Université de Montréal

Emerging Communication Between Competitive Agents

par Mikhail Noukhovitch

Département d'informatique et de recherche opérationnelle
Faculté des arts et des sciences

Mémoire présenté à la Faculté des arts et des sciences
en vue de l'obtention du grade de Maître ès sciences (M.Sc.)
en informatique

Dec, 2019

Université de Montréal
Faculté des arts et des sciences

Ce mémoire intitulé:

Emerging Communication Between Competitive Agents

présenté par:

Mikhail Noukhovitch

a été évalué par un jury composé des personnes suivantes:

Ioannis Mitliagkas,	président-rapporteur
Aaron Courville,	directeur de recherche
Yoshua Bengio,	membre du jury

Mémoire accepté le: .....

Résumé

Nous utilisons l'apprentissage automatique pour répondre à une question fondamentale: comment les individus peuvent apprendre à communiquer pour partager de l'information et se coordonner même en présence de conflits ? Cette thèse essaie de corriger l'idée qui prévaut à l'heure actuelle dans la communauté de l'apprentissage profond que les agents compétitifs ne peuvent pas apprendre à communiquer efficacement. Dans ce travail de recherche, nous étudions l'émergence de la communication dans les jeux coopératifs-compétitifs à travers un jeu expéditeur-receveur que nous construisons. Nous portons aussi une attention particulière à la qualité de notre évaluation. Nous observons que les agents peuvent en effet apprendre à communiquer, confirmant des résultats connus dans les domaines des sciences économiques. Nous trouvons également trois façons d'améliorer le protocole de communication appris. Premierement, l'efficacité de la communication est proportionnelle au niveau de coopération entre les agents, les agents apprennent à communiquer plus facilement quand le jeu est plus coopératif que compétitif. Ensuite, LOLA (Foerster et al., 2018a) peut améliorer la stabilité de l'entraînement et l'efficacité de la communication, principalement dans les jeux compétitifs. Et enfin, que les protocoles de communication discrets sont plus adaptés à l'apprentissage d'un protocole de communication juste et coopératif que les protocoles de communication continus.

Le chapitre 1 présente une introduction aux techniques d'apprentissage utilisées par les agents, l'apprentissage automatique et l'apprentissage par renforcement, ainsi qu'une description des méthodes d'apprentissage par renforcement propre aux systemes multi-agents. Nous présentons ensuite un historique de l'émergence du language dans d'autres domaines tels que la biologie, la théorie des jeux évolutionnaires, et les sciences économiques. Le chapitre 2 approndit le sujet de l'émergence de la communication entre agents compétitifs. Le chapitre 3 présente les conclusions de notre travail et expose les enjeux et défis de l'apprentissage de la communication dans un environment compétitif.

mots-clés: apprentissage profond, apprentissage par renforcement multiagents, émergence de la communication

Summary

We investigate the fundamental question of how agents in competition learn communication protocols in order to share information and coordinate with each other. This work aims to overturn current literature in machine learning which holds that unaligned, self-interested agents do not learn to communicate effectively. To study emergent communication for the spectrum of cooperative-competitive games, we introduce a carefully constructed sender-receiver game and put special care into evaluation. We find that communication can indeed emerge in partially-competitive scenarios, and we discover three things that are tied to improving it. First, that selfish communication is proportional to cooperation, and it naturally occurs for situations that are more cooperative than competitive. Second, that stability and performance are improved by using LOLA (Foerster et al., 2018a), a higher order "theory-of-mind" learning algorith, especially in more competitive scenarios. And third, that discrete protocols lend themselves better to learning fair, cooperative communication than continuous ones.

Chapter 1 provides an introduction to the underlying learning techniques of the agents, Machine Learning and Reinforcement Learning, and provides an overview of approaches to Multi-Agent Reinforcement Learning for different types of games. It then gives a background on language emergence by motivating this study and examining the history of techniques and results across Biology, Evolutionary Game Theory, and Economics. Chapter 2 delves into the work on language emergence between selfish, competitive agents. Chapter 3 draws conclusion from the work and points out the intrigue and challenge of learning communication in a competitive setting, setting the stage for future work.

Keywords: deep learning, multi-agent reinforcement learning, emergent communication

Table des matières

Résumé . . . . . **iii**

Summary . . . . . **iv**

Contents . . . . . **v**

Table des figures . . . . . **vii**

Liste des tables . . . . . **viii**

Liste des abréviations . . . . . **ix**

Acknowledgments . . . . . **x**

1 Introduction **1**
1.1 Machine Learning 1
1.2 Reinforcement Learning 2
1.2.1 Bandits 2
1.2.2 Markov Decision Processes 3
1.2.3 Policy Gradient 4
1.2.4 Deep Reinforcement Learning 5
1.3 Multi-Agent Reinforcement Learning 6
1.3.1 Fully Cooperative MARL 6
1.3.2 Fully Competitive 2-player MARL 6
1.3.3 General-Sum MARL 7
1.4 Language Emergence 9
1.4.1 Motivation 9
1.4.2 Signalling 9
1.4.3 Emergent Communication 10
1.4.4 Divergent Interests 12
1.4.5 Strategic Information Transmission 12
1.4.6 Competitive Emergent Communication 13

2 Selfish Emergent Communication **15**
2.1 Introduction 16

2.2 The Circular, Biased Sender-Receiver Game . . . 17
2.2.1 Description . . . 17
2.2.2 Proof of Purely Cooperative/Competitive Game . . . 18
2.2.3 Training Details . . . 20
2.3 Communication: Cooperation or Manipulation . . . 21
2.3.1 Evaluating Information Transfer . . . 21
2.3.2 Information Transfer vs Communication . . . 22
2.3.3 Cooperation vs. Manipulation . . . 22
2.3.4 Proof of L_2 Fairness . . . 23
2.4 Competitive Selfish Communication . . . 24
2.4.1 Communication Is Proportional To Cooperation . . . 24
2.4.2 Improving Competitive Communication With LOLA-DiCE . . . 25
2.4.3 Discrete vs Continuous Communication . . . 27
2.5 Extra Plots . . . 30

3 Conclusion . . . 36

Bibliography . . . 37

Table des figures

1.1 Basic Signalling Game 10

2.1 Circular Biased Sender-Receiver Game 17
2.2 Circular Biased Sender-Receiver Game with Bias = 180° 19
2.3 Competitive Communication Results using REINFORCE 25
2.4 Competitive Communication Results Using LOLA 26
2.5 Competitive Communication using Discrete vs. Continuous Channel 29
2.6 REINFORCE Sender, Deterministic Receiver, L_1 hyperparameter search 30
2.7 REINFORCE Sender, Deterministic Receiver, L_2 hyperparameter search. Note these are identical to Figure 2.6 except for $b = 150°$ . . 31
2.8 DiCE Sender, LOLA-1 Receiver 31
2.9 LOLA-1 Sender, Deterministic Receiver 32
2.10 LOLA-1 Sender, LOLA-1 Receiver 32
2.11 LOLA-2 Sender, LOLA-2 Receiver 33
2.12 LOLA-3 Sender, LOLA-3 Receiver 33
2.13 LOLA-4 Sender, LOLA-4 Receiver 34
2.14 Gaussian Sender. Deterministic Receiver playing the continuous game 34
2.15 Gaussian LOLA Sender. LOLA Receiver playing the continuous game 35
2.16 REINFORCE Sender, Deterministic Receiver for $b = 180°$ 35

Liste des tableaux

1.1 Example rewards for a single round of Prisoner's Dilemma 8

Liste des abréviations

AI	Artificial Intelligence
DiCE	Infinitely Differentiable Monte Carlo Estimator (Foerster et al., 2019)
LOLA	Learning with Opponent Learning Awareness (Foerster et al., 2018a)
MARL	Multi-Agent Reinforcement Learning
MDP	Markov Decision Process
ML	Machine Learning
MLP	Multi-Layer Perceptron aka fully-connected NN
NN	Neural Network
ReLU	Rectified Linear Unit (Nair and Hinton, 2010)
RL	Reinforcement Learning (as used by Sutton and Barto (2018))
SGD	Stochastic Gradient Descent

Acknowledgments

This thesis is dedicated to my friends, my family, and everyone else that selflessly tried to make me a better researcher and a better person. I am grateful, and I hope you don't regret wasting your time.

Just kidding. Thank you, I could not have done it without you.

1 Introduction

1.1 Machine Learning

Artificial Intelligence (AI) is a subfield of computer science that aims to design machines that act and/or think intelligently as defined by either the yardstick of human intelligence or some rational, correct action (Russell and Norvig, 2016). This thesis deals with a subfield of AI, Machine Learning, the study of algorithms that learn from data. In classical programming, we explicitly define rules that are applied to input data in order to achieve some effect. In machine learning, we instead observe data and improve the efficacy of our programs by learning computations based on what we observe.

Machine learning is generally split up into three categories that correspond to the signal given by the available input or *training* data. In **supervised learning**, we are given pairs of inputs and targets (or *labels*), where the goal is to learn the true mapping of inputs to targets. This covers tasks such as classification, if the targets follow a categorical distribution, and regression if the targets follow a continuous distribution. In **unsupervised learning**, we are only given inputs and the goal is to uncover underlying structures or patterns in the data. This covers tasks such as density estimation, where we estimate the distribution of the data, and clustering, where we try to group the data into a set of discrete categories. The final category of machine learning is what this thesis concerns itself with: **reinforcement learning**, where the data is an interactive environment that gives rewards for achieving certain goals. A common example is a video game where the task is to control the player and the reward is winning the game.

1.2 Reinforcement Learning

The term "Reinforcement Learning" (RL) has been used by various fields to describe many different techniques (Herrnstein, 1961) but we will use it in the way of Sutton and Barto (2018). Specifically, with RL we refer to the subfield of machine learning concerned with decision making in an environment guided by obtaining future rewards.

1.2.1 Bandits

The simplest RL environment is a **bandit** as exemplified by the classic problem of slot machines also known as "one-armed bandits". An agent is presented with some number of slot machines and at each time step t must choose a machine for which to pull the lever. We can consider our choice of machine as an action a so that the space of possible actions $\mathbb{A}$ is just the set of machines. For a lever pull, each machine has some distribution of winnings which we term **reward** $r \in \mathbb{R}$ and since this depends on the machine we chose, we say that the reward at each time step depends on the action $R_t : A_t \to \mathbb{R}$. The agent has some number of time steps ($T \in [1, \infty)$) and wishes to maximize their total reward $\sum_{t=0}^{T} r_t$. We can formally define the goal or objective we wish to maximize J

$$J = \sum_{t=0}^{T} \gamma^t R_t \tag{1.1}$$

where we can use a discount factor $\gamma \in (0, 1]$ to avoid an infinite sum in infinite-length environments. We term the discounted sum of future rewards to be the **return** $G_t = \sum_{k=0}^{T} \gamma^k R_{t+k+1}$.

To solve this problem, our goal is to come up with way of acting, a **policy** π, that chooses the best action at every time step. We can consider a policy to be a stochastic distribution over actions (machines). Looking for the best policy leads to the classic problem of balancing *exploration*, finding the machine with the best expected reward, and *exploitation*, maximizing the total reward by choosing the machine you currently think is best. Since our policy is stochastic, we can define the best policy as the one with the highest return *in expectation*

$$J(\pi) = E_{a \sim \pi}[G_1] \tag{1.2}$$

In bandits, this is equivalent to choosing the optimal action $A_t = a^*$

1.2.2 Markov Decision Processes

Apart from just rewards and actions, we can make our models more realistic by adding a **state** and if we make the Markov assumption about our states this makes our model a **Markov Decision Process** (MDP).[1] This is the full reinforcement learning problem in that we now have a state $s \in \mathcal{S}$ that can affect the possible actions available $A(s) \in \mathcal{A}$, a reward distributions for each state-action $R(s, a) \in \mathcal{R}$ and a transition distribution $P(s'|s, a)$ that accounts for the changing states by giving us the probability of a next state given the current state and action.

Therefore, we also change our policy to be dependent on the state $\pi(a|s)$. We can now also talk about the return from a state by calculating the actions and subsequent states. We call this the **value function** V of that state and using the transition function we can define it recursively.

$$V^\pi(s) = \mathbb{E}_{a\sim\pi}[G_t|S_t = s] \tag{1.3}$$
$$= \mathbb{E}_{a\sim\pi}\big[R_t + \gamma R_{t+1} + \gamma^2 R_{t+2} + \ldots\big] \tag{1.4}$$
$$= \mathbb{E}_{a\sim\pi}[R_t + \gamma V^\pi(s')] \tag{1.5}$$

Similarly we can define an action-value function that conditions on the current action as well $Q^\pi(s, a) = E_{a\sim\pi}[G_t|S_t = s, A_t = a]$. With this, we can rewrite our objective over states instead of over time. For the probability of each state we can use the stationary distribution of the markov chain $d^\pi(s)$.

1. The specific setup in Chapter 2 is only one time step long and can actually be modelled as a **contextual bandit**. Contextual bandits differ from MDPs in that you model actions as not affecting which future state is chosen and therefore the transition distribution $P(s'|s)$ does not depend on actions. This formulation is equivalent to the MDP for the task in Chapter 2 so to keep with standard RL explanations we use an MDP.

$$J(\pi) = \sum_{s \in S} d^{\pi}(s) V^{\pi}(s) \tag{1.6}$$

$$= \sum_{s \in S} d^{\pi}(s) \sum_{a \in A} \pi(a|s) Q^{\pi}(s, a) \tag{1.7}$$

$$\tag{1.8}$$

In practice, for finite-length MDPs we can sometimes consider $J(\pi)$ to marginalize over initial states and use the probability of initial states instead of the stationary distribution.

1.2.3 Policy Gradient

Now that we have an objective, we can treat this as an optimization problem to find the optimal policy π^*. In many cases, our policy is a function π_θ that is parametrized by some θ and finding the optimal policy means finding the optimal θ. We can more cleanly represent our objective $J(\pi_\theta)$ as $J(\theta)$

$$\theta^* = \max_{\theta} J(\theta) \tag{1.9}$$

One way to optimize this is by changing the parameters in a direction that improves our objective. A simple way is to follow the gradient of the objective using *gradient descent*, this is known as **policy gradient**. We can update our θ after each finished game (called an *episode*) by moving it a small step (or **learning rate**) α along the gradient of the objective. Since the full objective is not easily differentiable with respect to the policy parameters, we use the policy gradient theorem to reformulate it

$$\nabla_\theta J(\theta) = \nabla_\theta \sum_{s \in S} d^{\pi}(s) \sum_{a \in A} \pi_\theta(a|s) Q^{\pi}(s, a) \tag{1.10}$$

$$\propto \sum_{s \in S} d(s) \sum_{a \in A} Q^{\pi}(s, a) \nabla_\theta \pi_\theta(a|s) \tag{1.11}$$

$$= E_{s \sim d^{\pi}, a \sim \pi_\theta}[Q^{\pi}(s, a) \nabla_\theta \ln \pi_\theta(s|a)] \tag{1.12}$$

since $Q^\pi(s,a) = E_{a\sim\pi}[G_t|S_t = s, A_t = a]$

$$= E_{s\sim d^\pi, a\sim\pi_\theta}[G_t \nabla_\theta \ln \pi_\theta(s|a)] \tag{1.13}$$

One way we can measure G_t is empirically using Monte-Carlo sampling. By averaging the returns from different episodes, we can get a unbiased sample of G_t. Doing policy gradient in this way is known as **REINFORCE** (Sutton and Barto, 2018) (and elsewhere known as the score function estimator (Fu, 2006)). A common trick to improve performance is to subtract a baseline value from G_t to try and reduce the variance of the estimate while keeping the bias unchanged at 0.

1.2.4 Deep Reinforcement Learning

Deep learning (Goodfellow et al., 2016) has enjoyed much success in recent years (LeCun et al., 2015) so many RL models now use deep neural networks as powerful function approximators. One simple way is to use a deep neural network as the policy π. Since deep nets can be updated by backpropogation using a gradient (Rumelhart et al., 1986), they are easily integrated into policy gradient models. Our parametrized policy π_θ is therefore a network with weight parameters θ.

In practice, we can facilitate better learning by training on batches (allowing for a speed up by exploiting parallelization) and updating our parameters using **stochastic gradient descent** (SGD). With neural networks, we introduce variance in the form of initialization and *hyperparameters*, parameters we fix before training e.g. the learning rate α. Neural networks, particularly when used in RL, are sensitive to how they are randomly initialized and which hyperparameters are used (Henderson et al., 2018) so there are two common techniques to improve robustness: **random seed replicates** and **hyperparameter search**. To deal with variance in initialization, it is standard to replicate an experiment using a couple different random seeds thus varying the initialization and reporting averages and standard deviations across replicates to show robustness as well as performance. To deal with hyperparameter sensitivity, we run hyperparameter searches by randomly sampling from distributions of parameters (*random search*) [2]

2. Trying all possible combinations of parameters (*grid search*) is, in practice, no more effective than random sampling but much less efficient for neural networks (Bergstra and Bengio, 2012). Recent approaches also use population-based evolutionary learning during training to tune

1.3 Multi-Agent Reinforcement Learning

RL is further complicated when the environment is *non-stationary*: the reward distribution changes over time even when conditioning on an action and state. Essentially, non-stationary environment have factors in them that change over time and that we are not explicitly accounting for, making it more difficult to learn the best action to take. The non-stationarity discussed in this thesis is due to the presence of other agents, and this problem is known as **multi-agent reinforcement learning** (MARL).

In MARL, multiple agents are all acting and possibly also learning from their actions. The type of learning and evaluation necessary for MARL strongly depends on the game and its dynamics. We outline three scenarios that are relevant to the work in Chapter 2.

1.3.1 Fully Cooperative MARL

If all agents share the exact same goal (e.g. receive the same reward) then the game is fully cooperative and the aim is to achieve the maximum expected return much like single-agent RL. But unlike a single player that can coordinate all its actions seamlessly, cooperative MARL agents' main challenge is to coordinate their actions. In this way, cooperative MARL challenges usually involve games of difficult coordination such as the card game Hanabi (Bard et al., 2019b) or controlling two players in a Bomberman-like game (Resnick et al., 2018).

Since the goal is simply the maximum reward, in many cases we can choose our own team (Foerster et al., 2018b) and train in *self-play* – where all agents are differently initialized versions of the same architecture – and the problem becomes choosing the best *agent architecture* and training it in any reasonable way to achieve the highest return. This is similar to single-agent RL and so we can use some of the same optimization tricks and strategies (such as hyperparameter search).

1.3.2 Fully Competitive 2-player MARL

In contrast, 2-player fully-competitive games have agents trying to achieve completely opposing goals. These games are known in game theory as **constant sum**

hyperparameters (Jaderberg et al., 2017)

(a generalization of zero-sum) because the sum of rewards is constant and therefore any reward an agent receives is a reward it takes from its opponent. This class of games covers many modern strategy games such as chess, go (Silver et al., 2017), and starcraft (Vinyals et al., 2019).

Since agents are fully opposed, the move that is best for an agent is worst for its opponent. In this way any finite zero-sum game can be modelled as a minimax game where one agent tries to maximize it's reward and the other tries to minimize it (Von Neumann, 1928). Agents can therefore make the assumption that the opponent would take the move that will leave the agent worst off. This evaluation can be simplified by assuming that the agents' own evaluation of the game is equivalent to the opponent's and therefore they can use their own evaluation with a flipped sign as a model of their opponent's strategy.

However unlike cooperative MARL, choosing our own opponents is not an assumption we can make. Instead, the goal of competitive MARL is to find an agent that can outplay all opponents (Balduzzi et al., 2018) i.e. the best agent in the whole space of possible agents. Since playing against all possible strategies is infeasible, we must estimate which agent is the best from games against a sample of agents and even more importantly we must choose the space of opponents (Shoham et al., 2003). But even then simple self-play may not necessarily lead to better and better agents if the game is **non-transitive** such as rock-paper-scissors (Balduzzi et al., 2019). For some non-transitive games, we must demonstrate superiority against a large diverse selection of possible opponents (Vinyals et al., 2019) and to do that, we must play against opponents we have not trained with – **ad-hoc** play. This means that competitive games compare *learned agents*, not just architectures, and the goal is to find the best parameters as well as architecture.

1.3.3 General-Sum MARL

In Chapter 2, we investigate the partially competitive space between constant-sum and fully-cooperative games known as **general-sum games** where there is some amount of common interest and some amount of conflict. In this case, care must be taken in defining the "best" agent: it is not necessarily the agent that does as well or better than all opponents because agents must now maximize not only the conflict but also common interest. To achieve the highest possible reward in

a general-sum game, agents might have to cooperate to some extent (Leibo et al., 2017). For example, consider an agent playing *iterated prisoner's dilemma*, a game consisting of many sequential rounds of a prisoner's dilemma, see table 1.1. The agent that always defects will never have a reward worse than its opponent, but it will also not have a high reward against an intelligent opponent that also chooses to defect. A tit-for-tat agent that copies its opponent's previous move will achieve a similar reward against a defector but achieve a much higher reward against an agent that chooses to cooperate.

		Player 1	
		Cooperate	Defect
Player 2	Cooperate	3,3	1,4
	Defect	4,1	2,2

Table 1.1 – Example rewards for a single round of Prisoner's Dilemma

Since the maximum expected reward may only be possible by cooperating, agents must learn how to coordinate with each other. This can be done by training together or learning to understand opponent intentions at test time by observing their actions. The latter allows for ad-hoc comparison of *learned agents* at test time ; the former requires comparing *learning algorithms* trained together. The latter should then require a sequential/iterative game, so there is time to infer opponent intentions before acting (Fujimoto and Kaneko, 2019). But in MARL, there is no single standard for the *warmup time* needed to acclimate to an opponent (and facilitate coordination). Furthermore, agent architectures may also require meta-learning or other modifications as current self-play methods are insufficient to adapt to ad-hoc play even against different parametrizations of their own architecture (Bard et al., 2019a). Instead, work in general-sum MARL has mostly been on analysing *learning algorithms'* ability to cooperate and resolve social dilemmas (Foerster et al., 2018a; Letcher et al., 2019; Lerer and Peysakhovich, 2017). It is clear that general-sum games provide a complex learning ground where evaluation techniques are highly dependent not only on the games themselves but also on the questions being asked (Shoham et al., 2003). This thesis deals with the question of language and how it is emerged.

1.4 Language Emergence

1.4.1 Motivation

"How does an effective communication system arise among a collection of initially noncommunicating individuals?" (Wagner et al., 2003). This question is at the core of all research in language emergence across the fields of anthropology, linguistics, machine learning, and more. Motivations for studying this question can generally be broken down into two camps. The first is the scientific question of how animal communication and even human language has come about. Studying this generally involves modelling real world constraints and environments to see what factors are important in language emergence. The second is the engineering perspective that understanding the fundamentals of communication can help us improve communication between software systems (e.g. networking protocols) and in multi-agent interactions (e.g. coordinating MARL (Foerster et al., 2016)).

This thesis looks at learning communication between self-interested agents in a competitive environment. This question could be motivated by the former (e.g. animals communicate but can not be perfectly aligned) but is more applicable to the latter. Specifically, it is clear that both competition and cooperation are necessary in many real world multi-agent games (e.g. Risk, Settlers of Catan) and situations (e.g. salary negotiation). Furthermore, we cannot be sure to limit agent coordination to existing human protocols. So it is reasonable that future agents acting in the real world will have to coordinate and communicate with others agents that have different goals and targets. We wish to explore whether communication protocols can be learned for those situations and what properties we can imbue into the protocols and therefore into the interactions.

1.4.2 Signalling

Previous work on the evolution of communicative capacities has generally taken advantage of the signalling-game framework also known as **sender-receiver games**. Signalling games are a class of two-player games of incomplete information first introduced by Lewis (1969) to explore learning meaning by convention from initially random signals. In the basic signalling game, signals are used by an agent to disambiguate from a number of possible referents. There are two agents, called

the *sender* and *receiver* and some state of the world. It may consist of an external state, such as a predator being present in the case of animal signalling, or it may consist in some internal state of the sender, e.g their intention or preference. The sender observes the state of the world and sends a message to the receiver; the receiver observes the message, but cannot observe the state of the world. The receiver takes an action where each action is appropriate for a single state. In this basic game, there is only one action for each of the states which provides a positive payoff to the sender and receiver (see Fig 1.1)

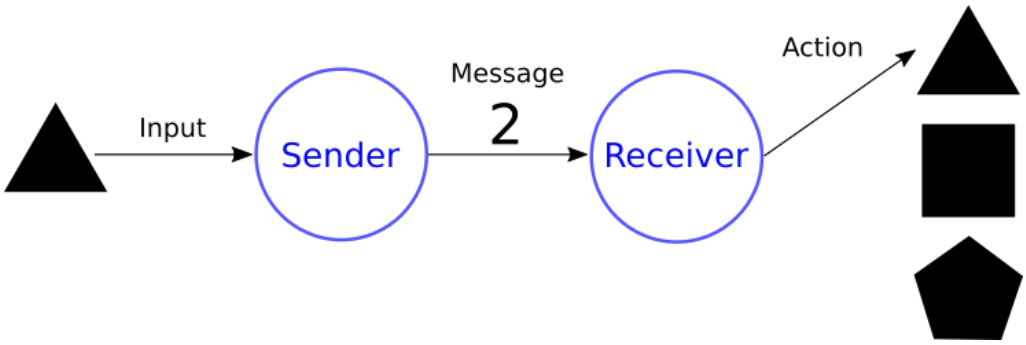


Figure 1.1 – Basic Signalling Game example where the sender is given one of three shapes and the receiver must guess the shape given only the sender's message

The signalling game has been modelled using evolutionary game theory to explain how meaning emerges via population dynamics or low-rationality learning dynamics (Skyrms, 2014/1996, 2010). This setup is used widely in economics (Riley, 2001), philosophy (Lewis, 1969; Skyrms, 2010), evolutionary biology (Zahavi, 1975; Spence, 1973), and political science (Banks, 1991), among others.

1.4.3 Emergent Communication

In recent years, the signalling game and language emergence have been tackled by more powerful learning algorithms and techniques, namely deep learning and deep reinforcement learning, and this field has been termed **emergent communication**. Beginning with Foerster et al. (2016); Sukhbaatar et al. (2016), the field of language emergence was brought into the fold of modern machine learning methods. Initially, setups had tightly-integrated agents trained together (Sukhbaatar et al., 2016) possibly passing gradient as well as messages (Foerster et al., 2016). Follow-up work focused on more natural communication using more decoupled execution

(Lowe et al., 2017), learning from images to capture linguistic properties (Lazaridou et al., 2016), changing communication to use variable-length sequences of discrete tokens (Havrylov and Titov, 2017), and allowing for multi-turn communication (Evtimova et al., 2017). Following claims of approaching natural language using these setups (Mordatch and Abbeel, 2018), Kottur et al. (2017) showed that many of these setups were contrived and brittle, demonstrating that an emerged natural language was much more difficult than it seemed.

Next works therefore focused on emergent communication multi-agent coordination in simple shape-based 2D games (Lowe et al., 2017), classic games such as Bomberman (Resnick et al., 2018), multi-step negotiation (Cao et al., 2018). With the increase of work in the field and complexity of environment, Lowe et al. (2019) pointed out common issues that trivialized results and set a simple but effective standard of measuring successful communication: if returns using a communication channel exceed those without a communication channel.

Despite the progress, one assumption consistently made was that emergent communication is a task between purely cooperative agents Lanctot et al. (2017) with no works exploring agents under conflict of interest until Cao et al. (2018). Singh et al. (2018) claimed to learn in mixed cooperative-competitive scenarios. However, their setup uses parameter sharing between opponents; their "mixed" case is non-competitive (and implicitly cooperative); their competitive game is actually two stages—one fully cooperative and one fully competitive; and their results in the competitive scenario are simply to mask out all communication. Overall, we cannot confirm their claim of studying competitive emergent communication nor the significance of their results. Cao et al. (2018) did test competitive agents but found that successful communication did not arise and argue that communication under competition was not possible. Later Jaques et al. (2019) argued that successful communication between agents in a partially competitive game (Leibo et al., 2017) did not arise without their complex learning rule. In both cases, the level of competition between agents was not quantified and the prevailing notion is that communication does not arise between competitive agents.

1.4.4 Divergent Interests

Though language emergence in machine learning has not strongly investigated cooperation as a factor, this study has a long history in biology, evolutionary game theory, and classical games. Lewis (1969) initial formulation of the signalling game assumes *pure cooperation* but it builds off the work of Shelling (1960), who noted that games should be understood to range on a spectrum—with games of *pure cooperation* on the one end and games of *pure-conflict* (zero-sum games) on the other.

In biology, empirical evidence in nature also demonstrates communication differing between levels of closeness and cooperation. For example, vervet monkeys will not produce alarm calls when there are no other monkeys present (Cheney and Seyfarth, 1985). Similarly, ground squirrels will only produce calls when kin are present (Sherman, 1977). In evolutionary game theory, Skyrms (2010) looks at a small number of signalling games where the players' interests are imperfectly aligned and Wagner (2012, 2014) shows that it is still technically possible for meaning to be conveyed in a zero-sum game, though the resultant dynamics will be chaotic. Martínez and Godfrey-Smith (2016) investigated a dynamic analysis of signalling with conflict of interest but limited their learning to using the *replicator dynamic* (Taylor and Jonker, 1978). Research in economics has investigated how *costly* signalling can stabilize communication in competitive environments (Spence, 1973).

1.4.5 Strategic Information Transmission

Work in emergent communication – and this work in question – is most similar to the economic concept of **cheap talk** (Crawford and Sobel, 1982; Farrell and Rabin, 1996). This is a model of communication that is costless, non-binding (does not limit strategic choices after speaking), and unverifiable (utterances can't be verified to be true). Relating these qualities to environments in RL, we can simply say that agent utterances do not directly influence the environment (i.e. the transition function or the reward function) and *may* only influence the beliefs of other agents. Therefore, the prevailing view of emergent communication under competition being impossible seems at odds with Strategic Information Transmission (Crawford and Sobel, 1982) – a seminal work in classical game theory – that proves possible cheap

talk communication protocols under competition. Crawford and Sobel (1982) study possible *fixed* communication equilibria under competition by giving the sender s and receiver r targets that differ by some bias b and creating a conflict of interest. Translating their setup to RL reward functions:

$$R_s(s, a) = -(a - s - b)^2 \tag{1.14}$$

$$R_r(s, a) = -(a - s)^2 \tag{1.15}$$

for some state $s \in \mathbb{R}$. Their focus is on static analysis and finding Nash equilibria — points where neither agent can improve by changing only their own strategy. They show there exist equilibria where the amount of information communicated is proportional to the alignment between the players' interests ($\frac{1}{b}$); however no informative equilibrium exists when interest diverge too greatly. Though they show that informative (and therefore successful) communication is possible under a conflict of interest, there are strong assumptions that come with classical game theory. First and foremost, the proof is not *feasibility* to achieve communication but simply *existence* of a Nash equilibrium where informative communication exists. In neural networks, this is equivalent to proving a point of convergence and its properties but having no guarantees about whether that point is feasibly achieved with SGD. Secondly, game theory assumes perfectly rational agents that know the payoffs of their opponents as well as the structure of the game. In RL parlance, that is equivalent to knowing the reward function of all agents as well as the transition function. Indeed, this is necessary in order for an agent to know they are in a Nash equilibrium and not diverge from it. This assumption is at odds with the fundamental approach to RL and is one of the reasons why MARL should not aim to converge to equilibria (Shoham et al., 2003).

1.4.6 Competitive Emergent Communication

We wish to come at this problem from the perspective of RL where agents learn through trial and error without being given a model of the world or other agents. Therefore, this work focuses on a dynamic analysis, not a static analysis, and aims to show the practical *feasibility* of learning communication under a conflict of interest. Following Shoham et al. (2003) and recent work in emergent communication

(Jaques et al., 2019) as well as evolutionary signalling (Skyrms, 2010) we do not explicitly aim for equilibrium but look at average information transfer of the dynamical system. The question of MARL evaluation raised in section 1.3.3 is even more difficult for emergent communication. Agents that are not trained together could not reasonably communicate with each other and even current approaches to meta-learning communication are still insufficient to adapt to new protocols (Ryan Lowe, 2020). Therefore, in order to investigate feasibility of language emergence it is reasonable to follow Cao et al. (2018); Jaques et al. (2019) and evaluate *agent architectures* trained together. Yet an issue with Cao et al. (2018); Jaques et al. (2019) was the lack of quantification of competitiveness, so following Lowe et al. (2019), it is necessary to guarantee that all communication is through the communication channel. We would like to avoid that agents *situated* in an environment with non-verbal actions bypass communication and implicitly coordinate through that environment (e.g. running towards the opponent's goal in soccer to communicate your teammate should pass you the ball). This is easily accomplished by having *non-situated* agents that can only interact through the communication channel. In this way, we have outlined the qualities necessary for a setup investigating practical feasibility and learnability of communication to also be as quantifiable and controllable.

2 Selfish Emergent Communication

Authors: Michael Noukhovitch*, Travis LaCroix, Angeliki Lazaridou, and Aaron Courville.

Contributions: MN conceptualized the idea, wrote all the code, performed all the experiments, managed the project schedule, ran all the meetings, proposed TL and AL as collaborators, wrote the first draft and the final paper. TL advised with domain knowledge in game theory (classical and evolutionary) and philosophy of language, proposed solutions to roadblocks, refined the idea to meet game theoretic standards, wrote a first draft of the paper, and wrote and edited the final paper. AL advised with domain knowledge in emergent communication and multi-agent RL, proposed solutions and best practices, and gave clarity on research direction. AC supervised the project, helped define the scope, refined the idea, provided in-depth feedback on encountered issues, checked calculations, proposed new avenues of research, and helped write and edit the final paper.

Affiliation

— Michael Noukhovitch, Mila, Université de Montréal
— Travis LaCroix, University of California Urvine, Mila
— Angeliki Lazaridou, Deepmind
— Aaron Courville, Mila, Université of Montréal

2.1 Introduction

The principles involved in the evolution of effective communication are essential for artificial intelligence research since they may lead to innovative communication methods for use by interacting AI agents and multi-robot systems. AI agents need a common language to coordinate with one another and to communicate successfully with humans (Wagner et al., 2003). The emergence of communication protocols between learning agents has seen a surge of interest in recent years, but most work tends to study fully-cooperative agents that share a reward (Foerster et al., 2016; Havrylov and Titov, 2017; Lazaridou et al., 2016). Work on selfish agents, that separately optimise their own reward, has been limited, with results suggesting that selfish agents do not learn to use a communication channel effectively (Cao et al., 2018; Jaques et al., 2019). This has contributed to a perspective in the field that emergent communication is a purely cooperative pursuit Lanctot et al. (2017). This is in contrast to theoretical results in game theory that show it is possible to use an existing cheap-talk communication channel effectively to resolve situations of partial conflict (Farrell and Rabin, 1996). We aim to reconcile these different findings and establish the degree of cooperation necessary for useful communication to emerge.

To study this problem in detail, we look at the simplest case of communication: a sender-receiver game (Lewis, 1969). This is a game of incomplete information, wherein a sender obtains private knowledge and communicates this to a receiver via a signal, or message. The receiver uses the message to inform its action in the environment. Though messages are arbitrary, and initially meaningless, the players can coordinate upon a conventional meaning for the signal (Skyrms, 2010). While the classic game is fully cooperative, we introduce an arbitrary level of conflict between our two players and investigate whether communication can emerge for each level of conflict. Contrary to current literature in machine learning, we find evidence that communication emerges in competitive scenarios—provided that the agents' interests are at least partially aligned. We further find that using LOLA (Foerster et al., 2018a)—an effective strategy for resolving social dilemmas—to explicitly model opponent learning yields more effective communication protocols. Finally, we consider the difference between continuous and discrete emergent communication and find that discrete communication lends itself better to cooperative

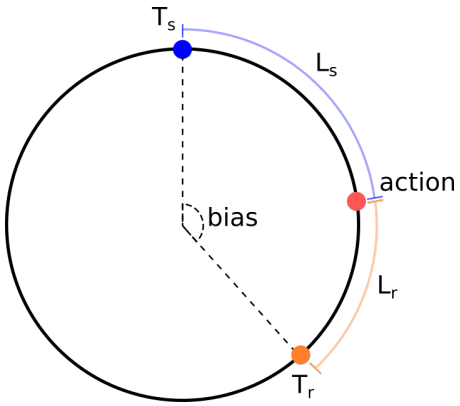


Figure 2.1 – Circular Biased Sender-Receiver Game has both agents given targets T_R, T_S that are b apart and choose an action a to receive the L_1 losses L_1^r, L_1^S

communication.

2.2 The Circular, Biased Sender-Receiver Game

2.2.1 Description

To investigate a range of competitive scenarios, we introduce a modified sender-receiver game with a continuous-bias variable, b, that represents the agents' conflict of interest, ranging from fully cooperative to fully competitive. The two players—the Sender (S) and the Receiver (R)—have corresponding targets (T_s and T_r), which are represented by angles on a circle that are b degrees apart: $T_r = (T_s + b) \mod 360°$.

The game starts with the sender's target being sampled uniformly from the circle $T_s \sim \text{Uniform}[0, 360)$. The sender is given its target as input and outputs a message, $m = S(T_s)$, consisting of a single, discrete token from a vocabulary $m \in V$. The receiver is given the message and outputs a scalar action $a = R(m)$. The goal of each agent is to make the receiver's action as close as possible to its *own* target value. After the receiver acts, both players get a loss between the action and their respective targets, $L^i = L(a, T_i)$. In this way, the sender can implicitly see the action of the receiver by seeing its effect. By using an L_1 loss between the angle of the target and action $L_1^i(T_i, a) = \min(|T_i - a|, 360° - |T_i - a|)$, it is

evident that a game with $b = 0°$ is fully cooperative ($L_1^r = L_1^s$) and a game with the maximum bias $b = 180°$ is fully competitive or constant-sum (a generalisation of zero-sum), see proof in section 2.2.2. All values in-between, $b \in (0°, 180°)$, represent the spectrum of partially cooperative/competitive *general-sum* games. Figure 2.1 gives an instance of this game; the game's algorithm is given in Algorithm 1. This can be seen as the game from Crawford and Sobel (1982) modified to cover the range of cooperative/competitive games.

Algorithm 1 Circular Biased Game Round

procedure TRAINING BATCH(b)
- $T_s \sim \text{Uniform}(0, 360)$
- $T_r \leftarrow T_s + b$
- $m \sim \text{Categorical}(S(T_s))$
- $a \leftarrow R(m)$
- $L_s \leftarrow L_1(T_s, a) = \min(|T_s - a|, 360 - |T_s - a|)$
- $L_r \leftarrow L_1(T_r, a) = \min(|T_r - a|, 360 - |T_r - a|)$
- R is updated with SGD
- S is updated with REINFORCE or DiCE

2.2.2 Proof of Purely Cooperative/Competitive Game

For $b = 0$, $T_s = T_r$ so trivially $L_s = L_r$ and the game is fully cooperative.

For $b = 180°, T_r = T_s + 180 \mod 360°$ we provide a visual demonstration in Figure 2.2 that the sum is always $L_s + L_r = 180°$ and therefore the game is constant-sum and fully competitive. We can also think of this as moving the actions distance d towards one agent's target means moving it distance d away from the other agent's target.

Aside from visual, we provide a formal proof of a constant-sum game. We know that $0 \leq T_r, T_s, a \leq 360°$. Assume without loss of generality $T_s < T_r$ so $T_r = T_s + 180°$ and $T_s \leq 180° \leq T_r \leq 360°$

$$\begin{aligned} L_s + L_r &= L_1(T_s, a) + L_1(T_r, a) \\ &= \min(|T_s - a|, 360° - |T_s - a|) + \min(|T_r - a|, 360° - |T_r - a|) \end{aligned}$$

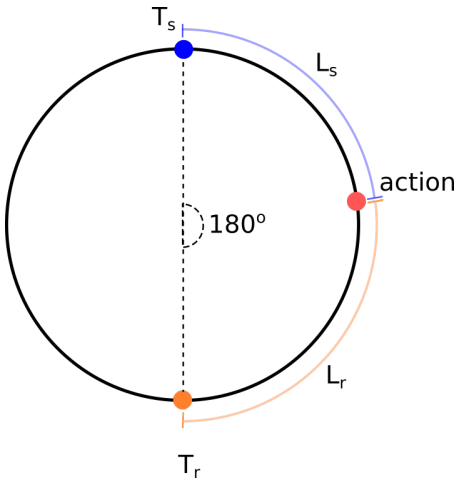


Figure 2.2 – The game with maximal bias 180° showing the sum of L_1 losses $L_r + L_s = 180°$

case 1: $|T_s - a| < 360° - |T_s - a|$

$$\begin{aligned} \min(|T_s - a|, 360° - |T_s - a|) &= |T_s - a| \\ |T_s - a| &< 180° \end{aligned}$$

subcase a: $T_s >= a$

$$\begin{aligned} |T_s - a| &= T_s - a \\ \because T_r &= T_s + 180° \\ T_r &\geq a + 180° \\ \therefore \min(|T_r - a|, 360° - |T_r - a|) &= 360 - (T_r - a) \\ L_s + L_r &= T_s - a + 360° - (T_r - a) \\ &= T_s - a + 360° - T_s - 180° + a \\ &= 180 \end{aligned}$$

subcase b: $T_s < a$

$$
\begin{aligned}
|T_s - a| &= a - T_s \\
a - T_s &< 180^\circ \\
a &< T_s + 180^\circ \\
a &< T_r \\
\therefore |T_r - a| &= T_r - a \\
\because T_r &= T_s + 180^\circ \\
T_r &< a + 180^\circ \\
T_r - a &< 180^\circ \\
2(T_r - a) &< 360^\circ \\
T_r - a &< 360^\circ - (T_r - a) \\
\therefore \min(|T_r - a|, 360^\circ - |T_r - a|) &= T_r - a \\
\therefore L_s + L_r &= (a - T_s) + (T_r - a) \\
&= T_r - T_s \\
&= 180^\circ
\end{aligned}
$$

We can extend the proof by symmetry (on the circle) for $|T_s - a| \geq 360^\circ - |T_s - a|$, so the sum of L_1 losses $L_r + L_s$ always equals 180° so the game is constant-sum and therefore fully competitive.

2.2.3 Training Details

Both agents are implemented as MLPs with two hidden layers and ReLU (Nair and Hinton, 2010) nonlinearities between all layers. The targets are sampled from the circle, the sender takes its target, T_s, as input and outputs a categorical distribution over a vocabulary from which we sample a message—its output. The receiver takes the message as input and deterministically outputs its action, a. Errors are calculated using the L_1 loss on the circle. The sender estimates its loss using the score function estimator (Fu, 2006)—also known as REINFORCE (Williams, 1992)—and has an added entropy regularisation term. Since the loss is differentiable with respect to the receiver, it is trained directly with gradient descent, so we are training in the style of a stochastic computation graph (Schulman et al.,

2015).

We train for 30 epochs of 250 batches, with batch size 64, and set the circumference of our circle to 36 (so that a loss of 90° is an error of 9). Both agents are trained using Adam (Kingma and Ba, 2014). To evaluate, we use a fixed test set of 100 equidistant points $\in [0°, 180°]$ and take the arg max of output distributions instead of sampling. We do all hyperparameter searches with Oríon (Bouthillier et al., 2019), using random search with a fixed budget of (100) searches. We perform a hyperparameter search over both agents' learning rates, hidden layer sizes, the vocabulary size, and entropy regularisation (when used). We always report results for given hyperparameters averaged over 5 random seeds, and we average our metric for hyperparameter search over the last 10 epochs to capture some level of stability as well as performance. All hyperparameter search spaces are available in the config files of the code repository.

2.3 Communication: Cooperation or Manipulation

2.3.1 Evaluating Information Transfer

To evaluate the communication emerged with a cheap-talk channel, we can simply look to the sum of agents' L_1 losses. Under non-communication (or uninformative communication), we know that the receiver will just guess a point at random, and the average loss for both players is the expected value of the loss—given that it is drawn uniformly $\mathbb{E}_{\mathrm{x}\sim U(0,360)}[L_1^s(T_s, x)] = 90°$ Therefore, any error for either agent below 90° is evidence of information transfer (Lowe et al., 2019). Furthermore, since there is no other action space for agents to communicate in, the information transfer must be happening in the emergent communication space (Mordatch and Abbeel, 2018). Therefore, the lower $L_1^r + L_1^s$ is, the more informative the learned protocol is ; and, the most informative protocol will have the lowest loss $\min_{a\in(0,360)} L_1^r + L_1^s = b$. To show this comparison, we always plot the loss under uninformative communication (90°) and the loss for each agent if they were to both fairly split the bias ($b/2$).

2.3.2 Information Transfer vs Communication

While we have found evidence of information transfer, does that necessarily mean our agents have *learned to communicate*? For example, our hyperparameter search could find a minimal learning rate for the sender, such that it is essentially static, and a normal configuration for the receiver. The game would then become not one of learning a protocol between two agents, but rather just a receiver learning the sender's initial random mapping of targets to messages. The receiver could then dominate the sender by always choosing $a = T_r$, which would yield $L_1^r + L_1^s = b$; namely, the optimal sum of losses and, therefore, optimal information transfer. This situation is clearly not what we are looking for, but it would be permissible, or potentially even encouraged, under an information-transfer objective (as measured by the sum of agents' L_1 losses). It is, therefore, necessary to delineate the differences in communication; here, we can look to extant results in signalling (Skyrms, 2010).

2.3.3 Cooperation vs. Manipulation

One perspective on information transfer is that of *manipulation* of receivers by senders (Dawkins and Krebs, 1978) or vice-versa (Hinde, 1981); this manifests as the domination of one agent over the other. We note that these situations are modelled as *cue-reading* or *sensory manipulation*, respectively, and are distinct from *signalling*—i.e., *communication* (Barrett and Skyrms, 2017). Accordingly, communication requires *both* agents to receive a net benefit (Krebs and Dawkins, 1984), which implies some degree of *cooperation* (Lewis, 1969). For the fully-cooperative case, previous metrics of joint reward (Lowe et al., 2019), or even influence of communication (Jaques et al., 2019), are sufficient to drive the hyperparameter search. But for competitive scenarios, neither of these can distinguish between manipulation and cooperation (Skyrms and Barrett, 2018).

Since our focus is on the emergence of *cooperative communication*, we are looking for settings where both agents perform better than either their fully-exploited losses ($L_1^s < b$ and $L_1^r < b$) or the loss under non-communication ($L_1^s < 90°$ and $L_1^r < 90°$). With this goal in mind, we choose the sum of squared losses ($(L_2^s)^2 + (L_2^r)^2$) as our hyperparameter-search metric. We can view our partially competitive scenario as having a *common-interest loss* ($180° - b$), in which both agents are fully cooperative,

and a *conflict-of-interest loss* (b), in which both agents are fully competitive. The sum of L_1 losses optimises only for the common interest, whereas L_2 prefers a more fair division of the conflict-of-interest loss in addition to optimising common interest (see proof in 2.3.4). We use the L_2 metric only on hyperparameter search and keep L_1 as our game's loss to maintain a constant-sum game for the fully competitive case.

2.3.4 Proof of L_2 Fairness

Assume without loss of generality $T_s < T_r$, we are minimizing the sum of L_2 losses

$$\begin{aligned}\min_a L_s + L_r &= \min_a (T_s - a)^2 + (T_r - a)^2 \\ &= \min_a (T_s - a)^2 + (T_s + b - a)^2\end{aligned}$$

let $T_s = x$

$$\begin{aligned}&= \min_a (x - a)^2 + (x + b - a)^2 \\ &= \min_a x^2 - 2ax + a^2 + x^2 + 2bx + b^2 - 2ax - 2ab + a^2 \\ &= \min_a 2(x - 2ax + a^2 + bx - ab + b^2/4) + b^2/2 \\ &= \min_a 2(x - a + b/2)^2 + b^2/2 \\ a &= x + b/2\end{aligned}$$

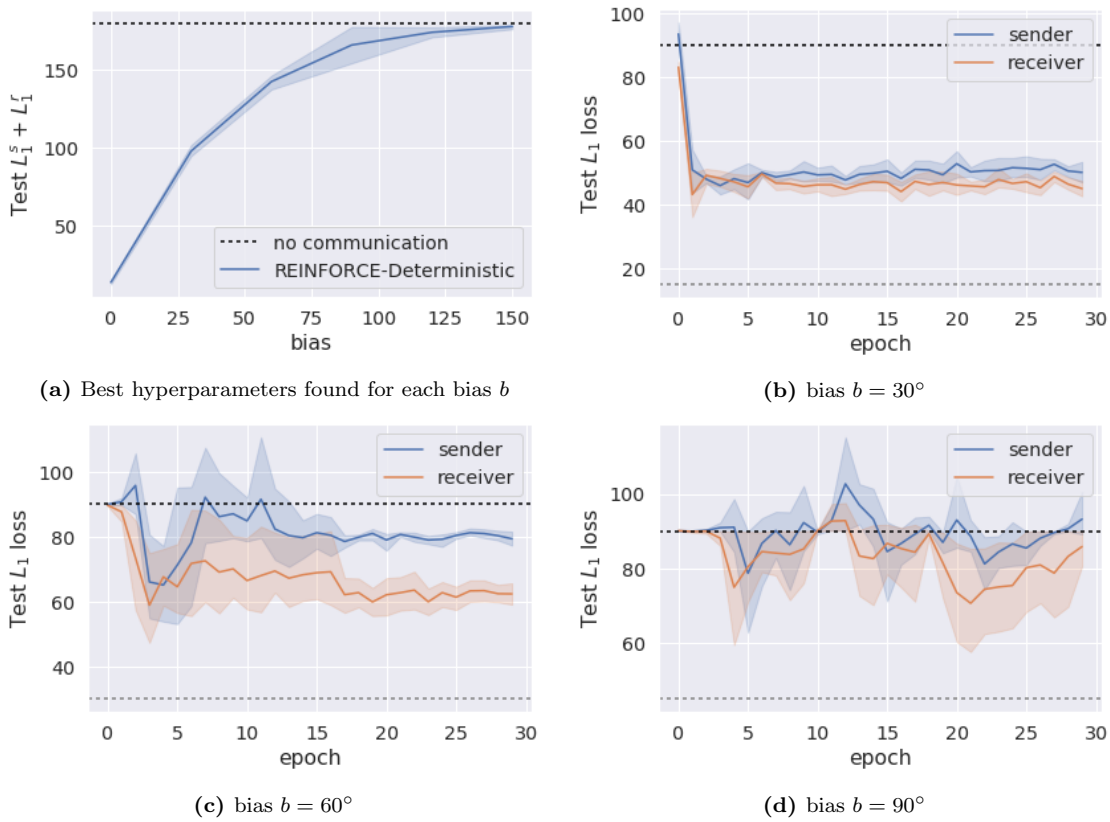
Sum of L_2 losses is minimized when the action is $T_s + b/2$ or halfway between both agents' targets.

2.4 Competitive Selfish Communication

2.4.1 Communication Is Proportional To Cooperation

We use six equidistant values of $b \in [0, 30°, 60°, 90°, 120°, 150°]$ and for each one, we do a hyperparameter search to find the lowest achievable $L_s^2 + L_r^2$. We do not usually test $b = 180°$ because the game is constant-sum and therefore trivially $L_1^s + L_1^r = 180°$, but for completeness you can see the results of a hyperparameter search with $b = 180°$ in Figure 2.16. We report our results in Figure 2.3 and find that agents do learn to cooperatively communicate without any special learning rules contrary to current literature. We can see that the performance decreases proportionately to the bias, meaning the sender is less informative with messages, forcing the receiver to be less accurate in its own guesses. This matches the theoretical results of Crawford and Sobel (1982) ; information transfer with communication is inversely proportional to the conflict of interest. Plots for each b are in shown in Figure 2.6. For the curve, we still plot the L_1 losses to maintain consistency and to make clearer the comparison to the no-communication baseline and the optimal information transfer (common interest maximisation).

We find that our results are basically unchanged between the different hyperparameter metrics ; a relatively fair and useful protocol is learned by the agents, but this deteriorates in more competitive scenarios. This is clear when comparing the stability and relative efficacy of protocols in $b = 30°, 60°$, shown in Figures 2.3b, 2.3c, and that of $b = 90°$ shown in Figure 2.3d. We can understand this through the lens of honest communication, which can be taken advantage of in highly competitive scenarios. If, for example, the sender communicates, with complete honesty, its own coordinates, then the receiver can take advantage of this and choose its location exactly so that $L_r = 0$ and $L_s = b$. Comparing this situation to non-communication ($L_r = L_s = 90°$), it is clear that even fully-exploited communication is a strictly dominant strategy for $b < 90°$ (i.e, when the game is *more cooperative than competitive*).



(a) Best hyperparameters found for each bias b

(b) bias $b = 30°$

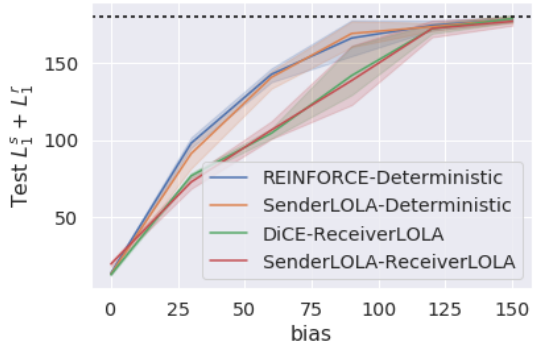
(c) bias $b = 60°$

(d) bias $b = 90°$

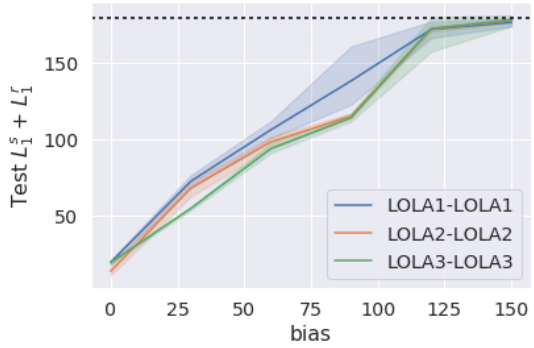
Figure 2.3 – Figure 2.3a plots the lowest $L_1^r + L_1^s$ test loss found with a hyperparameter search for $b \in [0, 30°, 60°, 90°, 120°, 150°]$, demonstrating that informative communication (below the dashed line) is indeed learned by selfish agents. Note that performance even in the fully cooperative $b = 0$ is not optimal because of the bottleneck of discrete communication. For $b \in [30°, 60°, 90°]$ we show the training curve of the best hyperparameters found in 2.3b,2.3c,2.3d. We plot the test loss over training epochs and showing the mean and standard deviation over 5 seeds, finding that for $b < 90°$ we find stable and relatively fair communication is naturally learned

2.4.2 Improving Competitive Communication With LOLA-DiCE

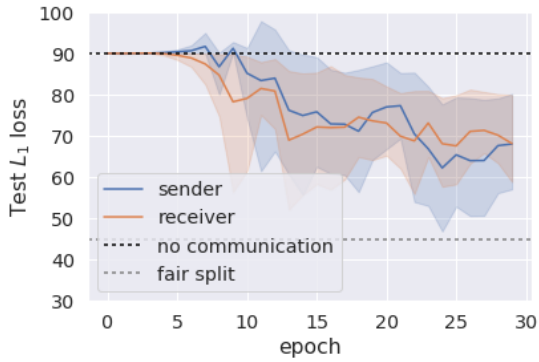
For more competitive cases, fully-exploitable communication is no longer dominant, and active communication now requires both agents to cautiously cooperate. To achieve this cooperation, we propose using LOLA (Foerster et al., 2018a)—a learning rule, resembling theory-of-mind, that allows us to backpropogate through n steps of the opponent's *learning*. LOLA was able to emerge cooperative behaviour in an iterated prisoner's dilemma, so it is a prime candidate for resolving our game in a similar situation. We experiment with LOLA in three configurations—



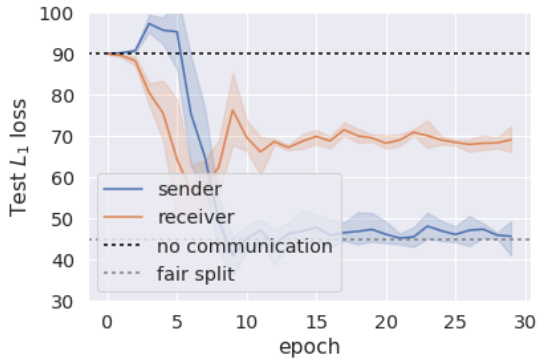
(a) Comparing the original setup, LOLA on the sender, LOLA on the receiver, and LOLA on both



(b) Comparing 1, 2, and 3-step LOLA on both agents



(c) 1-step LOLA on Sender and Receiver for $b = 90°$



(d) 2-step LOLA on Sender and Receiver for $b = 90°$

Figure 2.4 – LOLA improves learning to communicate and it is especially visible at $b = 90°$ where our original setup does very poorly. 2.4b shows that higher step LOLA improves slightly further but not past 2-step. Best feasible communication protocols found for $b = 90°$ using 1-step (2.4c) and 2-step LOLA (2.4d) on both agents demonstrates that the gains in performance over the basic setup shown in Figure 2.3d are not just from one agent doing better (though the sender is doing better) but both agents improving in performance and stability. Shaded area is standard deviation over 5 seeds

LOLA on the sender, LOLA on the receiver, LOLA on both—and do a similar hyperparameter search, with the added search space of the LOLA learning rate. Per the improvements made by Foerster et al. (2019), we replace the receiver's score function estimate with the DiCE estimator, and we backpropogate through exact copies of opponents. Per Foerster et al. (2018a), these results should be similar but lower variance compared to using opponent modelling. We show our results in Figure 2.4a with extended plots in Figures 2.8, 2.9, 2.10.

We find that LOLA on the sender is ineffective, but LOLA on the receiver and on both agents does indeed lead to better performance. This implies that emerging communication in competitive scenarios necessitates cooperation and that this cooperation can be found through explicit opponent modelling. Furthermore, comparing the curves of basic agents (Figure 2.3d) with those of LOLA agents (Figure 2.4c) shows that gains in performance are not from one agent dominating the other, but from both agents improving and increasing stability. We also look at the performance of n-step LOLA, which backpropogates through $n > 1$ steps of opponent learning. Figure 2.4b demonstrates that 2-step LOLA slightly outperforms 1-step LOLA, but 3-step LOLA does not provide any increase over 2-step. We see from Figure 2.4d that the increase comes mostly from stability of learning and slight improvement on the part of the sender.

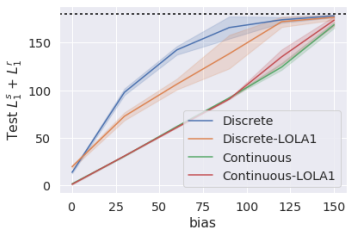
2.4.3 Discrete vs Continuous Communication

Another axis to consider is whether discrete or continuous communication lends itself better to learning with selfish agents. To compare, we make the sender's message a real-valued scalar and appropriately change its output distribution to be a Gaussian, for which it learns the mean and variance, concretely described in Algorithm 2. We, again, run hyperparameter searches, and we consider training our baseline training with a REINFORCE Sender and deterministic receiver as well as training both agents with 1-step LOLA. Our results in Figure 2.5a suggest that the learned protocols for continuous communication are all highly informative and near optimal. However, in all cases, the receiver is learning to manipulate the sender, and there is little evidence of cooperative communication. Indeed, we found no cases of both agents having a net benefit ($L_r, L_s < 90°$) in *any* of the hyperparameter runs for continuous messages between a Gaussian REINFORCE sender and

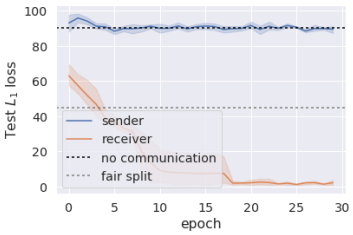
deterministic receiver past $b = 90°$, and only two cases of net benefit for LOLA-1 agents. Comparing this to discrete communication with the same LOLA-1 agents in Figure 2.5f, we can clearly see that they have a preference for more cooperative behaviour. Thus, we find that discrete messages generated with a Categorical distribution are an important component in emerging cooperative self-interested communication, as compared to continuous messages with a Gaussian distribution.

Algorithm 2 Continuous Circular Biased Sender-Receiver Game

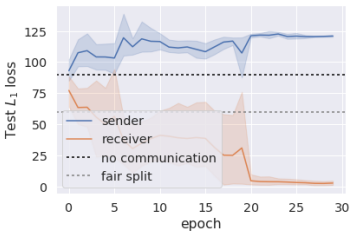
procedure TRAINING BATCH(b)
$T_s \sim \text{Uniform}(0, 360)$
$T_r \leftarrow T_s + b$
$\mu, \sigma \leftarrow S(T_s)$
$m \sim \text{Gaussian}(\mu, \sigma)$
$a \leftarrow R(m)$
$L_s \leftarrow L_1(T_s, a) = \min(|T_s - a|, 360 - |T_s - a|)$
$L_r \leftarrow L_1(T_r, a) = \min(|T_r - a|, 360 - |T_r - a|)$
R is updated with SGD
S is updated with REINFORCE or DiCE



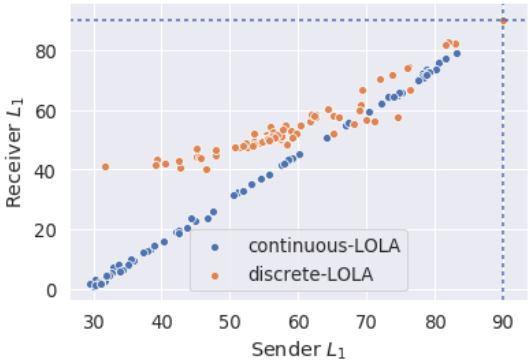
(a) Comparing discrete and continuous communication



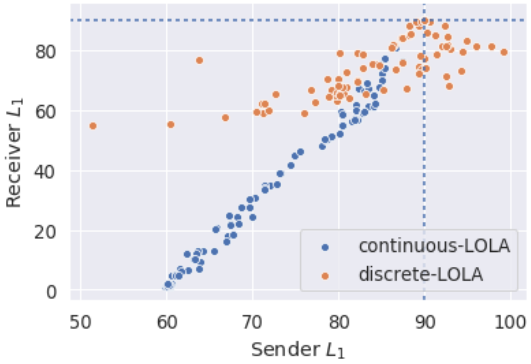
(b) Continuous Game for $b = 90°$



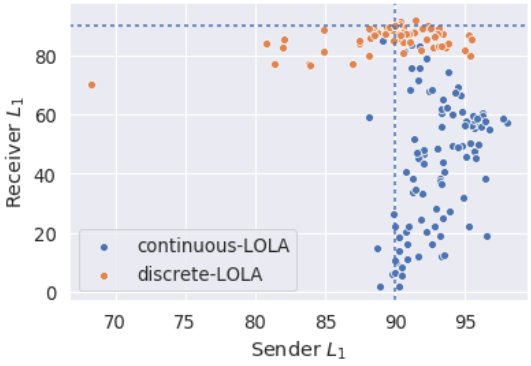
(c) Continuous Game for $b = 120°$



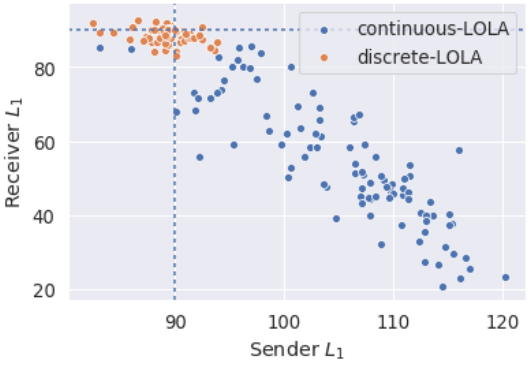
(d) L_s vs. L_r for all 100 searches for $b = 30°$



(e) L_s vs. L_r for all 100 searches for $b = 60°$



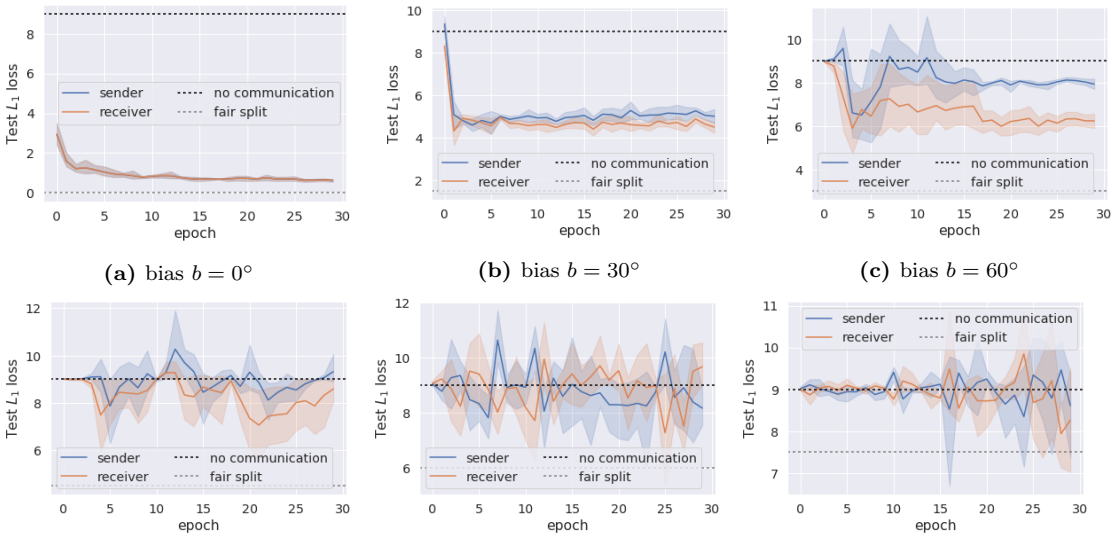
(f) L_s vs. L_r for all 100 searches for $b = 90°$



(g) L_s vs. L_r for all 100 searches for $b = 120°$

Figure 2.5 – The comparison between discrete and continuous communication for both the REINFORCE-deterministic setup as well as 1-step LOLA agents is shown in Figure 2.5a. We see that though overall continuous communication can achieve highest information transfer, the gains in performance seem to mostly from manipulation of the sender by the receiver. Two examples are shown for REINFORCE agents in Figures 2.5b,2.5c. To find a trend, we plot all 100 hyperparameter runs for $b \in [3, 6, 9, 12]$ between continuous and discrete communication using 1-step LOLA agents in Figures 2.5d,2.5e,2.5f,2.5g. We find that manipulation is the common result in continuous communication though individual cooperative points can sometimes be found. In general, continuous communication does not lend itself to cooperative communication

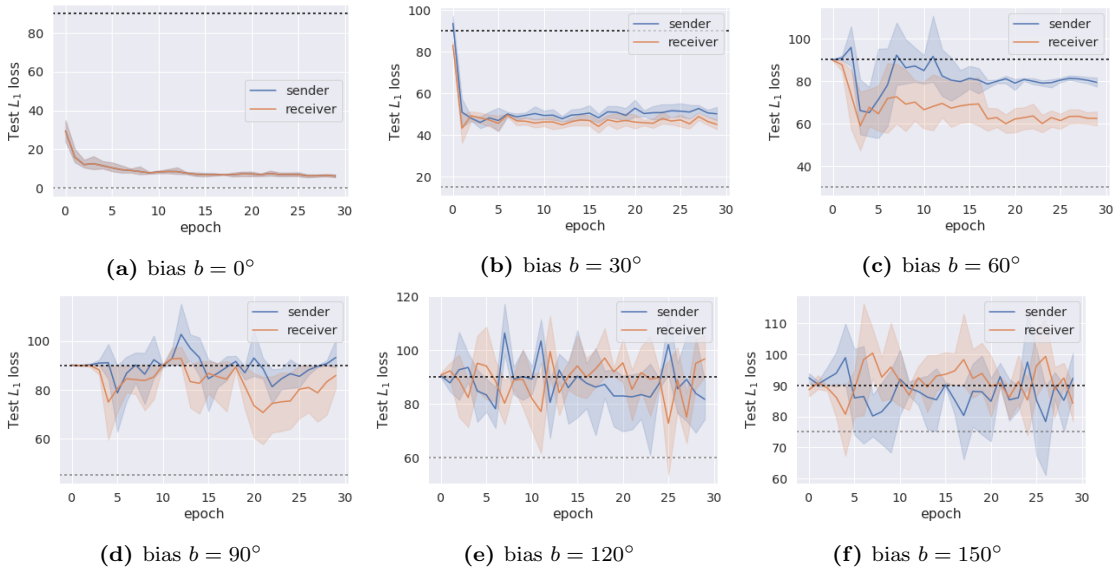
2.5 Extra Plots



(a) bias $b = 0°$ **(b)** bias $b = 30°$ **(c)** bias $b = 60°$

(d) bias $b = 90°$ **(e)** bias $b = 120°$ **(f)** bias $b = 150°$

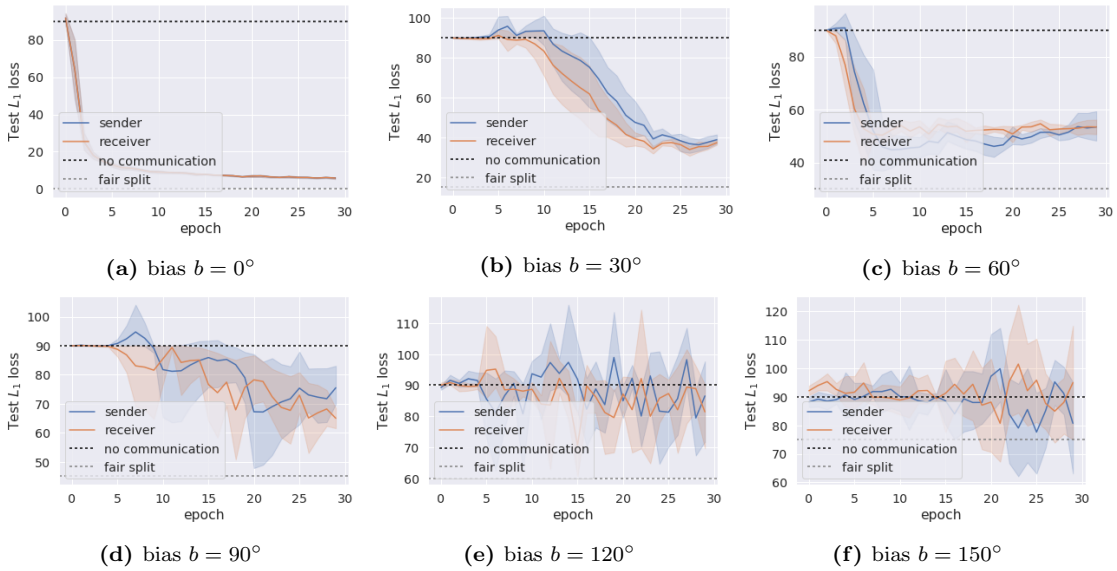
Figure 2.6 – REINFORCE Sender, Deterministic Receiver, L_1 hyperparameter search



(a) bias $b = 0°$ **(b)** bias $b = 30°$ **(c)** bias $b = 60°$

(d) bias $b = 90°$ **(e)** bias $b = 120°$ **(f)** bias $b = 150°$

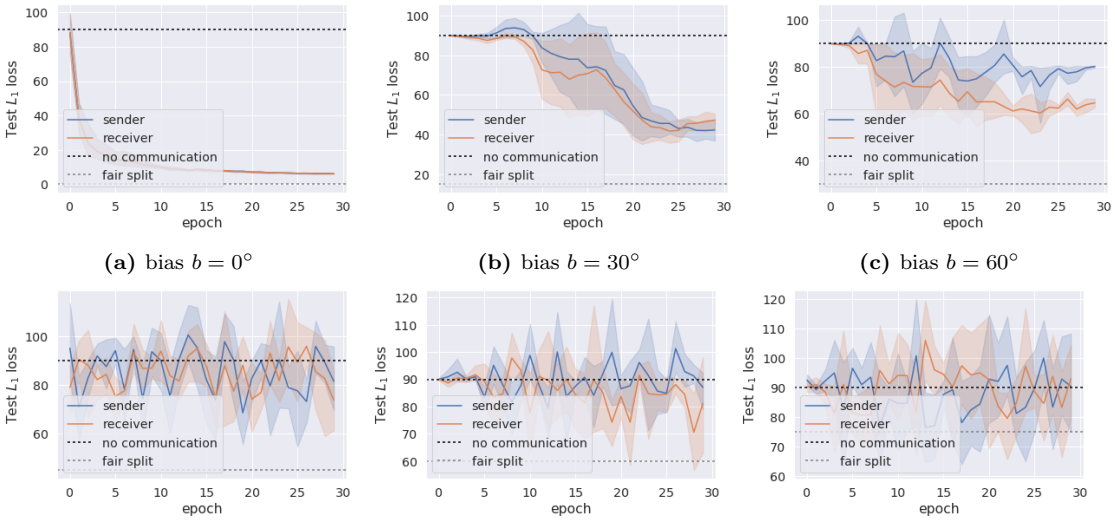
Figure 2.7 – REINFORCE Sender, Deterministic Receiver, L_2 hyperparameter search. Note these are identical to Figure 2.6 except for $b = 150°$



(a) bias $b = 0°$ **(b)** bias $b = 30°$ **(c)** bias $b = 60°$

(d) bias $b = 90°$ **(e)** bias $b = 120°$ **(f)** bias $b = 150°$

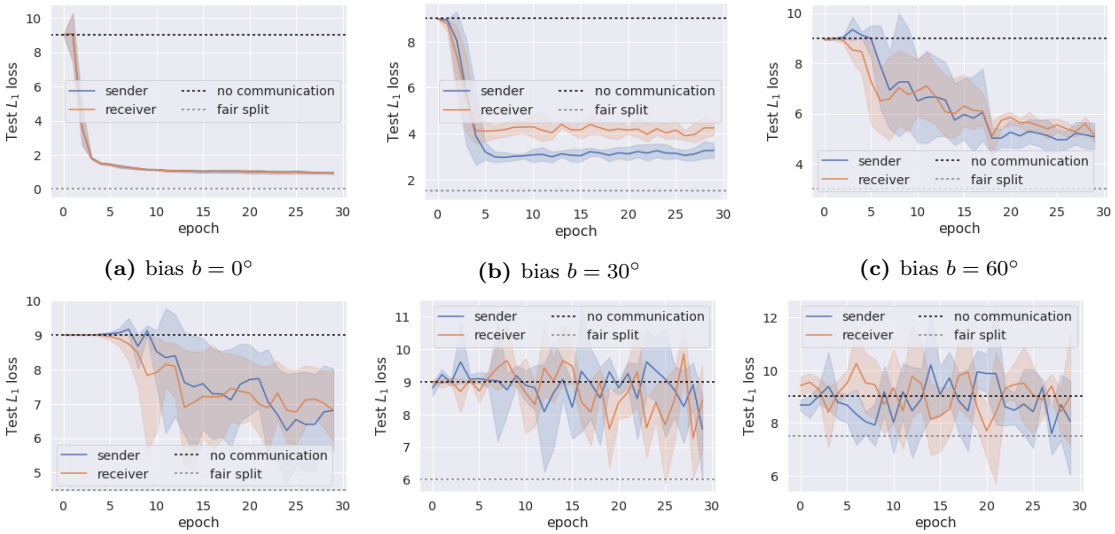
Figure 2.8 – DiCE Sender, LOLA-1 Receiver



(a) bias $b = 0°$ **(b)** bias $b = 30°$ **(c)** bias $b = 60°$

(d) bias $b = 90°$ **(e)** bias $b = 120°$ **(f)** bias $b = 150°$

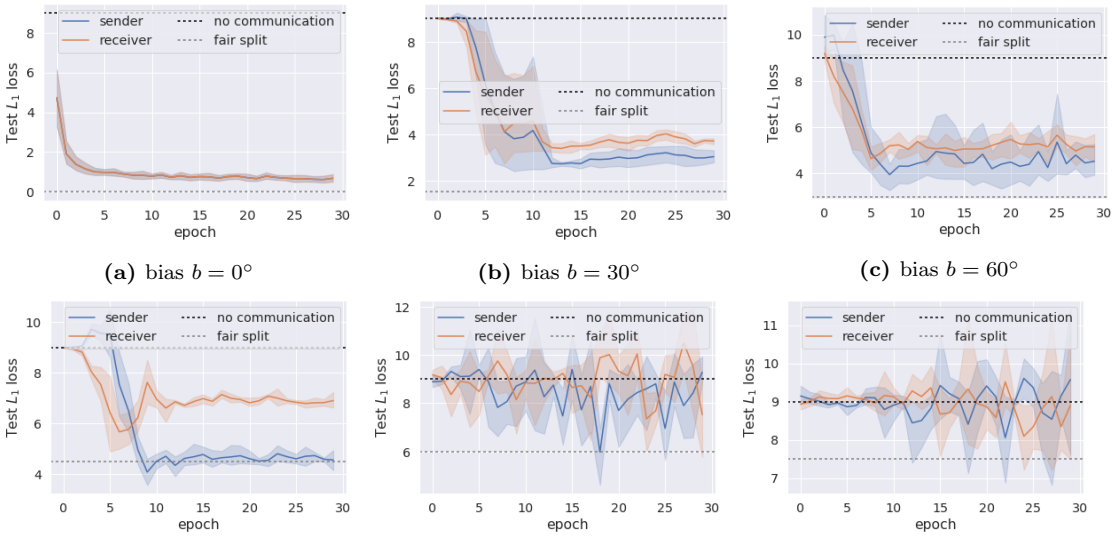
Figure 2.9 – LOLA-1 Sender, Deterministic Receiver



(a) bias $b = 0°$ **(b)** bias $b = 30°$ **(c)** bias $b = 60°$

(d) bias $b = 90°$ **(e)** bias $b = 120°$ **(f)** bias $b = 150°$

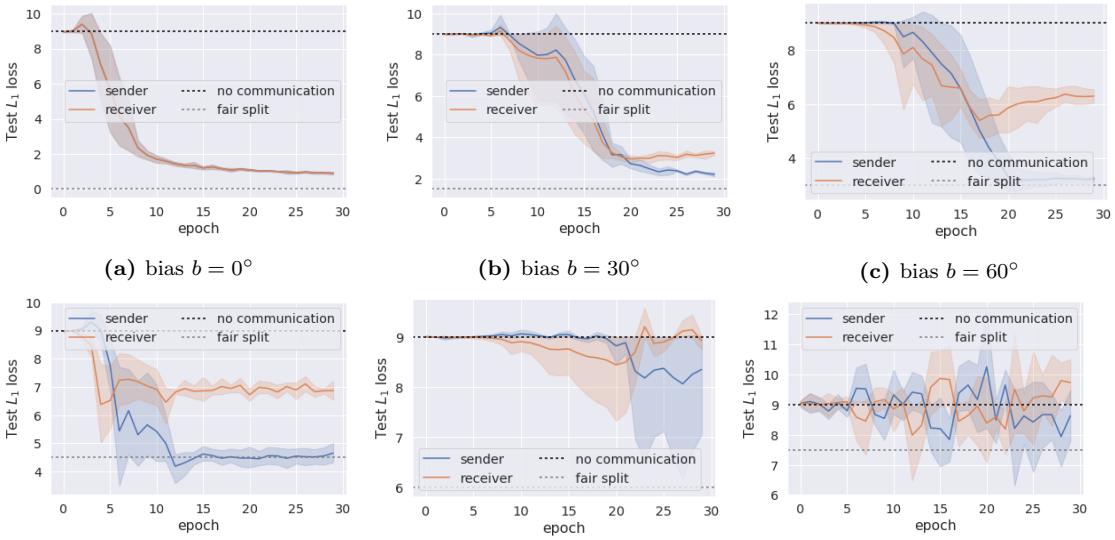
Figure 2.10 – LOLA-1 Sender, LOLA-1 Receiver



(a) bias $b = 0°$ **(b)** bias $b = 30°$ **(c)** bias $b = 60°$

(d) bias $b = 90°$ **(e)** bias $b = 120°$ **(f)** bias $b = 150°$

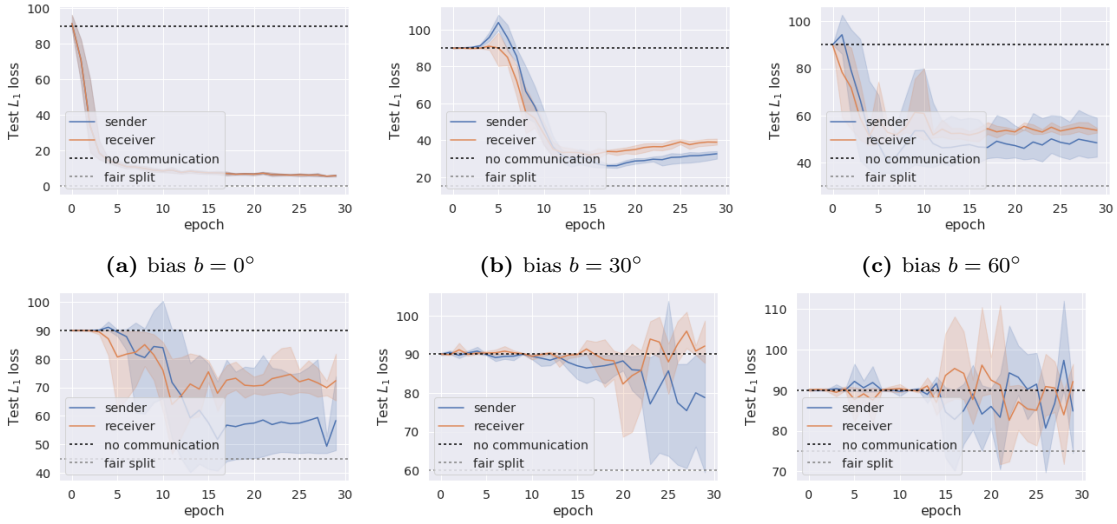
Figure 2.11 – LOLA-2 Sender, LOLA-2 Receiver



(a) bias $b = 0°$ **(b)** bias $b = 30°$ **(c)** bias $b = 60°$

(d) bias $b = 90°$ **(e)** bias $b = 120°$ **(f)** bias $b = 150°$

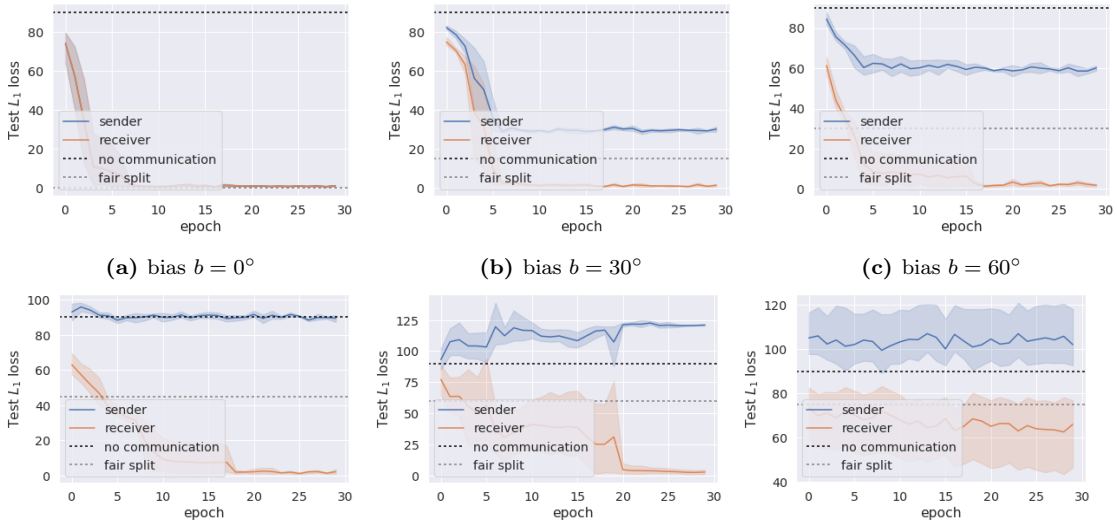
Figure 2.12 – LOLA-3 Sender, LOLA-3 Receiver



(a) bias $b = 0°$ **(b)** bias $b = 30°$ **(c)** bias $b = 60°$

(d) bias $b = 90°$ **(e)** bias $b = 120°$ **(f)** bias $b = 150°$

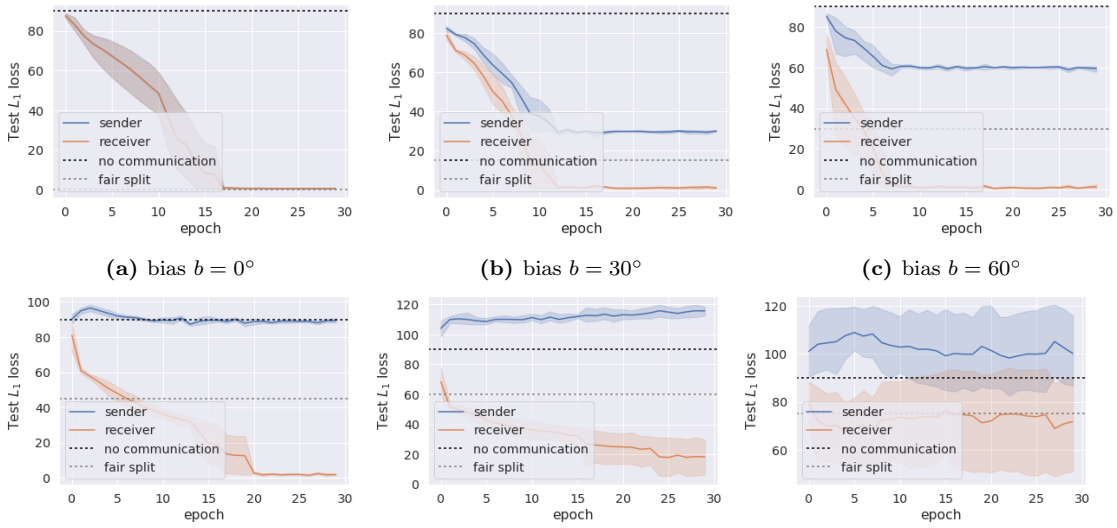
Figure 2.13 – LOLA-4 Sender, LOLA-4 Receiver



(a) bias $b = 0°$ **(b)** bias $b = 30°$ **(c)** bias $b = 60°$

(d) bias $b = 90°$ **(e)** bias $b = 120°$ **(f)** bias $b = 150°$

Figure 2.14 – Gaussian Sender. Deterministic Receiver playing the continuous game



(a) bias $b = 0°$ **(b)** bias $b = 30°$ **(c)** bias $b = 60°$

(d) bias $b = 90°$ **(e)** bias $b = 120°$ **(f)** bias $b = 150°$

Figure 2.15 – Gaussian LOLA Sender. LOLA Receiver playing the continuous game

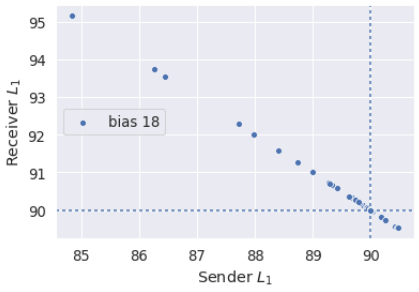


Figure 2.16 – REINFORCE Sender vs Deterministic Receiver errors for all hyperparameter runs with $b = 180°$ shows that agents are mostly fair in the fully competitive constant-sum game

3 Conclusion

First and foremost, we show evidence against the current notion that selfish agents do not learn to communicate, and we hope our findings encourage more research into communication under competition. We have shown three important properties of communication. First, a game being *more cooperative than competitive* is sufficient to naturally emerge communication. Second, we've found that *LOLA improves effective selfish communication* and, using our metric, we find it does so by improving both agents' performance and stability. Third, we've compared a categorical distribution over a *discrete communication channel* and a single Gaussian with a *continuous communication channel* finding that the former better encourages the learning of cooperative communication, whereas the latter lends itself to non-cooperative manipulation. In order to make these experiments we've also clarified the distinction between information transfer, communication, and manipulation. This extends the work of Lowe et al. (2019) to competitive scenarios, providing a better understanding of quantitative metrics for measure emergent communication in competitive environments.

In fully-cooperative emergent communication, both agents fully trust each other, so cooperatively *learning a protocol* is mutually beneficial. In competitive MARL, the task is *using an existing protocol* (or action space) to compete with each other. However, selfish emergent communication combines these two since the inherent competitiveness of using the protocol to win is tempered by the inherent cooperativeness of learning it ; without somewhat agreeing to meanings, agents cannot use those meanings to compete (Searcy and Nowicki, 2005; Skyrms and Barrett, 2018). Thus, the agents must both *learn* a protocol and *use* that protocol simultaneously. In this way, even while competing, selfish agents emerging a communication protocol must learn to cooperate.

Bibliographie

David Balduzzi, Karl Tuyls, Julien Pérolat, and Thore Graepel. Re-evaluating evaluation. In *NeurIPS*, 2018.

David Balduzzi, Marta Garnelo, Yoram Bachrach, Wojciech M Czarnecki, Julien Perolat, Max Jaderberg, and Thore Graepel. Open-ended learning in symmetric zero-sum games. In *ICML*, 2019.

Jeffrey Banks. *Signalling games in political science*. Routledge, London and New York, 1991.

Nolan Bard, Jakob N. Foerster, A. P. Sarath Chandar, Neil Burch, Marc Lanctot, H. Francis Song, Emilio Parisotto, Vincent Dumoulin, Subhodeep Moitra, Edward Hughes, Iain Dunning, Shibl Mourad, Hugo Larochelle, Marc G. Bellemare, and Michael H. Bowling. The hanabi challenge: A new frontier for ai research. *ArXiv*, abs/1902.00506, 2019a.

Nolan Bard, Jakob N. Foerster, Sarath Chandar, Neil Burch, Marc Lanctot, H. Francis Song, Emilio Parisotto, Vincent Dumoulin, Subhodeep Moitra, Edward Hughes, and et al. The hanabi challenge: A new frontier for ai research. *Artificial Intelligence*, Nov 2019b.

Jeffrey A. Barrett and Brian Skyrms. Self-Assembling Games. *The British Journal for the Philosophy of Science*, 68(2):329–353, 2017.

James Bergstra and Yoshua Bengio. Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, 13(Feb):281–305, 2012.

Xavier Bouthillier, Christos Tsirigotis, François Corneau-Tremblay, Pierre Delaunay, Michael Noukhovitch, Reyhane Askari, Peter Henderson, Dendi Suhubdy, Frédéric Bastien, and Pascal Lamblin. Oríon - Asynchronous Distributed Hyperparameter Optimization. https://github.com/Epistimio/orion, September 2019.

Kris Cao, Angeliki Lazaridou, Marc Lanctot, Joel Z. Leibo, Karl Tuyls, and Stephen Clark. Emergent communication through negotiation. In *ICLR*, 2018.

Dorothy Cheney and Robert Seyfarth. Vervet monkey alarm calls: Manipulation through shared information ? *Behaviour*, 94(1):150–166, 1985.

Vincent P. Crawford and Joel Sobel. Strategic information transmission. *Econometrica*, 50(6):1431–1451, 1982. URL https://doi.org/10.2307/1913390.

Richard Dawkins and John R. Krebs. Animal signals: Information or Manipulation ? In J. R. Krebs and N. B. Davies, editors, *Behavioural Ecology*, pages 282–309. Blackwell Scientific Publications, Oxford, 1978.

Katrina Evtimova, Andrew Drozdov, Douwe Kiela, and Kyunghyun Cho. Emergent communication in a multi-modal, multi-step referential game. *arXiv preprint arXiv:1705.10369*, 2017.

Joseph Farrell and Matthew Rabin. Cheap talk. *The Journal of Economic Perspectives*, 10(3):103–118, 1996.

Jakob Foerster, Ioannis Alexandros Assael, Nando de Freitas, and Shimon Whiteson. Learning to communicate with deep multi-agent reinforcement learning. In *Advances in Neural Information Processing Systems*, pages 2137–2145, 2016.

Jakob N. Foerster, Richard Y. Chen, Maruan Al-Shedivat, Shimon Whiteson, Pieter Abbeel, and Igor Mordatch. Learning with opponent-learning awareness. In *AAMAS*, 2018a.

Jakob N. Foerster, H. Francis Song, Edward Hughes, Neil Burch, Iain Dunning, Shimon Whiteson, Matthew M Botvinick, and Michael H. Bowling. Bayesian action decoder for deep multi-agent reinforcement learning. *ArXiv*, abs/1811.01458, 2018b.

Jakob N. Foerster, Gregory Farquhar, Maruan Al-Shedivat, Tim Rocktäschel, Eric P. Xing, and Shimon Whiteson. Dice: The infinitely differentiable monte-carlo estimator. In *ICML*, 2019.

Michael Fu. Chapter 19 gradient estimation. *Handbooks in Operations Research and Management Science*, 13, 12 2006. doi: 10.1016/S0927-0507(06)13019-4.

Yuma Fujimoto and Kunihiko Kaneko. Functional dynamic by intention recognition in iterated games. *New Journal of Physics*, 21(2):023025, 2019.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. http://www.deeplearningbook.org.

Serhii Havrylov and Ivan Titov. Emergence of language with multi-agent games: Learning to communicate with sequences of symbols. In *Advances in neural information processing systems*, pages 2149–2159, 2017.

Peter Henderson, Riashat Islam, Philip Bachman, Joelle Pineau, Doina Precup, and David Meger. Deep reinforcement learning that matters. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

Richard J Herrnstein. Relative and absolute strength of response as a function of frequency of reinforcement 1, 2. *Journal of the experimental analysis of behavior*, 4(3):267–272, 1961.

Robert A. Hinde. Animal Signals: Ethological and Games-Theory Approaches are Not Incompatible. *Animal Behavior*, 29:535–542, 1981.

Max Jaderberg, Valentin Dalibard, Simon Osindero, Wojciech M Czarnecki, Jeff Donahue, Ali Razavi, Oriol Vinyals, Tim Green, Iain Dunning, Karen Simonyan, et al. Population based training of neural networks. *arXiv preprint arXiv:1711.09846*, 2017.

Natasha Jaques, Angeliki Lazaridou, Edward Hughes, Caglar Gulcehre, Pedro A. Ortega, DJ Strouse, Joel Z. Leibo, and Nando de Freitas. Social influence as intrinsic motivation for multi-agent deep reinforcement learning. In *ICML*, 2019.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014.

Satwik Kottur, José Moura, Stefan Lee, and Dhruv Batra. Natural language does not emerge 'naturally' in multi-agent dialog. In *EMNLP*, 2017.

John R. Krebs and Richard Dawkins. Animal Signals: Mind Reading and Manipulation. In J. R. Krebs and N. B. Davies, editors, *Behavioural Ecology*, pages 380–402. Sinauer Associates, Sunderland, MA, 1984.

Marc Lanctot, Vinicius Zambaldi, Audrunas Gruslys, Angeliki Lazaridou, Karl Tuyls, Julien Perolat, David Silver, and Thore Graepel. A Unified Game-Theoretic Approach to Multiagent Reinforcement Learning. In *NIPS*, 2017.

Angeliki Lazaridou, Alexander Peysakhovich, and Marco Baroni. Multi-agent cooperation and the emergence of (natural) language. *arXiv preprint arXiv:1612.07182*, 2016.

Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521 (7553):436–444, 2015.

Joel Z. Leibo, Vinicius Zambaldi, Marc Lanctot, Janusz Marecki, and Thore Graepel. Multi-agent reinforcement learning in sequential social dilemmas. In *AAMAS*, 2017.

Adam Lerer and Alexander Peysakhovich. Maintaining cooperation in complex social dilemmas using deep reinforcement learning. *arxiv*, 2017. URL http://arxiv.org/abs/1707.01068.

Alistair Letcher, Jakob N. Foerster, David Balduzzi, Tim Rocktäschel, and Shimon Whiteson. Stable opponent shaping in differentiable games. In *ICLR*, 2019.

David Lewis. *Convention: A Philosophical Study*. Wiley-Blackwell, Oxford, 1969.

Ryan Lowe, Yi Wu, Aviv Tamar, Jean Harb, Pieter Abbeel, and Igor Mordatch. Multi-agent actor-critic for mixed cooperative-competitive environments. In *NIPS*, pages 6379–6390, 2017.

Ryan Lowe, Jakob N. Foerster, Y-Lan Boureau, Joelle Pineau, and Yann Dauphin. On the pitfalls of measuring emergent communication. In *AAMAS*, 2019.

Manolo Martínez and Peter Godfrey-Smith. Common interest and signaling games: a dynamic analysis. *Philosophy of Science*, 83(3):371–392, 2016.

Igor Mordatch and Pieter Abbeel. Emergence of grounded compositional language in multi-agent populations. In *AAAI*, 2018.

Vinod Nair and Geoffrey Hinton. Rectified linear units improve restricted boltzmann machines vinod nair. In *ICML*, volume 27, pages 807–814, 06 2010.

Cinjon Resnick, Wes Eldridge, David Ha, Denny Britz, Jakob Foerster, Julian Togelius, Kyunghyun Cho, and Joan Bruna. Pommerman: A multi-agent playground. *arxiv*, 2018. URL http://arxiv.org/abs/1809.07124.

John G Riley. Silver signals: Twenty-five years of screening and signaling. *Journal of Economic literature*, 39(2):432–478, 2001.

David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *nature*, 323(6088):533–536, 1986.

Stuart J Russell and Peter Norvig. *Artificial intelligence: a modern approach*. Malaysia; Pearson Education Limited,, 2016.

Jakob Foerster Douwe Kiela Joelle Pineau Ryan Lowe, Abhinav Gupta. On the interaction between supervision and self-play in emergent communication. In *ICLR*, 2020.

John Schulman, Nicolas Manfred Otto Heess, Theophane Weber, and Pieter Abbeel. Gradient estimation using stochastic computation graphs. In *NIPS*, 2015.

William A Searcy and Stephen Nowicki. *The evolution of animal communication: reliability and deception in signaling systems*. Princeton University Press, Princeton, NJ, 2005.

Thomas C. Shelling. *The Strategy of Conflict*. Harvard University Press, Cambridge, MA, 1960.

Paul W. Sherman. Nepotism and the evolution of alarm calls. *Science*, 197:1246–1253, 1977.

Yoav Shoham, Rob Powers, and Trond Grenager. Multi-agent reinforcement learning:a critical survey. *Tech Report, Stanford University*, 2003.

David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lynette R. Baker, Matthew Lai, Adrian Bolton, Yutian Chen, Timothy P. Lillicrap, Fan Hui, Laurent Sifre, George van den Driessche, Thore Graepel, and Demis Hassabis. Mastering the game of go without human knowledge. *Nature*, 550:354–359, 2017.

Amanpreet Singh, Tushar Jain, and Sainbayar Sukhbaatar. Learning when to communicate at scale in multiagent cooperative and competitive tasks. In *ICLR*, 2018.

Brian Skyrms. *Signals: Evolution, Learning, & Information*. Oxford University Press, Oxford, 2010.

Brian Skyrms. *Evolution of the Social Contract*. Cambridge University Press, Cambridge, 2014/1996.

Brian Skyrms and Jeffrey A. Barrett. Propositional content in signals, June 2018. URL http://philsci-archive.pitt.edu/14774/.

Michael Spence. Job market signaling. *The Quarterly Journal of Economics*, 87 (3):355–374, 1973.

Sainbayar Sukhbaatar, Rob Fergus, et al. Learning multiagent communication with backpropagation. In *Advances in Neural Information Processing Systems*, pages 2244–2252, 2016.

Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, second edition, 2018.

P. Taylor and L. Jonker. Evolutionarily stable strategies and game dynamics. *Mathematical Biosciences*, 40:145–156, 1978.

Oriol Vinyals, Igor Babuschkin, Junyoung Chung, Michael Mathieu, Max Jaderberg, Wojciech M Czarnecki, Andrew Dudzik, Aja Huang, Petko Georgiev, Richard Powell, et al. Alphastar: Mastering the real-time strategy game starcraft ii. *DeepMind Blog*, 2019.

John Von Neumann. On the theory of parlor games. *Mathematische Annalen*, 100: 295–320, 1928.

Elliott O. Wagner. Deterministic chaos and the evolution of meaning. *British Journal for the Philosophy of Science*, 63:547–575, 2012.

Elliott O. Wagner. Conventional semantic meaning in signalling games with conflicting interests. *British Journal for the Philosophy of Science*, 66(4):751–773, 2014.

Kyle Wagner, James A Reggia, Juan Uriagereka, and Gerald S Wilkinson. Progress in the simulation of emergent communication and language. *Adaptive Behavior*, 11(1):37–69, 2003.

Ronald J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8:229–256, 1992.

Amotz Zahavi. Mate selection: A selection for a handicap. *Journal of Theoretical Biology*, 53(1):205–214, 1975.