Promoting Resilience of Multi-Agent Reinforcement Learning via Confusion-Based Communication

Ofir Abu[1] , **Sarah Keren**[1,2] , **Matthias Gerstgrasser**[2] and **Jeffrey S. Rosenschein**[1]

[1]School of Computer Science and Engineering, Hebrew University of Jerusalem, Israel
[2]School of Engineering and Applied Sciences, Harvard University, USA

Abstract

Agents operating in real-world settings are often faced with the need to adapt to unexpected changes in their environment. Recent advances in *multi-agent reinforcement learning (MARL)* provide a variety of tools to support the ability of RL agents to deal with the dynamic nature of their environment, which may often be increased by the presence of other agents. In this work, we measure the *resilience* of a group of agents as the group's ability to adapt to unexpected perturbations in the environment. To promote resilience, we suggest facilitating collaboration within the group, and offer a novel *confusion-based* communication protocol that requires an agent to broadcast its local observations that are least aligned with its previous experience. We present initial empirical evaluation of our approach on a set of simulated multi-taxi settings.

1 Introduction

Reinforcement Learning (RL) agents are typically required to operate in dynamic environments, and must develop an ability to quickly adapt to unexpected perturbations in those environments. Promoting this ability is challenging, even for single-agent settings [11]. For a group of agents this becomes even more of a challenge; in addition to the dynamic nature of the environment, the agents need to deal with high variance caused by changes in the behavior of other agents [14; 19; 9].

In this work, we measure the *resilience* of a group of agents according to the group's ability to adapt to perturbations in the environment. Contrary to the investigation of *transfer learning* [21; 7] or *curriculum learning* [13], we do not have a target domain in which the group of agents is going to be deployed, nor do we have a training phase dedicated to preparing agents for the deployment environment. Instead, we aim to equip a group with the ability to adapt to unexpected changes that can occur at random times.

Recent literature is rich with a variety of different definitions of resilience and robustness, for both single and multi-agent settings [20; 18; 12]. Our measure for resilience is inspired by work done in robotics [16], which views resilience as the ability of a group to reach an agreement about estimated values of variables in the presence of non-cooperative members. In our setting, resilience measures the agents' performance in the presence of unexpected perturbations.

To promote group resilience, we introduce a communication protocol that defines the information that each agent shares with the group. Recent work demonstrates how communication in multiagent reinforcement learning (MARL) settings allows a group to learn and operate efficiently in complex but non-perturbed environments [6; 5]. To promote a group's ability to adapt to a randomly perturbed environment, we present a novel *confusion-based* communication protocol, that requires an agent to broadcast its current observations that are least aligned with its previous model of the environment.

Example 1 *Consider Figure 1, which depicts a multi-agent variation of the single Taxi domain [3]. In this setting, taxis are associated with one operator, but each taxi receives a direct payment from each passenger that it drops off at her destination. In addition to the taxis, we posit a* designer, *representing the taxi's operator, that aims to maximize the group's total revenue.*

The designer monitors the taxis' performance (revenue), and notices that taxis sometimes deviate from their usual routes. This may happen, for example, due to road construction, road congestion, or many other reasons. Trying to enable the group to maximize group performance despite these changes, and with the intention of not overwhelming the communication channel, the designer instructs taxis to broadcast 'confusing' experiences to the others, corresponding to actions that have unexpected outcomes. Thus, for example, when a taxi encounters a blocked road, and fails to drive through a street it has driven through regularly, it will broadcast this unsuccessful attempt to the other taxis. Similarly, when the taxi drives through a typically busy street, and finds that it is not busy, it will broadcast this information. This may relieve the effect these perturbations will have on the other taxis.

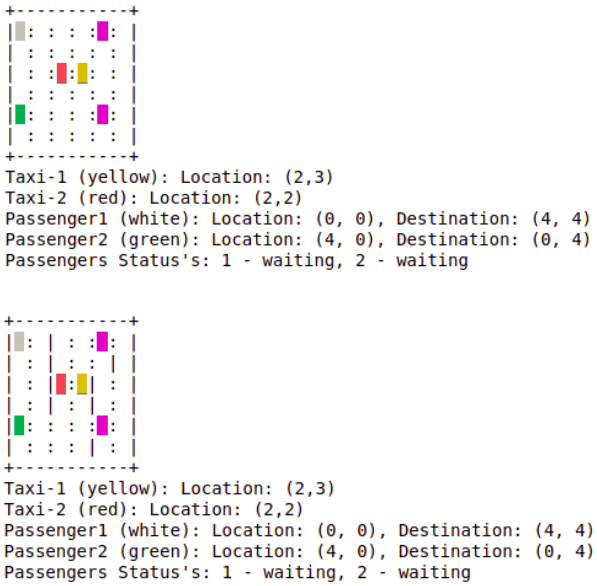

Figure 1: An illustration of the multiagent taxi domain. Perturbations are represented by introducing non-traversable walls in the environment on the right.

Our key contributions in this work are threefold. First, we suggest a new measure of group resilience that corresponds to the group's ability to adapt to unexpected changes. As a second contribution, and in order to promote resilience, we offer a new confusion-based communication protocol according to which all agents are notified about notable changes in an agent's surroundings. Lastly, we offer an initial empirical evaluation that demonstrates how agents that operate according to the confusion-based protocol are more resilient when compared to agents that are not using the suggested protocol.

2 Background

A *Markov Decision Process (MDP)* [2] is a widely used formalization of sequential decision making in stochastic environments. One standard formulation of an MDP is a tuple $M = \langle S, A, R, P, \gamma \rangle$ where S is a finite set of states, A is a finite set of actions, R is a reward function $R : S \times A \mapsto \mathbb{R}$, P is a transition function $P : S \times A \times S \mapsto [0, 1]$ defining a probability distribution $\mathbb{P}_s^a[S]$ over next states given the current state $s \in S$ and action $a \in A$, and $\gamma \in (0, 1]$ is a discount factor describing the rate by which reward deprecates over time. When the state is only partially observable by the agent, the definition also includes a set of observation tokens $\mathcal{O}$, and an observation function $O : S \mapsto \mathcal{O}$ that defines the observation that is done by an agent given the current state of the environment. In some cases, the definition of an MDP also includes an initial state $s_0 \in S$.

A *Markov game*, or a *stochastic game*, is a generalization of the MDP to multi-agent settings [8]. A Markov game is defined as a tuple $\langle S, \mathcal{A}, \mathcal{R}, \mathcal{T}, \gamma \rangle$. Where $\mathcal{A} = \{A^i\}_{i=1}^n$ is a collection of action sets A^i , one for each of the n agents in the environment, $\mathcal{R} = \{R^i\}_{i=1}^n$ is a collection of reward functions R^i defining the reward $r^i(a_t, s_t)$ that each agent receives when the joint action a_t is performed at state s_t, and $\mathcal{T}$ describers the probability distribution over next states when a joint action is performed. In the partially observable case, the definition also includes a joint observation function: $O : S \mapsto \mathcal{O}$, defining the observation for each agent at each state. In this work, we refer to Markov games as MDPs, games, domains, or environments interchangeably.

Reinforcement learning (RL) is a branch of machine learning that aims to learn optimal policies for partially informed agents operating in some environment, typically modeled as an MDP for the single agent case, or a Markov Game, for multi-agent settings. The behavior of an RL agent is described by a *policy* $\pi : S \times A \to [0, 1]$ which associates a probability of taking some action a at state s.

One common objective for RL agents is to maximize the expected discounted accumulated reward: $\mathbb{E}_{a_t \sim \pi(\cdot|s_t)}[\sum_{t \geq 0} \gamma^t R(s_t, a_t, s_{t+1})]$. Many RL algorithms maximize this objective by estimating a *value function* $V_\pi(s) = \mathbb{E}_{a_t \sim \pi(\cdot|s_t)}[\sum_{t \geq 0} \gamma^t R(s_t, a_t, s_{t+1})|s_0 = s]$, that estimates the accumulated discounted reward of following some policy π from a given state. Another function commonly used is the *action-value function* or *Q-function* $Q_\pi(s, a) = \mathbb{E}_{a_t \sim \pi(\cdot|s_t)}[\sum_{t \geq 0} \gamma^t R(s_t, a_t, s_{t+1})|a_0 = a, s_0 = s]$, representing the expected accumulated reward of taking action a at state s and then following policy π.

Multi-Agent Reinforcement Learning (MARL) extends RL to multi-agent settings, which are typically modeled using a Markov game. The performance (utility) $\mathcal{U}$ of a group can be defined in various ways, but is typically measured by the total discounted accumulated reward of the agents, or the average individual utility. In this work we use the terms group *performance* and group *utility* interchangeably.

A key challenge in MARL is that the policy of each agent changes during training, and the environment is non-stationary from the perspective of any individual agent, which may not be fully aware of the other agents and of their effect on the environment. This results in unstable learning and prevents the straightforward use of past experiences when deciding how to behave [9]. This high variance in the agent's experience is intensified in the settings we consider here, in which the environment is randomly perturbed.

3 Measuring Group Resilience

We aim to promote the ability of a group of agents to adapt to random perturbations in their environment. We refer to this ability as *resilience*, and describe it below. We will then suggest ways to promote group resilience by offering methods by which agents can collaborate effectively.

To measure the resilience of a group of agents we

take inspiration from the field of robotics. The work of Saulnier et al. [16] presents a control policy that allows a team of mobile robots to achieve desired performance in the presence of faults and attacks on individual members of the group. A group of robots achieves *resilient consensus* if the cooperative robots' performance is in some desired range, even in the presence of up to a bounded number of non-cooperative robots. In essence, we want *resilience* to mean that if an environment undergoes an unexpected (but somehow bounded-in-magnitude) perturbation, then agents can still achieve a fixed fraction of their original performance.

Our definition of resilience relies on some user-specified distance measure $\delta(M, M')$ that quantifies the magnitude of the change between an original MDP M and the modified MDP M'. It also relies on the specification of a utility measure $\mathcal{U}(M')$, quantifying the performance of a group of agents in a given MDP. Based on these measures, we define several notions of resilience. The strongest would not place any conditions on M' other than distance:

Definition 1 (Relative to Optimum C_K-resilience) *Given an MDP M and a bound $K \in \mathbb{R}$, we say that a group of agents α is* universally C_K-resilient *in M if*

$$\forall M' : \delta(M, M') \leq K \implies$$

$$\mathcal{U}(M') \geq C_K \cdot \mathcal{U}^*(M')$$

where $\mathcal{U}^*(M') = \max_\pi \mathcal{U}(\pi, M')$

Our definition above relies on the ability to compute the optimal value in a given environment, and comparing it to the actual utility achieved by the group of agents. For settings where this is not possible, we offer a *Relative to Origin* resilience measure that compares the performance of the group before and after the environment is perturbed.

Definition 2 (Relative to Origin C_K-resilience) *Given an MDP M and a bound $K \in \mathbb{R}$, we say that a group of agents α is* universally C_K-resilient *in M if*

$$\forall M' : \delta(M, M') \leq K \implies$$

$$\mathcal{U}(M') \geq C_K \cdot \mathcal{U}(M)$$

It is important to note that Definition 2 compares the performance of the agent in the perturbed environment against its performance in the original one, without considering its value. This means that a group of agents that follows a non-efficient policy (e.g., performing a no-op action repeatedly) will have high resilience. If this measure is used it is important to consider the utility achieved as an orthogonal measure. Moreover, the above definitions may not be very useful, as (depending on the distance measure used) we might not have any way to construct the set of arbitrary MDPs within a certain distance from the original MDP. A more restrictive definition would be to explicitly restrict resilience to be over a given set of environments of interest.

Definition 3 (Relative to Origin C_K-resilience over $\mathcal{M}$) *Given an MDP M, a set of perturbed MDPs $\mathcal{M}$, and a bound $K \in \mathbb{R}$, we say that a group of agents is* universally C_K-resilient *in M over $\mathcal{M}$ if*

$$\forall M' \in \mathcal{M} : \delta(M, M') \leq K \implies$$

$$\mathcal{U}(M') \geq C_K \cdot \mathcal{U}(M)$$

Resilience over $\mathcal{M}$ allows us to choose the set $\mathcal{M}$ of environments such that the distance condition is more easily verified. However, this condition still requires that the bound on performance holds for *any* $M' \in \mathcal{M}$, which may be unreasonably strong and impractical in many cases. We therefore further define resilience-in-expectation over a set equipped with a uniform probability distribution or another probability distribution Ψ generating MDPs M'. In other words, we require the expected performance of agents in MDPs to fulfill a performance guarantee, where the expectation is uniformly over all MDPs in $\mathcal{M}$ that are within K-distance of M. Similarly, we can define this notion for arbitrary distributions over MDPs:

Definition 4 (C_K-resilience in expectation over Ψ) *Given an MDP M, a distribution over MDPs Ψ, and a bound $K \in \mathbb{R}$, we say that a group of agents is* C_K-resilient in expectation *in M over Ψ if*

$$\mathbb{E}_{[M' \sim \Psi | \delta(M, M') \leq K]} \mathcal{U}(M') \geq C_K \cdot \mathcal{U}(M)$$

As a known result, it is easy to see that polynomially-many samples from Ψ are sufficient to achieve arbitrarily close approximations of the true expectation, with arbitrarily high probability.[1]

In this work, we are interested in settings in which we have an initial environment and a set of *perturbations* that can occur to the environment. In general, a perturbation $\phi : \mathcal{M} \mapsto \mathcal{M}$ is a function transforming a source MDP into a modified MDP. An *atomic perturbation* is a perturbation that changes only one of the basic elements of the original MDP. In the following, given an MDP $M = \langle S, A, R, P, \gamma \rangle$ and perturbation ϕ, the resulting MDP after applying ϕ is denoted by $M^\phi = \langle S^\phi, A^\phi, R^\phi, P^\phi, \gamma^\phi \rangle$.

Among the variety of perturbations that may occur, we focus here on three types of atomic perturbations: *transition function perturbations* modify the distribution over next states for a single state-action pair, *reward function perturbations* modify the reward of a single state-action pair, and *initial state perturbations* change the initial state of the MDP (if that state is defined).

Definition 5 (Transition Function Perturbation) *A perturbation ϕ is a* transition function perturbation *if for every MDP $M = \langle S, A, R, P, \gamma \rangle$, M^ϕ is identical to*

[1] Assume that the utility function $\mathcal{U}(M')$, considered as a random variable with M' drawn from $[M' \sim \Psi | \delta(M, M') \leq K]$ as above, is i.i.d. for a random draw of M' and has a finite variance σ^2. Then it follows from Chebychev's inequality that in order to be within ϵ of the true mean with probability at least δ, it is sufficient to collect at least $\frac{\sigma^2}{\epsilon^2 \cdot (1-\delta)}$ samples.

M except that for a single action state pair $s \in S$ and $a \in A$, $\mathbb{P}^a_s[S] \neq \mathbb{P}^{\phi}{}^a_s[S]$.

Definition 6 (Reward Function Perturbation) *A perturbation ϕ is a* reward function perturbation *if for every MDP $M = \langle S, A, R, P, \gamma \rangle$, M^ϕ is identical to M except that for a single action state pair $s \in S$ and $a \in A$, $r^a_s \neq r^{\phi}{}^a_s$.*

Definition 7 (Initial State Perturbation) *A perturbation ϕ is a* transition function perturbation *if for every MDP $M = \langle S, s_0, A, R, P, \gamma \rangle$, M^ϕ is identical to M except that $s_0 \neq s^\phi_0$.*

Example 1 (continued) *In our multi-taxi domain from Example 1, a road blockage is a perturbation comprised of atomic transition function perturbations that reduce to zero the probability of transitioning to the blocked cell from any adjacent cell. If the destination of a passenger changes, this is represented by two atomic perturbations, one that replaces the reward for a dropoff at the original destination with a negative reward, and one that adds a positive reward for a dropoff at the new destination.*

There are a variety of metrics for measuring the distance between two MDPs [17; 1]. We want the *magnitude of a perturbation* to represent the extent by which a perturbed environment is different from the original one. Intuitively, the bigger the magnitude, the harder it would be for a set of RL agents to adapt.

A straightforward way to measure the distance between two MDPs is to count the minimal number of atomic perturbations that transition the original MDP into the transformed one. Another measure is the one suggested by Song et al. [17], where the distance between two MDPs M and $M^{'}$ is calculated by computing the accumulated distance between every state in M and its corresponding state in M'. This definition holds for a setting where the two MDPs are *homogeneous*, such that there exists a correspondence (mapping) between the states, action spaces, and reward functions of the two MDPs.

Given two homogeneous MDPs M and M', the distance $d(s, s')$ between any two states $s \in S_M$ and $s' \in S_{M'}$ is defined as follows:

$$d(s, s') = \max_{a \in A} \{ |r^a_s - r'^a_{s'}| + cT_k(d)(\mathbb{P}^a_s[S_M], \mathbb{P}'_{s'}{}^a[S_{M'}]) \}$$

where $r^a_s, \mathbb{P}^a_s[S_M], r'^a_{s'}, \mathbb{P}'^a_{s'}[S_{M'}]$ are the immediate reward and the transition function for M and M' respectively, $T_k(d)$ is the Kantorovich distance [4] between the two probability distributions, and $c \in [0, 1]$ is some hyper-parameter defining the significance of the distance between the distributions.

In our setting, we focus on perturbations Φ that do not change the state space nor the action space, so $\Phi(M)$ and M are homogeneous according to Song et al. [17], and each state $s \in S_M$ corresponds to the same state in $S_\Phi(M)$. We therefore use this measure to estimate the distance between an MDP and its perturbed variations in our empirical evaluation below.

4 Promoting Group Resilience Via Confusion-Based Communication

Equipped with a way to assess the resilience of a group of agents, we aim at maximizing the resilience of a group of RL agents. Recent work in MARL suggests various approaches for promoting efficient collaboration within a group of agents. Such approaches include introducing a communication protocol [6; 9] or an ability to model other agents' policies [10; 15]. The focus in these frameworks is on helping agents deal with the non-stationarity of the environment that stems from the existence of other agents in it. Instead of promoting collaboration in order to maximize a group's performance in a given environment, we suggest promoting collaboration as a way to promote resilience. We hypothesize that agents that learn to collaborate will adapt more quickly to a changing environment.

Inspired by the success of communication protocols in increasing the performance of a group of RL agents [6; 5], we suggest a *confusion-based communication* protocol. We refer to *confusion* as the level by which the immediate reward observed by an agent is misaligned with its estimated reward. After every fixed number of time steps, each agent notifies the other agents in the system about its most confusing observations since its last broadcast. Our protocol relies on the existence of a limited communication channel that is shared among all agents, so that every message is broadcast to all other agents.

Formally, we define the level of confusion of an agent in a specific state by using its Q function. Let π_p be the policy of agent p, and let s_j be the next state the environment transitioned to after taking action a_i in s_i. The reward $\hat{r_i}$ of taking a_i in s_i is estimated by: $\hat{r_i} \approx Q_{\pi_p}(s_i, a_i) - Q_{\pi_p}(s_j, \pi_p(s_j))$. The confusion level for a given state and agent is defined as follows:

Definition 8 (Level of Confusion) *Let r_i and $\hat{r_i}$ be the observed and estimated reward of agent p after taking $a_i = \pi_p(s_i)$ in s_i. The level of confusion of agent p at s_i, $J^p_{s_i}$ is defined as:*

$$J^p_{s_i} = J^p_{s_i, \pi_p(s_i)} = J^p_{s_i, a_i} = \frac{|r_i - \hat{r_i}|}{r_i}$$

where J_{s_i, a_i} represents the level of confusion of the agent at state s_i after taking action a_i.

When communicating, the agent broadcasts its experience from the most confusing encountered states, meaning the states with the highest $J^p_{s_i}$. Specifically, the agent shares the state, action taken, reward collected, and next encountered state. The number of state experiences communicated is determined by a parameter m_l that represents the communication channel bandwidth.

For example, let π_i be the policy of agent i, and Q_i be the Q function associated with it after interacting with environment M. When agent i interacts with the perturbed environment $\Phi(M)$, high confusion is expressed as the difference between the estimated and observed

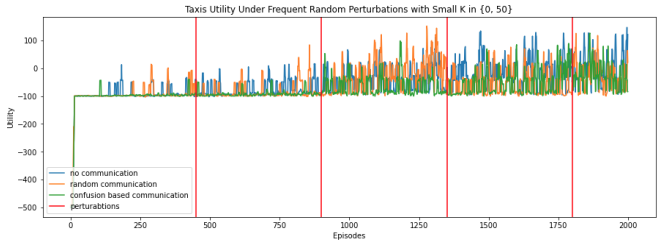

Figure 2: Average utility for a small $K \in \{0, 50\}$

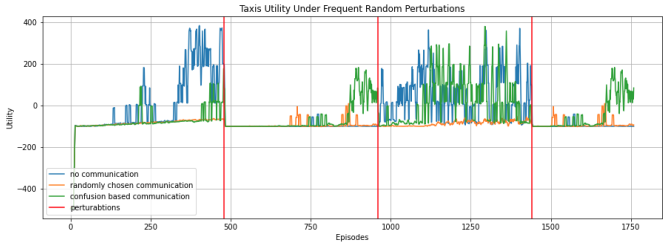

Figure 3: Average utility for $K = 150$.

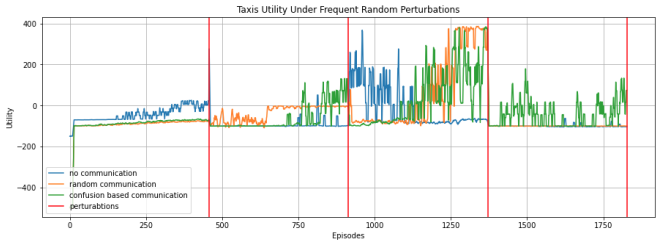

Figure 4: Average utility for $K = 200$

immediate rewards achieved by taking some action a_i at some state s_j. We use confusion to decide what knowledge should be broadcast to other agents, so that they can help one another adapt to the perturbed environment.

As we want to focus in this paper entirely on the aspect of resilience and confusion-based communication, we chose a specific implementation of communication that avoids any problems of emergent communications. Specifically, we take the state experience that an agent broadcasts, and insert it directly into the receiving agents' replay buffers. This also illustrates that we consider resilience to be a feature of a system of learning agents, rather than of final policies to be deployed. We see no reason why our approach couldn't also work with suitable frozen policies however, for instance using LSTM policies, and leave this as an interesting direction for future work.

5 Empirical Evaluation

The objective of our empirical evaluation is to assess the effect collaboration has on the resilience of a group of agents. Specifically, we examine a MARL setting where agents use different forms of communication, according to the group's performance in perturbed environments.

5.1 Environments

Our dataset consists of instances of the multi-taxi domain described in Example 1.[2] The environment is episodic, resetting when all passengers arrive at their destinations. At the beginning of each episode, the taxis appear at random start locations on the grid, while passengers' source and destination locations are chosen at random from a set of possible locations. Each taxi receives a high positive reward for each passenger that is successfully dropped off at her location, a small negative reward for each step in the environment, and a high negative reward when trying to drive through a wall. For each instance, the environment is represented by a 5×5 to 8×8 grid, with 2–3 taxis and 2–3 passengers.

Agents have a communication protocol according to which they broadcast their experience to the other agents. We experimented with channel bandwidths of $m_l \in [4, 15]$ and a transmission frequency (the rate by which messages are broadcast) $m_f \in [10, 30]$ time steps. The perturbations that can occur include the addition or removal of walls (modeled as transition function perturbation), passengers appearing in unexpected (initial state perturbations) locations, and new passenger destinations. In addition to this, we evaluated our approach in several other environments. We discuss these domains and results of our experiments in the supplementary material.

5.2 Setup

We experimented with perturbation bounds $K \in \{0, 50, 150, 200\}$, measured according to the distance measure suggested by Song et al. [17], randomly chosen to define the number of episodes the agents interact with the environment. Each episode is capped at 100 time steps, and each experimental setting is run for a fixed number of $T \in [1500, 3000]$ episodes. We use DQN agents parameterized by 4 sequential linear and ReLU layers; the agents are decentralized and self-interested. Group utility $\mathcal{U}$ is the accumulated reward of all agents.

In each environment instance, we compare the following groups of agents:

1. **No communication:** each agent interacts with the environment relying only on its own observations. Agents are not explicitly aware of one another, and do not communicate.
2. **Collaboration via communication of randomly selected observations:** agents broadcast a random subset of their observations at some fixed frequency and bandwidth.
3. **Collaboration via confusion-based communication:** the communication protocol we have introduced, according to which agents broadcast the top m_l most-confusing observations they encountered since the last message broadcast.

[2] Our dataset, source code and complete empirical results can be found in the supplementary material of this submission.

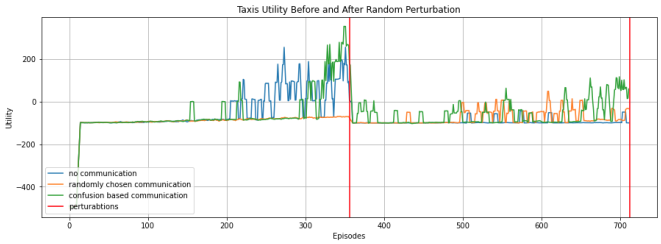

Figure 5: Learning curve of a group of agents in original and perturbed MDP, for $K = 150$.

Communication Protocol	**K=50**	**K=150**	**K=200**
No communication	0.801	0.450	0.238
Communication of randomly selected observations	0.822	0.242	0.187
Confusion based communication	**0.888**	**0.55**	**0.347**

Table 1: C-resilience values for groups of agents with different communication protocols, calculated over 10 experiments.

We considered two experimental setups. In the first, for each value of K, we constructed a set M_K by introducing randomly chosen perturbations to some initial environment. The resilience of the group is measured as the expected performance over the sampled group (Definition 4). In the second setup, we choose at random the frequency $t_{pert} \in [250, 500]$ by which perturbations are introduced into the environment. We let the agents learn and communicate in some initial environment M for t_{pert} episodes, and introduce a perturbation after t_{pert} episodes, transforming $M \mapsto M'$. After another t_{pert} episodes, the environment reverts back to the original environment, and so on. Our experiments were run on a university cluster using a mix of recent CPUs; each of our experiments ran in about 24 hours on 4 cores using 16GB RAM.

5.3 Results

Table 1 presents the average C_K-resilience in expectation calculated over 10 samples taken within $K = 50$, $K = 150$, and $K = 200$. As can be concluded from the results, our confusion based communication protocol achieves the highest resilience value. It is also interesting to see that the protocol that randomly selects the observations to broadcast achieves in average lower resilience than the no communication setting. This shows that merely sharing experiences between agents is not sufficient to achieve resilience. On the other hand, our specific confusion-based approach does lead to significantly improved resilience, showing the merit of using confusion to determine what information to share.

Figure 5 shows a learning curve for a group of agents before and after the environment is perturbed, as indicated by the vertical red bar in the plot. It is easy to see that the confusion-based group of agents (green line) achieves highest performance in the perturbed environment.

5.4 Outlook: Resilience in constantly-changing environments

The definitions we presented in Section 3 specify resilience based on the performance of agents after a single perturbation event leaves them in an altered environment. As a theoretical tool we believe this approach is most useful. In practice, however, we also investigate the merit of our confusion-based communication protocol in a wider range of settings. In particular, we evaluated the performance of confusion communication in *constantly-changing* environments. In this, the environment is perturbed every t_{pert} episodes. Then, after t_{pert} perturbed episodes, the environment is transitioned back to the original environment, before being perturbed again.

Figures 2, 3, and 4 show the average utility achieved by agents in these constantly-changing settings for $K \in (0 - 50)$, $K = 150$, and $K = 200$, respectively. Similarly to our single-perturbation-event results, we see that confusion-based communication clearly outperforms both non-communication and random communication in the perturbed environment (the time region between the 1st and 2nd red bar, and between and 3rd and 4th red bar).

This suggests the applicability of our approach to a wider range of problems of learning in frequently-changing environments. This could include more generally non-MDP environment, at least as long as the non-stationarity is coming from an underlying process that is in some form similar to occasionally introducing perturbations to an MDP.

5.5 Discussion

The results presented above show that the collaboration, and the use of *confusion based communication* in particular, achieves higher resilience in the presence of perturbations of large magnitude in the environment. We conclude that *confusing* observations encapsulate valuable information that helps agents learn about perturbed environments.

In addition, examining the achieved C-resilience values in Table-1 along side with Figures 3 and 4, we conclude that our suggested measure for the group's resilience, C-resilience, is a useful quantitative measure of group resilience. We also note that random communication i.e. sharing a random sample of experiences between agents, did not lead to high C-resilience, even though in principle even a random sample of another agent's experiences could be useful. This supports our belief that confusion-based communication is a useful tool to support resilience, rather than experience sharing in general.

It is interesting to note that at the beginning of each experiment the groups that use a communication channel in the non perturbed environment learn more slowly and achieve lower utility than the no communication group. This could be explained by the fact that during the initial learning phase in which exploration is high, messages

transferred by other agents can hinder the learning process. This could be resolved by learning a confusion threshold θ_{conf} that will be used to decide when to listen to incoming messages, or by delaying the activation of the communication channel until after the exploration phase of the individual agents completes. These are two interesting aspects we intend to investigate in future work.

6 Conclusion

We introduced novel formulations to evaluate the resilience of a group of agents by assessing the group's ability to adapt to perturbations of the environment. In addition, we suggested a novel *confusion-based communication* protocol to promote group resilience. To the best of our knowledge, this is the first measurement of group resilience that is relevant to multi-agent RL settings. Our evaluation shows that collaboration via our confusion-based communication protocol improves group resilience over naive baselines.

The ability of autonomous agents individually, or as a group in a multiagent system, to adapt to changes in the environment is obviously highly desirable in real-world settings. Dynamic environments are the rule, not the exception; if a system is to reliably pursue its objective function, it should be able to handle unexpected environmental changes. Those who have delegated tasks to autonomous agents stand to benefit from those agents being more likely to succeed. Societal benefit of resilience is thus clear, assuming the original tasks were of societal benefit themselves. It is noteworthy that the recent global pandemic perturbed many aspects of the environments in which we operated. In such cases, people used to certain kinds of collaboration before the pandemic may have found it easier adjusting to the unfamiliar constraints that were imposed. We believe our results reflect a quite specific kind of benefit that automated agents can derive from collaborating with one other. We do note that many usual caveats on AI research apply, for instance where the original task itself is not of societal benefit; we also note that communication sharing experience could potentially open questions of privacy. We leave this for future work and note potential solutions in existing work on differential privacy or federated learning.

The communication protocol we suggest here is imposed on the agents by a designer. In future work we intend to investigate other approaches for facilitating collaboration. In addition, we intend to examine settings in which instead of imposing a collaboration method, the designer merely introduces a new collaboration method and agents can choose whether or not to adopt it. In such settings, the designer may incentivize agents to collaborate, for example by using *reward shaping*.

References

[1] H. B. Ammar, E. Eaton, M. E. Taylor, D. C. Mocanu, K. Driessens, G. Weiss, and K. Tüyls. An automated measure of MDP similarity for transfer in reinforcement learning. *AAAI Workshop - Technical Report*, WS-14-07:31–37, 2014.

[2] D. P. Bertsekas. *Dynamic Programming: Deterministic and Stochastic Models*. Prentice-Hall, Inc., USA, 1987.

[3] T. G. Dietterich. An overview of MAXQ hierarchical reinforcement learning. *Lecture Notes in Artificial Intelligence (Subseries of Lecture Notes in Computer Science)*, 1864:26–44, 2000.

[4] R. Dobrushin. Prescribing a system of random variables by conditional distributions. *Theory of Probability and Its Applications*, 15:458–486, 1970.

[5] J. N. Foerster, Y. M. Assael, N. De Freitas, and S. Whiteson. Learning to communicate with deep multi-agent reinforcement learning. *Advances in Neural Information Processing Systems*, pages 2145–2153, 2016.

[6] N. Jaques, A. Lazaridou, E. Hughes, C. Gulcehre, P. A. Ortega, D. J. Strouse, J. Z. Leibo, and N. de Freitas. Social influence as intrinsic motivation for multi-agent deep reinforcement learning. *36th International Conference on Machine Learning, ICML 2019*, 2019-June:5372–5381, 2019.

[7] Y. Liang and B. Li. Parallel knowledge transfer in multi-agent reinforcement learning. *arXiv*, 2020.

[8] M. L. Littman. Markov games as a framework for multi-agent reinforcement learning. *Machine Learning Proceedings 1994*, pages 157–163, 1994.

[9] R. Lowe, Y. Wu, A. Tamar, J. Harb, P. Abbeel, and I. Mordatch. Multi-agent actor-critic for mixed cooperative-competitive environments. *Advances in Neural Information Processing Systems*, 2017-December:6380–6391, 2017.

[10] A. Mahajan, T. Rashid, M. Samvelyan, and S. Whiteson. MAVEN: Multi-agent variational exploration, 2019.

[11] S. Padakandla. A Survey of Reinforcement Learning Algorithms for Dynamically Varying Environments. *arXiv*, pages 1–15, 2020.

[12] A. Pattanaik, Z. Tang, S. Liu, G. Bommannan, and G. Chowdhary. Robust Deep Reinforcement Learning with adversarial attacks. *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS*, 3:2040–2042, 2018.

[13] R. Portelas, C. Colas, L. Weng, K. Hofmann, and P. Y. Oudeyer. Automatic curriculum learning for deep RL: A short survey. *IJCAI International Joint Conference on Artificial Intelligence*, 2021-Janua:4819–4825, 2020.

[14] Y. Qian, J. Wu, R. Wang, F. Zhu, and W. Zhang. Survey on Reinforcement Learning Applications in Communication Networks. *Journal of Communications and Information Networks*, 4(2), 2019.

[15] T. Rashid, M. Samvelyan, C. S. De Witt, G. Farquhar, J. Foerster, and S. Whiteson. QMIX: Monotonic value function factorisation for deep multi-agent reinforcement Learning. In *35th International Conference on Machine Learning, ICML 2018*, volume 10, 2018.

[16] K. Saulnier, D. Saldana, A. Prorok, G. J. Pappas, and V. Kumar. Resilient Flocking for Mobile Robot Teams. *IEEE Robotics and Automation Letters*, 2(2):1039–1046, 2017.

[17] J. Song, Y. Gao, H. Wang, and B. An. Measuring the distance between finite Markov decision processes. *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS*, pages 468–476, 2016.

[18] E. Vinitsky, Y. Du, K. Parvate, K. Jang, P. Abbeel, and A. Bayen. Robust Reinforcement Learning using Adversarial Populations. *arXiv*, 2020.

[19] C. Z. Xu, J. Rao, and X. Bu. URL: A unified reinforcement learning approach for autonomic cloud management. *Journal of Parallel and Distributed Computing*, 72(2):95–105, 2012.

[20] T. Zhang, W. Zhang, and M. M. Gupta. Resilient robots: Concept, review, and future directions. *Robotics*, 6(4):1–14, 2017.

[21] Z. Zhu, K. Lin, and J. Zhou. Transfer learning in Deep Reinforcement Learning: A survey, sep 2020.

7 Appendix

In this section we present results of our methodology in another domain, which we call "apple picking". In this domain, agents have to visit as many "apple" states as they can to collect a reward at each such state. Once all apples have been picked, the episode ends.

The perturbations we experimented with are similar as in section 5: adding obstacles and changing the possible "apple" locations.

We note that this environment in a way is easier than the original one, as the agents get reward for even completing part of the task. It is interesting to see in Table-2 that in this setting, where the objective is less complex, the different group of agents achieve similar resilience levels. However, confusion-based communication still achieves the highest resilience throughout our experiments.

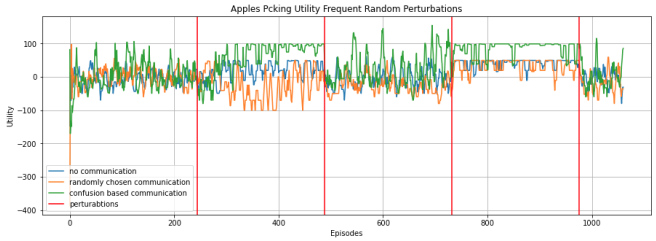

Figure 6: Apple Picking domain, Average utility for $K = 150$

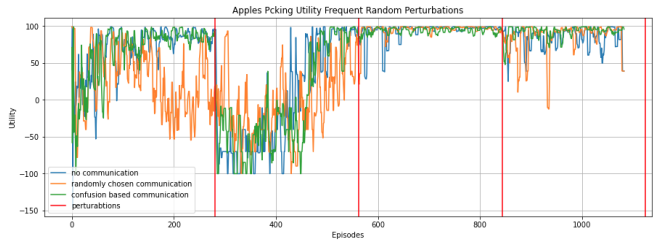

Figure 7: Apple Picking domain, Average utility for $K = 50$

Table 2: C-resilience values for groups of agents with different communication protocols in the "apple picking" domain, calculated over 10 experiments.

Communication Protocol	**K=50**	**K=150**	**K=200**
No communication	0.844	0.418	0.265
Communication of randomly selected observations	0.719	0.433	0.262
Confusion based communication	**0.881**	**0.451**	**0.327**

Table 3: C-resilience values for groups of agents with different communication protocols in the "apple picking" domain, calculated over 10 experiments.