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

Matteo Bettini[a,1], **Ajay Shankar**[a], **and Amanda Prorok**[a]

[a]Department of Computer Science and Technology, University Of Cambridge, Cambridge CB3 0FD, United Kingdom

This manuscript was compiled on May 4, 2023

Evolutionary science provides evidence that diversity confers resilience. 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, individual agents may develop diverse behaviors, resulting in emergent complementarity that benefits the system. Despite this feat, there is a surprising lack of tools that measure behavioral diversity in systems of learning agents. Such techniques would pave the way towards understanding the impact of diversity in collective resilience and performance. In this paper, we introduce System Neural Diversity (SND): a measure of behavioral heterogeneity for multi-agent systems where agents have stochastic policies. We discuss and prove its theoretical properties, and compare it with alternate, state-of-the-art behavioral diversity metrics used in cross-disciplinary domains. Through simulations of a variety of multi-agent tasks, we show how our metric constitutes an important diagnostic tool to analyze latent properties of behavioral heterogeneity. By comparing SND with task reward in static tasks, where the problem does not change during training, we show that it is key to understanding the effectiveness of heterogeneous vs homogeneous agents. In dynamic tasks, where the problem is affected by repeated disturbances during training, we show that heterogeneous agents are first able to learn specialized roles that allow them to cope with the disturbance, and then retain these roles when the disturbance is removed. SND allows a direct measurement of this latent resilience, while other proxies such as task performance (reward) fail to.

Diversity | Multi-Agent Reinforcement Learning | Heterogeneity Measure

Diversity is fundamental to life and commonplace in natural systems (1). In studying natural collective systems, biologists and ecologists have shown how *functional diversity* (2), which characterizes species with respect to differences in their behavioral traits, impacts ecosystem survival. Indeed, the lack of diversity has been shown to have devastating effects (3). The scientific field devoted to the analysis of diversity is, therefore, of paramount importance.

To study behavioral diversity, we first need to be able to measure it. While a variety of measures of diversity exist for natural systems (4), there is a lack of works studying *behavioral diversity* in engineered (artificial) systems. In this research, we are interested in measuring diversity and its impact in multi-agent systems, where collective intelligence is needed to complete a task. Collective intelligence has been shown to exist in human teams (5) and not to be dependent on maximum individual intelligence or average individual intelligence, but to be highly correlated with the social sensitivity of team members. It has been observed that "teams with moderately diverse cognitive styles usually perform better than those that are very similar in cognitive styles and also those that differ too much" (6). Several similarities can be drawn between the realm of humans and that of learning agents, as these characteristics are also observable in the context of multi-agent learning. A metric for diversity in artificial multi-agent systems would allow us to measure previously unobservable aspects of learned collective intelligence and thus better analyze the impact of diversity.

Multi-agent learning is a powerful computational paradigm which can provide effective solutions to hard problems (7). It has been employed in a variety of domains including rule compliance (8), robotics (9), traffic-control (10), and smart city energy management (11). Among learning paradigms, Multi-Agent Reinforcement Learning (MARL) (12) stands out due to its resemblance to learning in natural systems. In MARL, agents learn from experiences and rewards collected through interactions with the world. Such interactions are performed using policies, which are behavioral functions, usually represented as neural networks, mapping an agent's observation to a probabilistic action distribution. Traditional MARL algorithms constrain policies to be identical for agents with the same objective in order to improve training sample efficiency (13–15). This causes the agents to become behaviorally homogeneous. On the other hand, when policies can diverge, and thus output different action distributions for the same observation, we refer to them as heterogeneous. MARL paradigms for training heterogeneous policies have recently been proposed (16) and shown to have resilience benefits.

While current methods allow to enable heterogeneity in MARL, they lack a structured measure of behavioral diversity. The development of a reliable behavioral heterogeneity metric

Significance Statement

Behavioral diversity in multi-agent learning can be non-trivial to quantify. Yet, a reliable metric for it can help guide the design and regulation of system heterogeneity, even in cases where its advantages may not be directly observable. We introduce System Neural Diversity (SND), a metric that addresses this gap. Modeling heterogeneity as behavioral dispersion, SND is the first metric to allow a diversity comparison across multi-agent systems of different sizes and a quantification of behavioral redundancy. This makes SND a reliable proxy measure for task heterogeneity requirements. In cases where agents learn heterogeneous behaviors for no immediately apparent reason, SND can detect the presence of latent resilience skills (that become useful when the system is put under stress).

Author contributions: M.B. and A.P. formulated the main ideas; M.B. implemented the system and performed all evaluations; A.S. and A.P. provided guidance in the analysis of the evaluations. M.B., A.S., and A.P. wrote the paper. A.P initiated the project and provided the funding.

The authors declare no competing interest.

[1]To whom correspondence should be addressed. E-mail: mb2389@cl.cam.ac.uk

is a necessary stepping stone that would allow: (1) an analysis of the impact of diversity, and (2) its control. Current solutions designed to control diversity in MARL (17–20) tackle problem (2) through proxy measures (e.g., reward). This approach, however, is inefficient as it is not able to measure heterogeneity directly and thus it cannot model the true relationship between diversity and performance. For these reasons, we are interested in developing a system behavioral heterogeneity metric. In addition to satisfying the formal properties of a *metric*, we define two key properties: (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) would allow us to compare heterogeneity across different team sizes and Property (2) would allow the metric to measure behavioral redundancy.

In this paper, we introduce System Neural Diversity (SND), a measure of behavioral heterogeneity in multi-agent learning. With the term *neural diversity* we refer to behavioral heterogeneity between learning agents, as their behaviors are represented by neural networks. To compute SND, we firstly define a pairwise inter-agent behavioral distance. Pairwise distances are then aggregated into a system-wide SND. We prove that SND follows Property (1) and (2). Furthermore, we juxtapose SND with Hierarchic Social Entropy (HSE) (21), a state-of-the-art behavioral heterogeneity metric used in robotics, and show that HSE does not follow the properties defined, thus providing a complementary view of behavioral diversity.

We perform a set of evaluations using the proposed metric in order 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) multi-agent problems yield the following insights:

1. An SND of 0 indicates that homogeneous training should be preferred to benefit from sample efficiency.
2. An SND greater than 0, when heterogeneous paradigms obtain higher rewards than homogeneous ones, indicates that heterogeneity enables performance.
3. An SND greater than 0, when homogeneous and heterogeneous paradigms obtain the same reward, indicates that heterogeneity enables resilience.
4. In cases where the task undergoes repeated dynamic disruptions, SND is able to expose how heterogeneity allows agents to learn and maintain latent skills for resilience.

SND allows us to make these insights by providing a structured measure of behavioral heterogeneity, a quantity frequently overlooked in previous work.

Contributions. We claim the following contributions:

- We introduce System Neural Diversity, a metric of behavioral heterogeneity for multi-agent learning.
- We study the impact of system neural diversity, measuring the previously unobservable benefits of heterogeneity in collective static and dynamic tasks.

Related Work. In this section, we give an overview of existing diversity measures used in different domains.

Diversity Measures in MARL. Diversity in MARL has gained increasing attention, with recent works proposing architectures to enable diversity in multi-agent teams. In HetGPPO (16) the authors present an taxonomy of diversity classes and use it to classify existing heterogeneous MARL solutions. We refer the reader to this work for an overview of available heterogeneous MARL methods and their differences. While some solutions are able to control diversity as a function of training performance (17–20), they do so without being able to measure the resulting heterogeneity (i.e., open-loop control). The problem of developing a reliable diversity metric is frequently overlooked. Such a metric can then provide the feedback signal necessary to control diversity (i.e., for closed-loop control).

The works that most relate to this research are the ones that tackled the problem of developing a diversity metric for MARL. In (22) the authors introduce a diversity metric that uses sampled discrete actions from agents' policies to approximate their distributions. Total variation distance is used to compute the divergence among the resulting discrete distributions. Similarly, in (23), behavioral distances are obtained using sampled action datapoints. These methods use approximated distributions, leading to a decrease in the metric's accuracy. Another action-based diversity metric was introduced in (24). The authors propose to use symmetric Kullback–Leibler (KL) divergence to measure behavioral distances. However, symmetric KL does not satisfy the triangle inequality property, and hence, it is not a statistical metric. In (25) 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. Lastly a behavioral metric in embedding space is proposed in (26), where mappings between policies and embeddings are learned.

In parallel, a branch of single-agent population-based RL has started to employ methods from Quality-Diversity optimization (27, 28). This is a population-based evolutionary technique, in which populations are 'bred' to maximize both performance and diversity. A range of techniques have been proposed to analyze diversity in this domain (29, 30). In (31), 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 among sampled actions over randomly sampled states, which can lead to policies being evaluated in states they have not been trained on.

Diversity Index. 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 (4). They measure the distribution of n elements into c classes, where the proportion of elements in each class $h \quad \{1, c\}$ is noted as p_h. Most of these indices are special cases of the *true diversity index* or *Hill numbers* (32). In particular, Shannon entropy (33) is equivalent to the logarithm of the true diversity of order 1 and takes the following form:

$$E = -\sum_{h=1}^{c} p_h \log_2 p_h. \qquad [1]$$

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.

Diversity Measures in Robotics. When agents are embodied as robots, physical differences can emerge, leading to different capabilities. In (34), Prorok et al. 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 be identified. Diversity matrices can then be compared to measure the difference between two agents.

Behavioral differences, however, are not always correlated with physical ones and need a dedicated measure in policy space in order to be captured. Li et al. (35) propose a behavioral specialization metric that is, however, obtained through a correlation with team performance. Twu et al. (36) propose a computation-light heterogeneity metric that assumes a fixed number of diversity classes. Close to our work, Balch proposed Hierarchic Social Entropy (HSE) (21). This measure 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 (Eq. 1) 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

$$\text{HSE} = \int_0^\infty E(l)dl. \qquad [2]$$

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 our proposed metric in the following sections, as the two measures provide complementary tools to assess diversity.

Problem Formulation

We now formulate the multi-agent MARL problem analyzed in this work. To do so, we first introduce the multi-agent extension of a Partially Observable Markov Decision Process (POMDP) (37).

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

$$\left\langle N, S, \{O_i\}_{i\ N}, \{\sigma_i\}_{i\ N}, \{A_i\}_{i\ N}, \{R_i\}_{i\ N}, T, \gamma \right\rangle,$$

where $N = \{1, \ldots, n\}$ denotes the set of agents, S is the state space, shared by all agents and, $\{O_i\}_{i\ N}$ and $\{A_i\}_{i\ N}$ are the observation and action spaces, with $O_i \quad S, \quad i \quad N$. Further, $\{\sigma_i\}_{i\ N}$ and $\{R_i\}_{i\ N}$ are the agent observation and reward functions (potentially identical for all agents), such that $\sigma_i : S \to O_i$, and, $R_i : S \times \{A_i\}_{i\ N} \times S \to \mathbb{R}$. T is the stochastic state transition model, defined as $T : S \times \{A_i\}_{i\ N} \times S \to [0, 1]$, which outputs the probability $T(s^t, \{a_i^t\}_{i\ N}, s^{t+1})$ of transitioning to state s^{t+1} given the current state s^t and actions $\{a_i^t\}_{i\ N}$. Lastly, 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 (16). Using GPPO we perform homogeneous training via parameter sharing, thus $\theta = \theta_1 = \ldots = \theta_n$. 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.

In this work, we do not consider homogeneous policies which leverage *explicit behavioral typing* (16) to emulate heterogeneous policies. Such a typing occurs when a shared homogeneous policy is able to type agents based on a unique identifier concatenated to the input. Using this trick, a homogeneous policy can learn a multi-behavioral policy, conditioned on those unique inputs (e.g., indices). For an in-depth discussion on the drawbacks of this technique and other behavioral typing techniques, we refer the reader to (16).

Objective. The goal of this work is to measure the neural diversity (behavioral heterogeneity) in a system of leaning agents with heterogeneous policies π_{θ_i}. Our objective is then to develop a System Neural Diversity metric,

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

that takes as input the agents' policies and outputs a scalar value representing the behavioral diversity of the system.

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 maintaining 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. Firstly, we define a measure of inter-agent pairwise **behavioral distance**. We structure pairwise behavioral distances obtained with this measure in a behavioral distance matrix, which gives an overview of the distribution of agents' policies in behavioral space. Secondly, we aggregate the behavioral distance matrix into a single neural diversity value, which represents the **neural diversity metric** of the whole system. This metric is used to measure behavioral heterogeneity of a learning multi-agent system. In the following, we discuss the choice of the functions used for these two tasks, as well as showcasing and proving the respective properties of interest.

Behavioral Distance. Heterogeneity is a collective concept. Thus, 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. Therefore, we need to think about the simplest possible case of measuring behavioral

diversity and answer the question "How should the behavior of two agents be compared?". Motivated by this, we need to develop a mathematical metric that measures the behavioral distance of two agents $d : N \times N \to \mathbb{R}_{\geq 0}$. The behavioral distance of agents i and j is then given by $d(i, j)$. We want $d(i, j)$ to follow the properties of a mathematical metric (38).

Definition 1 (Properties of the behavioral distance metric (38))**.** *For the distance d to be a metric, it has to satisfy the following properties $i, j, k \quad N$:*

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):* $d(i, j) \leq d(i, k) + d(k, j)$

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

Behavioral distance among two agents can be measured through their policies. Since we are comparing policies of different agents, we assume that all agents have the same observation space $O = O_1 = \ldots = O_n$ and action space $A = A_1 = \ldots = A_n$. This allows for physical and behavioral diversity while making sure that the policies have the same input and output spaces. Heterogeneous policies will map at least one observation $o \quad O$ to two different action distributions $\pi_{\theta_i}(o) = \pi_{\theta_j}(o)$, while homogeneous ones will output the same distribution $\pi_\theta(o)$. Note that differences in parameter space do not necessarily map to differences in behavioral space (28, 39), 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 $\pi_{\theta_j}(o)$.

Our goal is to develop a distance

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

where a function f, providing a distance between two policies for a given observation, is evaluated over all possible observations. In order to develop such a distance we have to tackle two main tasks: (1) f has to be a measure between the two probability distributions outputted by the polices 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.

Distance for Stochastic Policies. Stochastic policies map a given observation to an action distribution. Thus, a comparison of two agent behaviors can be made through a comparison of their action distributions over a set of observations. To avoid sampling from such distributions during distance evaluation, we can use a closed-form statistical distance computed over their parameters. 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*. Among divergences (and particularly f-divergences) the Kullback–Leibler (KL) divergence has been used extensively in machine learning applications. Despite its wide adoption, this divergence has several practical problems,

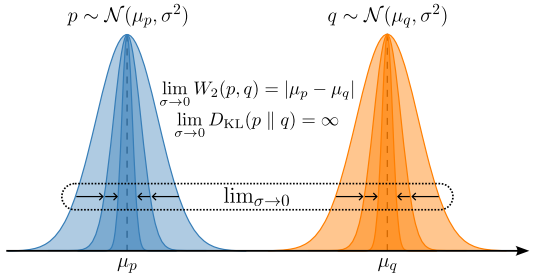


Fig. 1. Wasserstein metric $W_2(p, q)$ and KL divergence $D_{\mathrm{KL}}(p \parallel q)$ of two univariate distributions p, q as their standard deviation approaches 0. The value of W_2 in this scenario approaches the absolute difference of their means while KL divergence approaches infinity.

For instance, even if it can be made symmetric, it does not follow triangle inequality (prop. 4 of a metric). Triangle inequality is useful for determining an upper bound estimate of the behavioral distance of two agents when their respective distances to a third agent are available. Among metrics, on the other hand, we focus our attention on the Wasserstein metric (40) and on the Hellinger distance (41). If polices π_{θ_i} output multivariate Gaussian action distributions, as in this work, these metrics can be computed in closed-form using the distributions' parameters. They measure complementary aspects of the distance between two distributions p and q. The Wasserstein metric measures the minimum cost required to move all the probability mass from p to q in an optimal transport problem formulation. The Hellinger distance increases inversely proportionally to the probability that p assigns to every interval to which q assigns a positive probability. This distance is bounded between 0 and 1 and assumes its maximum value when the two distributions do not have any overlapping probability mass.

Fig. 1 depicts an illustrative example to further elucidate the differences between the Wasserstein metric and KL divergence. In this example, $p \quad N(\mu_p, \sigma^2)$ and $q \quad N(\mu_q, \sigma^2)$ are univariate Gaussian distributions. $W_2(p, q)$ represents the Wassertein metric and $D_{\mathrm{KL}}(p \quad q)$ 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), $D_{\mathrm{KL}}(p \quad q) \to \infty$ while $W_2(p, q) \to |\mu_p - \mu_q|$. This shows that, in the case the distributions assign increasing probability mass to their mean, 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 (42) where the model architecture significantly benefited from the use of W_2 over KL.

Following the reasons above, we use the Wasserstein metric in this work to measure the probability distance between two policies, since, unlike Hellinger, it provides an unbounded measure ranging to infinity (as diversity can)*. The resulting behavioral distance takes the following form:

$$d(i, j) = \int_O W_2(\pi_{\theta_i}(o), \pi_{\theta_j}(o))\, do. \qquad [3]$$

Observation Sampling for Distance Evaluation. Now that we have chosen the distance used to compare policies, we turn our

* In our code we provide implementations of several metrics and distances, including Hellinger, which can alternatively be used.

attention to the integral in Eq. 3 used to compute the distance over the set of all observations O. Since in this work we are considering continuous observation spaces, we need to develop a sampling technique that allows us to create a finite subset of observations $B \subset O$, over which we can evaluate the behavioral distance between policies $\pi_{\theta_i}(o)$. We choose to create 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 $(\{o_i^t\}_{i \in N}, \{a_i^t\}_{i \in N}, \{o_i^{t+1}\}_{i \in N})$ with time $t \in [0, T]$. We denote the sets of agents' observations and actions at time t as $\mathbf{o}^t = \{o_i^t\}_{i \in N}$ and $\mathbf{a}^t = \{a_i^t\}_{i \in N}$, respectively. Thus, we construct $B = \{\mathbf{o}^t\}_{t \in [0,T]}$.

When building 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 (43) provides a high likelihood that these observations were seen previously during training. While it has zero sampling bias, it is know to have a high variance in the states visited. We reduce variance 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 computational cost of evaluating the distance. Therefore, we can write the final formulation of the behavioral distance d as:

$$d(i,j) = \frac{1}{|B||N|} \sum_{\mathbf{o}^t \in B} \sum_{k \in N} W_2(\pi_{\theta_i}(o_k^t), \pi_{\theta_j}(o_k^t)). \qquad [4]$$

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

Behavioral Distance Matrix. Having defined the behavioral distance $d(i,j)$ used in this work, we now structure the inter-agent distances in a behavioral distance matrix. Let

$$\mathbf{d}(i) = [d(i,1), \ldots, d(i,n)] \qquad n = |N|$$

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

$$\mathbf{D} = \left[\mathbf{d}(1) \ , \ldots, \mathbf{d}(n) \ \right].$$

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 $j \in N \setminus \{1\}$ and $d(i,j) = 0$, for all $i,j \in N \setminus \{1\}$, we get a contribution of $\sum_{j \in N} d(1,j) = \frac{x(n-1)}{n}$ for agent 1 and a contribution of $\sum_{j \in N} d(i,j) = \frac{x}{n}$ for all other agents $i \in N \setminus \{1\}$. By then calculating the relative fraction over the resulting values, we can express the percentage agent contributions to the team heterogeneity.

Lastly, as in Balch's HSE, we can define ϵ-homogeneity.

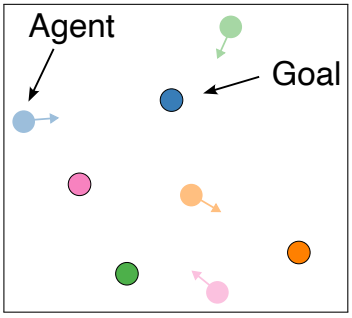


Fig. 2. *Multi-Agent Goal Navigation* example. Agents are spawned at random positions in a 2D workspace and take velocity actions (colored arrows) to reach their assigned goal, also spawned at random positions.

Definition 2 (ϵ-homogeneity (21))**.** *A multi-agent system is ϵ-homogeneous if and only if for all $i,j \in N$, $d(i,j) \leq \epsilon$.*

Example: Multi-Agent Goal Navigation. Let us now look at an experimental case study to complement the theoretical discussion on behavioral distance. This example will be used throughout other parts of this section to provide experimental evidence that supports and illustrates the theoretical insights.

In this *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 their 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. Agents are trained using HetIPPO (HetGPPO (16) without communication) and the scenario is created in the VMAS simulator[†] (44).

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

- *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. The matrices also show that, for instance, in the 3 goals case, agents $\{1, 2\}$ are 0.15-homogeneous, while in the 1 goal case, the entire system is 0.08-homogeneous. This shows that the behavioral distance matrix already provides an important diagnostic tool when assessing the diversity of a multi-agent system.

System Neural Diversity. Inter-agent behavioral distance constitutes a fine-grained diagnostic tool to assess system diversity. However, the behavioral distance matrix has size $n \times n$ and thus grows quadratically with the number of agents. This

[†] https://github.com/proroklab/VectorizedMultiAgentSimulator

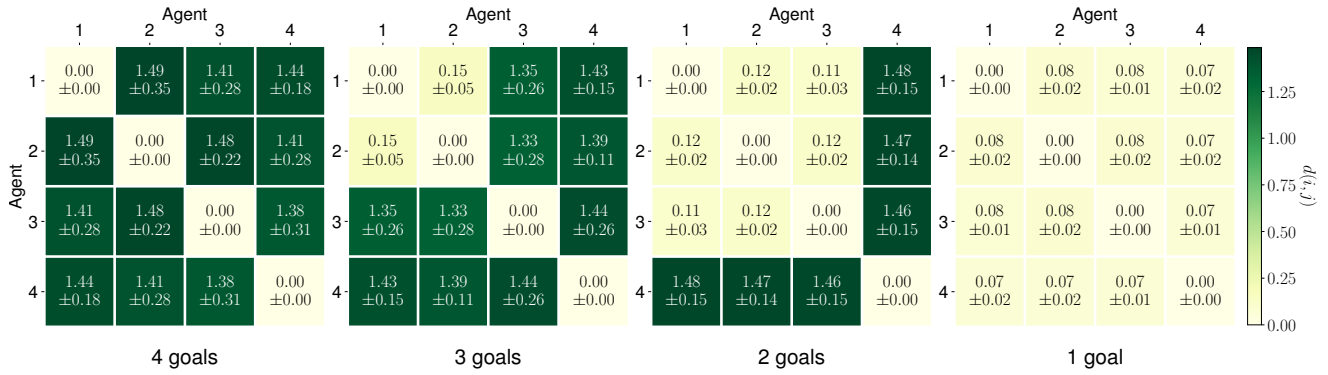


Fig. 3. Behavioral distance matrices for the four experiments run on *Multi-Agent Goal Navigation*. We can observe how, when agents are assigned the same goal, they become homogeneous and thus reduce their behavioral distance. We report mean and standard deviation for $d(i, j)$ over 5 random seeds for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

dependence on the number of agents is not practical when we want to read system heterogeneity at a glance. Therefore, we need to aggregate the information contained in the behavioral distance matrix into a single scalar output. Balch's HSE (21) is a metric introduced for this purpose, which clusters agents hierarchically based on the behavioral distance matrix and then computes the Shannon entropy at each hierarchical level by treating clusters as species. The entropies at the different levels are then summed together to obtain a single value. HSE is a valuable solution to compute system diversity, but presents some undesirable properties that limit its information content in some use cases. We will highlight and contrast these properties with the ones of our proposed metric.

In the following, we introduce System Neural Diversity (SND), a diversity metric that maps a behavioral distance matrix to a single diversity value. SND takes inspiration from the Gini coefficient (45) 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 would like to represent diversity as behavioral dispersion using distances $d(i, j)$. Furthermore, we want to let SND values range form zero to infinity. The proposed SND takes the form:

$$\mathrm{SND}(\mathbf{D}) = \frac{2\sum_{i=1}^{n}\sum_{j=i+1}^{n} d(i,j)}{n(n-1)}, \qquad [5]$$

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.

Invariance in the Number of Equidistant Agents. When measuring heterogeneity, a core question could be raised: "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 economical 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.*

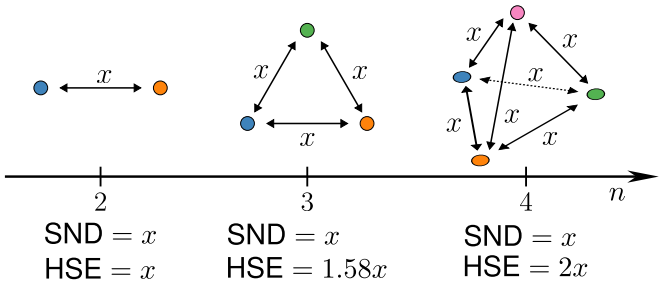


Fig. 4. Invariance in the number of equidistant agents. Agents (circles) at behavioral distance x are represented in behavioral space for $n = 2, 3, 4$. The reported values of SND and HSE show that SND remains invariant in the number of equidistant agents, while HSE increases.

Table 1. 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

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

Proof. Given that $d(i, j) = x$, i, j N with $i = j$, 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 $\mathrm{SND}(\mathbf{D}) = \frac{2n(n-1)x}{2n(n-1)} = x$ which is not dependent on n. □

Fig. 4 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.

To showcase the property experimentally, we implement HSE (using our Wasserstein behavioral distance to account for stochastic policies) and run experiments in the n goals multi-agent navigation scenario (Fig. 2) with increasing values of n. Tab. 1 shows that SND value remains invariant to the increasing addition of new agents with new goals, while HSE grows.

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. In any of these cases, redundancy in the number of agents with the same role may provides resilience in the event of unmodeled disruptions. While it seems intuitive that behavioral redundancy should lead to a decrease in system diversity, this aspect is not captured in HSE. HSE clusters 0-homogeneous 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. 5)**.** *Given a behavioral distance matrix* $\mathbf{D}$*, where* n *agents are divided equally in* n_c *behavioral clusters* $C = \{C_1, \dots, C_{n_c}\}$ *with* $|C_h| = \frac{n}{n_c}$ $\quad \mathbb{N}_{>0}$*,* $C_h \quad C$*,* $c(i) : N \to C$ *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*

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

Proof. Given that the 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 $\mathrm{SND}(\mathbf{D}) = \frac{2xn^2(n_c-1)}{2n_c n(n-1)} = x\frac{n(n_c-1)}{n_c(n-1)}$, 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, if n agents are equally distributed in n_c behavioral clusters, SND will increase when n_c is increased and decrease when n is increased. Fig. 5 depicts this property by showing the SND and HSE as a function of n for $n_c = 2, 3, 4$.

To showcase the property experimentally, we modify the *Multi-Agent Goal Navigation* scenario (Fig. 2) 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. 2 shows that the SND value decreases with the number of agents, following the described function. On the other hand, HSE not only does not decrease, but slightly increases, as agents with the same goal will be at behavioral distance epsilon ≈ 0.

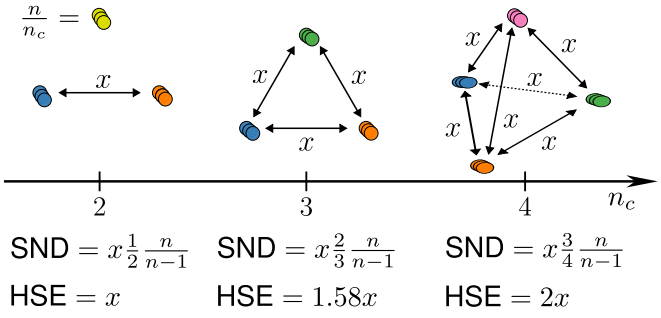


Fig. 5. Redundancy measure. Agents (circles) are divided into n_c behavioral clusters (circle stacks) at distance x in behavioral space for $n_c = 2, 3, 4$. The reported values show that, for each value of n_c, SND decreases with n, while HSE is invariant to n. Fig. 4 is a special case of this figure where $n = n_c$.

Table 2. 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

Evaluations

We are now interested in using the SND metric as an insightful heterogeneity index during a MARL training phase. For this, we run a series of experiments on various multi-robot tasks. The simulation environments representing these tasks are partly new implementations and partly adapted from existing environments in the VMAS benchmark set (16, 44). 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.).

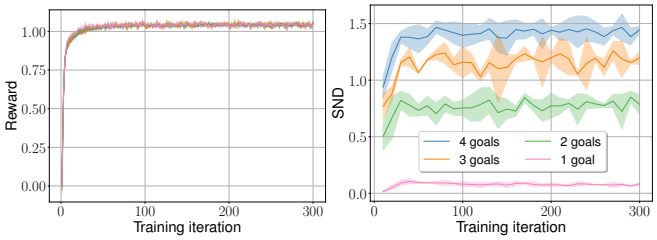
Experimental Setup. Simulations are performed in the VMAS (44) simulator. Agents are trained using the (Het)GPPO model (16) with a fully-connected graph topology. We refer to the non-communication version of (Het)GPPO as Het(IPPO). Agent policies output a 2D continuous action distribution for each observation. The distribution is parameterized using two univariate Gaussians (one for each dimension). Training is performed in RLlib (46) using PyTorch (47) and a multi-agent implementation of the PPO algorithm (48). The training parameters used are shown in Tab. 3.

Static Tasks. We refer with the term *static tasks* to multi-agent problems modeled by a fixed[‡] POMG. This is the type of prob-

[‡] A POMG that does not vary throughout the learning process

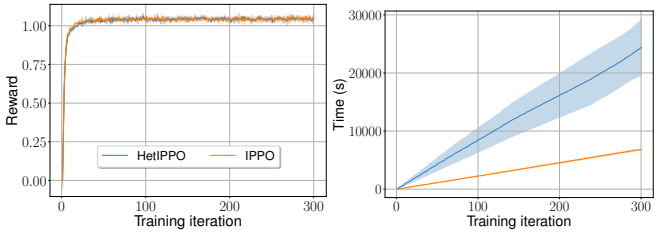
Table 3. Training parameters for all evaluations.

Training		PPO	
Batch size	60000		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



(a) Reward **(b)** System Neural Diversity

Fig. 6. SND in the *Multi-Agent Goal Navigation* scenario (Fig. 2). 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 3 random seeds for each experiment. The values are measured over 300 training iterations each performed over 600 episodes of experience.



(a) Reward **(b)** Cumulative training time

Fig. 7. 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 (Fig. 6b). 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.

lem traditionally used to benchmark MARL algorithms (49). In this section, 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.

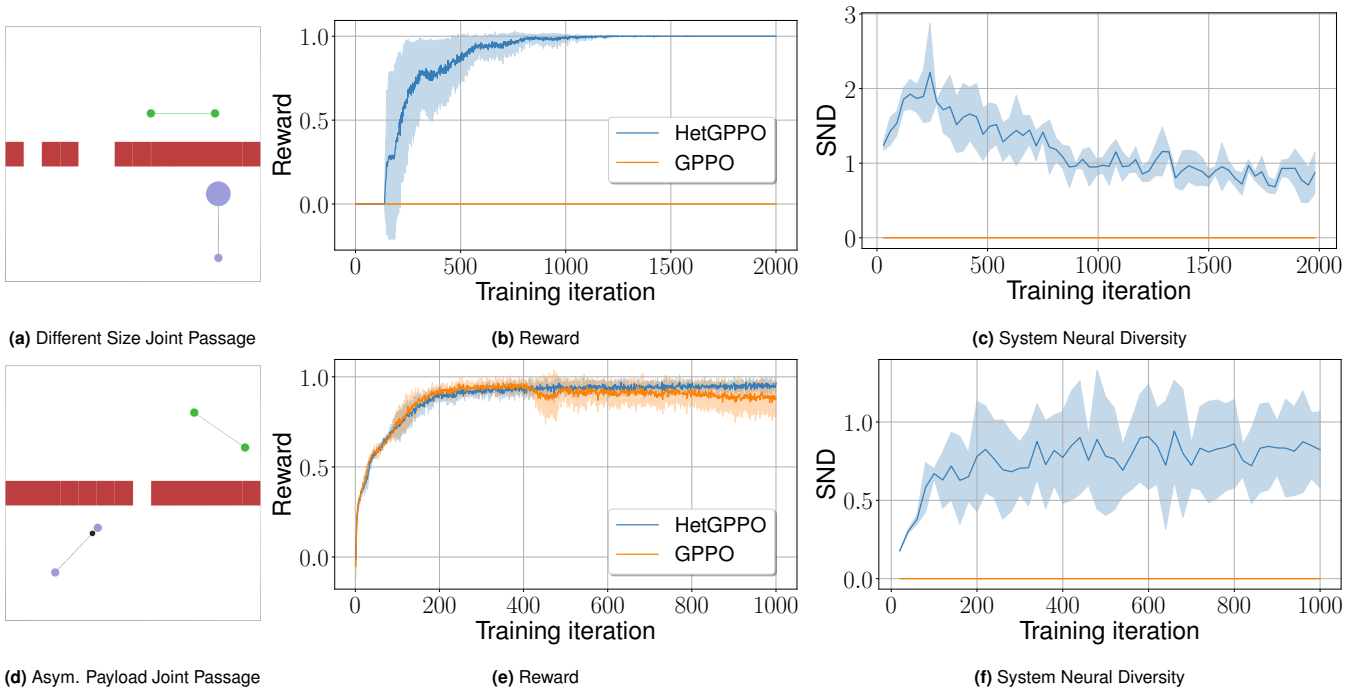
Multi-Agent Goal Navigation. We run a set of evaluations in the *Multi-Agent Goal Navigation* scenario described in the SND section (Fig. 2). We consider $n = 4$ agents with an increasing number of goals. This is the same setup used in Fig. 3. By observing the reward, shown in Fig. 6a, we can note that all agents converge to the maximum reward within the first 50 iterations. In Fig. 6b we report the SND measured throughout training. We can observe that SND 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 order to benefit from parameter sharing and increased sample efficiency. To demonstrate this, we run an additional evaluation in the *1 goal* setup, where SND is approximately 0, indicating no diversity. We compare homogeneous and heterogeneous training. In Fig. 7 we show that both paradigms obtain the same performance, with the homogeneous model being significantly more time-efficient.

Insight 1. *An* SND *of* 0 *indicates that homogeneous training should be preferred to benefit from sample efficiency.*

Different Size Joint Passage. This task, shown in Fig. 8a, 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 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. 8b 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, not being able 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 the SND plot in Fig. 8c, where we can see that the agents behave heterogeneously throughout training. They start with a high diversity that later converges to an SND of 1. This is due to the fact that the diversity here lies in the initial rotation of the joint (to align each agent with its gap) and the remaining navigation part of the task can be done homogeneously. This lets us make a key observation that diversity, in this case, is needed only during the initial 'assignment' action sequence that the agents take (to position themselves with respect to the correct gap). Since the rest of the navigation task is homogeneous, the agents observe the same states as they would in a homogeneous run, thus smoothing out their policy differences through training.

Insight 2. *An* SND *greater than* 0*, when heterogeneous paradigms obtain higher rewards than homogeneous ones, in-*



(a) Different Size Joint Passage **(b)** Reward **(c)** System Neural Diversity

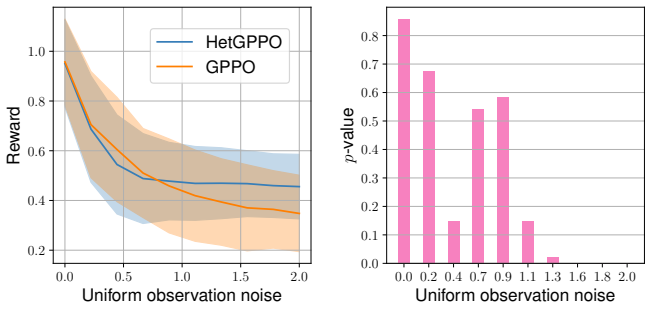
(d) Asym. Payload Joint Passage **(e)** Reward **(f)** System Neural Diversity

Fig. 8. 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. 9). We report mean and standard deviation over 3 random seeds for each experiment. Each training iteration is performed over 600 episodes of experience.

dicates that heterogeneity enables performance.

Asymmetric Payload Joint Passage. In the previous task considered, 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. 8d. 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. 8e, 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* (16) and thus learn a single multi-behavioral policy conditioned of these differences. In the heterogeneous model, on the other hand, two diverse

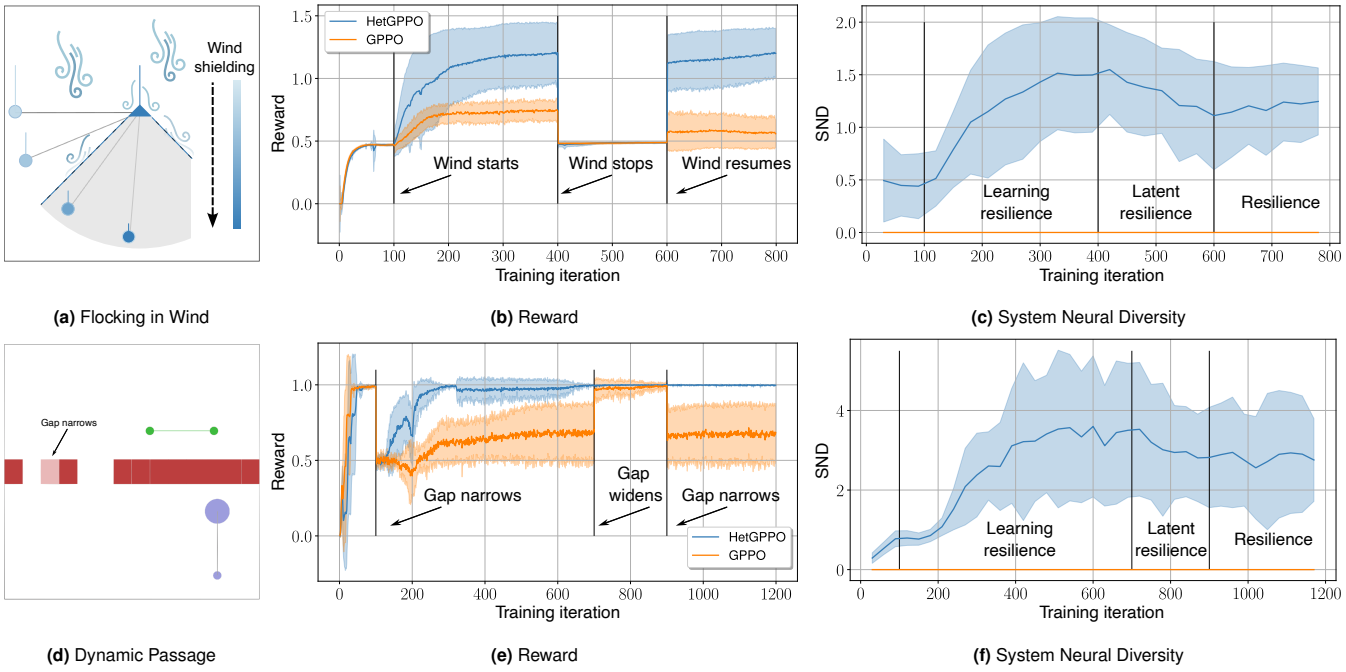


(a) Reward degradation **(b)** Welch's *t*-test

Fig. 9. Performance degradation in the *Asymmetric Payload Joint Passage* scenario in Fig. 8d 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 (50) for the means of the samples collected with the two models (right).

policies are learned, leading to a significant difference in SND (Fig. 8f). 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. 9, shows that the heterogeneous model proves more resilient to increasing noise[§]. From the p-values of a Welch's unequal variances t-test (50) we can observe that the perfor-

[§]Similar resilience results have been observed in (16) in various scenarios.



(a) Flocking in Wind **(b)** Reward **(c)** System Neural Diversity

(d) Dynamic Passage **(e)** Reward **(f)** System Neural Diversity

Fig. 10. Evaluations on dynamic tasks. **Top row (left to right)**: *Flocking in Wind* scenario with reward and SND. **Bottom row (left to right)**: *Dynamic Passage* scenario with reward and SND. 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 (wind) and 6 (passage) random seeds for each experiment. Each training iteration is performed over 300 (wind) and 200 (passage) episodes of experience.

mance 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.

Insight 3. *An* SND *greater than* 0*, when homogeneous and heterogeneous paradigms obtain the same reward, indicates that heterogeneity enables resilience.*

Dynamic Tasks. 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* (51). Heterogeneous training in MARL has previously been shown to have resilience properties in numerous simulated and real-world scenarios (16). In this section, we are interested in analyzing the adaptation of a multi-agent team to a disruption during the training process. In particular, we consider scenarios were such adaptation requires heterogeneous behaviors, and study how resilience relates to SND.

Flocking in Wind. Flocking is a long studied collective behavior observable in birds. The core behavioral rules of flocking were originally synthesized by Reynolds (52) and subsequently applied to multi-robot teams (53). In this scenario, shown in Fig. 10a, 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 from the other agent through communication.

This scenario can undergo a disruption, which manifests itself as an external wind force acting in the opposite direction of the agents' desired velocity. This results in the agents having to exert a higher force to track a desired velocity. However, the agents are physically different (see Fig. 10a): 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 the triangular agent behind or horizontal 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. 10b and Fig. 10c 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 (SND ≤ 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 is constrained 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, which cannot be perceived 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 maintain 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. 8c).

Insight 4. *In cases where the task undergoes repeated dynamic disruptions, SND is able to expose how heterogeneity allows agents to learn and maintain latent skills for resilience.*

Dynamic Passage. To further demonstrate how SND can act as an indicator of the latent resilience properties in heterogeneous agents, we analyze another dynamic task, shown in Fig. 10d. This scenario is a dynamic version of the *Different Size Joint Passage* scenario. In its base version, the two gaps in the wall have the same size and can fit either of the agents. The disturbance in this scenario occurs in the form of one of the gaps narrowing enough to block the larger agent. During this disturbance, the agents will need to perform a heterogeneous task assignment that directs them to their suitable gaps.

Once again, we train heterogeneous and homogeneous models for this scenario. In Fig. 10e and Fig. 10f we report the reward and SND, respectively. There is no disturbance through iterations 0 to 100 (i.e., both gaps fit either agent), and thus both paradigms learn to solve the task homogeneously. In fact, the heterogeneous model presents a low diversity of 0.6. At iteration 100, one of the gaps is reduced in size. With this disruption, both models drop to a reward of 0.5, meaning that the team passes the wall approximately 50% of the times. This indicates that the agent-to-gap assignment performed by the policies prior to the disruption followed a uniform random distribution. During iterations 100-700, the homogeneous model, unable to observe the physical differences of the agents, cannot perform heterogeneous gap assignment, and thus is only able to improve its performance slightly by leveraging tricks such as fitting both agents through the same gap. On the other hand, the heterogeneous agents learn two different behavioral roles and thus are able to increase their reward and SND until they restore the same performance observed prior to the disturbance. At iteration 700, the gap is widened again, restoring the task to its initial state. As in the wind scenario, we see that both paradigms regain the original performance. However, the high SND metric suggests that the heterogeneous model has learned a *latent resilient* skill. This is confirmed when the same disturbance is reintroduced again (iteration 900) – the performance of the heterogeneous agents remains unaffected, while the homogeneous counterpart suffers the same impairment as before.

Discussion

Behavioral diversity is a valuable skill in collective problems. Multi-agent systems that allow agents to specialize and learn unique (and potentially complementary) skills often demonstrate superior resilience towards disturbances (16). In this work, we presented a novel framework for measuring this heterogeneity with a System Neural Diversity (SND) metric. Our metric is able to measure behavioral diversity among agents with stochastic policies acting in a continuous state spaces. We show, theoretically and empirically, that SND allows to compare diversity across different system sizes and provides a measure of behavioral redundancy. These are two key properties that prior work (e.g., HSE) overlook, thus representing only a partial picture of system heterogeneity.

Our evaluations of the metric in a variety of multi-agent tasks (didactic and realistic) with different system sizes and different heterogeneity requirements establish the true representational power of our metric. The insights we draw from the various static and dynamic tasks point to the fact that SND provides a means to observe latent properties developed by a heterogeneous system during training. In our static tasks, we show how our metric can act as a key diagnostic tool to analyze heterogeneous systems. In particular, we derive three insights from the comparison of SND with task reward. These insights suggest that SND can inform on the effectiveness of heterogeneous learning agents. In our dynamic tasks, where the problem is affected by a repeated disturbance during training, we show that heterogeneous agents are first able to learn specialized roles that allow them to cope with the disturbance, and then retain these roles even when the disturbance is removed. This is beneficial for the agents when the disturbance reoccurs. SND allows a direct measurement of this latent resilience, while other proxies such as task performance (reward) fail to.

Finally, 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. This is already evident in the *Multi-Agent Goal Navigation* scenario when all agents are tasked towards the same goal. Fig. 7 shows that in this case, a homogeneous training offers an advantage in sample efficiency while achieving the same performance. Luckily, SND is able to measure the lack of heterogeneity in the

task and can thus inform our choice (see Insight 1). However, it is possible to construct other pathological scenarios where, just like in nature (54), excessive role specialization within a multi-agent system in fact 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.

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.

1. SR Kellert, *The value of life: Biological diversity and human society.* (Island press), (1997).
2. OL Petchey, KJ Gaston, Functional diversity: back to basics and looking forward. *Ecol. letters* **9**, 741–758 (2006).
3. MW Cadotte, K Carscadden, N Mirotchnick, Beyond species: functional diversity and the maintenance of ecological processes and services. *J. applied ecology* **48**, 1079–1087 (2011).
4. CM Tucker, et al., A guide to phylogenetic metrics for conservation, community ecology and macroecology. *Biol. Rev.* **92**, 698–715 (2017).
5. AW Woolley, CF Chabris, A Pentland, N Hashmi, TW Malone, Evidence for a collective intelligence factor in the performance of human groups. *science* **330**, 686–688 (2010).
6. AW Woolley, I Aggarwal, TW Malone, Collective intelligence and group performance. *Curr. Dir. Psychol. Sci.* **24**, 420–424 (2015).
7. DS Bernstein, R Givan, N Immerman, S Zilberstein, The complexity of decentralized control of markov decision processes. *Math. operations research* **27**, 819–840 (2002).
8. R Köster, et al., Spurious normativity enhances learning of compliance and enforcement behavior in artificial agents. *Proc. Natl. Acad. Sci.* **119**, e2106028118 (2022).
9. Q Li, F Gama, A Ribeiro, A Prorok, Graph neural networks for decentralized multi-robot path planning in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. (IEEE), pp. 11785–11792 (2020).
10. T Chu, J Wang, L Codecà, Z Li, Multi-agent deep reinforcement learning for large-scale traffic signal control. *IEEE Transactions on Intell. Transp. Syst.* **21**, 1086–1095 (2019).
11. JR Vázquez-Canteli, J Kämpf, G Henze, Z Nagy, Citylearn v1.0: An openai gym environment for demand response with deep reinforcement learning in *Proceedings of the 6th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation*, BuildSys '19. (Association for Computing Machinery, New York, NY, USA), p. 356–357 (2019).
12. K Zhang, Z Yang, T Başar, Multi-agent reinforcement learning: A selective overview of theories and algorithms. *Handb. Reinf. Learn. Control.* pp. 321–384 (2021).
13. JK Gupta, M Egorov, M Kochenderfer, Cooperative multi-agent control using deep reinforcement learning in *International conference on autonomous agents and multiagent systems*. (Springer), pp. 66–83 (2017).
14. T Rashid, et al., Qmix: Monotonic value function factorisation for deep multi-agent reinforcement learning in *International Conference on Machine Learning*. (PMLR), pp. 4295–4304 (2018).
15. S Sukhbaatar, R Fergus, , et al., Learning multiagent communication with backpropagation. *Adv. neural information processing systems* **29** (2016).
16. M Bettini, A Shankar, 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).
17. T Wang, et al., Rode: learning roles to decompose multi- agent tasks in *Proceedings of the International Conference on Learning Representations*. (2021).
18. F Christianos, G Papoudakis, MA Rahman, SV Albrecht, Scaling multi-agent reinforcement learning with selective parameter sharing in *International Conference on Machine Learning*. (PMLR), pp. 1989–1998 (2021).
19. L Chenghao, et al., Celebrating diversity in shared multi-agent reinforcement learning. *Adv. Neural Inf. Process. Syst.* **34** (2021).
20. T Wang, H Dong, V Lesser, C Zhang, Roma: Multi-agent reinforcement learning with emergent roles in *International Conference on Machine Learning*. (PMLR), pp. 9876–9886 (2020).
21. T Balch, Hierarchic social entropy: An information theoretic measure of robot group diversity. *Auton. robots* **8**, 209–238 (2000).
22. KR McKee, JZ Leibo, C Beattie, R Everett, Quantifying the effects of environment and population diversity in multi-agent reinforcement learning. *Auton. Agents Multi-Agent Syst.* **36**, 1–16 (2022).
23. Y Yu, H Jiang, Z Lu, Informative policy representations in multi-agent reinforcement learning via joint-action distributions. *arXiv e-prints* pp. arXiv–2106 (2021).
24. S Hu, C Xie, X Liang, X Chang, Policy diagnosis via measuring role diversity in cooperative multi-agent rl in *International Conference on Machine Learning*. (PMLR), pp. 9041–9071 (2022).
25. X Liu, et al., Towards unifying behavioral and response diversity for open-ended learning in zero-sum games. *Adv. Neural Inf. Process. Syst.* **34**, 941–952 (2021).
26. A Grover, M Al-Shedivat, J Gupta, Y Burda, H Edwards, Learning policy representations in multiagent systems in *International conference on machine learning*. (PMLR), pp. 1802–1811 (2018).
27. K Chatzilygeroudis, A Cully, V Vassiliades, JB Mouret, Quality-diversity optimization: a novel branch of stochastic optimization in *Black Box Optimization, Machine Learning, and No-Free Lunch Theorems*. (Springer), pp. 109–135 (2021).
28. KO Stanley, J Clune, J Lehman, R Miikkulainen, Designing neural networks through neuroevolution. *Nat. Mach. Intell.* **1**, 24–35 (2019).
29. S Doncieux, JB Mouret, Behavioral diversity measures for evolutionary robotics in *IEEE congress on evolutionary computation*. (IEEE), pp. 1–8 (2010).
30. MA Masood, F Doshi-Velez, Diversity-inducing policy gradient: Using maximum mean discrepancy to find a set of diverse policies. *arXiv preprint arXiv:1906.00088* (2019).
31. J Parker-Holder, A Pacchiano, KM Choromanski, SJ Roberts, Effective diversity in population based reinforcement learning. *Adv. Neural Inf. Process. Syst.* **33**, 18050–18062 (2020).
32. L Jost, Entropy and diversity. *Oikos* **113**, 363–375 (2006).
33. CE Shannon, A mathematical theory of communication. *The Bell system technical journal* **27**, 379–423 (1948).
34. A Prorok, MA Hsieh, V Kumar, The impact of diversity on optimal control policies for heterogeneous robot swarms. *IEEE Transactions on Robotics* **33**, 346–358 (2017).
35. L Li, A Martinoli, YS Abu-Mostafa, Learning and measuring specialization in collaborative swarm systems. *Adapt. Behav.* **12**, 199–212 (2004).
36. P Twu, Y Mostofi, M Egerstedt, A measure of heterogeneity in multi-agent systems in *2014 American Control Conference*. (IEEE), pp. 3972–3977 (2014).
37. LP Kaelbling, ML Littman, AR Cassandra, Planning and acting in partially observable stochastic domains. *Artif. intelligence* **101**, 99–134 (1998).
38. K Menger, K Menger, Statistical metrics. *Sel. Math. Vol. 2* pp. 433–435 (2003).
39. C Kaplanis, M Shanahan, C Clopath, Policy consolidation for continual reinforcement learning in *International Conference on Machine Learning*. (PMLR), pp. 3242–3251 (2019).
40. LN Vaserstein, Markov processes over denumerable products of spaces, describing large systems of automata. *Probl. Peredachi Informatsii* **5**, 64–72 (1969).
41. E Hellinger, Neue begründung der theorie quadratischer formen von unendlichvielen veränderlichen. *J. für die reine und angewandte Math.* **1909**, 210–271 (1909).
42. M Arjovsky, S Chintala, L Bottou, Wasserstein generative adversarial networks in *International conference on machine learning*. (PMLR), pp. 214–223 (2017).
43. RS Sutton, AG Barto, *Reinforcement learning: An introduction*. (MIT press), (2018).
44. M Bettini, R Kortvelesy, J Blumenkamp, 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).
45. C Gini, Variabilità e mutabilità. *Repr. Memorie di metodologica statistica (Ed. Pizetti E* (1912).
46. E Liang, et al., Rllib: Abstractions for distributed reinforcement learning in *International Conference on Machine Learning*. (PMLR), pp. 3053–3062 (2018).
47. A Paszke, et al., Pytorch: An imperative style, high-performance deep learning library. *Adv. neural information processing systems* **32** (2019).
48. J Blumenkamp, A Prorok, The emergence of adversarial communication in multi-agent reinforcement learning in *Conference on Robot Learning*. (PMLR), pp. 1394–1414 (2021).
49. M Samvelyan, et al., The starcraft multi-agent challenge in *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*. pp. 2186–2188 (2019).
50. BL Welch, The generalization of 'student's'problem when several different population varlances are involved. *Biometrika* **34**, 28–35 (1947).
51. A Prorok, et al., Beyond robustness: A taxonomy of approaches towards resilient multi-robot systems. *arXiv preprint arXiv:2109.12343* (2021).
52. CW Reynolds, Flocks, herds and schools: A distributed behavioral model in *Proceedings of the 14th annual conference on Computer graphics and interactive techniques*. pp. 25–34 (1987).
53. K Saulnier, D Saldana, A Prorok, GJ Pappas, V Kumar, Resilient flocking for mobile robot teams. *IEEE Robotics Autom. letters* **2**, 1039–1046 (2017).
54. ML McKinney, Extinction Vulnerability and Selectivity: Combining Ecological and Paleontological Views. *Annual Review of Ecology and Systematics* **28**, 495–516 (1997).