Planning Not to Talk: Multiagent Systems that are Robust to Communication Loss

Mustafa O. Karabag*
The University of Texas at Austin
Texas, USA
karabag@utexas.edu

Cyrus Neary*
The University of Texas at Austin
Texas, USA
cneary@utexas.edu

Ufuk Topcu
The University of Texas at Austin
Texas, USA
utopcu@utexas.edu

ABSTRACT

In a cooperative multiagent system, a collection of agents executes a joint policy in order to achieve some common objective. The successful deployment of such systems hinges on the availability of reliable inter-agent communication. However, many sources of potential disruption to communication exist in practice, such as radio interference, hardware failure, and adversarial attacks. In this work, we develop joint policies for cooperative multiagent systems that are robust to potential losses in communication. More specifically, we develop joint policies for cooperative Markov games with reach-avoid objectives. First, we propose an algorithm for the decentralized execution of joint policies during periods of communication loss. Next, we use the total correlation of the state-action process induced by a joint policy as a measure of the intrinsic dependencies between the agents. We then use this measure to lower-bound the performance of a joint policy when communication is lost. Finally, we present an algorithm that maximizes a proxy to this lower bound in order to synthesize minimum-dependency joint policies that are robust to communication loss. Numerical experiments show that the proposed minimum-dependency policies require minimal coordination between the agents while incurring little to no loss in performance; the total correlation value of the synthesized policy is one fifth of the total correlation value of the baseline policy which does not take potential communication losses into account. As a result, the performance of the minimum-dependency policies remains consistently high regardless of whether or not communication is available. By contrast, the performance of the baseline policy decreases by twenty percent when communication is lost.

KEYWORDS

Multiagent Systems; Communication Loss; Information Theory

ACM Reference Format:

Mustafa O. Karabag, Cyrus Neary, and Ufuk Topcu. 2022. Planning Not to Talk: Multiagent Systems that are Robust to Communication Loss. In *Proc. of the 21st International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2022), Online, May 9–13, 2022*, IFAAMAS, 9 pages.

1 INTRODUCTION

In a cooperative multiagent systems, a team of decision-making agents aims to achieve a common objective through repeated interactions with each other and with a shared environment. Such multiagent systems are ubiquitous; many applications of autonomous systems — such as the coordination of autonomous vehicles, the control of networks of mobile sensors, or the control of traffic lights — can be modeled as collections of interacting agents [6, 19].

Inter-agent communication plays an essential role in the successful deployment of such multiagent systems. In particular, the coordination between agents via communication – their agreement upon the particular actions to collectively take at any given point in time – is often necessary for the successful implementation of an optimal joint policy [5]. However, many possible sources of communication disruption exist in practice, such as radio interference, hardware failure, or even adversarial attacks intended to sabotage the team. Lost or unreliable communication can result in substantial degradation of the team's performance, because it removes the agents' ability to coordinate. Despite this reliance of the team's performance on communication, multiagent learning and planning algorithms typically do not offer robustness guarantees against possible losses in communication.

In this work, we study multiagent systems that are robust to such communication losses. In detail, we study sequential multiagent decision problems formulated as cooperative Markov games with reach-avoid objectives [1, 15]. We make the following contributions.

(1) Imaginary Play for Policy Execution During Intermittent Communication. During periods of lost communication, each agent maintains imaginary versions of their teammates' states and actions using the pre-agreed-upon joint policy and a model of the environment's stochastic dynamics. By maintaining such imaginary copies of their teammates, each agent may act according to a model of how their teammates are likely to behave, without receiving any communicated information from them. Once communication is re-established the agents share updates, correct their imaginary models, and proceed with policy execution as normal until communication is lost again, or until the team's task is complete.

(2) Theoretical Results: Total Correlation as a Measure of Policy Robustness to Communication Loss. We use the total correlation [27] – a generalization of the mutual information – of the stochastic state-action process induced by the joint policy as a measure of how reliant that particular policy is on communication. To relate this measure to the performance of the policy, we provide lower bounds on the value function achieved during intermittent communication, in terms of the total correlation of the policy and the value function it achieves when communication is available. In addition to the policy synthesis algorithm described below, this lower bound provides a means to select communication resources that are sufficient to achieve a particular performance while using noisy communication channels.

(3) Synthesis of Policies Robust to Intermittent Communication. To synthesize *minimum-dependency policies* that remain performant

*These authors contributed equally to this work.

Proc. of the 21st International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2022), P. Faliszewski, V. Mascardi, C. Pelachaud, M.E. Taylor (eds.), May 9–13, 2022, Online.

under intermittent communication, we present an algorithm that maximizes a proxy to the lower bound described above. This optimization problem is formulated as a difference of convex terms. We solve for local optima using the convex-concave procedure [29].

Numerical results empirically demonstrate the effectiveness of the proposed algorithms for communication-free policy execution and for the synthesis of minimum-dependency joint policies. When communication is not restricted, the synthesized minimum-dependency policies enjoy task performance that is similar to a baseline policy that does not take potential communication losses into account. However, the minimum-dependency policies require minimal coordination between agents; the total correlation value of their joint state-action processes is roughly one fifth of the total correlation value of the process induced by the baseline policy.

As a result, the performance of the minimum-dependency policies remain constant, even when communication between agents is restricted to be entirely unavailable. By contrast, we observe a twenty percent degradation in the performance of the baseline policy when communication is lost.

Outline. In §2 we discuss related work. In §3 we introduce preliminary material as well as the notation used throughout the paper. We present our problem statement and an illustrative running example in §4. The proposed algorithms for policy execution during communication losses are presented in §5. The paper's theoretical results and their implications are discussed in §6 and §7. We give the proofs of these theoretical results in [12]. In §8 we present the proposed formulation and solution to the policy synthesis problem, before presenting the experimental results in §9.

2 RELATED WORK

Multiagent decision-making problems have been formulated using several models, e.g., multiagent Markov decision processes (MDPs) [5]. Our problem setting, in which each agent has independent transitions and may only observe their own local state, is most similar to transition-independent decentralized MDPs (Dec-MDPs) [3]. However, while this work considers the fully decentralized setting – the agents cannot communicate at all – we consider the setting in which communication is allowed but unreliable. We note that Dec-MDPs are a special case of decentralized partially observable MDPs (Dec-POMDPs) [18], which are notoriously difficult to solve in general when the agents cannot communicate. In fact, even policy synthesis for finite-horizon transition-independent Dec-MDPs without communication is NP-complete [10].

Prior work for multiagent systems considers imposing specific communication structures between the agents, either as a dependency graph [11], or as a subset of joint states at which the agents may communicate [17]. In addition to these fixed communication structures, the papers [2, 28] consider communication as an explicit action that can be taken by the agents, leading to dynamic communication structures that change over time. While all of the above works consider synthesizing optimal behavior according to specific communication structures, our work studies multiagent systems that are robust to unpredictable communication losses.

To render the multiagent systems robust to communication loss, our work aims to minimize intrinsic dependencies between the agents. As a measure of such dependencies, we use the total correlation [27] – an information theoretic measure – of the state-action process induced by the joint policy. Information theoretic measures have been studied in single-agent MDPs [9, 14, 22, 25]. In particular, [25] synthesizes single-agent policies that minimize the transfer entropy from the state process to the action process with the purpose of minimizing the reliance of the policy on the underlying state process. By contrast, our work considers a multiagent setting and introduces information theoretic measures with the specific purpose of providing guarantees on the performance of the team under communication loss. The paper [9] proposes to minimize the mutual information between the underlying state process and the agent's action process in the context of single-agent reinforcement learning. By contrast, we study the multiagent setting and provide bounds on the performance of the entire team, when the agents have intermittent communication. Furthermore, we provide an optimization problem to synthesize joint policies that are robust to communication loss. In the multiagent reinforcement learning setting, [26] consider minimizing the mutual information between the state processes and the messages shared between the agents, but do not provide theoretical result on the performance of the team when communication is only intermittently available.

The centralized training decentralized execution paradigm in has recently drawn attention in multiagent reinforcement learning [16, 21, 23, 24]. These works enforce independence between the agents by imposing that the team's value function can be decomposed into local functions for each of the agents. In our work, we do not consider decomposition of the value function, but instead directly synthesize a joint policy that leads to intrinsic independence between agents. Another method to compute policies for decentralized execution, is to post-process a given joint policy. For example, [8] uses the rate-distortion framework [7] for this purpose. Our work does not assume a joint policy to be given a priori; we instead directly synthesize a joint policy that minimizes dependencies.

As discussed above, prior works tackle communication loss by making the policies fully decentralized [21, 23], or by having the agents maintain beliefs about their teammates [2, 28]. While belief-based myopic approaches lead to high reward for a single step, they do not guarantee optimality over entire paths. Instead of maintaining such belief distributions, in our work each agent creates imaginary copies of its teammates when communication is lost; this idea is similar in spirit to the concept of digital twins [4]. Combined with total correlation, the proposed imaginary play algorithm leads to performance guarantees over the entire path.

3 PRELIMINARIES

In this section, we outline several definitions and notation used throughout the paper. Given a finite collection of N agents – which we index by $i \in \{1, 2, \ldots, N\}$ – we model the dynamics of each individual agent using a Markov decision process (MDP) $\mathcal{M}^i$. An MDP is a tuple $\mathcal{M}^i = (\mathcal{S}^i, s^i_I, \mathcal{A}^i, \mathcal{T}^i)$. Here, $\mathcal{S}^i$ is a finite set of states, $s^i_I \in \mathcal{S}^i$ is an initial state, $\mathcal{A}^i$ is a finite set of available actions, and $\mathcal{T}^i : \mathcal{S}^i \times \mathcal{A}^i \to \Delta(\mathcal{S}^i)$ is a transition probability function. We use $\Delta(\mathcal{S}^i)$ to denote the set of all probability distributions over the state space $\mathcal{S}^i$. For notational brevity, we use $\mathcal{T}^i(s^i, a^i, y^i)$ to denote the probability of y^i given by the distribution $\mathcal{T}^i(s^i, a^i)$. A path ξ^i in the MDP $\mathcal{M}^i$ is an infinite sequence $\xi^i = s_0^i a_0^i s_1^i a_1^i \ldots$ of state-action pairs such that for every $t = 0, 1, \ldots, \mathcal{T}^i(s_t^i, a_t^i, s_{t+1}^i) > 0$.

Given such collection of agents, along with their corresponding MDPs $\mathcal{M}^i$, we formulate the team's decision problem as a cooperative Markov game $\boldsymbol{\mathcal{M}}$. The game is considered cooperative because all agents share a common objective. A cooperative Markov game involving N agents, each of which is modeled by an MDP $\mathcal{M}^i = (\mathcal{S}^i, s_I^i, \mathcal{A}^i, \mathcal{T}^i)$, is given by the tuple $\boldsymbol{\mathcal{M}} = (\boldsymbol{\mathcal{S}}, \boldsymbol{s_I}, \boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{T}})$. Here, $\boldsymbol{\mathcal{S}} = \mathcal{S}^1 \times \mathcal{S}^2 \times \ldots \times \mathcal{S}^N$ is the finite set of joint states, $\boldsymbol{s_I} = (s_I^1, \ldots, s_I^N)$ is the joint initial state, $\boldsymbol{\mathcal{A}} = \mathcal{A}^1 \times \mathcal{A}^2 \times \ldots \times \mathcal{A}^N$ is the finite set of joint actions, and $\boldsymbol{\mathcal{T}} : \boldsymbol{\mathcal{S}} \times \boldsymbol{\mathcal{A}} \to \Delta(\boldsymbol{\mathcal{S}})$ is the joint transition probability function. For notational brevity, we use $\boldsymbol{\mathcal{T}}(\boldsymbol{s}, \boldsymbol{a}, \boldsymbol{y})$ to denote the probability of $\boldsymbol{y}$ in the distribution $\boldsymbol{\mathcal{T}}(\boldsymbol{s}, \boldsymbol{a})$. The joint transition function $\boldsymbol{\mathcal{T}}$ is defined as $\boldsymbol{\mathcal{T}}(\boldsymbol{s}, \boldsymbol{a}, \boldsymbol{y}) = \prod_{i=1}^N \mathcal{T}(s^i, a^i, y^i)$ for all $\boldsymbol{s} = (s^1, \ldots, s^N), \boldsymbol{y} = (y^1, \ldots, y^N) \in \boldsymbol{\mathcal{S}}$ and $\boldsymbol{a} = (a^1, \ldots, a^N) \in \boldsymbol{\mathcal{A}}$. We note that the definition of the joint transition function $\boldsymbol{\mathcal{T}}$ assumes that the dynamics of the individual agents are independent. We use $\boldsymbol{\xi} = \boldsymbol{s}_0 \boldsymbol{a}_0 \boldsymbol{s}_1 \boldsymbol{a}_1 \ldots$ to denote the joint path of the agents. The joint path $\boldsymbol{\xi}$ is the union of individual paths $\xi^1, \ldots, \xi^N$.

A (stationary) joint policy $\pi_{joint} : \boldsymbol{\mathcal{S}} \to \Delta(\boldsymbol{\mathcal{A}})$ is a mapping from a particular joint state to a probability distribution over joint actions. We use $\pi_{joint}(\boldsymbol{s}, \boldsymbol{a})$ to denote the probability that action $\boldsymbol{a}$ is selected by π_{joint} given the team is in joint state $\boldsymbol{s}$.

In this work we consider team reach-avoid problems. That is, the team's objective is to collectively reach some target set $\boldsymbol{\mathcal{S}}_{\mathcal{T}} \subseteq \boldsymbol{\mathcal{S}}$ of states, while avoiding a set $\boldsymbol{\mathcal{S}}_{\mathcal{A}} \subseteq \boldsymbol{\mathcal{S}}$ of states. The centralized planning problem then, is to solve for a team policy π_{joint} maximizing the probability of reaching $\boldsymbol{\mathcal{S}}_{\mathcal{T}}$ from the team's initial joint state $\boldsymbol{s_I}$, while avoiding $\boldsymbol{\mathcal{S}}_{\mathcal{A}}$. We call this probability value the reach-avoid probability. More formally, we say that a *path* $\boldsymbol{\xi} = \boldsymbol{s}_1 \boldsymbol{a}_1 \boldsymbol{s}_2 \boldsymbol{a}_2 \ldots$ successfully reaches the target set $\boldsymbol{\mathcal{S}}_{\mathcal{T}}$ if there exists some time M such that $\boldsymbol{s}_M \in \boldsymbol{\mathcal{S}}_{\mathcal{T}}$ and for all $t < M$, $\boldsymbol{s}_t \notin \boldsymbol{\mathcal{S}}_{\mathcal{A}}$.

For notational convenience, we use $\boldsymbol{s}^{-i} \in \mathcal{S}^1 \times \ldots \times \mathcal{S}^{i-1} \times \mathcal{S}^{i+1} \times \ldots \times \mathcal{S}^N$ to denote the states of agent i's teammates, excluding agent i itself. By $\boldsymbol{\mathcal{S}}^{-i} = \mathcal{S}^1 \times \ldots \times \mathcal{S}^{i-1} \times \mathcal{S}^{i+1} \times \ldots \mathcal{S}^N$, we denote the set of all collections of the states of agent i's teammates. We similarly use $\boldsymbol{a}^{-i}$ and $\boldsymbol{\mathcal{A}}^{-i}$ to denote the actions of agent i's teammates and the set of all possible such collections of actions, respectively.

We use $x_{\boldsymbol{s},\boldsymbol{a}}$ to denote the occupancy measure of the state-action pair $(\boldsymbol{s}, \boldsymbol{a})$, i.e., the expected number of times that action $\boldsymbol{a}$ is taken at state $\boldsymbol{s}$. Similarly, x_{s^i,a^i} denotes the the occupancy measure of the state-action pair (s^i, a^i) for agent i. We note that $x_{s^i,a^i} = \sum_{\boldsymbol{s}^{-i} \in \boldsymbol{\mathcal{S}}^{-i}} \sum_{\boldsymbol{a}^{-i} \in \boldsymbol{\mathcal{A}}^{-i}} x_{s^i, \boldsymbol{s}^{-i}, a^i, \boldsymbol{a}^{-i}}$.

The entropy [7] of a discrete random variable Y with a support $\mathcal{Y}$ is $H(Y) = -\sum_{y \in \mathcal{Y}} \Pr(Y = y) \log(\Pr(Y = y))$. The Kullback-Leibler (KL) divergence [7] between discrete probability distributions Q^1 and Q^2 with supports $\boldsymbol{Q}^1$ and $\boldsymbol{Q}^2$, respectively, is

$$KL(Q^1 || Q^2) = \sum_{q \in \boldsymbol{Q}^1} Q^1(q) \log\left(\frac{Q^1(q)}{Q^2(q)}\right).$$

4 MULTIAGENT PLANNING AND COMMUNICATION

In this work we study the cooperative execution of joint policies when communication between the agents is intermittent, and in some cases entirely absent. We begin by discussing the inter-agent communication that is necessary, in general, for team policy execution before we present the problem statement.

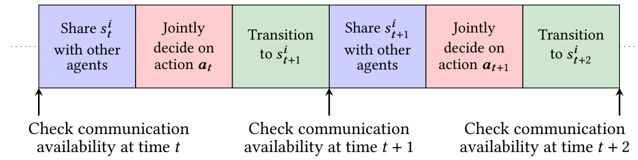

Figure 1: An illustration of the procedure for joint policy execution. At each decision step, all agents simultaneously check whether communication is available. If it is available, the agents share their local states in order to obtain the current joint state s_t before agreeing upon a joint action a_t sampled from the joint policy π_{joint}. Otherwise, the agents execute π_{joint} using imaginary play, outlined in Algorithm 1.

The agents operate in the environment by collectively executing a joint policy π_{joint}. Each agent only has access to its own local state and action information. The agents must communicate their local states s_t^i at each timestep t and use π_{joint} to collectively decide on a joint action $\boldsymbol{a}$, as is illustrated in Figure 1. Each agent executes its own local component a^i of the selected joint action and resultingly transitions to its next local state s_{t+1}^i.

We note that the joint policy requires communication between the agents at every time step. On the other hand, if the team suffers a communication failure at any given timestep, then they will not be able to share the necessary information to execute the joint policy in the manner described above.

Problem statement. (1) Create a planning algorithm that enables the agents to perform decentralized execution of the joint policy when communication is lost. (2) Quantify the performance of the team when such an algorithm is used during communication losses. (3) Synthesize joint policies that remain performant, even when communication is lost.

An illustrative example. We present the running example illustrated in Figure 2 to help motivate the above problems. Two robots R_1 and R_2 must simultaneously navigate to their respective targets T_1 and T_2. The robots must also maintain a pre-specified minimum distance from each other during navigation to reduce the risk of the robots colliding. Furthermore, rough terrain makes large portions of the navigation environment impassable, requiring the robots to navigate through one of two narrow valleys in order to reach their targets. Finally, a lake of water presents risk to the robots; if either of them accidentally falls into the water, then the team fails its task. The team's task is only considered complete once both robots have safely navigated to their respective targets. The objective of the agents is to complete this task with as high a probability as possible.

Given this team task, the robots may both choose to navigate through the bottom valley in order to reach their targets. This route is shorter than traveling through the top valley for both robots, and it avoids passing near the dangerous body of water. However, they must take turns when passing through the shared bottom valley to ensure that the robots never get too close to each other. Such behavior requires communication; both agents should share

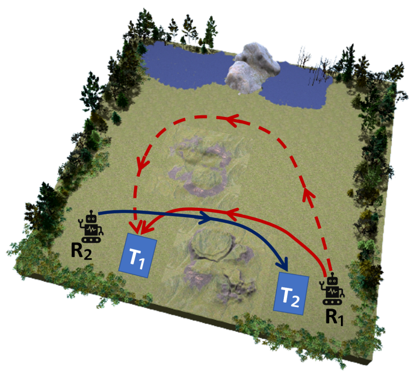

Figure 2: A two-agent navigation example. Two robots, R_1 and R_2, must navigate to their respective targets, T_1 and T_2, while avoiding collisions with each other. The terrain necessitates that each robot navigates through one of two valleys, while avoiding the water at the top of the map. During policy execution, each robot may only observe its own location, however, the agents communicate their locations with each other when such communication is possible. The colored curves illustrate different paths that the robots might to take, depending on the availability of communication.

their current location and intended next action in order to avoid simultaneously entering the valley.

By contrast, if no communication is available, the robots may instead choose to navigate through different valleys altogether. This joint behavior increases the risk that one might fall into the water, but it removes the requirement that the robots communicate.

5 DECENTRALIZED POLICY EXECUTION UNDER COMMUNICATION LOSS

Consider a scenario in which the team of agents lose communication during the execution of a joint policy. Under such circumstances, the agents cannot execute the policy as outlined in the previous section and as illustrated in Figure 1. Each agent must instead decide on a local action for itself, without knowing the local states or actions of its teammates. To achieve this decentralized execution of the joint policy, we propose to use *imaginary play*; each agent maintains imaginary copies of its teammates during periods of communication loss. That is, given the joint policy, the stochastic dynamics of the Markov game, and the states of their teammates at the last timestep before communication was lost, the agents maintain simulated copies of their teammates' states. Each agent then uses its own imaginary version of the entire team to sample a joint action from the policy, executes its own local component of that joint action, and then simulates the next states of its imaginary teammates. In the next time step, this process repeats.

Algorithm 1 details this process of joint policy execution through imaginary play. Before the communication breaks, every Agent i shares its state s_t^i with its teammates at every time step, and the agents collectively decide on a joint action $\boldsymbol{a}_t$. When the communication breaks at time t_{loss}, every agent i starts to play with imaginary teammates. That is, based on the last joint action $\hat{\boldsymbol{a}}_{t_{loss-1},i}$ prior to communication loss, every Agent i uses the joint transition function $\mathcal{T}$ to sample an imaginary state $\hat{s}^j_{t_{loss-1}+1,i}$ for each of its teammates. Here, $\hat{s}^j_{t,i}$ denotes Agent i's belief on Agent j's state at time t, and $\hat{\boldsymbol{a}}_{t,i}$ denotes Agent i's belief on the joint action at

Algorithm 1: Policy Execution with Imaginary Play

$t_{loss} = \infty$.
for $t = 0, 1, \ldots$ **do**
  if *Communication is possible* **then**
    For every $i \in [N]$ **do in parallel**
      Share s_t^i with other agents.
      Set $\hat{s}^j_{t,i} = s^j_t$ for all $j \neq i$.
      Jointly decide on an action $\boldsymbol{a}_t \sim \pi_{joint}(\boldsymbol{s}_t)$.
      Set $\hat{\boldsymbol{a}}_{t,i} = \boldsymbol{a}_t$.
      Execute a_t^i and transition to $s^i_{t+1} \sim \mathcal{T}(s^i_t, a^i_t)$.
  else
    Set $t_{loss} = t$. **break**
for $t = t_{loss}, t_{loss} + 1, \ldots$ **do**
  if $t = 0$ **then**
    Set $\hat{s}^j_{t,i} = s^j_I$ for all $j \neq i$.
  else
    Sample $\hat{s}^j_{t,i} \sim \mathcal{T}^j(\hat{s}^j_{t-1,i}, \hat{a}^j_{t-1,i})$ for all $j \neq i$.
  For every $i \in [N]$ **do in parallel**
    Sample $\hat{s}^j_{t,i} \sim \mathcal{T}^j(\hat{s}^j_{t-1,i}, \hat{a}^j_{t,i})$ for all $j \neq i$.
    Decide on an action
    $\hat{\boldsymbol{a}}_{t,i} \sim \pi_{joint}(\hat{s}^1_{t,i}, \ldots, \hat{s}^{i-1}_{t,i}, s^i_t, \hat{s}^{i+1}_{t,i}, \ldots, \hat{s}^N_{t,i})$.
    Execute $\hat{a}^i_{t,i}$ and transition to $s^i_{t+1} \sim \mathcal{T}(s^i_t, \hat{a}^i_{t,i})$.

Algorithm 2: Policy Execution with Intermittent Communication

for $t = 0, 1, \ldots$ **do**
  if *Communication is possible* **then**
    For every $i \in [N]$ **do in parallel**
      Share s_t^i with other agents.
      Set $\hat{s}^j_{t,i} = s^j_t$ for all $j \neq i$.
      Jointly decide on an action $\boldsymbol{a}_t \sim \pi_{joint}(\boldsymbol{s}_t)$.
      Set $\hat{\boldsymbol{a}}_{t,i} = \boldsymbol{a}_t$.
      Execute a_t^i and transition to $s^i_{t+1} \sim \mathcal{T}(s^i_t, a^i_t)$.
  else
    For every $i \in [N]$ **do in parallel**
      if $t = 0$ **then**
        Set $\hat{s}^j_{t,i} = s^j_I$ for all $j \neq i$.
      else
        Sample $\hat{s}^j_{t,i} \sim \mathcal{T}^j(\hat{s}^j_{t-1,i}, \hat{a}^j_{t-1,i})$ for all $j \neq i$.
      Decide on an action
      $\hat{\boldsymbol{a}}_{t,i} \sim \pi_{joint}(\hat{s}^1_{t,i}, \ldots, \hat{s}^{i-1}_{t,i}, s^i_t, \hat{s}^{i+1}_{t,i}, \ldots, \hat{s}^N_{t,i})$.
      Execute $\hat{a}^i_{t,i}$ and transition to $s^i_{t+1} \sim \mathcal{T}(s^i_t, \hat{a}^i_{t,i})$.

time t. Then, at every time step $t \geq t_{loss}$, every agent i samples a joint game action $\hat{\boldsymbol{a}}_{t,i}$ using the joint policy and these imagined teammate states $\hat{s}^1_{t,i}, \ldots, \hat{s}^N_{t,i}$. Every agent i then executes the local part $\hat{a}^i_{t,i}$ of its joint action $\hat{\boldsymbol{a}}_{t,i}$ and transitions to its next local state $s^i_{t+1,i}$. Based on its imagined joint action $\hat{\boldsymbol{a}}_{t,i}$ and the previous imagined teammate states $\hat{s}^1_{t,i}, \ldots, \hat{s}^N_{t,i}$, every Agent i also samples next imaginary states $\hat{s}^1_{t+1,i}, \ldots, \hat{s}^N_{t+1,i}$ for its teammates.

We remark that while every agent operates cooperatively with its imaginary teammates under a communication loss, the objective of the team is evaluated with respect to the true joint state.

In some scenarios, communication failures may be intermittent as opposed to being persistent. That is, the agents may re-gain communication capabilities after periods of communication loss. For such scenarios, we propose that the agents follow imaginary play whenever communication is lost, update their imaginary representations when communication is re-established, and coordinate directly with their real teammates for as long as communication remains available. Algorithm 2 describes this proposed approach for policy execution with intermittent communication.

6 MEASURING THE INTRINSIC DEPENDENCIES BETWEEN THE AGENTS

Given a joint policy, the team's performance under imaginary play will differ from the performance that would have been achieved under full communication. Recall that we measure the team's performance as their probability of reaching the set $\boldsymbol{\mathcal{S}}_{\mathcal{T}} \subseteq \boldsymbol{\mathcal{S}}$ of target joint states from the initial joint state $\boldsymbol{s}_I$, while avoiding $\boldsymbol{\mathcal{S}}_{\mathcal{A}} \subseteq \boldsymbol{\mathcal{S}}$.

Intuitively, the team's performance under imaginary play will depend on how much the behavior of any particular agent changes according to the behavior of its teammates, as well as on how much the behavior of an agent's imaginary teammates differs from that of its actual teammates. In other words, if the joint policy induces high intrinsic dependencies between the agents, then policy execution using imaginary play will lead to different outcomes than policy execution with fully available communication.

Total correlation [27] measures the amount of information shared between multiple random variables. Let X^i be a random variable over the paths $\xi^i = s^i_0 a^i_0 s^i_1 a^i_1 \ldots$ of Agent i and $\boldsymbol{X}$ be a random variable over the joint paths $\boldsymbol{\xi} = \boldsymbol{s}_0 \boldsymbol{a}_0 \boldsymbol{s}_1 \boldsymbol{a}_1 \ldots$ of all agents induced by the joint policy π_{joint} under full communication. We refer to the total correlation $C_{\pi_{joint}}$ of joint policy π_{joint} as

$$C_{\pi_{joint}} = \left[\sum_{i=1}^{N} H(X^i)\right] - H(\boldsymbol{X}).$$

There are two contributing factors to the value of the total correlation. Firstly, if the actions of a particular agent depend on the local states of its teammates, then this will increase the value of the total correlation. Secondly, if the joint policy is randomized and the agents need to coordinate on an action – the action of each agent depends on the actions simultaneously selected by its teammates – then this will also increase the value of the total correlation.

If the total correlation is 0, then there are no dependencies between the agents, i.e., the path of any given agent is independent from those of its teammates. As the dependencies between the agents increase, so too does the value of the total correlation. We additionally remark that when there are only two agents, the total correlation between the state-action processes of the agents is equivalent to the mutual information between them.

We accordingly propose to use total correlation as measure of the intrinsic dependencies between the agents induced by a particular joint policy. In the next section, we relate the value of total correlation to the team's performance under communication loss.

7 PERFORMANCE GUARANTEES UNDER COMMUNICATION LOSS

In this section, we provide lower bounds on the team's performance under a particular joint policy during communication loss. These theoretical results are accomplished by relating the total correlation of the joint policy to the distribution over paths induced by executing that policy using imaginary play. The proofs of all results are included in [12].

Relating total correlation to imaginary play. Let $\boldsymbol{\Gamma}^{full}$ be the distribution of joint paths induced by the joint policy executed with full communication. Also, let $\boldsymbol{\Gamma}^{img}_0$ be the distribution of joint paths under imaginary play with no communication, i.e., $t_{loss} = 0$ in Algorithm 1. By the definition of total correlation, we have

$$C_{\pi_{joint}} = \left[\sum_{i=1}^{N} H(X^i)\right] - H(\boldsymbol{X}) = KL(\boldsymbol{\Gamma}^{full} || \boldsymbol{\Gamma}^{img}_0).$$

From this definition, we observe that when $C_{\pi_{joint}} = 0$, the induced distributions $\boldsymbol{\Gamma}^{full}, \boldsymbol{\Gamma}^{img}_0$ must be the same since $KL(\boldsymbol{\Gamma}^{full} || \boldsymbol{\Gamma}^{img}_0) = 0$. Furthermore, as the value of $C_{\pi_{joint}}$ increases, the KL divergence between $\boldsymbol{\Gamma}^{full}$ and $\boldsymbol{\Gamma}^{img}_0$ increases as well.

On the closeness between path distributions induced by different communication availabilities. The value of $C_{\pi_{joint}}$ measures how much the distribution over paths $\boldsymbol{\Gamma}^{img}_0$ differs from $\boldsymbol{\Gamma}^{full}$ in the setting where the agents never communicate, i.e. $t_{loss} = 0$. We now consider a scenario in which the agents communicate and operate together for some time, then lose communication and switch to imaginary play at time $t_{loss} > 0$. Let $\boldsymbol{\Gamma}^{img}_{t_{loss}}$ be the distribution of joint paths for an arbitrary positive value of t_{loss}. Intuitively, we expect that the initial period of communication should not increase the KL divergence between $\boldsymbol{\Gamma}^{full}$ and $\boldsymbol{\Gamma}^{img}_{t_{loss}}$ in comparison with the case when $t_{loss} = 0$. Lemma 7.1 confirms this intuition.

Lemma 7.1. *For every $t_{loss} \in \{0, 1, \ldots\} \cup \{\infty\}$ in Algorithm 1,*

$$KL(\boldsymbol{\Gamma}^{full} || \boldsymbol{\Gamma}^{img}_0) \geq KL(\boldsymbol{\Gamma}^{full} || \boldsymbol{\Gamma}^{img}_{t_{loss}}).$$

We can similarly show that arbitrary intermittent communication does not increase the KL divergence between the induced path distributions. Let $\Lambda = \lambda_0, \lambda_1, \ldots$ be a sequence of binary values such that $\lambda_t = 1$ if and only if communication is available at time t. The KL divergence between $\boldsymbol{\Gamma}^{full}$ and $\boldsymbol{\Gamma}^{img}_0$ is not higher than that between $\boldsymbol{\Gamma}^{full}$ and $\boldsymbol{\Gamma}^{int}_{\Lambda}$, where $\boldsymbol{\Gamma}^{int}_{\Lambda}$ is the distribution of paths under intermittent communication with an arbitrary sequence Λ of communication availability. Furthermore, as shown in the second half of Lemma 7.2, when $\Lambda = \lambda_0, \lambda_1, \ldots$ is a random sequence of communication availabilities, the communication dropout rate q is related to the KL divergence between the distributions.

Lemma 7.2. *Let $\Lambda = \lambda_0, \lambda_1, \ldots$ be an arbitrary sequence of communication availability in Algorithm 2. Then,*

$$KL(\Gamma^{full}||\Gamma_0^{img}) \geq KL(\Gamma^{full}||\Gamma_\Lambda^{int}).$$

Let $\Lambda = \lambda_0, \lambda_1, \ldots$ be a random sequence of binary values such that every λ_t is independently sampled from a Bernoulli random variable with parameter $1-q$, and $\Gamma^{int} = \mathbb{E}_\Lambda\left[\Gamma_\Lambda^{int}\right]$. Then,

$$KL(\Gamma^{full}||\Gamma_0^{img}) \geq KL(\Gamma^{full}||\Gamma^{int})/q.$$

Lemmas 7.1 and 7.2 bound the KL divergence between path distributions when the communication availability is independent from the histories of the agents. In practice, communication availability may depend on the state-action processes of the agents. For example, in the multiagent navigation task depicted in Figure 2, the agents may not be able to communicate if they do not have line-of-sight, e.g., when they are on the opposite sides of the mountains. Lemma 7.3 shows a stronger result: The distribution over joint paths under imaginary play is close to Γ^{full} even when the communication availability is a function of the agents' histories.

Lemma 7.3. *Let $f : (\mathcal{S} \times \mathcal{A})^* \to \{0,1\}$ be an arbitrary function that determines the communication availability based on the team's joint history such that $\lambda_0 = f(\varepsilon)$ and $\lambda_t = f(\boldsymbol{s}_0\boldsymbol{a}_0 \ldots \boldsymbol{s}_{t-1}\boldsymbol{a}_{t-1})$. Let Γ_f^{img} be the distribution over joint paths induced by imaginary play (Algorithm 1) and communication availability dictated by f. Then,*

$$KL(\Gamma^{full}||\Gamma_0^{img}) \geq KL(\Gamma^{full}||\Gamma_f^{img}).$$

We remark that Algorithms 1 and 2 are agnostic to when the future communication failures happen. The lemmas do not assume a priori knowledge of the sequence of communication availability.

On the value of the reach-avoid probability under communication loss. We use the above results on the KL divergence between distributions of paths to derive bounds on the reach-avoid probability achieved by a particular joint policy under communication loss.

Let $\boldsymbol{v}^{full}$ be the reach-avoid probability induced by a joint policy with full communication, $\boldsymbol{v}^{img}$ be the reach-avoid probability of the same policy under imaginary play (Algorithm 1), and $\boldsymbol{v}^{int}$ be the reach-avoid probability under intermittent communication (Algorithm 2). Also, let $\mathcal{S}_\mathcal{D}$ be the states from which the probability of reaching $\mathcal{S}_\mathcal{T}$ is 0 under the joint policy. Define $len(\xi = \boldsymbol{s}_0\boldsymbol{a}_0\ldots) = \min\{t+1|\boldsymbol{s}_t \in \mathcal{S}_\mathcal{T} \cup \mathcal{S}_\mathcal{D}\}$ and $l^{full} = \mathbb{E}[len(\xi)|\xi \sim \Gamma^{full}]$.

Theorem 7.4 shows that the reach-avoid probability of a joint policy under imaginary play is lower-bounded by a function of the policy's reach-avoid probability with full communication and the value of $C_{\pi_{joint}}$, even when the communication availability depends on the agents' histories.

Theorem 7.4. *Let $f : (\mathcal{S} \times \mathcal{A})^* \to \{0,1\}$ be an arbitrary function that determines the communication availability based on the history of the agents such that $\lambda_t = f(\boldsymbol{s}_0\boldsymbol{a}_0 \ldots \boldsymbol{s}_{t-1}\boldsymbol{a}_{t-1})$. For this system,*

$$\boldsymbol{v}^{img} \geq \boldsymbol{v}^{full} - \sqrt{1-\exp(-C_{\pi_{joint}})}.$$

We now consider the setting in which the team's communication fails at some random time $t_{loss} \geq 0$ and does not recover thereafter. When t_{loss} follows a geometric distribution, we derive a stronger bound that relates the probability of communication failure at each time step to the reach-avoid probability under imaginary play.

Theorem 7.5. *Consider a communication system that fails with probability p at any communication step and never recovers, i.e., $\Pr(t_{loss} = t) = (1-p)^t p$ in Algorithm 1. For this system,*

$$\boldsymbol{v}^{img} \geq \max\left(\boldsymbol{v}^{full} - \sqrt{1-\exp(-C_{\pi_{joint}})}, \boldsymbol{v}^{full}(1-p)^{\frac{l^{full}}{\boldsymbol{v}^{full}}}\right).$$

When communication availability is intermittent, and can be modeled by a Bernoulli process, the reach-avoid probability under intermittent communication is directly lower-bounded by a function of the communication dropout rate q. We remark that the lower bound provides a means to select communication resources that are sufficient to achieve a particular performance while using noisy communication channels. In detail, consider a noisy communication channel on which the team must communicate. The code rate [7] can be adjusted according to the desired value of q, which in turn determines the value of the lower bound on $\boldsymbol{v}^{int}$.

Theorem 7.6. *Consider a communication system that fails with probability q at any communication step independent from the other communication steps. For this system,*

$$\boldsymbol{v}^{int} \geq \max\left(\boldsymbol{v}^{full} - \sqrt{1-\exp(-qC_{\pi_{joint}})}, \boldsymbol{v}^{full}(1-q)^{\frac{l^{full}}{\boldsymbol{v}^{full}}}\right).$$

The lower bounds in Theorems 7.4, 7.5, and 7.6 show that the reach-avoid probability of a joint policy under communication loss depends on the total correlation of the joint policy, the reach-avoid probability achieved with full communication, the communication dropout rate, and the expected path length under the joint policy. When the total correlation is 0, the reach-avoid probability under communication loss is the same as the reach-avoid probability with full communication. As the total correlation of the joint policy increases, the values of the lower bounds decrease. During intermittent communication, if value of the dropout rate is 0, then the reach-avoid probability of the joint policy executed using imaginary play (Algorithm 1) or intermittent communication (Algorithm 2) is the same as when the policy is executed with full communication. When the communication dropout rates are 1, the reach-avoid probability under communication loss depends on the value of the total correlation. We note that the bounds are tight when either the communication dropout rate or the total correlation is 0.

8 JOINT POLICY SYNTHESIS

In this section, we discuss the synthesis of minimum-dependency joint policies π_{MD} that are robust to communication failures.

Entropy of paths for a single agent. Given the Markov game, a stationary joint policy π_{joint} induces a Markov chain. This Markov chain generates a stationary process $\boldsymbol{X}$, which is the joint path of the agents. The entropy $H(\boldsymbol{X})$ of a stationary process has a closed form expression in terms of the occupancy measures $x_{\boldsymbol{s},\boldsymbol{a}}$ of the joint state-action pairs $(\boldsymbol{s},\boldsymbol{a})$ [22]. The path of a single agent, on the other hand, follows a hidden Markov model where $\boldsymbol{X}$ is the underlying process and X^i is the observed process. However, the entropy $H(X^i)$ of a process that follows a hidden Markov model does not admit a closed-form expression.

Let x_{s^i,a^i} be the occupancy measure for the state-action pair $(s^i,a^i) \in \mathcal{S}^i \times \mathcal{A}^i$ under the joint policy π_{joint}. Consider a stationary process $\bar{X}^i$ that induces the same occupancy measures x_{s^i,a^i}

as the joint policy. The entropy $H(\bar{X}^i)$ of the stationary process is greater than or equal to the entropy $H(X^i)$ of the original process [22]. Since $H(X^i)$ does not admit a closed form expression, we instead upper bound $C_{\pi_{joint}}$ using $H(\bar{X}^i)$. Formally, we have

$$\bar{C}_{\pi_{joint}} = \left[\sum_{i=1}^{N} H(\bar{X}^i)\right] - H(\boldsymbol{X}) \geq C_{\pi_{joint}} = \left[\sum_{i=1}^{N} H(X^i)\right] - H(\boldsymbol{X}).$$

The policy synthesis optimization problem. To optimize the reach-avoid probability under communication loss, we would like to maximize the lower bound given in Theorem 7.5. However, due to the complex nature of this lower bound, we propose to instead use the following optimization problem as a proxy to the original problem:

$$\sup_{\pi_{joint}} \boldsymbol{v}^{full} - \delta l^{full} - \beta \bar{C}_{\pi_{joint}} \tag{1}$$

where $\delta > 0$ and $\beta > 0$ are constants.

We now represent (1) in terms of occupancy measures and construct the optimization problem for synthesis. We first preprocess $\boldsymbol{\mathcal{M}}$ to ensure that $\bar{C}_{\pi_{joint}}$ is well-defined. Define $\boldsymbol{\mathcal{S}}_{\mathcal{D}} = \{\boldsymbol{s} | \max_{\pi_{joint}} \boldsymbol{v}_{joint} = 0$ when the path begins at $\boldsymbol{s}\}$, the set of all states from which the reach-avoid task is violated with probability 1. We note that $\boldsymbol{\mathcal{S}}_{\mathcal{D}} \supseteq \boldsymbol{\mathcal{S}}_{\mathcal{A}}$. For synthesis, we add an absorbing end state $\boldsymbol{s}_\epsilon = (s_\epsilon^1, \ldots, s_\epsilon^N)$ and a joint action $\epsilon = (\epsilon^1, \ldots, \epsilon^N)$ to $\boldsymbol{\mathcal{M}}$, which represent the end of the game in terms of the reach-avoid objective. Every $\boldsymbol{s} \in \boldsymbol{\mathcal{S}}_{\mathcal{T}} \cup \boldsymbol{\mathcal{S}}_{\mathcal{D}}$ has a single action ϵ, and $\boldsymbol{\mathcal{T}}(\boldsymbol{s}, \epsilon, \boldsymbol{s}_\epsilon) = 1$ for all $\boldsymbol{s} \in \boldsymbol{\mathcal{S}}_{\mathcal{T}} \cup \boldsymbol{\mathcal{S}}_{\mathcal{D}}$, i.e., the states in $\boldsymbol{\mathcal{S}}_{\mathcal{T}} \cup \boldsymbol{\mathcal{S}}_{\mathcal{D}}$ deterministically transitions to $\boldsymbol{s}_\epsilon$. For synthesis, we assume that every $\boldsymbol{s} \in \boldsymbol{\mathcal{S}} \setminus (\boldsymbol{\mathcal{S}}_{\mathcal{T}} \cup \boldsymbol{\mathcal{S}}_{\mathcal{D}})$ has a finite occupancy measure, i.e., $\sum_{\boldsymbol{a} \in \boldsymbol{\mathcal{A}}} x(\boldsymbol{s}, \boldsymbol{a}) \leq K$ for some $K \geq 0$.

In the previous sections, we assumed that the joint policy is stationary. The following proposition shows that stationary policies suffice to maximize (1) after the preprocessing step.

Proposition 8.1. *There exists a stationary joint policy that is a solution to* (1).

Given that the stationary policies suffice, we can rewrite (1) as an optimization problem in terms of the occupancy measures $x_{\boldsymbol{s},\boldsymbol{a}}$. The constraints of this optimization problem are as follows. State $\boldsymbol{s}_\epsilon$ has an occupancy measure of zero, i.e. $x_{\boldsymbol{s}_\epsilon,\boldsymbol{a}} = 0$ for all $\boldsymbol{a} \in \boldsymbol{\mathcal{A}} \cup \{\epsilon\}$. The other states have nonnegative occupancy measures, i.e., $x_{\boldsymbol{s},\boldsymbol{a}} \geq 0$ for all $\boldsymbol{s} \in \boldsymbol{\mathcal{S}}, \boldsymbol{a} \in \boldsymbol{\mathcal{A}} \cup \{\epsilon\}$. The occupancy measures satisfy the flow equations $\sum_{\boldsymbol{a} \in \boldsymbol{\mathcal{A}} \cup \{\epsilon\}} x_{\boldsymbol{s},\boldsymbol{a}} = \sum_{\substack{\boldsymbol{y} \in \boldsymbol{\mathcal{S}} \\ \boldsymbol{b} \in \boldsymbol{\mathcal{A}} \cup \{\epsilon\}}} x_{\boldsymbol{y},\boldsymbol{b}} \boldsymbol{\mathcal{T}}(\boldsymbol{y}, \boldsymbol{b}, \boldsymbol{s}) + \mathbb{1}_{\{\boldsymbol{s}_I = \boldsymbol{s}\}}$ for all $\boldsymbol{s} \in \boldsymbol{\mathcal{S}}$. The objective function is

$$\max_{x} \quad \boldsymbol{v}^{full} - \delta l^{full} - \beta \left(\sum_{i=1}^{N} H(\bar{X}^i) - H(\boldsymbol{X}) \right).$$

The reach-avoid probability $\boldsymbol{v}^{full}$ can be expressed as $\boldsymbol{v}^{full} = \sum_{\boldsymbol{s} \in \boldsymbol{\mathcal{S}} \setminus (\boldsymbol{\mathcal{S}}_{\mathcal{D}} \cup \boldsymbol{\mathcal{S}}_{\mathcal{T}})} \sum_{\boldsymbol{a} \in \boldsymbol{\mathcal{A}}} \sum_{\boldsymbol{y} \in \boldsymbol{\mathcal{S}}_{\mathcal{T}}} x_{\boldsymbol{s},\boldsymbol{a}} \boldsymbol{\mathcal{T}}(\boldsymbol{s}, \boldsymbol{a}, \boldsymbol{y})$. The expected path length is the expected time spent in the transient states, i.e., $l^{full} = \sum_{\boldsymbol{s} \in \boldsymbol{\mathcal{S}}} \sum_{\boldsymbol{a} \in \boldsymbol{\mathcal{A}} \cup \{\epsilon\}} x_{\boldsymbol{s},\boldsymbol{a}}$. The entropy $H(\boldsymbol{X})$ [22] of the joint state-action process until reaching state $\boldsymbol{s}_\epsilon$ is

$$\sum_{\substack{\boldsymbol{s} \in \boldsymbol{\mathcal{S}} \\ \boldsymbol{a} \in \boldsymbol{\mathcal{A}}}} x_{\boldsymbol{s},\boldsymbol{a}} \log \left(\frac{\sum_{\boldsymbol{b} \in \boldsymbol{\mathcal{A}}} x_{\boldsymbol{s},\boldsymbol{b}}}{x_{\boldsymbol{s},\boldsymbol{a}}} \right) + \sum_{\substack{\boldsymbol{s} \in \boldsymbol{\mathcal{S}} \\ \boldsymbol{a} \in \boldsymbol{\mathcal{A}}}} x_{\boldsymbol{s},\boldsymbol{a}} \sum_{\boldsymbol{y} \in \boldsymbol{\mathcal{S}}} \boldsymbol{\mathcal{T}}(\boldsymbol{s}, \boldsymbol{a}, \boldsymbol{y}) \log \left(\frac{1}{\boldsymbol{\mathcal{T}}(\boldsymbol{s}, \boldsymbol{a}, \boldsymbol{y})} \right).$$

The entropy $H(\bar{X}^i)$ [22] of the stationary state-action process $\bar{X}^i$ until reaching state $\boldsymbol{s}_\epsilon$ is

$$\sum_{\substack{s^i \in \mathcal{S}^i \\ a^i \in \mathcal{A}^i \cup \{\epsilon^i\}}} x_{s^i,a^i} \log \left(\frac{\sum_{b^i \in \mathcal{A}^i} x_{s^i,b^i}}{x_{s^i,a^i}} \right) + \sum_{\substack{s^i \in \mathcal{S}^i \\ a^i \in \mathcal{A}^i \cup \{\epsilon^i\}}} x_{s^i,a^i} \sum_{y^i \in \mathcal{S}^i \cup \{s_\epsilon^i\}} \mathcal{T}^i(s^i, a^i, y^i) \log \left(\frac{1}{\mathcal{T}^i(s^i, a^i, y^i)} \right).$$

The objective function of the optimization problem consists of convex, concave, and linear functions of the occupancy measures. $\boldsymbol{v}^{full}$ and $-\delta l^{full}$ are linear functions of the occupancy measures. $\beta H(\boldsymbol{X})$ is a concave function of occupancy measures, and $-\beta \sum_{i=1}^{N} H(\bar{X}^i)$ is a convex function of occupancy measures. Furthermore, the problem's constraints are linear. We use the concave-convex procedure [13, 29] to solve for a local optimum.

After solving for the optimal values $x^*_{\boldsymbol{s},\boldsymbol{a}}$ of the occupancy measure variables, we define the minimum-dependency joint policy as $\pi_{MD}(\boldsymbol{s}, \boldsymbol{a}) = x^*(\boldsymbol{s}, \boldsymbol{a}) / \sum_{\boldsymbol{b} \in \boldsymbol{\mathcal{A}}} x^*(\boldsymbol{s}, \boldsymbol{b})$ for all $\boldsymbol{s} \in \boldsymbol{\mathcal{S}} \setminus (\boldsymbol{\mathcal{S}}_{\mathcal{T}} \cup \boldsymbol{\mathcal{S}}_{\mathcal{D}}), \boldsymbol{a} \in \boldsymbol{\mathcal{A}}$ such that $\sum_{\boldsymbol{b} \in \boldsymbol{\mathcal{A}}} x^*(\boldsymbol{s}, \boldsymbol{b}) > 0$, and $\pi_{MD}(\boldsymbol{s}, \boldsymbol{a}) = 1/|\boldsymbol{\mathcal{A}}|$ otherwise [20]. We note that π_{MD} is stationary in the joint state space $\boldsymbol{\mathcal{S}}$.

9 NUMERICAL EXPERIMENTS

We apply the proposed policy synthesis algorithm to the two-agent navigation example illustrated in Figure 2. The setup and objective of this task are as described in §4. In all of the experiments, we compare the results of the minimum-dependency policy π_{MD}, synthesized by the algorithm presented in §8, to a baseline policy π_{base} which does not take communication into account; the baseline policy maximizes the probability that the team will complete its task, while assuming that communication will always be available. For further details surrounding the synthesis of the baseline policy and for an additional three-agent experiment, we refer the reader to [12]. Project code is available at github.com/cyrusneary/multi-agent-comms.

The common environment of the agents, illustrated in Figure 2, is discretized into a grid of cells, each of which corresponds to an individual local state. At any given timestep, each agent takes one of five separate actions: move left, move right, move up, move down, or remain in place. Each agent slips with probability 0.05 every time it takes an action, resulting in the agent moving instead to another one of its valid neighboring states. The resulting optimization problem has 15,625 variables and 16,087 constraints.

In all experiments, the values of the coefficients δ and β in the objective of the policy synthesis problem are set to 0.01 and 0.4 respectively. These values were selected to strike a balance between the optimization objective's three competing terms.

9.1 Fully Imaginary Play

Figure 3 compares the results of the minimum-dependency policy π_{MD} and the baseline policy π_{base} in two scenarios: when communication is either fully available or never available.

We observe from the top figure that the proposed policy synthesis algorithm is effective at reducing the total correlation of the induced

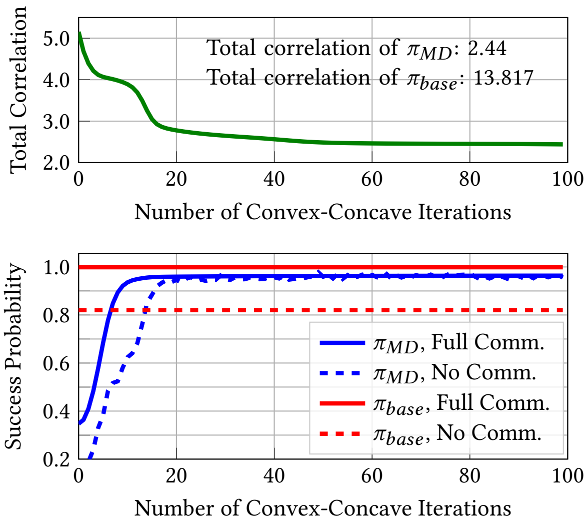

Figure 3: (Top) Total correlation value of the minimum-dependency policy π_{MD} as a function of the number of elapsed iterations of the convex-concave optimization procedure. (Bottom) Probability of task success for π_{MD}. For comparison, we plot the success probability resulting from both imaginary play execution (no communication) and centralized execution (full communication). To estimate the probability of task success, we perform rollouts of the joint policy and compute the empirical rate at which the team accomplishes its objective.

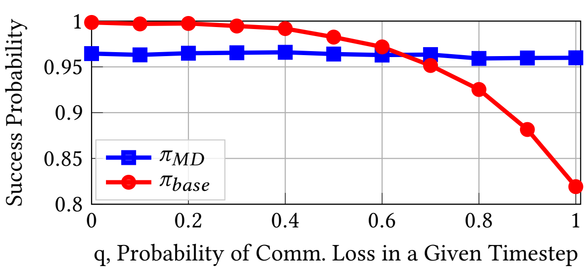

Figure 4: Success probability of intermittent communication for different values of q, which represents the probability of communication unavailability during any given timestep. When $q = 0$ communication is available at every timestep, and when $q = 1$ communication is never available.

stochastic state-action process; the total correlation value of π_{MD} is three orders of magnitude smaller than that of π_{base}.

The bottom figure shows the strong performance of π_{MD} when no communication is available between the agents. In particular, we observe that π_{MD} achieves a probability of task success of 0.97, regardless of whether the agents are able to communicate. That is, by minimizing the total correlation of the policy, π_{MD} ensures the agents may successfully execute the policy without communicating during execution. Conversely, while π_{base} achieves a 0.99 probability of task success when communication is available, this value falls to 0.82 if the agents lose the ability to communicate. This experiment empirically demonstrates the intuition of Theorem 7.5.

In addition to the quantitative results illustrated by Figure 3, we observe an interesting quantitative change in behavior between π_{base} and π_{MD}. In particular, π_{base} results in both of the agents navigating through the lower valley in order to arrive at their targets. This route relies heavily on teammate coordination; the agents must communicate at each timestep in order to safely take turns passing through the valley without colliding. By contrast, π_{MD} results in agent R_1 navigating through the top valley while R_0 takes the bottom valley. Intuitively, by navigating through separate valleys, this team behavior is much less likely to result in collisions even if the agent's don't share their locations with each other. As a result, teammate coordination is much less important to successfully execute the behavior of π_{MD} than it is to execute that of π_{base}.

9.2 Intermittent Communication

While the previous discussion focused on the empirical performance of π_{MD} in the setting where the agents cannot communicate at all, we now examine the setting in which random intermittent communication is available. More specifically, we assume that at each timestep communication fails with probability q, independently of whether or not communication is available during the other timesteps. In this setting, the agents execute the joint policy according to Algorithm 2. That is, if communication is available at a given timestep, all agents collectively share their local states and decide on a joint action. Conversely, when communication is not available, the agents execute the policy using imaginary play.

Figure 4 plots the team's probability of task success when they execute either π_{MD} or π_{base} using Algorithm 2, as a function of the probability of communication failure q. We observe that the probability of task success of the baseline policy π_{base} is very high when $q = 0$, however, it begins to significantly decrease as q increases beyond 0.4. Conversely, the proposed minimum-dependency policy π_{MD} does not suffer such a drop in performance; as q increases and communication becomes more sparse the task success probability of policy π_{MD} remains constant.

10 CONCLUSIONS

In this work, we develop multiagent systems that are robust to communication loss. We provide algorithms for decentralized policy execution when communication is lost and relate the performance of these algorithms to the performance achieved by the same policy under full communication. Using these theoretical results, we propose an optimization algorithm for the synthesis of joint policies that are robust to potential losses in communication. While the policy synthesis algorithm directly operates on the joint state space, future work will aim to use abstractions of the joint state-action space to scale the policy synthesis to larger problems.

ACKNOWLEDGMENTS

This work was supported in part by AFRL FA9550-19-1-0169, ARL ACC-APG-RTP W911NF1920333, and ARO W911NF-20-1-0140.

REFERENCES

[1] Christel Baier and Joost-Pieter Katoen. 2008. *Principles of model checking.* MIT Press.

[2] Raphen Becker, Alan Carlin, Victor Lesser, and Shlomo Zilberstein. 2009. Analyzing myopic approaches for multi-agent communication. *Computational Intelligence* 25, 1 (2009), 31–50.

[3] Raphen Becker, Shlomo Zilberstein, Victor Lesser, and Claudia V Goldman. 2003. Transition-independent decentralized Markov decision processes. In *Proceedings of the 2nd International Conference on Autonomous Agents and Multiagent Systems.* 41–48.

[4] Stefan Boschert and Roland Rosen. 2016. Digital twin—the simulation aspect. In *Mechatronic futures.* Springer, 59–74.

[5] Craig Boutilier. 1996. Planning, learning and coordination in multiagent decision processes. In *Proceedings of the 6th Conference on Theoretical Aspects of Rationality and Knowledge*, Vol. 96. 195–210.

[6] Yongcan Cao, Wenwu Yu, Wei Ren, and Guanrong Chen. 2012. An overview of recent progress in the study of distributed multi-agent coordination. *IEEE Transactions on Industrial Informatics* 9, 1 (2012), 427–438.

[7] Thomas M Cover and Joy A Thomas. 1991. *Elements of Information Theory.* John Wiley & Sons.

[8] Roel Dobbe, David Fridovich-Keil, and Claire Tomlin. 2017. Fully Decentralized Policies for Multi-Agent Systems: An Information Theoretic Approach. In *Proceedings of the 31st Conference on Neural Information Processing Systems.* 2945–2954.

[9] Ben Eysenbach, Russ R Salakhutdinov, and Sergey Levine. 2021. Robust Predictable Control. *Pre-proceedings of the 35th Conference on Neural Information Processing Systems.*

[10] Claudia V Goldman and Shlomo Zilberstein. 2004. Decentralized control of cooperative systems: Categorization and complexity analysis. *Journal of Artificial Intelligence Research* 22 (2004), 143–174.

[11] Carlos Guestrin, Daphne Koller, and Ronald Parr. 2002. Multiagent planning with factored MDPs. In *Proceedings of the 14th Conference on Neural Information Processing Systems.* 1523–1530.

[12] Mustafa O. Karabag, Cyrus Neary, and Ufuk Topcu. 2022. Planning Not to Talk: Multiagent Systems that are Robust to Communication Loss. arXiv:2201.06619 [cs.MA]

[13] Gert Lanckriet and Bharath K Sriperumbudur. 2009. On the convergence of the concave-convex procedure. In *Proceedings of the 21st Conference on Neural Information Processing Systems.* 1759–1767.

[14] Felix Leibfried and Jordi Grau-Moya. 2020. Mutual-information regularization in Markov decision processes and actor-critic learning. In *Proceedings of the 3rd Conference on Robot Learning.* 360–373.

[15] Michael L Littman. 1994. Markov games as a framework for multi-agent reinforcement learning. In *Proceedings of the 11th International Conference on Machine Learning.* 157–163.

[16] Anuj Mahajan, Tabish Rashid, Mikayel Samvelyan, and Shimon Whiteson. 2019. Maven: Multi-agent variational exploration. In *Prooceedings of the 33rd Conference on Neural Information Processing Systems.* 7613–7624.

[17] Francisco S Melo and Manuela Veloso. 2011. Decentralized MDPs with sparse interactions. *Artificial Intelligence* 175, 11 (2011), 1757–1789.

[18] Frans A Oliehoek and Christopher Amato. 2016. *A concise introduction to decentralized POMDPs.* Springer.

[19] Lynne E Parker, Daniela Rus, and Gaurav S Sukhatme. 2016. Multiple mobile robot systems. In *Springer Handbook of Robotics.* Springer, 1335–1384.

[20] Martin L Puterman. 2014. *Markov decision processes: discrete stochastic dynamic programming.* John Wiley & Sons.

[21] Tabish Rashid, Mikayel Samvelyan, Christian Schroeder, Gregory Farquhar, Jakob Foerster, and Shimon Whiteson. 2018. Qmix: Monotonic value function factorisation for deep multi-agent reinforcement learning. In *Proceedings of the 35th International Conference on Machine Learning.* 4295–4304.

[22] Yagiz Savas, Melkior Ornik, Murat Cubuktepe, Mustafa O Karabag, and Ufuk Topcu. 2019. Entropy maximization for Markov decision processes under temporal logic constraints. *IEEE Trans. Automat. Control* 65, 4 (2019), 1552–1567.

[23] Kyunghwan Son, Daewoo Kim, Wan Ju Kang, David Earl Hostallero, and Yung Yi. 2019. Qtran: Learning to factorize with transformation for cooperative multi-agent reinforcement learning. In *Proceedings of the 36th International Conference on Machine Learning.* 5887–5896.

[24] Peter Sunehag, Guy Lever, Audrunas Gruslys, Wojciech Marian Czarnecki, Vinicius Zambaldi, Max Jaderberg, Marc Lanctot, Nicolas Sonnerat, Joel Z Leibo, Karl Tuyls, et al. 2018. Value-decomposition networks for cooperative multi-agent learning based on team reward. In *Proceedings of the 17th International Conference on Autonomous Agents and Multiagent Systems.* 2085–2087.

[25] Takashi Tanaka, Henrik Sandberg, and Mikael Skoglund. 2021. Transfer-entropy-regularized Markov decision processes. *IEEE Trans. Automat. Control* (2021).

[26] Rundong Wang, Xu He, Runsheng Yu, Wei Qiu, Bo An, and Zinovi Rabinovich. 2020. Learning efficient multi-agent communication: An information bottleneck approach. In *Proceedings of the 37th International Conference on Machine Learning.* 9908–9918.

[27] Satosi Watanabe. 1960. Information theoretical analysis of multivariate correlation. *IBM Journal of research and development* 4, 1 (1960), 66–82.

[28] Feng Wu, Shlomo Zilberstein, and Xiaoping Chen. 2011. Online planning for multi-agent systems with bounded communication. *Artificial Intelligence* 175, 2 (2011), 487–511.

[29] Alan L Yuille and Anand Rangarajan. 2002. The concave-convex procedure (CCCP). In *Proceedings of the 14th Conference on Neural Information Processing Systems.* 1033–1040.