Task-Based Information Compression for Multi-Agent Communication Problems with Channel Rate Constraints

Arsham Mostaani, Thang X. Vu, Symeon Chatzinotas, and Björn Ottersten

Abstract—A collaborative task is assigned to a multiagent system (MAS) in which agents are allowed to communicate. The MAS runs over an underlying Markov decision process and its task is to maximize the averaged sum of discounted one-stage rewards. Although knowing the global state of the environment is necessary for the optimal action selection of the MAS, agents are limited to individual observations. The inter-agent communication can tackle the issue of local observability, however, the limited rate of the inter-agent communication prevents the agent from acquiring the precise global state information. To overcome this challenge, agents need to communicate their observations in a compact way such that the MAS compromises the minimum possible sum of rewards. We show that this problem is equivalent to a rate-distortion problem which we call the task-based information compression. We introduce a scheme for task-based information compression titled State aggregation for information compression (SAIC), for which a state aggregation algorithm is analytically designed. The SAIC is shown to be capable of achieving near optimal performance in terms of the achieved sum of discounted rewards. The proposed algorithm is applied to a rendezvous problem and its performance is compared with several benchmarks. Numerical experiments confirm the superiority of the proposed algorithm.

Index Terms—Task-based information compression, machine learning for communication, multiagent systems, reinforcement learning.

I. INTRODUCTION

This paper studies a subclass of distributed empirical risk minimization problem [1]. We investigate the control and communication policy selection of multiple agents who aim at cooperatively minimizing an expected risk. Given a time horizon $M$, a discount factor $\gamma$, discrete alphabets $\Omega_1, \ldots, \Omega_n, M_1, \ldots, M_n, \mathcal{C}$ and a stage reward function $r(\cdot): \prod_{i=1}^n \Omega_i \times \prod_{i=1}^n M_i \rightarrow \mathbb{R}$, each agent $i$ at each time step $t_0$ should minimize the expected risk

$$
\mathbb{E} \left\{ \sum_{t=t_0}^M -\gamma^{t-t_0} r(\mathbf{o}_1(t), \ldots, \mathbf{o}_n(t), \mathbf{m}_1(t), \ldots, \mathbf{m}_n(t)) \right\},
$$

(1)

with respect to $\mathbf{m}_i(t_0)$ whereas only the observation $\mathbf{o}_i(t_0) \in \Omega_i$ and the received communication $\hat{c}(t_0) \in \mathcal{C}$ are known to the agent $i$. Each agent $i \in \{1, \ldots, k\}$ at every iteration $t \in \{1, \ldots, M\}$ solves its own version of this problem in parallel to the other agents. We assume the special case where the transmitted message $c(t)$ - corresponding to the received message $\hat{c}(t)$ - is itself composed of $n$ broadcast messages $c(t) = \langle c_1(t), \ldots, c_n(t) \rangle$, where $c_i(t)$ is generated and transmitted by agent $i$ and encodes the observation $\mathbf{o}_i(t)$ of the agent.

As further discussed in section II, we assume an underlying Markov decision process according to which the observations of all agents are generated. The value of the reward function can thus be controlled or optimized by selecting proper control signals $\mathbf{m}_i(t)$ at each agent. This problem could be a classic multi-agent Markov decision process (MAMDP) [2] if, $\hat{c}(t)$ could include all the information inside all the observations $\mathbf{o}_1(t), \ldots, \mathbf{o}_n(t)$. We assume, however, the communication message $c(t)$ is sent over an imperfect communication channel with limited rate. This introduces an information constraint on the problem. Accordingly, one further question to ask is if we can perform the communication at each agent $i$, with the information constraint in place, such that it leads to the optimal performance of the multiagent system in minimizing the expected risk. Each agent $i$ have to jointly select the tuple $\langle \mathbf{m}_i(t), \hat{c}_i(t) \rangle$ at each time step $t$ with the aim of optimizing the expected risk $\mathbb{E}$. Due to the limited rate of the communication channel between the agents, it is necessary for agents to compactly represent their observations in communication messages. As we ultimately measure the performance of the multiagent system in terms of the expected risk, the loss of information caused by the compact representation of the agents’ observations needs to be managed in such a way that it minimally affects the obtained risk [3], [4]. As such, in this form of compression scheme which we call task-based information compression, the goal of abstraction is different from conventional compression schemes whose ultimate aim is to reduce the distortion between the original and compressed data $D(\mathbf{c}(t), \hat{\mathbf{c}}(t))$ [5].

We show that the task-based information compression problem is a generalized version of the rate-distortion problem, where the compression scheme can be designed to minimize the distortion $D(V(\mathbf{c}(t), \ldots), V(\hat{\mathbf{c}}(t), \ldots))$ between a function of an agent’s observation and a function of the approximated agent’s observation. It is also shown that the function $V(\cdot)$ measures the value of the information to be communicated and allows the clustering of the input space of the encoder to be done based on the value of the input data. In particular instead of clustering input data points based on how similar to each other they are, they will be clustered together based on how similarly valuable they are for the task at hand.

The considered problem is formulated in a general form which can be directly deployed in numerous interesting applications in telecommunications. Suitable applications for are those that include multiple cooperative decision makers that have to communicate through a communication network.
with limited resources. We assume rate-limited but error-free communications - a noisy channel where traditional channel coding is used to ensure error free communications with the achievable rate that is, in fact, the rate constraint of the channel.

A. Literature Review

From one point of view, the given examples fall in the general category of multi-agent communications, where in contrast to many other cooperative multi-agent system’s [6], the full state and action information is not available to each agent. Accordingly, agents are required to carry out communication to overcome these barriers [7]. Earlier works used to addressed the coordination of multiple agents through a noise-free communication channel, where the agents follow an engineered communication strategy [8]–[12]. Later the impact of stochastic delays in multiagent communication was considered on the multiagent coordination [11], while [13] considers event-triggered local communications. Deep reinforcement learning with communication of the gradients of the agents’ objective function was proposed in [13] to learn the communication among multiple agents. In contrast to the above mentioned works, the presence of noise in the inter-agent communication channel was first studied by [14] where exact reinforcement learning was used to design the inter-agent communications. Later, the authors of [4] proposed a deep reinforcement learning approach to address similar problem. Papers [4], [14]–[17] and [18] have contributed to the rapidly emerging literature on task-oriented communications [19]. Noteworthy are also some novel metrics that are introduced in [20] to measure the positive signaling and positive listening amongst agents which learn how to communicate [13], [14], [18].

The current work can also be seen as designing a state aggregation algorithm. State aggregation enables each agent to compactly represent its observations through communication messages while maintaining their performance in the collaborative task. Classical state aggregation algorithms have been used to reduce the complexity of the dynamic programming problems over MDPs [21]–[24] as well as Partially Observable MDPs [25]. One similar work is [26], which studies a task-based quantization problem. In contrast to our work, the assumption there is that the parameter to be quantized is only of, which provides an analytical approach to design task-based compression schemes. In particular, we use state-aggregation to design an information compression scheme to compactly represent the observation process of each agent in a multi-agent system.

Conventionally, the communication system design is disjoint from the distributed decision making design [8]–[11], [13], [28]. This work can also be interpreted as a demonstration of the potential of the joint design of the source coding (compression) and control policies. Herein, we introduce a scheme that can incorporate the features of the control task in the communication design problem, while it guarantees the performance of its solution to bear a close resemblance to that of the jointly optimal policies.

B. Related Works

The aforementioned problem is neither a classic MDP problem as we have the issue of partial observability, and nor is a partially observable MDP (POMDP) [29] as the action vector is not jointly selected at a single entity. The problem of interest is not even a decentralized POMDP (Dec-POMDP) [30] since agents have access to proper means for communications and they also have to optimize their communication strategy given channel limitations.

Nevertheless, similar class of problems - often referred to as task-oriented, goal-oriented or efficient communication approaches, has recently received significant attention from the communication society, see e.g., the extensive survey on similar problems in [19]. Table I positions the current work against some of the recent research that are closely related. To date, there is no work in the literature that we are aware of, which provides an analytical approach to design task-based communications for the coordination of multiple cooperative agents.

C. Contributions

The contributions of this paper are as follows:

- Firstly, we develop a general cooperative multiagent framework in which agents interact over an MDP environment. Unlike the existing works which assume perfect communication links [13], [18], [28], [31], we assume the practical rate-limited communications between the agents. We formulate a two-agents cooperative problem where agents interact over an underlying MDP and can communicate over a channel with limited rate. Our goal is to derive the optimal action selection and communication strategies to maximize the joint objective function.

- Secondly, we propose a new multi-agent state-aggregation algorithm which allows us to separately design the communication and control policies of agents. We will show that the performance of this algorithm in terms of the expected system return is near to the optimal control policy when agents can maintain perfect communications.

- Thirdly, We quantify the loss that is incurred on the expected return of the system due to the rate constraints
that are introduced on the inter-agent communication channels. In particular, our theoretical results show that, if our proposed method, SAIC, is applied the expected return of the multi-agent communication system with rate constraints in place can stay in a close proximity of the optimal expected return that is obtained under perfect communications.

- Finally, extensive numerical comparisons are made between the performance of the SAIC and several other benchmark schemes in terms of the optimality of the expected risk/return, for the rendezvous problem. One of these benchmark schemes, titled conventional communication, is the result of a theorem developed by [7], adapted to our problem. The second benchmark is the control of the multiagent system by a centralized controller which is assumed to have perfect communications with the agents and executes the Q-learning. When communications are carried out without rate constraint, these reference schemes provides an upper-bound to the numerical problem. It is shown, however, that when rate constraints are in place, SAIC is of significant advantages over CIC as well as other benchmarks.

Organization: Section II describes the system model for a cooperative multi-agent task with rate constrained inter-agent communications. Section III Proposes a scheme for the joint design of communication and control policies that takes the value of information into account to perform information compression. We also provide analytical results on how distant the result of this algorithm can be from the optimal centralized solution. The numerical results and discussions are provided in Section V. Finally, Section VI concludes the paper.

II. SYSTEM MODEL

For the reader’s convenience, a summary of the notation that we follow in this paper is given in Table I. Bold font is used for matrices or scalars which are random and their realizations follows simple font. We consider a two-agent system, where at any time step \( t \) each agent \( i \in \{1, 2\} \) makes a local observation \( o_i(t) \in \Omega \) on environment while the true state of environment is \( s(t) \in \mathcal{S} \). The alphabets \( \Omega \) and \( \mathcal{S} \) define observation space and state space, respectively. The true state of environment \( s(t) \) is controlled by the joint actions \( m_i(t), m_j(t) \in \mathcal{M} \) of the agents, where each agent \( i \) can only choose its local action \( m_i(t) \) which is selected from the local action space \( \mathcal{M} \).

The environment runs on discrete time steps \( t = 1, 2, ..., M \), where at each time step, each agent \( i \) selects its domain level action \( m_i(t) \) upon having an observation \( o_i(t) \) of environment. Dynamics of the environment are governed by a conditional probability mass function

\[
T(s(t+1)|s(t), m_i(t), m_j(t)) = \frac{p(s(t+1) = s(t+1)|s(t) = s(t), m_i(t) = m_i(t), m_j(t) = m_j(t))},
\]

which is unknown to the agents. \( T(\cdot) \) defines the transition probability of the environment to state \( s(t+1) \) given the current state of environment \( s(t) \) and the joint actions \( m_i(t), m_j(t) \). We recall that domain level actions \( m_i(t) \) can, for instance, be in the form of a movement or acceleration in a particular direction or any other type of action depending on the domain of the cooperative task. We consider a particular structure for agents’ observations, referred to as collective observations in the literature [7]. Under collective observability, individual observation of an agent provides it with partial information about the current state of the environment, however, having knowledge of the collective observations acquired by all of the agents is sufficient to realize the true state of environment. To further elaborate, at all time steps \( t \) agents’ observation processes \( o_i(t), o_j(t) \) follow eq. (5) and eq. (4). Note that even in the case of collective observability, for agent \( i \) to be able to observe the true state of environment at all times, it

| Paper | Information source | Joint coms and control | Distributed | Source/Channel coding | Implicit/Explicit cons | Analytical/Data-driven |
|-------|------------------|-----------------------|-------------|-----------------------|----------------------|-----------------------|
| [16], [17] | × | × | × | Quantization | N/A | Data-driven |
| [5] | ✓ (Linear) | ✓ | ✓ | Quantization | Explicit | Analytical |
| [13], [20] | N/A | ✓ | ✓ | Quantization | Explicit | Data-driven |
| [4], [14] | ✓ (Markov) | ✓ | ✓ | Channel Coding | Explicit | Data-driven |
| Our work | ✓ (Markov) | ✓ | ✓ | Quantization | Explicit | Analytical |

Table II

| Symbol | Meaning |
|--------|---------|
| \( x(t) \) | A generic random variable generated at time \( t \) |
| \( \mathcal{X} \) | Realization of \( x(t) \) |
| \(|\mathcal{X}|\) | Cardinality of \( \mathcal{X} \) |
| \( p_x(x(t)) \) | Shorthand for \( \text{Pr}(x(t) = x(t)) \) |
| \( H(x(t)) \) | Information entropy of \( x(t) \) (bits) |
| \( I(x(t); y(t)) \) | Mutual information of \( x(t) \) and \( y(t) \) |
| \( E_{p(x)}[x] \) | Expectation of the random variable \( X \) over the probability distribution \( p(x) \) |
| \( \text{tr}(t) \) | Realization of the system’s trajectory at time \( t \) |
| \( \delta(\cdot) \) | Dirac delta function |
| \( \lceil \cdot \rceil \) | Ceiling function |

Table I

COMPARISON BETWEEN OUR WORK AND THE RELATED PRIOR ART

| Symbol | Meaning |
|--------|---------|
| \( X \) | Algebra of \( x(t) \) |
| \( T \) | Alphabet of \( x(t) \) |
| \(|\mathcal{X}|\) | Cardinality of \( \mathcal{X} \) |
| \( p_x(x(t)) \) | Shorthand for \( \text{Pr}(x(t) = x(t)) \) |
| \( H(x(t)) \) | Information entropy of \( x(t) \) (bits) |
| \( I(x(t); y(t)) \) | Mutual information of \( x(t) \) and \( y(t) \) |
| \( E_{p(x)}[x] \) | Expectation of the random variable \( X \) over the probability distribution \( p(x) \) |
| \( \text{tr}(t) \) | Realization of the system’s trajectory at time \( t \) |
| \( \delta(\cdot) \) | Dirac delta function |
| \( \lceil \cdot \rceil \) | Ceiling function |
needs to have access to the observations of the other agent $j \neq i$ through communications at all times.

$$H(s(t)) > H(o_i(t)) = I(o_i(t); s(t)) > 0, \quad i \in \{1, 2\}, \quad (3)$$

$$H(o_i(t), o_j(t)) = H(s(t)), \quad j \neq i \quad (4)$$

$$I(o_i(t), s(t)) + I(o_j(t), s(t)) = H(s(t)) \quad (5)$$

A deterministic reward function $r(\cdot) : S \times M^2 \to \mathbb{R}$ indicates the reward of both agents at time step $t$, where the arguments of the reward function are the global state of the environment $s(t)$ and the domain-level actions $m_i(t), m_j(t)$ of both agents. We assume that the underlying environment over which agents interact can be defined in terms of an MDP determined by the tuple $\{S, M^2, r(\cdot), \gamma, T(\cdot)\}$, where $S$ and $M$ are discrete alphabets, $r(\cdot)$ is a function, $T(\cdot)$ is defined in (2) and the scalar $\gamma \in [0, 1]$ is the discount factor. The focus of this paper is on scenarios in which the agents are unaware of the state transition probability function $T(\cdot)$ and of the closed form of the function $r(\cdot)$. However we assume that, further to the literature of reinforcement learning, a realization of the function $r(s(t), m_i(t), m_j(t))$ will be accessible for both agents $i$ and $j$ at each time step $t$. Since the tuple $\{S, M^2, r(\cdot), \gamma, T(\cdot)\}$ is an MDP and the state process $s(t)$ is jointly observable by the agents, the system model of this cooperative multiagent setting is also referred to as a multiagent MDP (MAMDP or MMMDP) in the literature of multiagent decision making [2, 34, 35].

In what follows two problems regarding the above mentioned setup is detailed. The main intention of this paper is to address the decentralized control and inter-agent communications for a system of multiple agents. However, the problem of centralized control of the system is formalized in subsection II-A as we keep comparing the analytical and numerical results obtained by decentralized algorithms with the centralized algorithm. Moreover, the simpler nature and mathematical notations used for the centralized problem, allow the reader to have a smoother transition to the decentralized problem which is of more complex nature and notation.

### A. Centralized Control

We consider a scenario in which a central controller has instant access to the observations $o_i(t), o_j(t)$ of both agents through a free (with no cost on the objective function) and reliable communication channel. Following the Eq. (4) the joint observations $o_i(t), o_j(t)$ are representative of the true state of environment $s(t)$ at time $t$. From the central controller’s point of view, the environment is an MDP introduced by the tuples $\{S, M^2, r(\cdot), \gamma, T(\cdot)\}$. The goal of the centralized controller is to maximize the expected sum of discounted rewards. The expectation is computed over the joint PMF of the whole system trajectory $s(1), m_i(1), m_j(1), ..., s(M), m_i(M), m_j(M)$ from time $t = 1$ to $t = M$, where this joint probability mass function (PMF) is generated if agents follow policy $\pi(\cdot)$, eq. (7), for their action selections at all times and $s(1) \in S$ is randomly selected by the environment. To have a more compact notation to refer to the system trajectory, hereafter, we represent the realization of a system trajectory at time $t$ by $tr(t)$ which corresponds to the tuple $\langle o_i(t), o_j(t), m_i(t), m_j(t) \rangle$ and the realization of the whole system trajectory by $\{tr(t)\}_{t=1}^M$. Accordingly, the problem boils down to a single agent problem which can be denoted by

$$\max_{\pi(\cdot)} \mathbb{E}_{p^*_{\pi}} \left\{ \sum_{t=1}^M \gamma^{t-1} r(o_i(t), o_j(t), m_i(t), m_j(t)) \right\} \quad (6)$$

where the policy $\pi(\cdot)$ can be expressed by

$$\pi(m_i(t), m_j(t)|s(t)) = p(m_i(t) = m_i(t), m_j(t) = m_j(t)|s(t) = s(t)) \quad (7)$$

and $p_{\pi_i}(s(t+1)|s(t))$ is the probability of transitioning from $s(t)$ to $s(t+1)$ when the joint action policy $\pi(\cdot)$ is executed by the central controller. Similarly, $p_{\pi}(\{tr(t)\}_{t=1}^M)$ is the joint PMF of $tr(1), tr(2), ..., tr(M)$ when the joint action policy $\pi(\cdot)$ is followed by the central controller.

On one hand, problem (6) can be solved using single-agent Q-learning [33] and the solution $\pi^*(\cdot)$ obtained by Q-learning is guaranteed to be the optimal control policy, given some non-restricting conditions [36]. On the other hand, the use-cases of the centralized approach are limited to the applications in which there is a permanent communication link with unlimited rate between the agents and the controller. Whereas these conditions are not met in many remote applications, where there is no communication infrastructure to connect the agents to the central controller.

Given enough training time, and channels with sufficient rate of communication between the agents and the central controller, the centralized algorithm provides us with a performance upper bound in maximizing the objective function (6). The aim of this paper is, however, to introduce decentralized approaches which are run over rate-limited communication channels with comparable performance.

### B. Problem Statement

Here we consider a scenario in which the same objective function explained in Eq. (6) needs to be maximized by the two-agent system in a decentralized fashion, Fig. 1. Namely, agents with partial observability can only select their own actions. To prevail over the limitations imposed by the local observability, agents are allowed to have direct (explicit) communications, and not indirect (implicit) communications [32, 37]. However, the communication is done through a channel with limited rate

$$R = I(c_i(t), \tilde{c}_i(t)) \quad (8)$$

where $c_i(t) \in C = \{-1, 1\}^B$ stands for the communication message generated by agent $i$ before being encoded by error correction codes and $\tilde{c}_i(t) \in \{-1, 1\}^B$ corresponds to the same communication message after being decoded by the channel decoding block, of the agent $j$’s receiver. Note that instead of maximal achievable rate of the channel, $R$ (in bits) represents the channel maximal (non-)asymptotic achievable rate corresponding to the used channel coding (of arbitrary
code length). It should be noted that the design of the channel coding and modulation schemes are beyond the scope of this paper and the main focus is on the compression of agents’ generated communication messages. In this problem, the limited rate of the channel is accepted as a constraint which is imposed by the given channel coding, length of codewords, modulation scheme and the available bandwidth. Here we assume, the achievable rate of information exchange for both inter-agent communication channels to be equal to \( R \), i.e., the communication resources are split evenly amongst the two agents. In particular we consider \( R \) to be time-invariant and to follow:

\[
R < \min \{ H(o_i(t)), H(o_j(t)) \}. \tag{9}
\]

Now let the function \( g(t') \) denote the system’s return - negative of what was defined as risk earlier:

\[
g(t') = \sum_{t=1}^{M} \mathcal{E}_t^{t-1} r(o_i(t), o_j(t), m_i(t), m_j(t)). \tag{10}
\]

Note that \( g(t') \) is random variable and a function of \( t' \) as well as the trajectory \( \{ t(r) \}_{t=1}^{M} \). Due to the lack of space, here we drop a part of arguments of this function. Accordingly, the decentralized problem is formalized as

\[
\max_{\pi_i^n, \pi_j^n} \mathbb{E}_{P_{r^n}} \{ g(1) \}, \quad i \in \{1, 2\}, \ i \neq j
\]

s.t.

\[
I(c_j(t); \tilde{o}_j(t)) \leq R, \tag{11}
\]

where in its general form, the domain level action policy \( \pi_i^n : \mathcal{M} \times \mathcal{C} \times \mathcal{O} \rightarrow \{0, 1\} \) of each agent \( i \) is defined as

\[
\pi_i^n(m_i(t)|o_i(t), \tilde{c}_i(t)) = \Pr\{ m_i(t) = m(t)|o_i(t), \tilde{c}_i(t) = \tilde{c}_i(t) \}, \tag{12}
\]

and the communication policy \( \pi_j^n : \mathcal{C}^2 \times \mathcal{C} \rightarrow \{0, 1\} \) of each agent \( i \) follows:

\[
\pi_i^n(c_i(t)|o_i(t), \tilde{c}_i(t)) = \Pr\{ c_i(t) = c_i(t)|o_i(t), \tilde{c}_i(t) = \tilde{c}_i(t) \}. \tag{13}
\]

Similar to the centralized problem, \( p_{r^n} = \mathbb{E}_{P_{r^n}} \{ \{ t(r) \}_{t=1}^{M} \} \) is the joint probability mass function of \( t(1), t(2), \ldots, t(M) \) when each agent \( i \in \{1, 2\} \) follows the action policy \( \pi_i^n(\cdot) \) and the communication policy \( \pi_j^n(\cdot) \) and \( s(1) \in \mathcal{O} \) is randomly selected by the environment. To make the problem more concrete, further to \( 12 \) and \( 13 \), here we assume the presence of an instantaneous communication between agents, which is in contrast to delayed communication models \[14]. \[38\] Fig. 2 demonstrates this communication model. As such, each agent \( i \) at any time step \( t \) prior to the selection of its action \( m_i(t) \) receives a communication message \( \tilde{c}_i(t) \) that includes some information about the observations of agent \( j \) at time \( t \).

III. STATE AGGREGATION FOR INFORMATION COMPRESSION IN MULTIAGENT COORDINATION TASKS

This section tackles the constraint on the rate of inter-agent information exchange in the problem \( 11 \) by introducing state aggregation for compression of agents observations. State aggregation in this paper is applied as a method to carry out a task-based information compression. We design the state aggregation algorithm such that it can suppress part of observation information that results in the smallest possible loss in the performance of the multi-agent system, where this loss is measured in terms of regret from maximum achievable expected cumulative rewards. As mentioned before, we intend to perform compression in a manner that the entropy of compressed communication messages fits the rate of the inter-agent communication channel \( R \). Similar to some other recent papers with focus on multi-agent coordination \[13]. \[59\], here a centralized training phase for the two-agent system is required, however, the execution can still be done in a decentralized fashion.

Although in a more general approach the selection of communication actions \( c_i(t) \) could be conditioned on both \( o_i(t) \) and \( \tilde{c}_i(t) \), here we focus on just communication policies of type \( \pi_i^n(c_i(\cdot)|o_i(\cdot)) \), where communication actions of each agent at each time are selected only based on its observation at that time, further to the explanations given in subsection \[14\]. Here we assume that the communication resources are split evenly amongst the two agents, by considering the achievable rate of information exchange of both communication channels to be equal to \( R \). As such, both agents compress their observations to acquire communication messages of equal entropy. We also assume observations of both agents to have equal entropy \( H(o_i(t)) = H(o_j(t)) \).

Definition 1 (Task-based information compression (TBIC) problem).

Let the higher order function \( \Pi_i^n \) be a map from the vector space \( \mathcal{K}_C \) of all possible joint communication policies \( \{ \pi_i^n(\cdot), \pi_j^n(\cdot) \} \) to the vector space \( \mathcal{K}_M \) of optimal corresponding joint control policies \( \{ \pi_i^*(\cdot), \pi_j^*(\cdot) \} \). Upon the availability of \( \Pi_i^n \), by plugging it into the problem \( 11 \), we will have a new problem

\[
\max_{\pi_i^n} \mathbb{E}_{P_{r^n}} \{ g(1) \}, \quad i \in \{1, 2\}, \ i \neq j
\]

s.t.

\[
I(c_j(t); \tilde{o}_j(t)) \leq R, \tag{14}
\]
where we maximize the system’s return with respect to only communication policies $\langle \pi^1_\m, \pi^2_\m \rangle$. The problem is called here as TBCD problem.

In TBCIC problem, there is no need to find the joint communication and control policies since the optimal control policies $\langle \pi^m_\m, \pi^m_\o \rangle$ are automatically computed by the mapping $\Pi^m_\m (\pi^m_\o, \pi^m_\c)$. This allows us to separately solve the communication problem without any compromise on the optimality of the system’s expected return. SAIC follows the same approach to separately design the communication policy with the caveat that we introduce an approximated version of the TBCIC problem, and afterwards we find the communication policies $\langle \pi^m_\m, \pi^m_\c \rangle$ that optimize the approximated TBCIC problem. We will then show how close the optimal solution to the approximated problem can be to the original TBCIC problem where no communication constraint is in place.

Lemma 2. The objective function of the decentralized problem (11), which is a function of system trajectory $\{tr(t)\}_{t=1}^M$, can be expressed as

$$
\begin{align*}
&\mathbb{E}_{p_{\pi^m_\m, \pi^m_\c} (\{tr(t)\}_{t=1}^M)} \{g(1)\} = \\
&\mathbb{E}_{p_{\pi^m_\m, \pi^m_\c} (\{o_i(1), e_i(1)\})} \left\{ \mathbb{E}_{p_{\pi^m_\o, \pi^m_\c} (\{tr(t)\}_{t=1}^M)} g(1|o_i(1), e_i(1)) \right\}
&\mathbb{E}_{p_{\pi^m_\o, \pi^m_\c} (o_i(1), e_i(1))} \{V_{\pi^m_\o, \pi^m_\c} (o_i(1), e_i(1))\}
&\mathbb{E}_{p_{\pi^m_\o, \pi^m_\c} (o_i(1), e_i(1))} \{V_{\pi^m_\o, \pi^m_\c} (o_i(1), e_i(1))\},
\end{align*}
$$
(15)

where $V_{\pi^m_\o, \pi^m_\c} (o_i(t), e_i(t))$ is the unique solution to the Bellman equation corresponding to the joint action and communication policies $\pi^m_\o, \pi^m_\c$ of both agents.

for all $o_i(t) \in \Omega, e_i(t) \in C$. In light of eq. (15), the problem (11), when solved with respect to only the control policies $\langle \pi^m_\m, \pi^m_\c \rangle$ can be expressed as

$$
\max_{\pi^m_\o, \pi^m_\c} \mathbb{E}_{p_{\pi^m_\m, \pi^m_\c} (o_i(1), e_i(1))} \left\{ V_{\pi^m_\o, \pi^m_\c} (o_i(1), e_i(1)) \right\}, \quad \text{subject to} \quad I (e_i(t); e_j(t)) \leq R.
$$
(16)

Lemma 3, also let us obtain the solution of (16) by finding the optimal value function $V_{\pi^m_\o, \pi^m_\c} (o_i(1), e_i(1))$ obtained by Q-learning. It is important to note that the value function $V_{\pi^m_\o, \pi^m_\c} (o_i(1), e_i(1))$ obtained by Q-learning will be optimal only if the environment, explained by the tuple $\{\Omega \times C, \mathcal{M}, T, r(\cdot), \gamma\}$, is an MDP. Accordingly, we assume that the aggregated MDP denoted by $\{\Omega \times C, \mathcal{M}, T, \gamma\}$, which is obtained by doing state aggregation on the original MDP denoted by $\{\Omega, \mathcal{M}, T, \gamma\}$, is an MDP itself.

Lemma 3. The maximum of expectation of value function $V_{\pi^m_\o, \pi^m_\c} (o_i(t), e_i(t))$, over the joint distribution of $o_i(t), e_i(t)$ is equal to the expectation of value of optimal policy

$$
\max_{\pi^m_\o, \pi^m_\c} \mathbb{E}_{p_{\pi^m_\m, \pi^m_\c} (o_i(1), e_i(1))} \left\{ V_{\pi^m_\o, \pi^m_\c} (o_i(1), e_i(1)) \right\}
= \mathbb{E}_{p_{\pi^m_\o, \pi^m_\c} (o_i(1), e_i(1))} \left\{ V_{\pi^m_\o, \pi^m_\c} (o_i(1), e_i(1)) \right\}
$$
(17)

Remember that numerical methods such as value iteration or Q-learning, cannot normally provide parametric solutions which is in contrast to our requirements in SAIC, as explained earlier in this section. Lemma 2 allows us to acquire a parametric approximation of $V_{\pi^m_\o, \pi^m_\c} (o_i(1), e_i(1))$ by leveraging the value function $V^* (o_i(t), o_j(t))$ corresponding to the optimal solution of the centralized problem $\{6\}$. Note that policy $\pi^m_\o (\cdot)$ as described in Section $\{10\}$ is the optimal solution to the centralized problem $\{6\}$ which can be obtained using Q-learning. Accordingly, following lemma 2, we propose to derive an off-policy approximation of $V_{\pi^m_\o, \pi^m_\c} (o_i(t), e_i(t))$ following

$$
V^* (o_i(t), e_i(t)) = \sum_{o_j(t) \in \mathcal{O}_k} V^* (o_i(t), o_j(t)) p(o_j(t)|e_i(t)).
$$
(18)

Based on the results of lemma 1 and lemma 2, theorem 3 is constructed such that it allows us to compute the communication policies of agents independent from their action policies. The proposed communication policy by theorem 3, is the optimal communication policy for an approximated version of the problem $\{11\}$.

Theorem 5. The communication policy $\pi^*_{\o, \c} (\cdot) \in \Pi_{\o, \c}$ obtained by solving the k-median clustering problem

$$
\min_{\pi^*_{\o, \c} (\cdot)} \sum_{k=1}^{2^R} \sum_{o_i(t) \in \mathcal{O}_k} \left| V^* (o_i(t)) - \mu_k \right|
$$
(19)

is the optimal solution to an approximated version of the problem $\{11\}$. Note that $\pi^*_{\o, \c} (\cdot)$ corresponds to a unique $\pi^*_{\o, \c} (\cdot)$.

Proof. Further to eq. (15), problem (11) can be expressed by

$$
\max_{\pi^m_\o, \pi^m_\c} \mathbb{E}_{p_{\pi^m_\m, \pi^m_\c} (o_i(1), e_i(1))} \left\{ V_{\pi^m_\o, \pi^m_\c} (o_i(1), e_i(1)) \right\},
$$
(20)

subject to $I (e_i(t); e_j(t)) \leq R$, for $i \in \{1, 2\}, i \neq j$. We can now plug $\pi^m_\o (\cdot)$, which is the result of solving problem (16), into problem (20) to obtain the following problem

$$
\max_{\pi^m_\o} \mathbb{E}_{p_{\pi^m_\m, \pi^m_\c} (o_i(1), e_i(1))} \left\{ V_{\pi^m_\o, \pi^m_\c} (o_i(1), e_i(1)) \right\},
$$
(21)

subject to $I (e_i(t); e_j(t)) \leq R$, for $i \in \{1, 2\}, i \neq j$. By substituting $V_{\pi^m_\o, \pi^m_\c} (o_i(1), e_i(1))$ with its approximator $V^* (o_i(t), e_i(t))$, we will have

$$
\max_{\pi^m_\o (\cdot)} \mathbb{E}_{p_{\pi^m_\m, \pi^m_\c} (o_i(1), e_i(1))} \left\{ V^* (o_i(1), e_i(1)) \right\},
$$
(22)

subject to $I (e_i(t); e_j(t)) \leq R$. Note that the optimizers of the problem (22) and (23) are identical due to the fact that the additional term $\mathbb{E} \{ V^* (o_i(t), o_j(t)) \}$ is independent from the communication policy $\pi^m_\c (\cdot)$. Furthermore, the problem (23) is now expressed as a form of rate distortion problem with mean absolute difference of the value functions $V^* (o_i(t), o_j(t))$ and $V^* (o_i(t), e_i(t))$ as the measure of distortion. This interpretation of problem (23) can be understood later by seeing the eq. (24).

$$
\min_{\pi^m_\o, \pi^m_\c} \mathbb{E}_{p_{\pi^m_\m, \pi^m_\c} (o_i(1), e_i(1))} \left\{ V^* (o_i(1), o_j(1)) - V^* (o_i(1), e_i(1)) \right\}
$$
(23)

The expectation $\mathbb{E}_{p_{\pi^m_\m, \pi^m_\c} (o_i, e_i)} \left\{ V^* (o_i(1), o_j(1)) - V^* (o_i(1), e_i(1)) \right\}$ can be estimated by computing it over the empirical distribution of $o_i(1)$, $o_j(1)$. Note that the empirical joint distribution of $o_i(1)$, $e_i(1)$ can be obtained by following the communication policy $\pi^m_\c (\cdot)$ on the empirical distribution of $o_i(1), o_2(1)$. Therefore, the problem
can be rewritten as

$$\min_{\pi(t)} \sum_{o(t) \in \Omega} \sum_{o(t) \in \Omega} \left| V^*(o(t), o_j(t)) - V^*(o(t), \tilde{c}_i(t)) \right|$$

s.t. \( I(o(t); \tilde{c}_i(t)) \leq R. \quad (24) \)

Selection of the optimal \( \pi(t) \) here, means to extract all the information of \( o(t) \) which is useful in the task and to include them in \( \tilde{c}_i(t) \). This task-based information compression problem can be interpreted as a quantization problem in which distortion is defined in terms of value functions as seen in \( (24) \), where a certain number of quantization levels, \( 2^n \), is allowed. Note that the term \( V^*(o(t), o_j(t)) - V^*(o(t), \tilde{c}_i(t)) \), being always non-negative, is equal to its absolute value \( V^*(o(t), o_j(t)) - V^*(o(t), \tilde{c}_i(t)) \). To gain more insight about the meaning of this task-based information compression, we have also detailed a conventional quantization problem which is adapted to our problem setting in eq. \( (25) \), where \( \tilde{c}_i \sim \pi(t) \{ \tilde{c}_i(1) | o(t) \} \). In fact, the compression scheme applied in the CIC, explained later on in subsection \( \text{V-B} \), is obtained by solving a similar problem, with the caveat that distance in case of CIC is computed in terms of squared euclidean distance.

$$\min_{\pi(t)} \sum_{o(t) \in \Omega} \left| o(t) - \tilde{c}_i(t) \right|, \quad \text{s.t.} I(o(t); \tilde{c}_i(t)) \leq R. \quad (25)$$

Quantization levels are disjoint sets \( P_{i,k} \in \Omega \), where their union \( \cup_{k=1}^{2^R} P_{i,k} \) will cover the entire \( \Omega \). Each quantization level is represented by only one communication message \( \tilde{c}_i(t) = c_k \in C \). Further to lemma 2, the value of \( V^*(o(t), \tilde{c}_i(t)) \) can be computed by empirical mean \( \mu_k = \frac{1}{|P_{i,k}|} \sum_{o(t) \in P_{i,k}} V^*(o(t), \tilde{c}_i(t)) \).

The quantization problem \( (24) \) becomes a k-median clustering problem as illustrated by \( (26) \).

$$\min_{P_i} \sum_{o(t) \in \Omega} \sum_{k=1}^{2^R} \sum_{o(t) \in P_{i,k}} \left| V^*(o(t), o_j(t)) - \mu_k \right|.$$ 

where \( P_i = \{ P_{i,1}, ..., P_{i,2^R} \} \) is a partition of \( \Omega \).

By taking the mean of \( V^*(o(t), o_j(t)) \) over the empirical distribution of \( o(t) \) we can also marginalize out \( o_j(t) \). Again, it does not change the solution of the problem and we will have

$$\min_{P_i} \sum_{k=1}^{2^R} \sum_{o(t) \in \Omega} \left| V^*(o(t)) - \mu_k \right|,$$

in which \( \mu_k = \frac{1}{|o(t) \in \Omega|} \mu_k \) will approximate \( V^*(o(t)) \).

Theorem \( 5 \) allows us to find a communication/compression policy \( \pi^c \) by clustering the input space \( \Omega \) of the communication policy according to the values \( V^*(o(t)) \) of the input points. This policy was shown to be the optimal communication policy for the approximated task-based communication policy design problem. As mentioned in Theorem \( 5 \), one way to obtain \( V^*(o(t)) \) is to solve the centralized problem \( (26) \) by Q-learning. By solving this problem \( Q^*(o(t), o_j(t), m_i(t), m_j(t)) \) can be obtained. Accordingly, following Bellman optimality equation, we can compute \( V^*(o(t)) \) by

$$\max_m Q^*(o(t), m_i(t)) = V^*(o(t)),$$

where \( V^*(o(t)) \) can be expressed as

$$V^*(o(t)) = \mathbb{E}_{p_{o(t)}}\left[ g(1) | o(t) = o_i(1) \right] \quad (29)$$

and further to the law of iterated expectations, it can also be written as

$$V^*(o(t)) = \mathbb{E}_{p_{o(t)}}\left[ \mathbb{E}_{p_{o(t)}}\left( g(1) | o(t) = o_i(1) \right) \right].$$

Based on \( (29) \), \( V^*(o(t)) \) can be computed both analytically (if transition probabilities of environment are available) and numerically. A remarkable feature of computing \( V^*(o(t)) \) using this method is its independence from the used inter-agent communication algorithm.

On the availability of the exact \( \pi^c \), \( V^*(o(t)) \) can be computed both analytically (if transition probabilities of environment are available) and numerically. A remarkable feature of computing \( V^*(o(t)) \) using this method is its independence from the used inter-agent communication algorithm. By following this scheme, detailed in Algorithm 3, we first compute the value \( V^*(o) \) for all \( o \in \Omega \). Afterwards, by solving the k-median clustering problem \( (27) \), an observation aggregation scheme indicated by \( P_i \) is computed. By following this aggregation scheme, the observations \( o(t) \in \Omega \) will be aggregate such that the performance of the multi-agent system in terms of the the objective function it attains is optimized.

Remark 2: The optimal policy \( \pi^c \) is achievable within the centralized training phase. According to the conditions mentioned in eq. (4)-(5), the environment is fully observable for the central controller while the central controller posses the ability to jointly select the actions for all agents. The problem will thus reduce to a single agent Q-learning applied on an MDP with asymptotic convergence to the optimal policy \( \pi^c \).

Remark 3: During the decentralized training phase, each agent, instead of viewing the environment as the original underlying MDP denoted by \( \{ \Omega, \mathcal{M}, r, \gamma, T^* \} \), views an aggregated form of the original MDP denoted by \( \{ \Omega \times C, \mathcal{M}, r, \gamma, T^* \} \). The aggregated MDP will be an MDP itself, if the original MDP is lumpable with respect to the equivalence relation defined by \( P \).
Algorithm 1 State Aggregation for Information Compression (SAIC)

1: Input: $\gamma$, $\alpha$, $c$
2: Initialize all-zero table $N_i^n(o_i(t), \hat{c}_i(t), m_i(t))$, for $i = 1, 2$
3: and Q-table $Q_i^n(\cdot) \leftarrow Q_i^n(h_{i-1}(\cdot))$, for $i = 1, 2$
4: and all-zero Q-table $Q(o_i(t), o_j(t), m_i(t), m_j(t))$
5: Obtain $\pi_i^*$ and $Q_i^*$ by solving (33) using Q-learning [33]
6: Compute $V^*(o_i(t))$ following eq. (27) for $\forall o_i(t) \in \Omega$
7: Solve problem (27) by applying k-median clustering to obtain $\pi_i^*(\cdot)$, for $i = 1, 2$
8: for each episode $k = 1 : K$ do
9: Randomly initialize local observation $o_i(t = 1)$, for $i = 1, 2$
10: for $t_k = 1 : M$ do
11: Select $c_i(t)$ following $\pi_i^*\cdot()$, for $i = 1, 2$
12: Obtain message $\xi_i(t)$, for $i = 1, 2$, $j \neq i$
13: Update $Q_i^n(o_i(t - 1), \xi_i(t - 1), m_i(t - 1))$, for $i = 1, 2$
14: Select $m_i(t) \in M$ following UCB, for $i = 1, 2$
15: Increment $N_i^n(o_i(t), \xi_i(t), m_i(t))$, for $i = 1, 2$
16: Obtain reward $r(o_i(t), \xi_i(t), m_i(t), m_j(t))$, for $i = 1, 2$
17: Make a local observation $o_i(t)$, for $i = 1, 2$
18: $t_k = t_k + 1$
19: end
20: Compute $\sum_{t=1}^M \gamma^t r_i$ for the $t$th episode
21: end
22: Output: $Q_i^n(\cdot)$
23: and $\pi_i^n(m_i(t)|o_i(t), \xi_i(t))$ by following greedy policy for $i = 1, 2$

The resulting multiagent environment will be, according to the definition, a multiagent MDP (MMDP) [35].

Remark 5: Within the distributed training phase, distributed Q-learning is applied to a deterministic MMDP [2] which leads to an asymptotically optimal control policy [2], if the underlying MDP is deterministic.

Note that the control policy $\pi_{i,SAIC}^\pi(\cdot)$ that is obtained within the distributed training phase of SAIC is optimal for the given communication policy $\pi_{c,SAIC}(\cdot)$, that was obtained within the centralized training phase. Therefore, $\pi_{i,SAIC}^\pi(\cdot)$ is not necessarily an optimal solution to the problem (11). In theorem 6, however, we set an upper-bound on the possible loss on the expected return of the system due to the selection of $\pi_{i,SAIC}^\pi(\cdot)$ and $\pi_{c,SAIC}^\pi(\cdot)$ using to SAIC. Instead of comparing the communication and control policies $\pi_i^{c,SAIC}(), \pi_i^{c,SAIC}(), \pi_i^{SAIC}(), \pi_i^{SAIC}(), i \in \{1, 2\}$ with the jointly optimal policies for the problem (11), we compare the expected return that is attained by SAIC with that of the centralized controller with full observability which is even harder to achieve.

Definition 6. Given a positive number $\epsilon$ a subset $\mathcal{P}_{i,k} \subseteq \Omega$ is said to be $\epsilon$-cost-uniform with respect to the policy $\pi(\cdot)$ if the following condition holds for two arbitrary observations $o', o'' \in \mathcal{P}_{i,k}$:

$c_1: M_{\pi}(o') = M_{\pi}(o'')$ \hspace{1cm} \hspace{1cm} (33)
$c_2: \text{For any } m \in M_{\pi}(o') : |Q_i^\pi(o', m) - Q_i^\pi(o'', m)| < \epsilon$ \hspace{1cm} \hspace{1cm} (34)

where $M_{\pi}(o') = \{m \in M : \pi(m|o') > 0\}$

Theorem 7. Consider a multi-agent system in which agents are subject to local observability and local action selection. If agents are allowed to communicate through communication channels with limited rate $R$-bits at each time step, the maximum achievable expected return of the multi-agent system following SAIC algorithm will be in a short neighbourhood of the same multiagent system if it was controlled with a centralized unit that has perfect communications with all agents:

$$\mathbb{E}_{\pi = \pi^*}((\pi(t))_{t=0}^T) \{g(t_0)\} = \mathbb{E}_{\pi = \pi^*}((\pi(t))_{t=0}^T) \{g(t_0)\} < \frac{2 \epsilon}{(1 - \gamma)^2},$$

where $\gamma$ is the discount factor and $\epsilon$ should be computed according to lemma 5.

Lemma 8. Given the partition $\mathcal{P}_i = \{\mathcal{P}_{i,1}, ..., \mathcal{P}_{i,2^{\epsilon}}\}$ that is obtained by solving eq. (27) during the centralized training phase, all subsets $\mathcal{P}_{i,k}$ for $k \in \{1, 2, ..., 2^{\epsilon}\}$ are $\epsilon$-cost-uniform with respect to the optimal joint policy $\pi(\cdot)$ for each $i \in \{1, 2\}$.

Proof. Following definition 1 and eq. (27) the proof is straightforward.

IV. NUMERICAL RESULTS

In this section, we evaluate our proposed schemes via numerical results for the popular rendezvous problem [31, 40], in which the inter-agent communication channel is set to have a limited rate. Rendezvous problem is of particular interest as it allows us to consider a cooperative multiagent system comprising of two agents that are required to communicate for their coordination task. In particular, as detailed in subsection IV-A, if the communication between agents is not efficient, at any time step $t$ each agent will only have access to its local observation $o_i(t)$, which is its own location in the case of rendezvous problem. This mere information is insufficient for an agent to attain the larger reward $C_2$, but is sufficient to attain the smaller reward $C_1$. Accordingly, compared with cases in which no communication between agents is present, in the setup of the rendezvous problem, efficient communication policies can increase the attained objective function of the multiagent system up to six-folds, as will be seen in Fig. 4. The system operates in discrete time, with agents taking actions and communicating in each time step $t = 1, 2, ...$. We consider a variety of grid-worlds with different size values $N$ and different locations for the goal-point $\omega^T$. We compare the proposed SAIC and LBIC with (i) Centralized Q-learning scheme and (ii) the Conventional Information Compression (CIC) scheme which is explained in subsection IV-B.

A. Rendezvous Problem

As illustrated in Fig. 3, two agents operate on an $N \times N$ grid world and aim at arriving at the same time at the goal point on the grid. Each agent $i \in \{1, 2\}$ at any time step $t$ can only observe its own location $o_i(t)$ on the grid, where the observation space is $\Omega = \{0, 1, 2, ..., n^2 - 1\}$. Each episode terminates as soon as an agent or both visit the goal point which is denoted as $\omega^T \in \Omega$. That is, at any time step $t$ that the observations tuple $(o_1(t), o_2(t))$ is a member of $\Omega^T = \{\omega^T\} \times \Omega \cup \Omega \times \{\omega^T\}$, the episode will be terminated. At time $t = 1$, the initial position of both agents is randomly and uniformly selected amongst the non-goal states, i.e. for each agent $i \in \{1, 2\}$ the initial position of the agent is $o_i(1) \in \Omega - \{\omega^T\}$.

At any time step $t = 1, 2, ...$ each agent $i$ observes its position, or environment state, and acquires information about the position of the other agent by receiving a communication message $\hat{c}_i(t)$
sent by the other agent \( j \neq i \) at the time step \( t \). Based on this information, agent \( i \) selects its environment action \( m_i(t) \) from the set \( M = \{ \text{Right, Left, Up, Down, Stop} \} \), where an action \( m_i(t) \in M \) represent the horizontal/vertical move of agent \( i \) on the grid at time step \( t \). For instance, if an agent \( i \) is on a grid-world as depicted on Fig. 3(a), and observes \( o_i(t) = 4 \) and selects "Up" as its action, the agent’s observation at the next time step will be \( o_i(t+1) = 8 \). If the position to which the agent should be moved is outside the grid, the environment is assumed to keep the agent in its current position. We assume that all these deterministic state transitions are captured by \( T(o_i(t), o_j(t), m_i(t), m_j(t)) \), which can determine the observations of agents in the next time step \( t+1 \) following

\[
\langle o_i(t+1), o_j(t+1) \rangle = T(o_i(t), o_j(t), m_i(t), m_j(t)).
\]

Accordingly, given observations \( o_i(t), o_j(t) \) and actions \( m_i(t), m_j(t) \), both agents receive a single team reward

\[
r(o_i(t), o_j(t), m_i(t), m_j(t)) = \begin{cases} 
C_1, & \text{if } P_1 \\
C_2, & \text{if } P_2, \\
0, & \text{otherwise}, 
\end{cases}
\]

where \( C_1 < C_2 \) and the propositions \( P_1 \) and \( P_2 \) are defined as \( P_1 : T(o_i(t), o_j(t), m_i(t), m_j(t)) \in \Omega^2 - \{ o_i^2 \} \) and \( P_2 : T(o_i(t), o_j(t), m_i(t), m_j(t)) \in \{ o_i^2 \} \). When only one agent arrives at the target point \( \omega^2 \), the episode will be terminated with the smaller reward \( C_1 \) being obtained, while the larger reward \( C_2 \) is attained when both agents visit the goal point at the same time. Note that this reward signal encourages coordination between agents which in turn can benefit from inter-agent communications.

Furthermore, at each time step \( t \) agents choose a communication message to send to the other agent by selecting a communication action \( c_i(t) \in C = \{ 0,1 \}^R \) of \( R \) bits, where \( R \) is the maximum achievable rate of the inter-agent communication channel following specific channel conditions and error correction methods which are used. That is, \( R \) is imposed to the problem as a constraint.

The goal of the multiagent system is, thus, to maximize the average discounted cumulative rewards by solving the problem (11).

**B. Conventional Information Compression In Multiagent Coordination Tasks**

As a baseline, we consider a conventional scheme that selects communications and actions separately. For communication, each agent \( i \) sends its observation \( o_i(t) \) to the other agent by following policy \( \pi_i^c(t) \). According to this policy the agent’s observation \( o_i(t) \) will be mapped to a binary bit sequence \( c_i(t) \), using an injective (and not necessarily surjective) mapping \( f_i : \Omega \to \{-1,1\}^{C_i} \). Consequently, the communication policy \( \pi_i^c \) becomes deterministic and follows

\[
\pi_i^c(c_i(t+1)|o_i(t), \tilde{c}_j(t)) = \delta(c_i(t+1) - f_i(o_i(t))).
\]

Agent \( i \) obtains an estimate \( \tilde{c}_j(t) \) of the observation \( j \) by having access to a quantized version of \( o_j(t) \). This estimate is used to define the environment-state-action value function \( Q_i^c(o_i(t), \tilde{c}_j(t), m_i(t)) \). This function is updated using Q-learning and the UCB policy in a manner similar to Algorithm 2, with no communication policy to be learned.

This communication strategy is proven to be optimal [7], if the inter-agent communication does not impose any cost on the cooperative objective function and the communication channel is noise-free. Under these conditions, and when the dynamics of the environment are deterministic, each agent \( i \) can distributively learn the optimal policy \( \pi^{\text{opt}}_i(\cdot) \), using value iteration or its model-free variants e.g., Q-learning [2]. This communication policy requires a channel rate \( R \geq H(o_i) \), whereas in this paper, we are focused on the scenarios with \( R \leq H(o_i) \). Therefore, due to the limited rate of the communication channel, a form of lossy information compression is required to be carried out.

Note that compression before a converged action policy is not possible, since all observations are a priori equally likely. Thus, we first train the CIC on a communication channel with unlimited capacity. Afterwards, when a probability distribution for observations is obtained, by applying Lloyd’s algorithm, we define an equivalence relation on the observation space \( \Omega \) with \( 2^R \) numbers of equivalence classes \( \Omega_1, \ldots, \Omega_{2^R} \). According to the defined equivalence relation by Lloyd’s algorithm, we can uniquely define the mapping \( f_2 : \Omega \to \{-1,1\}^R \) that maps agent \( j \)’s observation \( o_j(t) \). The mapping \( f_2(\cdot) \) that maps agent \( j \)’s observation \( o_j(t) \) into a quantized communication \( c_j(t) \) is not an injective mapping anymore. That is, by receiving the communication message \( c_j(t) \in \Omega_k \subset \subset \) agent \( i \) cannot retrieve \( o_j(t) \) but understands the observation of agent \( j \) has been a member of \( \Omega_k \). Upon, an optimal performance of Lloyd’s algorithm in defining the equivalence relation, we expect this algorithm to perform optimally, as long as \( H(o_i(t)) \leq R \). Note that this algorithm has a limitation, as it requires the first round of training to be done over communication channels with unlimited capacity.

**C. Results**

To perform our numerical experiments, rewards of the rendezvous problem are selected as \( C_1 = 1 \) and \( C_2 = 10 \), while the discount factor is \( \gamma = 0.9 \). A constant learning rate \( \alpha = 0.07 \) is applied, and the UCB exploration rate \( c = 12.5 \). In any figure that the performance of each scheme is reported in terms of the averaged discounted cumulative rewards, the attained rewards throughout training iterations are smoothed using a moving average filter of memory equal to 20,000 iterations. Regardless of the grid-world’s size and goal location, the grids are numbered row-wise starting from the left-bottom as shown in Fig. 3. Fig. 4 illustrates the performance of the proposed SAIC as well as six other benchmark schemes

- Centralized Q-learning under perfect communications.
• Learning based information compression (LBIC) scheme performs the joint design of communication and control policies through reinforcement learning following an algorithm similar to the one proposed in [14].

• CIC, see the details of CIC in subsection IV-B

• Heuristic non-communicative (HNC) algorithm is a heuristic scheme which exploits domain knowledge to design a control policy where no communication is present. In HNC, agents approach the goal point and wait next to it for a large enough number of time-steps to make sure the other agent has also arrived there. Only after that, they will get into the goal point. Note that this scheme requires communication/coordination between agents prior to the starting point of the task.

• Heuristic optimal communication (HOC) algorithm is a heuristic scheme which exploits domain knowledge to design jointly optimal communication and control policies. In HNC, agents approach the goal point and wait next to it until they hear from the other agent it also has arrived there. Only after that, they will get into the goal point. Note that this scheme requires communication/coordination between agents prior to the starting point of the task.

• Hybrid scheme uses the abstract representation of agents’ observations according to SAIC with \( R = 2 \) bits and feeds these latent observations to a centralized controller. The central controller learns the joint action selection of both agents using Q-learning.

The performance is measured in terms of the expected sum of discounted rewards in a rendezvous problem. The grid-world is considered to be of size \( N = 8 \) and its goal location to be \( \omega = 22 \). The rate budget of the channel between the two agents is \( R = 2 \) bits per time step. Since centralized Q-learning is not affected by the limitation on channel’s achievable bit rate, it achieves optimal performance after enough training, 160k iterations. The CIC, due to insufficient rate of the communication channel never achieves the optimal solution. The LBIC, however, is seen to outperforms the CIC, although it is trained and executed fully distributedly. It is observed that the SAIC by less than 1% gap achieves optimal performance and does that remarkably fast, where the performance gap for the LBIC and CIC are roughly 20% and 30% respectively. The yellow curve showing the performance of the CIC with no communication between agents, would show us the best performance of distributed reinforcement learning that can be achieved if no communication between agents is in place. In fact, the better performance of any scheme compared with the yellow curve, is the sign that the scheme benefits from some effective communication between agents. Note that, when inter-agent communication is unavailable, i.e., \( R = 0 \) bit per time step, there would be no difference in the performance of the CIC, SAIC or LBIC as all of them use the same algorithm to find out the action policy \( \pi^*(\cdot) \). We also recall the fact that both the CIC and SAIC require a separate training phase which is not captured by Fig. 4. SAIC requires a centralized training phase and CIC a distributed training phase with unlimited capacity of inter-agent communication channels. The performance of these two algorithms in Fig. 4 is plotted after the first phase of training.

To understand the underlying reasons for the remarkable performance of the SAIC, Fig. 5 is provided so that equivalence classes computed by the SAIC can be seen, all the locations of the grid shaded with the same colour belonging to the same equivalence class. The SAIC is extremely efficient, in performing state aggregation such that the loss of observation information does not incur any loss on the achievable sum of discounted rewards. Fig. 5(a) illustrates the state aggregation obtained by the SAIC, for which the achievable sum of discounted rewards is illustrated in Fig. 4. It is illustrated in Fig. 5(a) that how the SAIC performs observation compression with ratio \( R_c = 3 : 1 \), while it leads to nearly no performance loss for the collaborative task of the multiagent system. Here the definition of compression ratio follows

\[
R_c = \left[ \frac{H(\pi_o(t))}{H(\pi_e(t))} \right].
\]

(39)

Fig. 6 allows us to see how precise the approximation of \( V_{\pi^*}^{(o)}(o_1, \tilde{c}_1(1)) \) by the value function \( V^*(o_1, \tilde{c}_1(1)) \) is. The figure illustrates the values for both \( V_{\pi^*}^{(o)}(o_1(1), \tilde{c}_1(1)) \) and \( V^*(o_1(1), o_1(t)) \), where \( o_1(1) = 21 \) and for all \( o_1(t) \) takes all possible values in \( \Omega \). For instance the values 7.2 mentioned on the right down corner of the grid demonstrates the value of \( V^*(o_1(1), o_1(t)) \) when \( o_1(1) = 20 \) and \( o_1(t) = 7 \).

We also investigate the impact of channel rate \( R \) on the achievable value of objective function for the LBIC, SAIC and CIC, in Fig. 7. In this figure, the normalized value of achieved objective function for any scheme at any given \( R \) is shown. As per (40), the average of the achieved return for the scheme of interest is computed by

\[
E_{p_{\pi^*}^{(o)}((t(1)))} \{ g(1) \}, \text{ where } \pi^*_1(\cdot) \text{ and } \pi^*(\cdot) \text{ are obtained by the scheme of interest after solving (37) with a given value of } R.
\]

The attained objective function for the scheme of interest is then normalized by dividing it to the average of objective function

\[
E_{p_{\pi^*}^{(o)}((t(1)))} \{ g(1) \} \text{ that is attained by the optimal centralized policy } \pi^*(\cdot).
\]

The policy policy \( \pi^*(\cdot) \) is the optimal solution to (6) under no communications constraint.

\[
\frac{E_{p_{\pi^*}^{(o)}((t(1)))} \{ g(1) \}}{E_{p_{\pi^*}^{(o)}((t(1)))} \{ g(1) \}}.
\]

(40)

Accordingly, when the normalized objective function of a particular scheme is seen to be close to the value 1, it implies that the scheme has been able to compress the observation information with almost zero loss in the achieved objective function. On one hand, it is demonstrated that the SAIC soon achieves the optimal performance, while it takes the CIC at least \( R = 4 \) bits to get to achieve a sub-optimal value of the objective function. The LBIC, on the other hand, provides more than 10% performance gain in very low rates of communication \( R \leq \{1, 2, 3\} \) bits per time step, compared with CIC and 20% performance gain compared with SAIC at \( R = 1 \) bits per time step.

Fig. 8 studies the normalized objective functions attained by the LBIC, SAIC and CIC under different compression ratios \( R_c \). A whopping 40% performance gain is acquired by the SAIC, in comparison to the CIC, at high compression ratio \( R_c = 3 : 1 \). This means 66% of data rate saving with no performance drop in attaining the collaborative objective function. The SAIC, however, underperforms the LBIC and CIC at very high compression ratio of
Figure 5. State aggregation for multi-agent communication in a two-agent rendezvous problem with grid-worlds of varied sizes and goal locations. The observation space is aggregated to four equivalence classes, $R = 2$ bits, and number of training episodes has been $K = 1500k$, $K = 1000k$ and $K = 500k$ for figure (a) and (b) and (c) respectively. Locations with similar color represent all the agents’ observations which are grouped into the same equivalence class. The information compression ratio $R^c_i$ has seen to be $6.2$, $5.2$ and $4.2$ in subplots a), b) and c) respectively.

Figure 6. Left grid-world shows the observation space $G$, amongst which one particular observation is chosen $o_i(t) = 20$. While agent $i$ makes this observation, agent $j$ can potentially be at any other 64 locations of the greed. The value function $V^i_o (o_i (t) = 20, o_j (t))$ for all $o_j (t) \in \Omega$ is depicted in the right grid-world, e.g. a number at location 22, shows the value function $V^i_o (o_i (t) = 20, o_j (t) = 22) = 10$. You can also see the values of $V^m_i (o_i (t), \tilde{c}_j (t))$ for $o_i (t) = 20$ and all possible $\tilde{c}_j (t) \in C$ with $R = 2$ bits.

Figure 7. A performance comparison between several multi-agent communication and control schemes under different achievable bit rates. All experiments are performed where $N = 8$ and $\omega^2 = 21$, similar to the grid-world of Fig.5-a. The number of training episodes/iterations for any scheme at any given channel rate $R$ has been $K = 200K$.

$R^c_i = 6 : 1$. This is due to the fact that the condition mentioned in remark 2 is not met at this high rate of compression. Moreover, the CIC scheme is seen not to achieve the optimal performance even at compression rate of $R^c_i = 6 : 5$ which is due to the fact that by exceeding the compression ratio $R^c_i = 1 : 1$ each agent $i$ may lose some information about the observation $o_j (t)$ of the other agent which can be helpful in taking the optimal action decision.

As demonstrated through a range of numerical experiments, the weakness of conventional schemes for compression of agents’ ob-

Figure 8. A performance comparison between several multi-agent communication and control schemes under different rates of information compression. All experiments are performed where $N = 8$ and $\omega^2 = 21$. The number of training episodes/iterations for any scheme at any given channel rate $R$ has been $K = 200K$.

servations is that they may lose/keep information regardless of how useful they can be towards achieving the optimal objective function. In contrast, the task-based compression schemes SAIC and LBIC, for communication rates (very) lower than the entropy of the observation process, manage to compress the observation information not to minimize the distortion but to maximize the achievable value of the objective function.

V. CONCLUSION

This paper has investigated a distributed multiagent reinforcement learning problem in which agents share a unique task, maximizing the average discounted cumulative one-stage rewards. Since we consider a limited rate for the multiagent communication channels, task-based compression of agents observations has been of the essence. The so-called task-based compression scheme, SAIC, introduced in this paper, is different from the conventional source coding algorithms in the sense that they do not aim at achieving minimum possible distortion given a communication rate, but rather, they maximize the objective function of the multiagent system given a communication rate between agents.

The proposed schemes are seen to outperform conventional source coding algorithms, by up to a remarkable 40% difference in the achieved objective function, when being imposed with (very) tight constraint on the communication rate. The introduced information compression schemes can have a substantial impact in many communication applications, e.g. device to device communications, where the ultimate goal of communication is not a reliable transfer of information between two ends but is to acquire information which is useful to improve an achievable team objective. Our scheme is of
more relevance to scenarios where observation process which is to be compressed is generated by an MDP and is not an i.i.d process. The studies in this paper have been limited to a team of two agents with symmetric constraints on the communication rates. Accordingly, considering a system composed of larger number of agents as well as non-symmetric rate constraints for the same problem can be useful avenues to extend the applicability of the introduced schemes.

REFERENCES

[1] V. Vapnik, “Principles of risk minimization for learning theory,” in Advances in neural information processing systems, 1992, pp. 831–838.

[2] M. Lauer and M. A. Riedmiller, “An algorithm for distributed reinforcement learning in cooperative multi-agent systems,” in Proc. Conference on Machine Learning. Morgan Kaufmann Publishers Inc., 2000.

[3] V. Kostina and B. Hassibi, “Rate-tradeoff in control,” IEEE Transactions on Automatic Control, vol. 64, no. 11, pp. 4525–4540, 2019.

[4] T.-Y. Tung, S. Kubus, J. R. Pujol, and D. Gunduz, “Effective communications: A joint learning and communication framework for multi-agent reinforcement learning over noisy channels,” arXiv preprint arXiv:2101.10369, 2021.

[5] S. Arimoto, “An algorithm for computing the capacity of arbitrary discrete memoryless channels,” IEEE Transactions on Information Theory, vol. 18, no. 1, pp. 14–20, 1972.

[6] D. Lee, N. He, P. Kamalaruban, and V. Cevher, “Optimization for communication-mediated multigain autonomous systems,” IEEE Signal Processing Magazine, vol. 37, no. 3, pp. 123–135, 2020.

[7] D. P. Bertsekas and D. A. Castanon, “Adaptive aggregation methods for infinite horizon dynamic programming,” IEEE Transactions on Automatic Control, vol. 34, no. 6, pp. 589–598, June 1989.

[8] D. P. Bertsekas, “Feature-based aggregation and deep reinforcement learning: A survey and some new implementations,” IEEE/CAA Journal of Automatica Sinica, vol. 6, no. 1, pp. 1–31, 2018.

[9] G. Rubino, “Interactive multi-agent communication with backpropagation,” in Proc. Advances in Neural Information Processing Systems, Barcelona, 2016, pp. 2244–2252.

[10] D. P. Bertsekas and D. A. Castanon, “Adaptive aggregation, rollout, and enhanced policy improvement for reinforcement learning,” arXiv preprint arXiv:1910.02426, 2019.

[11] H. Zou, C. Zhang, S. Lasaulce, and et al., “Decision-oriented communications: Application to energy-efficient resource allocation,” in Intl. Conf. on Wireless Networks and Mobile Communications. IEEE, 2018.

[12] H. Mao, Z. Zhang, Z. Xiao, Z. Gong, and Y. Ni, “Learning agent communication under limited bandwidth by message pruning,” arXiv preprint arXiv:1912.05304, 2019.

[13] S. Sukhbaatar, R. Fergus et al., “Learning multiagent communication with backpropagation,” in Proc. Advances in Neural Information Processing Systems, Barcelona, 2016, pp. 2244–2252.

[14] D. P. Bertsekas, “Adaptive aggregation methods for infinite horizon dynamic programming,” IEEE Transactions on Automatic Control, vol. 34, no. 6, pp. 589–598, June 1989.

[15] G. Rubino, “Interactive multi-agent communication with backpropagation,” in Proc. Advances in Neural Information Processing Systems, Barcelona, 2016, pp. 2244–2252.

[16] D. P. Bertsekas, “Feature-based aggregation and deep reinforcement learning: A survey and some new implementations,” IEEE/CAA Journal of Automatica Sinica, vol. 6, no. 1, pp. 1–31, 2018.

[17] G. Rubino, “Interactive multi-agent communication with backpropagation,” in Proc. Advances in Neural Information Processing Systems, Barcelona, 2016, pp. 2244–2252.

[18] D. P. Bertsekas, “Adaptive aggregation methods for infinite horizon dynamic programming,” IEEE Transactions on Automatic Control, vol. 34, no. 6, pp. 589–598, June 1989.

[19] G. Rubino, “Interactive multi-agent communication with backpropagation,” in Proc. Advances in Neural Information Processing Systems, Barcelona, 2016, pp. 2244–2252.

[20] D. P. Bertsekas, “Adaptive aggregation methods for infinite horizon dynamic programming,” IEEE Transactions on Automatic Control, vol. 34, no. 6, pp. 589–598, June 1989.
APPENDIX B
PROOF OF LEMMA 1

Proof. We know that by following the optimal action policy \( \pi^*_i \), the following holds

\[
V^{n_m,*}_{\pi^*_i} (o_i(t), c_i(t)) \geq V^{n_m,*}_{\pi^*_i} (o_i(t), \tilde{c}_i(t)),
\]

\( \forall \pi^*_i (\cdot), \ \forall c_i(t) \in \{ -1, 1 \}, \ \forall o_i(t) \in \Omega. \) \hfill (42)

According to \( \text{(42)} \), a weighted sum of \( V^{n_m,*}_{\pi^*_i} (o_i(t), \tilde{c}_i(t)) \) will always remain larger than or equal to the sum of \( V^{n_m,*}_{\pi^*_i} (o_i(t), c_i(t)) \) weighted with the same coefficients:

\[
E_{p(o_i(t), \tilde{c}(t))} \left\{ V^{n_m,*}_{\pi^*_i} (o_i(t), \tilde{c}_i(t)) \right\} \geq \sum_{o_i(t) \in \Omega} E_{p(o_i(t), \tilde{c}(t))} \left\{ V^{n_m,*}_{\pi^*_i} (o_i(t), c_i(t)) \right\},
\]

\( \text{where } p(o_i(t), \tilde{c}_i(t)) \text{ can be any joint probability mass function of } o_i(t), \tilde{c}_i(t). \) \hfill (43)

That is to say

\[
E_{p(o_i(t), \tilde{c}(t))} \left\{ V^{n_m,*}_{\pi^*_i} (o_i(t), \tilde{c}_i(t)) \right\} = \max_{\pi^*_i} \sum_{o_i(t) \in \Omega} E_{p(o_i(t), \tilde{c}(t))} \left\{ V^{n_m,*}_{\pi^*_i} (o_i(t), c_i(t)) \right\}. \hfill (44)
\]

APPENDIX C
PROOF OF LEMMA 2

Proof. \( V^*(o_i(t'), \tilde{c}_i(t')) = E_{p([tr]^{M} | o_i(t), \tilde{c}_i(t))} \left\{ \sum_{t=0}^{M} \gamma^{-1} r(o_i(t), o_j(t), m_i(t), m_j(t))o_i(t), \tilde{c}_i(t) \right\}, \)

\( \text{where the conditional probability } p([tr]^{M} | o_i(t'), \tilde{c}_i(t')) \text{ can be extended following the law of total probabilities:} \)

\[
V^*(o_i(t'), \tilde{c}_i(t')) = \sum_{o_i(t') \in \Omega} p([tr]^{M} | o_i(t'), o_j(t'), \tilde{c}_i(t')) p(o_i(t)|\tilde{c}_i(t)) \]

\( \text{in which } o_i(t'), o_j(t') \text{ are sufficient statistics and the second summation can be shifted to have} \)

\[
V^*(o_i(t'), \tilde{c}_i(t')) = \sum_{o_j(t') \in \Omega} \sum_{o_i(t) \in [tr]^{M}} g(t') p([tr]^{M} | o_i(t'), o_j(t'), \tilde{c}_i(t')) p(o_i(t)|\tilde{c}_i(t)). \hfill (48)
\]

APPENDIX D
PROOF OF THEOREM \[7\]

Proof. According to the \[23\] (Lemma 1), optimal state values of the aggregated MDPs (the environment as is seen by one agent during the decentralized training phase of SAIC) are in a short neighbourhood of the optimal values corresponding to the optimal solution to the original underlying MDP:

\[
\forall o_j \in \Omega \text{ and } \forall i \in \{1, 2\} : \quad |V^*(o_i, o_j) - V^*_i (o_i, \tilde{c}_i)| < \frac{2\epsilon}{(1-\gamma)^2}, \hfill (50)
\]

where \( V^*_i (\cdot) \) is the value function corresponding to \( \pi_i^{SAIC} (\cdot) \). Following the eq. \[15\], one can write the expected return of the system under centralized scheme as:

\[
E_{p_{o_i, o_j}} \{ g(t_o) \} = E \left\{ V^*(o_i(t_o), o_j(t_o)) \right\} \sum_{o_j \in \Omega} \sum_{o_i \in \Omega} V^*(o_i(t_o), o_j(t_o)) p_{o_i, o_j} (o_i(t_o), o_j(t_o)), \hfill (51)
\]

where the second expectation is taken over the joint probability distribution \( p_{o_i, o_j} (o_i(t_o), o_j(t_o)) \) of \( o_i \) and \( o_j \) when following the action policy \( \pi^*(\cdot) \). Similarly, following the eq. \[15\], one can write the expected return of the system under SAIC scheme as:

\[
E_{p_{o_i, \tilde{c}_i}} \{ g(t_o) \} = \sum_{\tilde{c}_i \in \tilde{c}_i} V^*(o_i(t_o), \tilde{c}_i(t_o)) p_{o_i, \tilde{c}_i} (o_i(t_o), \tilde{c}_i(t_o)). \hfill (52)
\]

We can rewrite the joint probability \( p_{o_i, \tilde{c}_i} (o_i(t_o), \tilde{c}_i(t_o)) \) as

\[
p_{o_i, \tilde{c}_i} (o_i(t_o), \tilde{c}_i(t_o)) = \sum_{o_j(t_o) \in P_{i,k}} p_{o_i, o_j} (o_i(t_o), o_j(t_o)). \hfill (53)
\]

where the subset \( \mathcal{P}_{i,k} \subset \Omega \) stands for the set of all observation realizations \( o_j \) that are represented by \( \tilde{c}_i(t_o) \) according to the policy \( \pi_i^{SAIC} (\cdot) \). Further to equations \[51\]–[53], the difference between the achievable expected return of the centralized scheme and SAIC can be explained by

\[
E_{p_{o_i, \tilde{c}_i}} \{ g(t_o) \} - E_{p_{o_i, o_j}} \{ g(t_o) \} = \sum_{o_j \in \Omega} \sum_{o_i \in \Omega} V^*(o_i(t_o), o_j(t_o)) p_{o_i, o_j} (o_i(t_o), o_j(t_o)) - \sum_{o_j \in \Omega} \sum_{o_i \in \Omega} V^*(o_i(t_o), \tilde{c}_i(t_o)) p_{o_i, \tilde{c}_i} (o_i(t_o), \tilde{c}_i(t_o)). \hfill (54)
\]

We now proceed by factorizing the joint probability \( p_{o_i, o_j} (o_i(t_o), o_j(t_o)) \) which yields

\[
E_{p_{o_i, \tilde{c}_i}} \{ g(t_o) \} - E_{p_{o_i, \tilde{c}_i}} \{ g(t_o) \} = \sum_{o_j \in \Omega} \sum_{o_i \in \Omega} V^*(o_i(t_o), o_j(t_o)) - \sum_{o_j \in \Omega} \sum_{o_i \in \Omega} V^*(o_i(t_o), \tilde{c}_i(t_o)). \hfill (55)
\]

Since \( |V^*(o_i(t_o), o_j(t_o)) - V^*(o_i(t_o), \tilde{c}_i(t_o))| \) is upper-bounded by a constant term \( \frac{2\epsilon}{(1-\gamma)^2} \), its weighted sum is also upper bounded by the same term \( \frac{2\epsilon}{(1-\gamma)^2} \). Thus we conclude the proof of theorem \[7\].