Learning to Communicate Implicitly by Actions

Zheng Tian,[1] Shihao Zou,[1] Ian Davies,[1] Tim Warr,[1] Lisheng Wu,[1]
Haitham Bou Ammar,[1,2] Jun Wang[1]
[1]University College London, [2]Huawei R&D UK
{zheng.tian.11, shihao.zou.17, ian.davies.12, tim.warr.17, lisheng.wu.17}@ucl.ac.uk
haitham.bouammar@huawei.com
jun.wang@cs.ucl.ac.uk

Abstract

In situations where explicit communication is limited, human collaborators act by learning to: (i) infer meaning behind their partner's actions, and (ii) convey private information about the state to their partner implicitly through actions. The first component of this learning process has been well-studied in multi-agent systems, whereas the second — which is equally crucial for successful collaboration — has not. To mimic both components mentioned above, thereby completing the learning process, we introduce a novel algorithm: Policy Belief Learning (PBL). PBL uses a belief module to model the other agent's private information and a policy module to form a distribution over actions informed by the belief module. Furthermore, to encourage communication by actions, we propose a novel auxiliary reward which incentivizes one agent to help its partner to make correct inferences about its private information. The auxiliary reward for communication is integrated into the learning of the policy module. We evaluate our approach on a set of environments including a matrix game, particle environment and the non-competitive bidding problem from contract bridge. We show empirically that this auxiliary reward is effective and easy to generalize. These results demonstrate that our PBL algorithm can produce strong pairs of agents in collaborative games where explicit communication is disabled.

Introduction

In collaborative multi-agent systems, communication is essential for agents to learn to behave as a collective rather than a collection of individuals. This is particularly important in the imperfect information setting, where private information becomes crucial to success. In such cases, efficient communication protocols between agents are needed for private information exchange, coordinated joint-action exploration, and true world-state inference.

In typical multi-agent reinforcement learning (MARL) settings, designers incorporate explicit communication channels hoping to conceptually resemble language or verbal communication which are known to be important for human interaction (Baker et al. 1999). Though they can be used for facilitating collaboration in MARL, explicit communication channels come at additional computational and memory costs, making them difficult to deploy in decentralized control (Roth, Simmons, and Veloso 2006).

Environments where explicit communication is difficult or prohibited are common. These settings can be synthetic such as those in games, e.g., bridge and Hanabi, but also frequently appear in real-world tasks such as autonomous driving and autonomous fleet control. In these situations, humans rely upon implicit communication as a means of information exchange (Rasouli, Kotseruba, and Tsotsos 2017) and are effective in learning to infer the implicit meaning behind others' actions (Heider and Simmel 1944). The ability to perform such inference requires the attribution of a mental state and reasoning mechanism to others. This ability is known as theory of mind (Premack and Woodruff 1978). In this work, we develop agents that benefit from considering others' perspectives and thereby explore the further development of machine theory of mind (Rabinowitz et al. 2018).

Previous works have considered ways in which an agent can, by observing an opponent's behavior, build a model of opponents' characteristics, objectives or hidden information either implicitly (He et al. 2016; Bard et al. 2013) or explicitly (Raileanu et al. 2018; Li and Miikkulainen 2018). Whilst these works are of great value, they overlook the fact that an agent should also consider that it is being modeled and adapt its behavior accordingly, thereby demonstrating a theory of mind. For instance, in collaborative tasks, a decision-maker could choose to take actions which are informative to its teammates, whereas, in competitive situations, agents may act to conceal private information to prevent their opponents from modeling them effectively.

In this paper, we propose a generic framework, titled policy belief learning (PBL), for learning to cooperate in imperfect information multi-agent games. Our work combines opponent modeling with a policy that considers that it is being modeled. PBL consists of a *belief module*, which models other agents' private information by considering their previous actions, and a *policy module* which combines the agent's current observation with their beliefs to return a distribution over actions. We also propose a novel auxiliary reward for encouraging communication by actions, which is integrated

into PBL. Our experiments show that agents trained using PBL can learn collaborative behaviors more effectively than a number of meaningful baselines without requiring any explicit communication. We conduct a complete ablation study to analyze the effectiveness of different components within PBL in our bridge experiment.

Related Work

Our work is closely related to (Albrecht and Stone 2017; Lowe et al. 2017; Mealing and Shapiro 2017; Raileanu et al. 2018) where agents build models to estimate other agents' hidden information. Contrastingly, our work enhances a "flat" opponent model with recursive reasoning. "Flat" opponent models estimate only the hidden information of opponents. Recursive reasoning requires making decisions based on the mental states of others as well as the state of the environment. In contrast to works such as I-POMDP (Gmytrasiewicz and Doshi 2005) and PR2 (Wen et al. 2019) where the nested belief is embedded into the training agent's opponent model, we incorporate level-1 nested belief "I believe that you believe" into our policy by a novel auxiliary reward.

Recently, there has been a surge of interest in using reinforcement learning (RL) approaches to learn communication protocols (Foerster et al. 2016; Lazaridou, Peysakhovich, and Baroni 2016; Mordatch and Abbeel 2017; Sukhbaatar, Szlam, and Fergus 2016). Most of these works enable agents to communicate via an explicit channel. Among these works, Mordatch and Abbeel (2017) also observe the emergence of non-verbal communication in collaborative environments without an explicit communication channel, where agents are exclusively either a sender or a receiver. Similar research is also conducted in (de Weerd, Verbrugge, and Verheij 2015). In our setting, we do not restrict agents to be exclusively a sender or a receiver of communications – agents can communicate mutually by actions. Knepper et al. (2017) propose a framework for implicit communication in a cooperative setting and show that various problems can be mapped into this framework. Although our work is conceptually close to (Knepper et al. 2017), we go further and present a practical algorithm for training agents. The recent work of (Foerster et al. 2018) solves an imperfect information problem as considered here from a different angle. We approach the problem by encouraging agents to exchange critical information through their actions whereas Foerster et al. train a public player to choose an optimal deterministic policy for players in a game based on publicly observable information. Implicit communication is also considered in the human-robot interaction community. In Cooperative Inverse RL (CIRL) where robotic agents try to infer a human's private reward function from their actions (Hadfield-Menell et al. 2016), optimal solutions need to produce behavior that conveys information.

Dragan, Lee, and Srinivasa (2013) consider how to train agents to exhibit legible behavior (i.e. behavior from which it is easy to infer the intention). Their approach is dependent on a hand-crafted cost function to attain informative behavior. Mutual information has been used as a means to promote coordination without the need for a human engineered cost function. Strouse et al. (2018) use a mutual information objective to encourage an agent to reveal or hide its intention. In a related work, Jaques et al. (2019) utilize a mutual information objective to imbue agents with social influence. While the objective of maximal mutual information in actions can yield highly effective collaborating agents, a mutual information objective in itself is insufficient to necessitate the development of implicit communication by actions. Eccles et al. (2019) introduce a reciprocity reward as an alternative approach to solve social dilemmas.

A distinguishing feature of our work in relation to previous works in multi-agent communication is that we do not have a predefined explicit communication protocol or learn to communicate through an explicit channel. Information exchange can only happen via actions. In contrast to previous works focusing on unilaterally making actions informative, we focus on bilateral communication by actions where information transmission is directed to a specific party with potentially limited reasoning ability. Our agents learn to communicate through iterated policy and belief updates such that the resulting communication mechanism and belief models are interdependent. The development of a communication mechanism therefore requires either direct access to the mental state of other agents (via centralized training) or the ability to mentalize, commonly known as theory of mind. We investigate our proposed algorithm in both settings.

Problem Definition

We consider a set of agents, denoted by $\mathcal{N}$, interacting with an unknown environment by executing actions from a joint set $\mathcal{A} = \{\mathcal{A}_1, \dots, \mathcal{A}_N\}$, with $\mathcal{A}_i$ denoting the action space of agent i, and N the total number of agents. To enable models that approximate real-world scenarios, we assume private and public information states. Private information states, jointly (across agents) denoted by $\mathcal{X} = \{\mathcal{X}_1, \dots, \mathcal{X}_N\}$ are a set of hidden information states where $\mathcal{X}_i$ is only observable by agent i, while public states $\mathcal{O}$ are observed by all agents. We assume that hidden information states at each time step are sampled from an unknown distribution $\mathcal{P}_x : \mathcal{X} \to [0, 1]$, while public states evolve from an initial distribution $\mathcal{P}_o : \mathcal{O} \to [0, 1]$, according to a stochastic transition model $\mathcal{T} : \mathcal{O} \times \mathcal{X} \times \mathcal{A}_1 \times \cdots \times \mathcal{A}_N \times \mathcal{O} \to [0, 1]$. Having transitioned to a successor state according to $\mathcal{T}$, agents receive rewards from $\mathcal{R} : \mathcal{S} \times \mathcal{A}_1 \times \cdots \times \mathcal{A}_N \to \mathbb{R}$, where we have used $\mathcal{S} = \mathcal{O} \times \mathcal{X}$ to denote joint state descriptions that incorporate both public and private information. Finally, rewards are discounted over time by a factor $\gamma \in (0, 1]$. With this notation, our problem can be described succinctly by the tuple: $\langle \mathcal{N}, \mathcal{A}, \mathcal{O}, \mathcal{X}, \mathcal{T}, \mathcal{R}, \mathcal{P}_x, \mathcal{P}_o, \gamma \rangle$, which we refer to as an imperfect information Markov decision process (I2MDP)[1]. In this work, we simplify the problem by assuming that hidden information states $\mathcal{X}$ are temporally static and are given at the beginning of the game.

We interpret the joint policy from the perspective of agent i such that $\pi = (\pi^i(a^i|s), \pi^{-i}(a^{-i}|s))$, where $\pi^{-i}(a^{-i}|s)$

[1]We also note that our problem can be formalized as a decentralized partially observable Markov decision process (Dec-POMDP) (Bernstein, Zilberstein, and Immerman 2013).

is a compact representation of the joint policy of all agents excluding agent i. In the collaborative setting, each agent is presumed to pursue the shared maximal cumulative reward expressed as

$$\max \ \ \eta^i(\pi) = \mathbb{E}\left[\sum_{t=1}^{\infty} \gamma^t R(s_t, a_t^i, a_t^{-i})\right], \tag{1}$$

where $s_t = [x_t^i, x_t^{-i}, o_t]$ is the current full information state, (a_t^i, a_t^{-i}) are joint actions taken by agent i and all other agents respectively at time t and γ is a discount factor.

Policy Belief Learning

Applying naive single agent reinforcement learning (SARL) algorithms to our problem will lead to poor performance. One reason for this is the partial observability of the environment. To succeed in a partially observable environment, an agent is often required to maintain a belief state. Recall that, in our setting, the environment state is formed from the union of the private information of all agents and the publicly observable information, $s_t = [x_t^i, x_t^{-i}, o_t]$. We therefore learn a belief module $\Phi^i(x_t^{-i})$ to model other agents' private information x_t^{-i} which is the only hidden information from the perspective of agent i in our setting. We assume that an agent can model x_t^{-i} given the history of public information and actions executed by other agents $h_t^i = \{o_{1:t-1}, a_{1:t-1}^{-i}\}$. We use a NN to parameterize the belief module which takes in the history of public information and produces a belief state $b_t^i = \Phi^i(x_t^{-i}|h_t^i)$. The belief state together with information observable by agent i forms a sufficient statistic, $\hat{s}_t^i = [x_t^i, b_t^i, o_t]$, which contains all the information necessary for the agent to act optimally (Åström 1965). We use a separate NN to parameterize agent i's policy $\pi^i(a_t^i|\hat{s}_t^i)$ which takes in the estimated environment state $\hat{s}_t^i$ and outputs a distribution over actions. As we assume hidden information is temporally static, we will drop the time script for it in the rest of the paper.

The presence of multiple learning agents interacting with the environment renders the environment non-stationary. This further limits the success of SARL algorithms which are generally designed for environments with stationary dynamics. To solve this, we adopt centralized training and decentralized execution, where during training all agents are recognized as one central representative agent differing only by their observations. Under this approach, one can imagine belief models $\Phi^i(x^{-i}|h_t^i)$ and $\Phi^{-i}(x^i|h_t^{-i})$ sharing parameters $\boldsymbol{\phi}$. The input data, however, varies across agents due to the dependency on both h_t^i and h_t^{-i}. In a similar fashion, we let policies share the parameters $\boldsymbol{\theta}$. Consequently, one may think of updating $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$ using one joint data set aggregated across agents. Without loss of generality, in the remainder of this section, we discuss the learning procedure from the point of view of a single agent, agent i.

We first present the learning procedure of our belief module. At iteration k, we use the current policy $\pi_{[k]}(a^i|\hat{s})$ to generate a data set of size M, $\Omega_{[k]} = \{(x_j^{-i}, h_j^i)_{j=1}^M\}$, using self-play and learn a new belief module by minimizing:

$$\boldsymbol{\phi}_{[k]} := \underset{\boldsymbol{\phi}}{\operatorname{argmin}}\, \mathbb{E}_{(x^{-i},h^i)\sim\Omega_{[k-1]}}\left[\mathrm{KL}(x_j^{-i}||b_j^i(h_j^i;\boldsymbol{\phi})\right], \tag{2}$$

where $\mathrm{KL}(\cdot||\cdot)$ is the Kullback–Leibler(KL) divergence and we use a one-hot vector to encode the ground truth, x_j^{-i}, when we calculate the relevant KL-divergence.

With updated belief module $\Phi_{[k]}$, we learn a new policy for the next iteration, $\pi_{[k+1]}$, via a policy gradient algorithm. Sharing information in multi-agent cooperative games through communication reduces intractability by enabling coordinated behavior. Rather than implementing expensive protocols (Heider and Simmel 1944), we encourage agents to *implicitly communicate* through actions by introducing a novel auxiliary reward signal. To do so, notice that in the centralized setting agent i has the ability to consult its opponent's belief model $\Phi^{-i}(x^i|h_t^{-i})$ thereby exploiting the fact that other agents hold beliefs over its private information x_i. In fact, comparing b_t^{-i} to the ground-truth x^i enables agent i to learn which actions bring these two quantities closer together and thereby learn informative behavior. This can be achieved through an auxiliary reward signal devised to encourage informative action communication:

$$r_{c,t}^i = \mathrm{KL}(x^i||b^{-i,*}) - \mathrm{KL}(x^i||b_{t+1}^{-i}), \tag{3}$$

where $b^{-i,*} = \Phi_{[k]}^{-i}(x^i|h_{t,*}^{-i})$ is agent $-i$'s best belief (so-far) about agent i's private information:

$$b^{-i,*} = \operatorname{argmin}\ \mathrm{KL}(x^i||b_u^{-i})\ \forall\ u \le t.$$

In other words, $r_{c,t}^i$ encourages communication as it is proportional to the improvement in the opponent's belief (for a fixed belief model $\Phi_{[k]}^{-i}(x^i|h_{t+1}^{-i})$), measured by its proximity to the ground-truth, resulting from the opponent observing agent i's action a_t^i. Hence, during the policy learning step of PBL, we apply a policy gradient algorithm with a shaped reward of the form:[2]

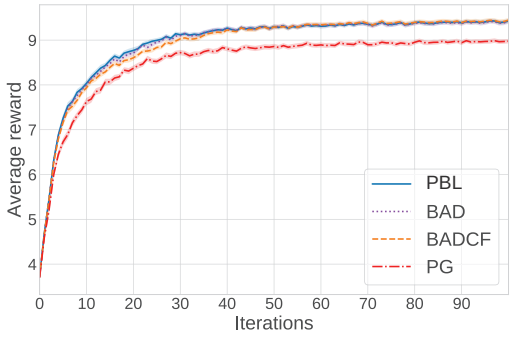
$$r = r_e + \alpha r_c, \tag{4}$$

where r_e is the reward from the environment, r_c is the communication reward and $\alpha \ge 0$ balances the communication and environment rewards.

Initially, in the absence of a belief module, we pre-train a policy $\pi_{[0]}$ naively by ignoring the existence of other agents in the environment. As an agent's reasoning ability may be limited, we may then iterate between Belief and Policy learning multiple times until either the allocated computational resources are exhausted or the policy and belief modules converge. We summarize the main steps of PBL in Algorithm 1. Note that, although information can be leaked during training, as training is centralized, distributed test-phase execution ensures hidden-private variables during execution.

[2]Please note, we omit the agent index i in the reward equation, as we shape rewards similarly for all agents.

(a) Payoff for the matrix game



(b) Learning curves of PBL and baselines over 100 runs

Figure 1: Matrix game experiment and results.

Algorithm 1 Per-Agent Policy Belief Learning (PBL)

Initialize: Randomly initialize policy π_0 and belief Φ_0
Pre-train π_0
for $k = 0$ to max_iterations **do**
  Sample episodes for belief training using self-play forming the data set $\Omega_{[k]}$
  Update belief network using data from $\Omega_{[k]}$ solving Equation 2
  Given updated beliefs $\Phi_{[k+1]}(\cdot)$, update policy $\pi(\cdot)$ (policy gradients with rewards from Equation 4)
end for
Output: Final policy, and belief model

Machine Theory of Mind

In PBL, we adopt a centralized training and decentralized execution scheme where agents share the same belief and policy models. In reality, however, it is unlikely that two people will have exactly the same reasoning process. In contrast to requiring everyone to have the same reasoning process, a person's success in navigating social dynamics relies on their ability to attribute mental states to others. This attribution of mental states to others is known as theory of mind (Premack and Woodruff 1978). Theory of mind is fundamental to human social interaction which requires the recognition of other sensory perspectives, the understanding of other mental states, and the recognition of complex non-verbal signals of emotional state (Lemaignan and Dillenbourg 2015). In collaboration problems without an explicit communication channel, humans can effectively establish an understanding of each other's mental state and subsequently select appropriate actions. For example, a teacher will re-iterate a difficult concept to students if she infers from the students' facial expressions that they have not understood. The effort of one agent to model the mental state of another is characterized as Mutual Modeling (Dillenbourg 1999).

In our work, we also investigate whether the proposed communication reward can be generalized to a distributed setting which resembles a human application of theory of mind. Under this setting, we train a separate belief model for each agent so that $\Phi^i(x^{-i}|h_t^i)$ and $\Phi^{-i}(x^i|h_t^{-i})$ do not share parameters ($\phi^i \neq \phi^{-i}$). Without centralization, an agent can only measure how informative its action is to others with its own belief model. Assuming agents can perfectly recall their past actions and observations, agent i computes its communication reward as:[3]

$$r_{c,t}^i = \mathrm{KL}(x^i||\tilde{b}_t^{i,*}) - \mathrm{KL}(x^i||\tilde{b}_{t+1}^i),$$

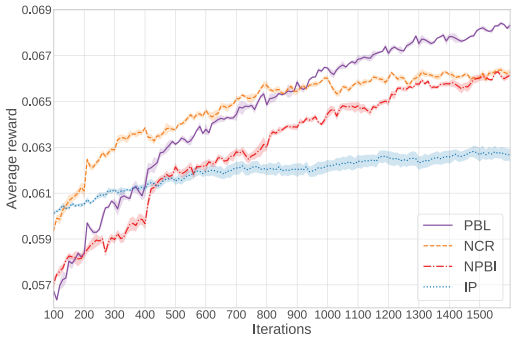
where $\tilde{b}_t^{i,*} = \Phi^i(x^i|h_{t,*}^{-i})$ and $\tilde{b}_{t+1}^i = \Phi^i(x^i|h_{t+1}^{-i})$. In this way, an agent essentially establishes a mental state of others with its own belief model and acts upon it. We humbly believe this could be a step towards machine theory of mind where algorithmic agents learn to attribute mental states to others and adjust their behavior accordingly.

The ability to mentalize relieves the restriction of collaborators having the same reasoning process. However, the success of collaboration still relies on the correctness of one's belief about the mental states of others. For instance, correctly inferring other drivers' mental states and conventions can reduce the likelihood of traffic accidents. Therefore road safety education is important as it reduces variability among drivers reasoning processes. In our work, this alignment amounts to the similarity between two agents' trained belief models which is affected by training data, initialization of weights, training algorithms and so on. We leave investigation of the robustness of collaboration to variability in collaborators' belief models to future work.

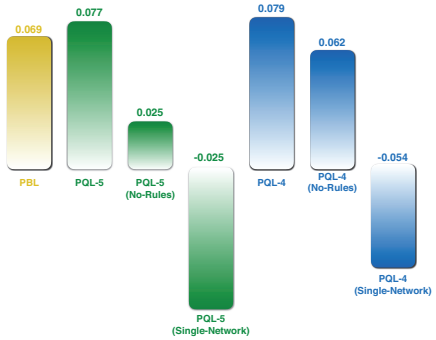
Experiments & Results

We test our algorithms in three experiments. In the first, we validate the correctness of the PBL framework which integrates our communication reward with iterative belief and policy module training in a simple matrix game. In this relatively simple experiment, PBL achieves near optimal performance. Equipped with this knowledge, we further apply PBL to the non-competitive bridge bidding problem to verify its scalability to more complex problems. Lastly, we investigate the efficacy of the proposed communication reward in a distributed training setting.

[3]Note the difference of super/sub-scripts of the belief model and its parameters when compared to Equation 3.



(a) Learning curves for non-competitive bridge bidding



(b) Comparison of PBL and PQL

Figure 2: a) Learning curves for non-competitive bridge bidding with a warm start from a model trained to predict the score distribution (average reward at warm start: 0.038). Details of warm start provided in our supplementary material. b) Bar graph comparing PBL to variants of PQL, with the full version of PQL results as reported in (Yeh and Lin 2016).

Matrix Game

We test our PBL algorithm on a matrix card game where an implicit communication strategy is required to achieve the global optimum. This game is first proposed in (Foerster et al. 2018). There are two players and each player receives a card drawn from $\{\text{card 1}, \text{card 2}\}$ independently at the beginning of the game. Player 1 acts first and Player 2 responds after observing Player 1's action. Neither player can see the other's hand. By the design of the payoff table (shown in Figure. 1a), Player 1 has to use actions C and A to signify that it holds Cards 1 and 2 respectively so that Player 2 can choose its actions optimally with the given information. We compare PBL with algorithms proposed in (Foerster et al. 2018) and vanilla policy gradient. As can be seen from Figure 1b, PBL performs similarly to BAD and BAD-CF on this simple game and outperforms vanilla policy gradient significantly. This demonstrates a proof of principle for PBL in a multi-agent imperfect information coordination game.

Contract Bridge Case-Study

Non-competitive contract bridge bidding is an imperfect information game that requires information exchange between agents to agree high-quality contracts. Hence, such a game serves as an ideal test-bed for PBL. In bridge, two teams of two (North-South vs East-West) are situated in opposing positions and play a trick-taking game using a standard 52-card deck. Following a deal, bidding and playing phases can be effectively separated. During the bidding phase, players sequentially bid for a contract until a final contract is reached. A *PASS* bid retains previously proposed contracts and a contract is considered final if it is followed by three consecutive *PASS* bids. A non-*PASS* bid proposes a new contract of the form $\langle \texttt{integer}, \texttt{suit} \rangle$, where `integer` takes integer values between one and seven, and `suit` belongs to $\{\clubsuit, \diamondsuit, \heartsuit, \spadesuit, \text{NT}\}$. The number of tricks needed to achieve a contract are $6 + \texttt{integer}$, and an NT suit corresponds to bidding to win tricks without trumps. A contract-declaring team achieves points if it fulfills the contract, and if not, the points for the contract go to the opposing team. Bidding must be non-decreasing, meaning `integer` is non-decreasing and must increase if the newly proposed trump suit precedes or equals the currently bid suit in the ordering $\clubsuit < \diamondsuit < \heartsuit < \spadesuit < NT$.

In this work, we focus on non-competitive bidding in bridge, where we consider North (N) and South (S) bidding in the game, while East (E) and West (W) always bid *PASS*. Hence, the declaring team never changes. Thus, each deal can be viewed as an independent episode of the game. The private information of player $i \in \{N, S\}$, x^i, is its hand. x^i is a 52-dimensional binary vector encoding player i's 13 cards. An agent's observation at time step t consists of its hand and the bidding history: $o_t^i = \{x_t^i, h_t^i\}$. In each episode, Players N and S are dealt hands x^N, x^S respectively. Their hands, together, describe the full state of the environment $s = \{x^N, x^S\}$, which is *not fully observed* by either of the two players. Since rolling out via self-play for every contract is computationally expensive, we resort to double dummy analysis (DDA) (Haglund 2010) for score estimation. Interested readers are referred to (Haglund 2010) and the supplementary material for further details. In our work, we use standard Duplicate bridge scoring rules (League 2017) to score games and normalize scores by dividing them by the maximum absolute score.

Benchmarking & Ablation Studies: PBL introduces several building blocks, each affecting performance in its own right. We conduct an ablation study to better understand the importance of these elements and compare against a state-of-the-art method in PQL (Yeh and Lin 2016). We introduce the following baselines:

1. **Independent Player (IP)**: A player bids independently without consideration of the existence of the other player.
2. **No communication reward (NCR)**: One important question to ask is how beneficial the additional communication auxiliary reward r_c is in terms of learning a good bidding strategy. To answer this question, we implement a baseline using the same architecture and training schedule as PBL but setting the communication reward weighting to zero, $\alpha = 0$.
3. **No PBL style iteration (NPBI)**: To demonstrate that

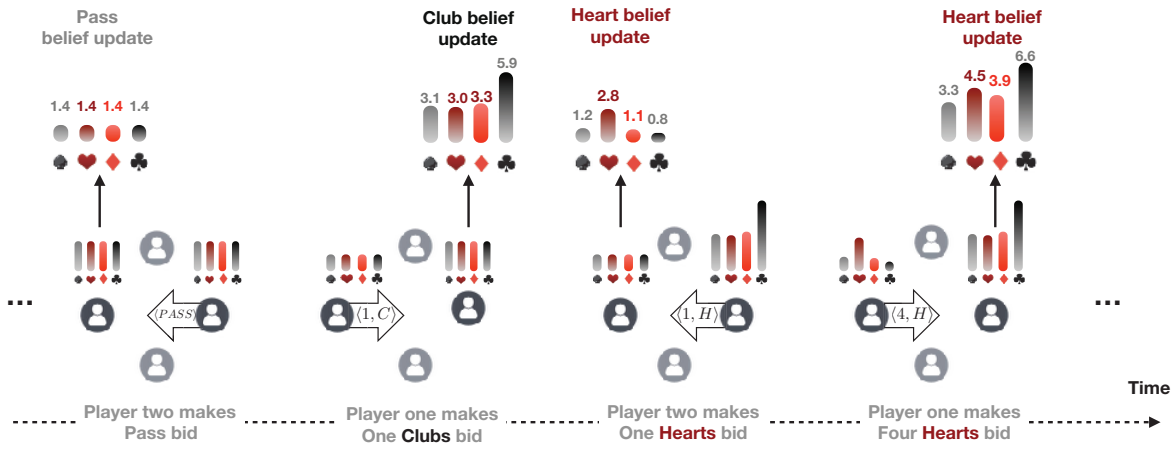


Figure 3: An example of a belief update trace showing how PBL agents use actions for effective communication. Upon observing $\langle PASS \rangle$ from East, West decreases its HCP belief in all suits. When West bids $\langle 1, \mathrm{C} \rangle$, East improves belief in clubs. Next, East bids $\langle 1, \mathrm{H} \rangle$. West recalculates its belief from last time step and increases its HCP belief in hearts.

multiple iterations between policy and belief training are beneficial, we compare our model to a baseline policy trained with the same number of weight updates as our model but no further PBL iterations after training a belief network Φ_0 at PBL iteration $k = 0$.

4. **Penetrative Q-Learning (PQL)**: PQL as proposed by Yeh and Lin (2016) as the first bidding policy for non-competitive bridge bidding without human domain knowledge.

Figure 2a shows the average learning curves of our model and three baselines for our ablation study. We obtain these curves by testing trained algorithms periodically on a pre-generated test data set which contains 30,000 games. Each point on the curve is an average score computed by Duplicate bridge scoring rules (League 2017) over 30,000 games and 6 training runs. As can been seen, IP and NCR both initially learn faster than our model. This is reasonable as PBL spends more time learning a communication protocol at first. However, IP converges to a local optimum very quickly and is surpassed by PBL after approximately 400 learning iterations. NCR learns a better bidding strategy than IP with a belief module. However, NCR learns more slowly than PBL in the later stage of training because it has no guidance on how to convey information to its partner. PBL outperforming NPBI demonstrates the importance of iterative training between policy and belief modules.

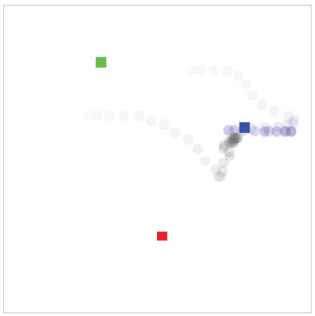
Restrictions of PQL: PQL (Yeh and Lin 2016) is the first algorithm trained to bid in bridge without human engineered features. However, its strong bidding performance relies on heavy adaption and heuristics for non-competitive Bridge bidding. First, PQL requires a predefined maximum number of allowed bids in each deal, while using different bidding networks at different times. Our results show that it will fail when we train a single NN for the whole game, which can been seen as a minimum requirement for most DRL algorithms. Second, PQL relies on a rule-based function for selecting the best contracts at test time. In fact, removing this second heuristic significantly reduces PQL's performance as reported in Figure 2b. In addition, without pre-processing the training data as in (Yeh and Lin 2016), we could not reproduce the original results. To achieve state-of-the-art performance, we could use these (or other) heuristics for our bidding algorithm. However, this deviates from the focus of our work which is to demonstrate that PBL is a general framework for learning to communicate by actions.

Belief Update Visualization: To understand how agents update their beliefs after observing a new bid, we visualize the belief update process (Figure 3). An agent's belief about its opponent's hand is represented as a 52-dimensional vector with real values which is not amenable to human interpretation. Therefore, we use *high card points* (HCPs) to summarize each agent's belief. For each suit, each card is given a point score according to the mapping: **A=4, K=3, Q=2, J=1, else=0**. Note that while agents' beliefs are updated based on the entire history of its opponent's bids, the difference between that agent's belief from one round to the next is predominantly driven by the most recent bid of its opponent, as shown in Figure 3.

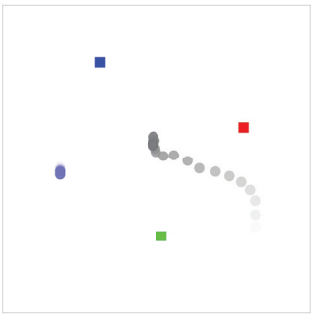
Learned Bidding Convention: Whilst our model's bidding decisions are based entirely on raw card data, we can use high card points as a simple way to observe and summarize the decisions which are being made. For example, we observe our policy opens the bid with 1♠ if it has HCPs of spade 4.5 or higher but lower HCPs of any other suits. We run the model on the unseen test set of 30,000 deals and summarize the learned bidding convention in the supplementary material.

Imperfect Recall of History: the length of the action history players can recall affects the accuracy of the belief models. The extent of the impact depends on the nature of the game. In bridge, the order of bidding encodes important information. We ran an ablation study where players can only recall the most recent bid. In this setting, players do worse (average score 0.065) than players with perfect recall. We conjecture that this is because players can extract less information and therefore the accuracy of belief models drops.

(a) Learning curves for Silent Guide



(b) CR



(c) NCR

Figure 4: a) Learning curves for Silent Guide. Guide agent trained with communication reward (CR) significantly outperforms the one trained with no communication reward (NCR). b) A trajectory of Listener (gray circle) and Guide (blue circle) with CR. Landmarks are positioned randomly and the Goal landmark (blue square) is randomly chosen at the start of each episode. c) A trajectory of Listener and Guide with NCR. Trajectories are presented with agents becoming progressively darker over time.

Silent Guide

We modify a multi-agent particle environment (Lowe et al. 2017) to test the effectiveness of our novel auxiliary reward in a distributed setting. This environment also allows us to explore the potential for implicit communication to arise through machine theory of mind. In the environment there are two agents and three landmarks. We name the agents Guide and Listener respectively. Guide can observe Listener's goal landmark which is distinguished by its color. Listener does not observe its goal. However, Listener is able to infer the meaning behind Guide's actions. The two agents receive the same reward which is the negative distance between Listener and its goal. Therefore, to maximize the cumulative reward, Guide needs to tell Listener the goal landmark color. However, as the "Silent Guide" name suggests, Guide has no explicit communication channel and can only communicate to Listener through its actions.

In the distributed setting, we train separate belief modules for Guide and Listener respectively. The two belief modules are both trained to predict a naive agent's goal landmark color given its history within the current episode but using different data sets. We train both Guide and Listener policies from scratch. Listener's policy takes Listener's velocity, relative distance to three landmarks and the prediction of the belief module as input. It is trained to maximize the environment reward it receives. Guide's policy takes its velocity, relative distance to landmarks and Listener's goal as input. To encourage communication by actions, we train Guide policy with the auxiliary reward proposed in our work. We compare our method against a naive Guide policy which is trained without the communication reward. The results are shown in Figure 4. Guide when trained with communication reward (CR) learns to inform Listener of its goal by approaching to the goal it observes. Listener learns to follow. However, in NCR setting, Listener learns to ignore Guide's uninformative actions and moves to the center of three landmarks. While Guide and Listener are equipped with belief models trained from different data sets, Guide manages to use its own belief model to establish the mental state of Listener and learns to communicate through actions judged by this constructed mental state of Listener. We also observe that a trained Guide agent can work with a naive RL listener (best reward -0.252) which has no belief model but can observe PBL guide agent's action. The success of Guide with CR shows the potential for machine theory of mind. We obtain the learning curves by repeating the training process five times and take the shared average environment reward.

Conclusions & Future Work

In this paper, we focus on implicit communication through actions. This draws a distinction of our work from previous works which either focus on explicit communication or unilateral communication. We propose an algorithm combining agent modeling and communication for collaborative imperfect information games. Our PBL algorithm iterates between training a policy and a belief module. We propose a novel auxiliary reward for encouraging implicit communication between agents which effectively measures how much closer the opponent's belief about a player's private information becomes after observing the player's action. We empirically demonstrate that our methods can achieve near optimal performance in a matrix problem and scale to complex problems such as contract bridge bidding. We conduct an initial investigation of the further development of machine theory of mind. Specifically, we enable an agent to use its own belief model to attribute mental states to others and act accordingly. We test this framework and achieve some initial success in a multi-agent particle environment under distributed training. There are a lot of interesting avenues for future work such as exploration of the robustness of collaboration to differences in agents' belief models.

References

Albrecht, S. V., and Stone, P. 2017. Autonomous Agents Modelling Other Agents: A Comprehensive Survey and Open Problems. *arXiv e-prints*.

Åström, K. J. 1965. Optimal control of Markov Processes with incomplete state information. *Journal of Mathematical Analysis and Applications* 10:174–205.

Baker, M.; Hansen, T.; Joiner, R.; and Traum, D. 1999. The role of grounding in collaborative learning tasks. *Collaborative learning: Cognitive and computational approaches*.

Bard, N.; Johanson, M.; Burch, N.; and Bowling, M. 2013. Online implicit agent modelling. In *Proceedings of the 2013 International Conference on Autonomous Agents and Multi-agent Systems*, AAMAS '13, 255–262.

Bernstein, D. S.; Zilberstein, S.; and Immerman, N. 2013. The Complexity of Decentralized Control of Markov Decision Processes. *arXiv e-prints* arXiv:1301.3836.

de Weerd, H.; Verbrugge, R.; and Verheij, B. 2015. Higher-order theory of mind in the tacit communication game. *Biologically Inspired Cognitive Architectures*.

Dillenbourg, P. 1999. What do you mean by collaborative learning? In Dillenbourg, P., ed., *Collaborative-learning: Cognitive and Computational Approaches.* Oxford: Elsevier.

Dragan, A. D.; Lee, K. C.; and Srinivasa, S. S. 2013. Legibility and predictability of robot motion. In *Proceedings of the 8th ACM/IEEE international conference on Human-robot interaction*, 301–308. IEEE Press.

Eccles, T.; Hughes, E.; Kramár, J.; Wheelwright, S.; and Leibo, J. Z. 2019. Learning reciprocity in complex sequential social dilemmas. *arXiv preprint arXiv:1903.08082*.

Foerster, J. N.; Assael, Y. M.; Freitas, N.; and Whiteson, S. 2016. Learning to communicate with deep multi-agent reinforcement learning. *CoRR* abs/1605.06676.

Foerster, J. N.; Song, F.; Hughes, E.; Burch, N.; Dunning, I.; Whiteson, S.; Botvinick, M.; and Bowling, M. 2018. Bayesian Action Decoder for Deep Multi-Agent Reinforcement Learning. *arXiv e-prints* arXiv:1811.01458.

Gmytrasiewicz, P. J., and Doshi, P. 2005. A framework for sequential planning in multi-agent settings. *J. Artif. Int. Res.*

Hadfield-Menell, D.; Russell, S. J.; Abbeel, P.; and Dragan, A. 2016. Cooperative inverse reinforcement learning. In *Advances in neural information processing systems*, 3909–3917.

Haglund, B. 2010. Search algorithms for a bridge double dummy solver.

He, H.; Boyd-Graber, J.; Kwok, K.; and III, H. D. 2016. Opponent modeling in deep reinforcement learning. In *ICML '16*.

Heider, F., and Simmel, M. 1944. An experimental study of apparent behavior. *The American journal of psychology*.

Jaques, N.; Lazaridou, A.; Hughes, E.; Gulcehre, C.; Ortega, P.; Strouse, D.; Leibo, J. Z.; and De Freitas, N. 2019. Social influence as intrinsic motivation for multi-agent deep reinforcement learning. In *International Conference on Machine Learning*, 3040–3049.

Knepper, R. A.; I.Mavrogiannis, C.; Proft, J.; and Liang, C. 2017. Implicit communication in a joint action. In *Proceedings of the 2017 ACM/IEEE International Conference on Human-Robot Interaction*, HRI '17. ACM.

Lazaridou, A.; Peysakhovich, A.; and Baroni, M. 2016. Multi-agent cooperation and the emergence of (natural) language. *CoRR* abs/1612.07182.

League, A. C. B. 2017. Laws of duplicate bridge.

Lemaignan, S., and Dillenbourg, P. 2015. Mutual modelling in robotics: Inspirations for the next steps. *2015 10th ACM/IEEE International Conference on Human-Robot Interaction (HRI)* 303–310.

Li, X., and Miikkulainen, R. 2018. Dynamic adaptation and opponent exploitation in computer poker. In *Workshops at the 32nd AAAI Conference on Artificial Intelligence*.

Lowe, R.; Wu, Y.; Tamar, A.; Harb, J.; Abbeel, P.; and Mordatch, I. 2017. Multi-agent actor-critic for mixed cooperative-competitive environments. *Neural Information Processing Systems (NIPS)*.

Mealing, R., and Shapiro, J. L. 2017. Opponent modeling by expectation–maximization and sequence prediction in simplified poker. *IEEE Transactions on Computational Intelligence and AI in Games* 9(1):11–24.

Mordatch, I., and Abbeel, P. 2017. Emergence of grounded compositional language in multi-agent populations. *CoRR* abs/1703.04908.

Premack, D., and Woodruff, G. 1978. Does the chimpanzee have a theory of mind? *Behavioral and Brain Sciences* 1.

Rabinowitz, N. C.; P., F.; Song, H. F.; Zhang, C.; Eslami, S. M. A.; and Botvinick, M. 2018. Machine theory of mind. *CoRR* abs/1802.07740.

Raileanu, R.; Denton, E.; Szlam, A.; and Fergus, R. 2018. Modeling others using oneself in multi-agent reinforcement learning. *CoRR* abs/1802.09640.

Rasouli, A.; Kotseruba, I.; and Tsotsos, J. 2017. Agreeing to cross: How drivers and pedestrians communicate. In *Intelligent Vehicles Symposium (IV), 2017 IEEE*. IEEE.

Roth, M.; Simmons, R.; and Veloso, M. 2006. What to communicate? execution-time decision in multi-agent pomdps. In *DARS '06*. Springer. 177–186.

Strouse, D. J.; Kleiman-Weiner, M.; Tenenbaum, J.; Botvinick, M.; and Schwab, D. 2018. Learning to share and hide intentions using information regularization. NIPS'18.

Sukhbaatar, S.; Szlam, A.; and Fergus, R. 2016. Learning multiagent communication with backpropagation. *CoRR*.

Wen, Y.; Yang, Y.; Luo, R.; Wang, J.; and Pan, W. 2019. Probabilistic recursive reasoning for multi-agent reinforcement learning. In *ICLR*.

Yeh, C., and Lin, H. 2016. Automatic bridge bidding using deep reinforcement learning. *CoRR* abs/1607.03290.