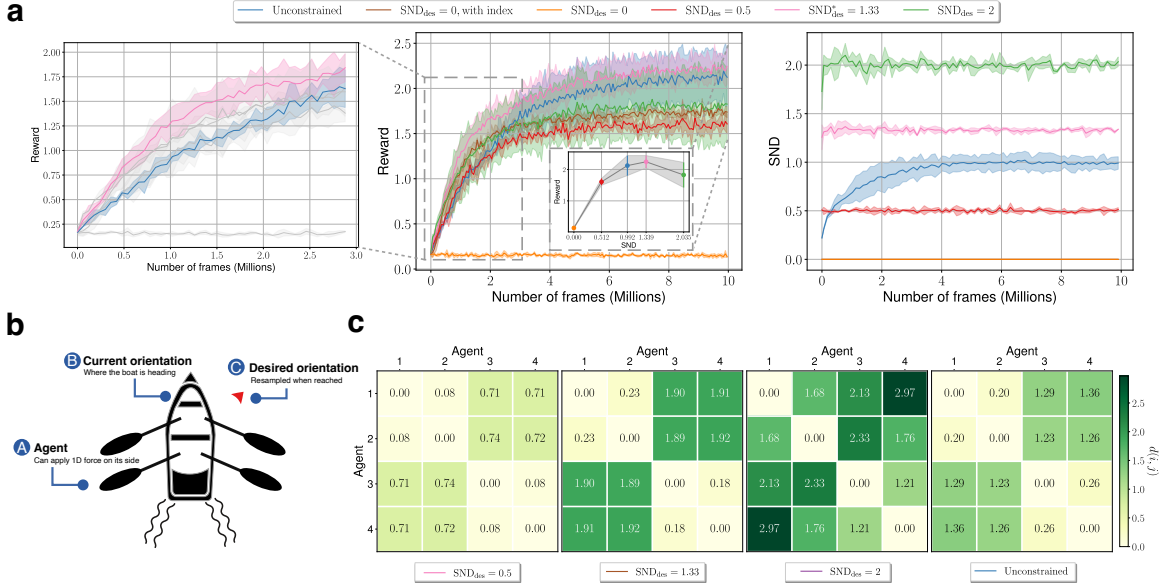


Figure 9: Controlling SND in the *Differential Steering* case study. **a**, Reward and SND for different levels of controlled diversity and for unconstrained heterogeneous and homogeneous agents. **b**, The setup of the task. **c**, behavioral distance matrices for the four heterogeneous runs. The figure shows how we can bootstrap exploration by controlling SND, and how too much or too little SND can be detrimental to the task. We report mean and standard deviation over 3 random seeds for each experiment.

5.4 Controlling SND: Differential Steering

Now that we have seen how SND can be used as an insightful diversity measurement tool, we are interested in showing how it can be used to *control* diversity. Towards this end, we employ DiCo (Bettini et al., 2024a), a diversity control paradigm that uses SND as the reference metric, constraining the multi-agent system to a desired diversity value SND_{des} . We run experiments in BenchMARL (Bettini et al., 2024b) using the MAPPO algorithm.

To illustrate the effects of controlling SND, we create the *Differential Steering* case study, shown in Fig. 9b. In this task, we position $\frac{n}{2}$ agents on each side of a rowboat. Agents can apply a 1D force that affects the orientation of the vessel through a torque. The effects of this action are flipped for agents on opposite sides, meaning a forward force applied by agents on the right will cause counter-clockwise rotation while the same force applied by agents on the left will cause a clockwise rotation. The net torque on the boat is given by the sum of the forces applied by all agents. The goal of the agents is to steer the boat to a desired orientation which is randomly resampled when reached. The agents need to do so by observing only the orientation of the boat, with knowledge of the desired orientation.

The optimal solution requires agents from different sides to take maximally different actions at all times, while agents on the same side take maximally similar actions. Since the agents do not observe which side they are on, naïve homogeneous policies are not able to solve the task. Given that actions are in the range $[-1, 1]$, two agents that converge to Delta distributions on the opposite extremes of this range will have a W_2 distance of 2. Therefore, we know that the optimal policies for this tasks consist of two behavioral

clusters of $\frac{n}{2}$ homogeneous agents, with the clusters being at behavioral distance 2. Recalling Prop. 2, we can compute the optimal system SND for this task, replacing $x = 2$ (distance between clusters), $n_c = 2$ (number of clusters), $n = 4$ (the number of agents we use in the experiment), obtaining:

$$\text{SND}(\mathbf{D}) = x \frac{n(n_c - 1)}{n_c(n - 1)} = \frac{n}{n - 1} = 1.\bar{3}.$$

We use this derived optimal value $\text{SND}_{\text{des}}^* = 1.33$ to control SND in the steering task described above. We also run experiments with: unconstrained heterogeneous agents (HetMAPPO), homogeneous agents ($\text{SND}_{\text{des}} = 0$), homogeneous agents that observe a one-hot index (to allow them to condition on this to emulate diverse behavior), insufficient controlled diversity ($\text{SND}_{\text{des}} = 0.5$), and excess controlled diversity ($\text{SND}_{\text{des}} = 2$).

From the results (Fig. 9), we confirm that agents constrained at $\text{SND}_{\text{des}} = \text{SND}_{\text{des}}^* = 1.33$ obtain the best reward and are able to bootstrap the exploration phase thanks to the diversity constraint. Unconstrained heterogeneous agents also arrive at a similar behavior with a comparable SND, but, being not controlled, take longer to converge to this solution. We also observe that controlling SND to values that are too low (0.5) or too high (2) can be detrimental to performance as the agents are forced to allocate too little or too much diversity with respect to what is needed for the task. We also note how, in this case, homogeneous agents that are able to learn multimodal behavior by conditioning on an explicit index, converge to suboptimal performance, probably due to the large discontinuity required to switch between the two behavioral roles. From the behavioral matrices in Fig. 9, we observe how $\text{SND}_{\text{des}} = 1.33$ more closely aligns with our theoretical analysis, with an inter-cluster behavioral distance ≈ 2 and an intra-cluster distance ≈ 0 .

This simple case study shows how controlling SND to a known optimal value helps in reducing the search space of unconstrained heterogeneous policies, and consequently improve the overall performance of the learned policies. While in this case we were able to compute the optimal $\text{SND}_{\text{des}}^*$ given the POMG, this may not be always possible. For more complex tasks, where such priors are not available, it is possible to apply the constraint as an inequality ($\text{SND} \geq \text{SND}_{\text{des}}$), which can still provide the aforementioned benefits without constraining the policies to a specific SND value. For more examples and extensive experiments on controlling SND, we refer the reader to Bettini et al. (2024a).

6. Discussion

Behavioral diversity is a valuable skill in collective problems. As seen from examples above, multi-agent systems that allow agents to specialize and learn unique (and potentially complementary) skills often demonstrate superior resilience towards disturbances. This work presented a novel framework for capturing this behavioral heterogeneity with a System Neural Diversity (SND) metric. Our metric is able to measure behavioral diversity among agents with stochastic policies acting in continuous state spaces. Prior work in designing such diversity metrics overlooked two key properties of interest – the ability to compare diversity across varying team sizes, and the ability to represent redundancy – thus capturing only a partial picture of heterogeneity. We show, theoretically and empirically, that SND functions as a complementary tool that enables these comparisons. Our evaluations of the

metric in a variety of didactic and realistic multi-robot tasks with different system sizes and different heterogeneity requirements establish the representational power of our metric.

The insights we draw from the various tasks indicate that SND provides a means to observe latent properties developed by a heterogeneous system during training, allowing a direct measurement of latent resilience, where other proxies, such as task performance (reward), fail to. In particular, we highlight the following insights:

1. When training heterogeneous policies, observing $\text{SND} \approx 0$ indicates that agents are behaving homogeneously, and that homogeneous training could instead be used to benefit from higher sample efficiency (Sec. 4.2.3).
2. In tasks where heterogeneous paradigms obtain higher rewards, an $\text{SND} > 0$ acts as confirmation that team heterogeneity enables performance (Sec. 5.2.1). Conversely, when heterogeneous policies add resilience to noise (Sec. 5.2.2), an observed $\text{SND} > 0$ explains this diversity while the rewards alone do not.
3. In cases where the task undergoes repeated dynamic disruptions, SND is able to expose how heterogeneity allows agents to learn and maintain latent skills that confer resilience (Sec. 5.3).
4. Controlling SND helps bootstrapping the exploration phase leading to faster convergence to optimal policies (Sec. 5.4).

It is important to note that the computational cost of SND grows quadratically with the number of agents, as all agent pairs need to be evaluated. In large-scale systems, where only neighboring agents might be relevant for the task at hand, it is possible to compute SND only within sub-groups. The topology of sub-groups can be defined in various ways, considering proximity, physicality, or objective criteria, or it can be given directly as input from human priors. Computing a global diversity value from group values is an interesting open problem that allows many design possibilities.

Further, while our evaluations suggest that heterogeneous training is a powerful paradigm for multi-agent problems, we caution that this may not always be the case. In fact, it is possible to construct pathological scenarios where, just like in nature (McKinney, 1997), excessive role specialization within a multi-agent system becomes the cause of failure during unforeseen changes. For instance, highly specialized agents are not able to dynamically swap roles, and thus, without behavioral redundancy, the system performance may suffer from agent faults. SND can still play an important role in such cases by aiding the analysis and regulation of a trade-off between specialization and redundancy.

In this context, a core open-problem is determining the behavioral diversity requirements of a given task. Early investigation in this domain suggests that it might be possible to analyze certain theoretical properties of the reward function to determine the heterogeneity need of a simple multi-agent decision process (Amir et al., 2025). In more complex settings, where such theoretical assumptions do not hold, monitoring the evolution of SND throughout training can aid in empirically understanding this task property and even aid in the formulation of novel theoretical conjectures.

Acknowledgments

This work was supported by Army Research Laboratory (ARL) Distributed and Collaborative Intelligent Systems and Technology (DCIST) Collaborative Research Alliance (CRA) W911NF-17-2-0181 and the European Research Council (ERC) Project 949940 (gAIa). We thank Alex McAvoy for reviewing a draft of this manuscript and providing helpful feedback.

Appendix A. Further Related Works

In this section, we present further related works about diversity indices from the biological domain.

Diversity indices are quantitative measures of the number of different species in a community. They are commonly used in ecology and biology to represent different aspects of diversity such as richness, evenness, and divergence (Tucker et al., 2017). They measure the distribution of n elements into c classes, where the proportion of elements in each class $h \in \{1, c\}$ is noted as p_h . Most of these indices are special cases of the *true diversity index* or *Hill numbers* (Jost, 2006). In particular, Shannon entropy (Shannon, 1948) is equivalent to the logarithm of the true diversity of order 1 and takes the following form:

$$E = - \sum_{h=1}^c p_h \log_2 p_h. \quad (5)$$

This index is adopted across a wide range of domains as it weights each class by its proportional abundance.

Despite the wide adoption of diversity indices, they are not directly applicable to measuring behavioral diversity in MARL. This is because they rely on a predefined number of classes or species, while the multi-agent systems we consider reside in a continuous time-varying behavioral (i.e., policy) space.

Appendix B. Sample efficiency of homogeneous training

In this section, we are interested in demonstrating the benefits of homogeneous training in the *Multi-Agent Goal Navigation*, when all agents are assigned the same goal and SND approaches 0. In such a case, the metric acts an important diagnostic tool, suggesting that a homogeneous training strategy should be preferred in subsequent experiment iterations in order to benefit from parameter sharing and increased sample efficiency.

To demonstrate this, we run an additional evaluation. We compare homogeneous and heterogeneous training. In Fig. 10 we show that both paradigms obtain the same performance, with the homogeneous model being significantly more time-efficient.

Appendix C. Experimental setup

Simulations are performed in the VMAS (Bettini et al., 2022) simulator. Agents are trained using the (Het)GPPO model (Bettini et al., 2023) with a fully-connected graph topology. We refer to the non-communication version of (Het)GPPO as Het(IPPO). Agent policies

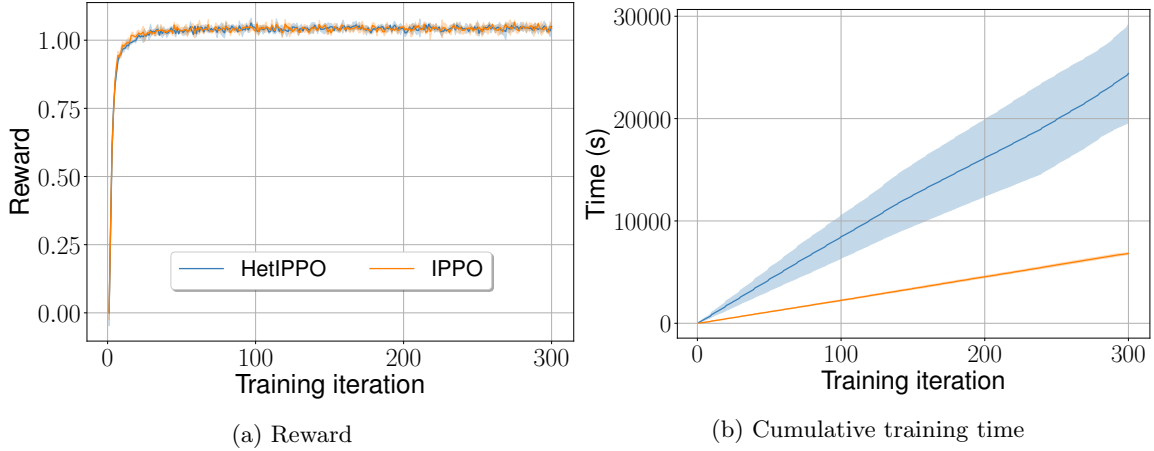


Figure 10: Comparison of homogeneous (IPPO) and heterogeneous (HetIPPO) training in the *Multi-Agent Goal Navigation* scenario with 1 goal, where heterogeneous models have been shown to have approximately 0 SND. In this case, we can observe that homogeneous training should be preferred as a more time-efficient solution thanks to its higher sample efficiency. We report mean and standard deviation over 3 random seeds for each experiment. The values are measured over 300 training iterations each performed over 600 episodes of experience.

Table 4: Training parameters for all evaluations.

Training		PPO	
Batch size	60000	ϵ	0.2
Minibatch size	4096	γ	0.99
SDG Iterations	40	λ	0.9
# Workers	5	Entropy coeff	0
# Envs per worker	50	KL coeff	0.01
Learning rate	5e-5	KL target	0.01

output a 2D continuous action distribution for each observation. The distribution is parameterized using two univariate Gaussians (one for each dimension). Training is performed in RLlib (Liang et al., 2018) using PyTorch (Paszke et al., 2019) and a multi-agent implementation of the PPO algorithm (Blumenkamp and Prorok, 2021). The training parameters used are shown in Tab. 4.

The code for the experiments used in this paper is publicly available at <https://github.com/proroklab/HetGPP0>.

References

- M. Amir, M. Bettini, and A. Prorok. When is diversity rewarded in cooperative multi-agent learning? *arXiv preprint arXiv:2506.09434*, 2025.
- M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein generative adversarial networks. In *International conference on machine learning*, pages 214–223. PMLR, 2017.
- T. Balch. Hierarchic social entropy: An information theoretic measure of robot group diversity. *Autonomous robots*, 8(3):209–238, 2000.
- M. Bettini, R. Kortvelesy, J. Blumenkamp, and A. Prorok. VMAS: A Vectorized Multi-Agent Simulator for Collective Robot Learning. In *Proceedings of the 16th International Symposium on Distributed Autonomous Robotic Systems, DARS '22*. Springer, 2022.
- M. Bettini, A. Shankar, and A. Prorok. Heterogeneous Multi-Robot Reinforcement Learning. In *Proceedings of the 22nd International Conference on Autonomous Agents and Multiagent Systems, AAMAS '23*. International Foundation for Autonomous Agents and Multiagent Systems, 2023.
- M. Bettini, R. Kortvelesy, and A. Prorok. Controlling behavioral diversity in multi-agent reinforcement learning. In *Forty-first International Conference on Machine Learning*, 2024a. URL <https://openreview.net/forum?id=qQjUgItPq4>.
- M. Bettini, A. Prorok, and V. Moens. Benchmarl: Benchmarking multi-agent reinforcement learning. *Journal of Machine Learning Research*, 25(217):1–10, 2024b. URL <http://jmlr.org/papers/v25/23-1612.html>.
- J. Blumenkamp and A. Prorok. The Emergence of Adversarial Communication in Multi-Agent Reinforcement Learning. In *Conference on Robot Learning*, pages 1394–1414. PMLR, 2021.
- M. W. Cadotte, K. Carscadden, and N. Mirotchnick. Beyond species: functional diversity and the maintenance of ecological processes and services. *Journal of applied ecology*, 48(5):1079–1087, 2011.
- K. Chatzilygeroudis, A. Cully, V. Vassiliades, and J.-B. Mouret. Quality-Diversity Optimization: a novel branch of stochastic optimization. In *Black Box Optimization, Machine Learning, and No-Free Lunch Theorems*, pages 109–135. Springer, 2021.
- L. Chenghao, T. Wang, C. Wu, Q. Zhao, J. Yang, and C. Zhang. Celebrating diversity in shared multi-agent reinforcement learning. *Advances in Neural Information Processing Systems*, 34, 2021.
- F. Christianos, G. Papoudakis, M. A. Rahman, and S. V. Albrecht. Scaling multi-agent reinforcement learning with selective parameter sharing. In *International Conference on Machine Learning*, pages 1989–1998. PMLR, 2021.
- S. Doncieux and J.-B. Mouret. Behavioral diversity measures for evolutionary robotics. In *IEEE congress on evolutionary computation*, pages 1–8. IEEE, 2010.

- W. Fu, C. Yu, Z. Xu, J. Yang, and Y. Wu. Revisiting Some Common Practices in Cooperative Multi-Agent Reinforcement Learning. In *International Conference on Machine Learning*, pages 6863–6877. PMLR, 2022.
- C. Gini. Variabilità e mutabilità. *Reprinted in Memorie di metodologica statistica (Ed. Pizetti E)*, 1912.
- J. K. Gupta, M. Egorov, and M. Kochenderfer. Cooperative multi-agent control using deep reinforcement learning. In *International conference on autonomous agents and multiagent systems*, pages 66–83. Springer, 2017.
- E. Hellinger. Neue begründung der theorie quadratischer formen von unendlichvielen veränderlichen. *Journal für die reine und angewandte Mathematik*, 1909(136):210–271, 1909.
- S. Hu, C. Xie, X. Liang, and X. Chang. Policy Diagnosis via Measuring Role Diversity in Cooperative Multi-agent RL. In *International Conference on Machine Learning*, pages 9041–9071. PMLR, 2022.
- M. Jaderberg, W. M. Czarnecki, I. Dunning, L. Marris, G. Lever, A. G. Castaneda, C. Beattie, N. C. Rabinowitz, A. S. Morcos, A. Ruderman, et al. Human-level performance in 3D multiplayer games with population-based reinforcement learning. *Science*, 364(6443):859–865, 2019.
- J. Jiang and Z. Lu. The emergence of individuality. In *International Conference on Machine Learning*, pages 4992–5001. PMLR, 2021.
- L. Jost. Entropy and diversity. *Oikos*, 113(2):363–375, 2006.
- L. P. Kaelbling, M. L. Littman, and A. R. Cassandra. Planning and acting in partially observable stochastic domains. *Artificial intelligence*, 101(1-2):99–134, 1998.
- C. Kaplanis, M. Shanahan, and C. Clopath. Policy Consolidation for Continual Reinforcement Learning. In *International Conference on Machine Learning*, pages 3242–3251. PMLR, 2019.
- S. R. Kellert. *The value of life: Biological diversity and human society*. Island press, 1997.
- L. Li, A. Martinoli, and Y. S. Abu-Mostafa. Learning and measuring specialization in collaborative swarm systems. *Adaptive Behavior*, 12(3-4):199–212, 2004.
- E. Liang, R. Liaw, R. Nishihara, P. Moritz, R. Fox, K. Goldberg, J. Gonzalez, M. Jordan, and I. Stoica. RLlib: Abstractions for distributed reinforcement learning. In *International Conference on Machine Learning*, pages 3053–3062. PMLR, 2018.
- X. Liu, H. Jia, Y. Wen, Y. Hu, Y. Chen, C. Fan, Z. Hu, and Y. Yang. Towards unifying behavioral and response diversity for open-ended learning in zero-sum games. *Advances in Neural Information Processing Systems*, 34:941–952, 2021.
- A. Lupu, B. Cui, H. Hu, and J. Foerster. Trajectory diversity for zero-shot coordination. In *International conference on machine learning*, pages 7204–7213. PMLR, 2021.

- A. Mahajan, T. Rashid, M. Samvelyan, and S. Whiteson. Maven: Multi-agent variational exploration. *Advances in neural information processing systems*, 32, 2019.
- M. A. Masood and F. Doshi-Velez. Diversity-inducing policy gradient: Using maximum mean discrepancy to find a set of diverse policies. *arXiv preprint arXiv:1906.00088*, 2019.
- K. R. McKee, J. Z. Leibo, C. Beattie, and R. Everett. Quantifying the effects of environment and population diversity in multi-agent reinforcement learning. *Autonomous Agents and Multi-Agent Systems*, 36(1):1–16, 2022.
- M. L. McKinney. Extinction Vulnerability and Selectivity: Combining Ecological and Paleontological Views. *Annual Review of Ecology and Systematics*, 28(1):495–516, 1997. doi: 10.1146/annurev.ecolsys.28.1.495.
- K. Menger. Statistical Metrics. *Proceedings of the National Academy of Sciences*, 28(12): 535–537, 1942. doi: 10.1073/pnas.28.12.535. URL <https://www.pnas.org/doi/abs/10.1073/pnas.28.12.535>.
- J. Parker-Holder, A. Pacchiano, K. M. Choromanski, and S. J. Roberts. Effective diversity in population based reinforcement learning. *Advances in Neural Information Processing Systems*, 33:18050–18062, 2020.
- A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.
- A. Prorok, M. A. Hsieh, and V. Kumar. The impact of diversity on optimal control policies for heterogeneous robot swarms. *IEEE Transactions on Robotics*, 33(2):346–358, 2017.
- A. Prorok, M. Malencia, L. Carlone, G. S. Sukhatme, B. M. Sadler, and V. Kumar. Beyond robustness: A taxonomy of approaches towards resilient multi-robot systems. *arXiv preprint arXiv:2109.12343*, 2021.
- T. Rashid, M. Samvelyan, C. Schroeder, G. Farquhar, J. Foerster, and S. Whiteson. Qmix: Monotonic value function factorisation for deep multi-agent reinforcement learning. In *International Conference on Machine Learning*, pages 4295–4304. PMLR, 2018.
- C. E. Shannon. A mathematical theory of communication. *The Bell system technical journal*, 27(3):379–423, 1948.
- K. O. Stanley, J. Clune, J. Lehman, and R. Miikkulainen. Designing neural networks through neuroevolution. *Nature Machine Intelligence*, 1(1):24–35, 2019.
- S. Sukhbaatar, R. Fergus, et al. Learning multiagent communication with backpropagation. *Advances in neural information processing systems*, 29, 2016.
- R. S. Sutton and A. G. Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

- C. M. Tucker, M. W. Cadotte, S. B. Carvalho, T. J. Davies, S. Ferrier, S. A. Fritz, R. Grenyer, M. R. Helmus, L. S. Jin, A. O. Mooers, et al. A guide to phylogenetic metrics for conservation, community ecology and macroecology. *Biological Reviews*, 92(2):698–715, 2017.
- P. Twu, Y. Mostofi, and M. Egerstedt. A measure of heterogeneity in multi-agent systems. In *2014 American Control Conference*, pages 3972–3977. IEEE, 2014.
- L. N. Vaserstein. Markov processes over denumerable products of spaces, describing large systems of automata. *Problemy Peredachi Informatsii*, 5(3):64–72, 1969.
- T. Wang, J. Wang, Y. Wu, and C. Zhang. Influence-Based Multi-Agent Exploration. In *International Conference on Learning Representations*, 2019.
- T. Wang, H. Dong, V. Lesser, and C. Zhang. ROMA: Multi-Agent Reinforcement Learning with Emergent Roles. In *International Conference on Machine Learning*, pages 9876–9886. PMLR, 2020.
- T. Wang, T. Gupta, B. Peng, A. Mahajan, S. Whiteson, and C. Zhang. RODE: learning roles to decompose multi- agent tasks. In *Proceedings of the International Conference on Learning Representations*, 2021.
- B. L. Welch. The generalization of ‘STUDENT’S’ problem when several different population variances are involved. *Biometrika*, 34(1-2):28–35, 1947.
- A. W. Woolley, I. Aggarwal, and T. W. Malone. Collective intelligence and group performance. *Current Directions in Psychological Science*, 24(6):420–424, 2015.
- Y. Yu, H. Jiang, and Z. Lu. Informative Policy Representations in Multi-Agent Reinforcement Learning via Joint-Action Distributions. *arXiv e-prints*, pages arXiv–2106, 2021.