From Misunderstanding to Cooperation: Understanding and Expressing Intentions Through Non-Communicative Actions

# # #

ABSTRACT

Solving situations of misunderstanding requires two abilities: to build a coherent model of others in order to understand them, and to build a model of "me" perceived by others in order to be understood. Having an image of me seen by others requires two recursive orders of modeling, known in psychology as first and second orders of theory of mind. It becomes especially difficult to find an understanding when agents don't have a common language to communicate and have to learn and share each others intentions through their behaviors. In this paper, we present a cognitive architecture based on both Reinforcement Learning and Inverse Reinforcement Learning that aims to reach mutual understanding in multi-agent scenarios. We study different conditions of empathy or gratitude that lead to cooperation in prisoner's dilemma.

CCS Concepts

•Computing methodologies → Multi-agent systems;

Keywords

AAMAS proceedings, LaTeX, text tagging

1. INTRODUCTION

In any interaction that requires cooperation agents need to understand each other's intentions. A common ground [2], for instance a spoken language – or even a gestural language, helps to reach such mutual understanding [16][12]. Without any such ground, agents can still analyse their respective motivations through their goal-directed behaviors in order to cooperate or intentionally defect [10]. Moreover, they can adapt their behavior in order to be more easily understood (NEED REF). We investigate in this paper a cognitive architecture enabling virtual agents of such analysis.

As humans, we have different strategies to exhibit understanding or to resolve a misunderstanding. As an example, if someone is talking about a visual object, we alternatively gaze between the object and the person to make sure he saw that we gazed at the object. Or if we detect that the other person has not understood a gesture (e.g. pointing at an object) we would probably exaggerate the gesture. But even in this example, we have recourse to a known common gestural signal. TO CONTINUE (need for mutual understanding in HRI)

Developed by Baron-Cohen and Leslie [1], the Theory of Mind (ToM) describes the ability to attribute mental states and knowledge to others. In interaction, humans are permanently collecting and analysing huge quantity of information to stay aware of emotions, goals and understandings of their fellows. In this work, we focus on a generalization of this notion: *Mutual Modelling* (MM) is the reciprocal ability to establish a mental model of the other [13].

2. MUTUAL MODELING

2.1 Non-recursive approach

Many ToM-based architecture have been developed in order to study multi-agent behaviors in social games. But so far, most of these applications where limited to first order of modeling (agents does not take into account how they are modeled by others) while higher orders leads to better performances in a range of simple social games [14][5][22][6][7][8]. However, 2nd order (an agent has a model of itself viewed by others) seams sufficient to generate rich social behaviors [18]. Higher orders (an agent has a model of its own theory of mind imagined by other agents), although they outperform 2nd order in some cases (especially fourth [7]) do not seams to bring important advantages [22].

We introduce a cognitive architecture enabling second order of modeling. In contrast with previous approaches [14][6][18], this one is not recursive. In recursive modeling, an agent re-use its own architecture (that allows theory of mind) to model other agents [9]. In multi-agent framework, recursions need to be stopped at a given depth. Otherwise it would lead to an infinite loop of mutual modeling, known as infinite regress in epistemic logic [3]. Such approaches have limits: it is difficult to process in parallel reasoning of all agents, it becomes heavy in computation beyond second order of modeling, and different agents or their images (perceived by others) may have different reasoning and may adopt different behaviors facing similar observations.

In our non-recursive approach, one agent has three models: a model of itself, a model of others and a model of itself seen by others. None of these models are performing theory of mind. At any instant, the agent updates its three models given his observations and use them to make predictions and decisions.

Appears in: *Proceedings of the 16th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2017), S. Das, E. Durfee, K. Larson, M. Winikoff (eds.), May 8–12, 2017, São Paulo, Brazil.*

2.2 Model of itself

An agent i models itself as a RL agent: at a time t, it chooses an action a_i^t. Depending on this decision and all other agent's decisions $\{a_j^t\}_{j \neq i}$, it receives an observation $o_i^{t+1} = O(a_1^t, a_2^t, ..a_n^t)$ (n being the number of agents) and a reward that only depend on this observation $r_i^{t+1} = R_i(o^{t+1})$. Each agent has its own reward function that is unknown by other agents.

As in [20], this framework is simplified by the agent as a *Markovian Decision Process* (MDP) [17] where the observations are assumed to be states that just depend on its previous observation and action following an unknown probability distribution:

$$o_i^{t+1} = O(a_1^t, a_2^t, ..a_n^t) \sim \mathbb{P}[o_i^{t+1} | a_i^t, o_i^t]$$

Hence, at the beginning, the decision making of the agent is performed by Q-learning [21]. Given the observation o^{t+1}, the agent learns the best new action a^{t+1} in order to maximize its future rewards (see algo. 1).

Algorithm 1: Q-learning. *TD* stands for *Temporal Difference.*

```
Initialize $Q(o, a)$
Initialize $o^0$
forall iterations $t$ do
    Choose $a^t$ from $o^t$ using policy derived from Q
    Take action $a^t$, receive $r^{t+1}$, $o^{t+1}$
    $TD = r^{t+1} + \gamma \max_{a^{t+1}} Q(o^{t+1}, a^{t+1}) - Q(o^t, a^t)$
    $Q(o^t, a^t) \leftarrow Q(o^t, a^t) + \eta \, TD$
end
```

2.3 Model of others

At the same time, it receives actions and observations of other agents $\{a_j^t\}_{j \neq i}$ and $\{o_j^t\}_{j \neq i}$. Given this information, it can infer their reward functions $\{R_j\}_{j \neq i}$ by *Inverse Reinforcement Learning* (IRL) [15]. It this setup, the IRL must be performed on-line. In [11] they provide an incremental algorithm for on-line IRL in a MDP framework. This method is efficient but not so intuitive. As our final goal is to develop agents that could interact with humans, we want to adopt a less efficient but more intuitive approach that looks like how any human (or even a child) would infer the intentions of others. And maybe the simplest way is the following:

If I liked I repeat, otherwise I change: Supposing agent j receives an observation *"light"* and chooses action *"press-the-button"*. Then it receives new observation *"glass-of-juice"*. If it liked the juice, probably the next time it will observe the signal *"light"* it would again choose action *"press-the-button"*. Otherwise it would choose another possible action.

In order to formalize this approach, we denote as $\hat{r}_{i:j}^t = \hat{R}_{i:j}(o_j^t)$ the reward of agent j at time t inferred by agent i. Agent i memorizes, for each possible observation o_j of agent j, the last action $A_{i:j}(o_j)$ it chose facing o_j. Agent i also memorizes, for each observation o_j, the next observation $O_{i:j}(o_j)$ perceived as a consequence of choosing action $A_{i:j}(o_j)$. If at time t, agent j observes o_j^t and chooses once again the action $a_j^t = A_{i:j}(o_j^t)$, it means agent j "liked" the previous consequence of this choice, says $o_{prev} = O_{i:j}(o_j^t)$. In that case, agent i increments its inferred reward function $\hat{R}_{i:j}(o_j^t)$ for agent j as follow:

$$\hat{R}_{i:j}(o_{prev}) \leftarrow (1 - \frac{1}{\sqrt{n(o_j^t)}}).\hat{R}_{i:j}(o_{prev}) + \frac{1}{\sqrt{n(o_j^t)}}$$

Where $n(o_j^t)$ is the number of times agent i observed agent j observing o_j^t. Contrariwise, if it chooses a different action $a_j^t \neq A_{i:j}(o_j^t)$, agent i decrements the estimated reward function $\hat{R}_{i:j}(o_j^t)$ for agent j:

$$\hat{R}_{i:j}(o_{prev}) \leftarrow (1 - \frac{1}{\sqrt{n(o_j^t)}}).\hat{R}_{i:j}(o_{prev}) - \frac{1}{\sqrt{n(o_j^t)}}$$

Then, given the inferred reward functions, an agent can predict the next action of other agents. Such a prediction can be used to adapt its own next decision in consequence, and also to evaluate how it is able to model other agents. This intuitive IRL process is summarized in (see algo. 2).

Algorithm 2: Intuitive on-line IRL. Agent i is inferring the reward function of agent j.

```
Initialize $R_{i:j}(o)$
forall iterations $t$ do
    Agent $j$ observes $o_j^t$ and takes action $a_j^t$
    if $o_j^t$ has already been observed by $j$ then
        Remember:
        $a_{prev} = A_{i:j}(o_j^t)$ previous action of $j$ after $o_j^t$
        $o_{prev} = O_{i:j}(o_j^t)$ previous consequence
        if $a_j^t = a_{prev}$ then
            $\hat{R}_{i:j}(o_{prev}) \leftarrow (1 - \frac{1}{\sqrt{n(o_j^t)}}).\hat{R}_{i:j}(o_{prev}) + \frac{1}{\sqrt{n(o_j^t)}}$
        else
            $\hat{R}_{i:j}(o_{prev}) \leftarrow (1 - \frac{1}{\sqrt{n(o_j^t)}}).\hat{R}_{i:j}(o_{prev}) - \frac{1}{\sqrt{n(o_j^t)}}$
        end
    end
    Agent $j$ then observes the new consequence $o_j^{t+1}$
    Update memories:
    $A_{i:j}(o_j^t) = a_j^t$
    $O_{i:j}(o_j^t) = o_j^{t+1}$
    $n(o_j^t) \leftarrow n(o_j^t) + 1$
end
```

2.4 Model of itself seen by others

In order to model itself perceived by other agents, an agent processes exactly the same way used to model other agents. Thus, it infers its own reward function R_i given its previous actions and observations in order to estimate how other agents would infer its reward function. In the following sections, we denote as $\hat{R}_{i:(j:i)}$ this estimated function. As in (ref section 2.3), agent i uses its memories of its previous choices of action $A_{i:(j:i)}(o_i)$ and consequences $O_{i:(j:i)}(o_i)$ observed by another agent j for all possible observations o_i in order to update $\hat{R}_{i:(j:i)}$. Note that if all agents are aware of all the true observations of others and have the same initial estimations of others rewards (for instance, $R_{i:(j:i)}^0(o_i) = R_{j:i}^0(o_j) = 0 \; \forall i, j, o_i, o_j$), we then have the equality:

$$\hat{R}_{i:(j:i)}^t = \hat{R}_{j:i}^t \quad \forall t$$

3. EXPRESSING INTENTIONS

At this moment, our agents are just behaving in an “egoist” way, trying to maximize their own rewards and optionally they model others and themselves seen by others. But in order to promote cooperation, we provide any agent with the possibility to help other agent to infer its reward function. In that purpose, an agent can, each time it has not the expected observation and reward, move next time to another action even if the last one was in average the optimal choice. In other words, imagine the agent has learned that when it sees an observation *“light”* the optimal action is *“press-the-button”* and leads with probability 0.5 to *“glass-of-wine”* associated with positive reward ($R(\mathit{wine}) = 1$) and with probability 0.5 to *“electric-chock”* that is associated with negative reward ($R(\mathit{chock}) = -0.9$), while another action *“do-nothing”* always leads to *“nothing”* associated with a null reward ($R(\mathit{nothing}) = 0$). In that case, a RL-based behavior would always chose action “press-the-button” that leads in average to a positive reward ($\hat{r} = 0.1$). But another agent observing this behavior, with no additive information, would infer that both *“glass-of-wine”* and *“electric-chock”* are associated with positive rewards while *“nothing”* is associated with negative (or null) reward. Now, if the agent, each time it receives the electric chock, do nothing the very next time it sees the light, it becomes possible for an observer to guess it does not like the chock but tried the action because it wanted the glass of wine.

Formally, the agent is using its model of itself seen by others: when agent i is perceiving o_i, it looks at the true reward associated with the last consequence $O_{i:(j:i)}(o_i)$:

$$r = R_i\left(O_{i:(j:i)}(o_i)\right)$$

if this reward was acceptable (*e.g.* superior to a fixed threshold) the agent repeat the last action it did after observing o_i, hence $A_{i:(j:i)}(o_i)$. Otherwise, it chooses the best of the remaining actions (according to Q-values).

That way we enable agents to help each others in inferring their reward functions. Now our agents have the choice between two possible behaviors: the classical Q-learning or this expressing-intentions behavior (described step by step in algo. 3).

4. EMPATHY AND GRATITUDE

We finally provide our agents with intrinsic rewards (ref intrinsic motivation) that depend on how they estimate the rewards of other agents. We define two different intrinsic rewards that an agent can feel observing the other ones:

Empathy $e^t_{i:j}$ of an agent i observing an agent j at a time t is proportional to its estimation of the reward that j received:

$$e^t_{i:j} \propto \hat{R}_{i:j}(o^t_j)$$

Gratitude $g^t_{i:(j:i)}$ of an agent i observing an agent j at a time t is proportional to its estimation of *how j would infer i's own reward*:

$$g^t_{i:(j:i)} \propto \hat{R}_{i:(j:i)}(o^t_i)$$

Our model of empathy is based on de Waal's *Action-Percept Model* framework [4]. In this context, agents have a common set of possible actions or observations. Then empathy describes the capacity to be affected by and share the emotional state of another (inferred through this common set of action-perception).

Algorithm 3: Intuitive on-line IRL. Agent i is inferring its own reward function as it could be estimated by agent j.

```
Initialize $R_{i:j}(o)$
forall iterations $t$ do
    Agent $i$ observes $o_i^t$
    if $o_i^t$ has already been observed by $i$ then
        Remember:
        $a_{prev} = A_{i:(j:i)}(o_i^t)$ previous action of $i$ after $o_i^t$
        $r_{prev} = R_i\left(O_{i:(j:i)}(o_i)\right)$ previous reward
        if $r_{prev} > \theta$ then
            Repeat previous action $a_i^t = a_{prev}$
        else
            Choose a different action $a_i^t \neq a_{prev}$ using Q
        end
        Take action $a_i^t$, receive $r^{t+1}$, $o^{t+1}$
        Update memories:
        $A_{i:(j:i)}(o_i^t) = a_i^t$
        $O_{i:(j:i)}(o_i^t) = o_i^{t+1}$
        $n(o_i^t) \leftarrow n(o_i^t) + 1$
    end
end
```

The intrinsic reward for gratitude is based on the idea that “*it's the thought that counts*”, expression used to indicate that it is the kindness behind an act that matters, however imperfect or insignificant the act may be.

Now, at time t, as agent i observes a signal o^t_i, it receives a total reward $\mathfrak{r}^t_i$, that is sum of extrinsic ($R_i(o^t_i)$) and intrinsic rewards (empathy and gratitude):

$$\mathfrak{r}^t_j = R_i(o_i) + \sum_{j \neq i} \alpha_i \, e^t_{i:j} + \beta_i \, g^t_{i:(j:i)}$$

Where α and β are control parameters that are used to try different situations. For example, we can compare agents that only feel empathy ($\beta = 0$) or only gratitude ($\alpha = 0$). We can also explore negative values of α and β that could lead to aggressive behaviors.

5. PRISONER'S DILEMMA

The Prisoner's Dilemma (PD) is an ideal game to study the social behavior of our agents. In that context, we focus on a 2-agents system. Each agent has the choice between two actions *defect* or *cooperate*. If both agent choose to cooperate, they receive a reward $\mathcal{R}$. If they both defect, they receive a smaller reward $\mathcal{P}$. If one agent defect while the other cooperate, the agent that defected receives the highest reward $\mathcal{T}$ and the agent that cooperated receives the smallest reward $\mathcal{S}$. The generalized form of PD requires following conditions:

$$\mathcal{T} > \mathcal{R} > \mathcal{P} > \mathcal{S}$$

The payoff relationship $\mathcal{R} > \mathcal{P}$ implies that mutual cooperation is superior to mutual defection, while the payoff relationships $\mathcal{T} > \mathcal{R}$ and $\mathcal{P} > \mathcal{S}$ imply that defection is the dominant strategy for both agents. We implemented the iterated version of this game (IPD), where agents successively play this game and remember previous actions of their opponent. Classical RL-learning would always tend to the Nash equilibrium [19] that consists in always choosing defection.

We implemented an IPD with payoff $\mathcal{T} = 1$, $\mathcal{R} = 0.6$, $\mathcal{P} = 0$, $\mathcal{S} = -1$. Table 5 displays the payoff matrix of this game. Each game last 1000 iterations. At each iteration t, agent i chooses action $a_i^t \in \{cooperate,\ defect\}$ and receives a signal $o_i^t \in \{\mathcal{O}_\mathcal{R}, \mathcal{O}_\mathcal{S}, \mathcal{O}_\mathcal{T}, \mathcal{O}_\mathcal{P}\}$ associated with the corresponding reward ($R_i(\mathcal{O}_\mathcal{S}) = \mathcal{S}$, etc). Agent i also receive the action and the observation of the other agent, a_j^t and o_j^t. But agent are not aware of the payoff matrix that defines the reward of the other (in fact, it is the same).

	Cooperate	Defect
Cooperate	0.6, 0.6	-1, 1
Defect	1, -1	0, 0

Table 1: IPD payoff matrix

The Q-learning behavior of agents was implemented with parameters $\gamma = 0.8$, $\eta = 0.05$ and actions were chosen using the Gibbs softmax method:

$$a \sim \mathbb{P}[a|o] = \frac{e^{\tau Q(o,a)}}{\sum_b e^{\tau Q(o,b)}}$$

With temperature parameter $\tau = 5$.

6. RESULTS AND DISCUSSION

6.1 Pure Q-learning

We first tried to let our agents behave without expressing intention and with no intrinsic reward for empathy or gratitude ($\alpha = \beta = 0$). As expected, agents quickly tend to the Nash equilibrium and always defect (ref to fig1). Each agent learned a wrong reward function for the other. Table 6.1 shows the average resulting reward functions learned by the agents over 50 IPD game with 1000 iterations. We can see that with a small variance agents successfully learned that the other has negative reward $\mathcal{S}$, but as the other was always defecting at the end, both thought that the other had strong positive reward $\mathcal{P}$ that is, in fact, null (see column $\mathcal{P}$ of table 6.1).

	$\mathcal{T}$	$\mathcal{R}$	$\mathcal{P}$	$\mathcal{S}$
Truth	1	0.6	0	-1
$\hat{R}_{1:2}$	0.34	-0.13	0.90	-0.75
σ^2	0.39	0.52	0.16	0.13
$\hat{R}_{2:1}$	0.25	-0.16	0.92	-0.77
σ^2	0.36	0.46	0.15	0.094

Table 2: $\alpha = 0$, $\beta = 0$. Average learned other's rewards functions by agents 1 and 2 over 50 IPD games and variances. We can see that with a small variance agents successfully learned that the other has negative reward $\mathcal{S}$, but as the other was always defecting at the end, both thought that the other had strong positive reward $\mathcal{P}$ that is, in fact, null (see yellow cells).

6.2 Q-learning with empathy & gratitude

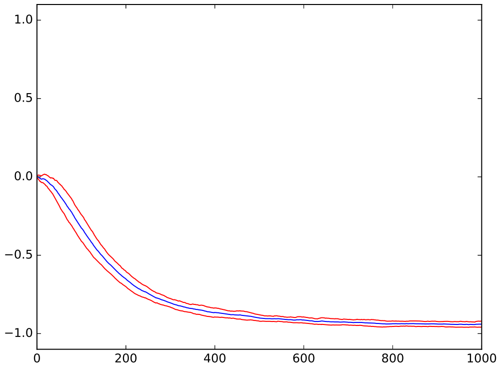

Figure 1: $\alpha = 0$, $\beta = 0$. Average trajectory of defect-cooperate ratio over 50 IPD games and variances. +1 represent a full cooperation (both agents cooperate) and -1 represent a full defection (both agents defect). the trajectory is computed with an exponential moving average of this ratio.

Gentle vs gentle: Here we look at the behavior of agents where both receive positive intrinsic reward for empathy and gratitude ($\alpha = 0.9$, $\beta = 0.3$). Paradoxically, it sped up Nash equilibrium's attraction (ref to fig2). As in pure Q-learning situation, both agents learned a false reward function where $\mathcal{P}$ is hight for the other (see column $\mathcal{P}$ of table 6.2). Thus, they were intrinsically rewarded by empathy while they where defecting. Furthermore, as they also learned that the other is punished while they both cooperate (see column $\mathcal{R}$ of table 6.2), they where intrinsically punished while they cooperated.

	$\mathcal{T}$	$\mathcal{R}$	$\mathcal{P}$	$\mathcal{S}$
Truth	1	0.6	0	-1
$\hat{R}_{1:2}$	0.39	-0.28	1.	-0.69
σ^2	1.9e-01	2.2e-01	1.7e-27	7.8e-02
$\hat{R}_{2:1}$	0.56	-0.32	0.99	-0.62
σ^2	1.6e-01	2.5e-01	7.7e-08	9.6e-02

Table 3: $\alpha = 0.9$, $\beta = 0.3$. Average learned other's rewards functions by agents 1 and 2 over 50 IPD games and variances. We can see that with a small variance agents successfully learned that the other has negative reward $\mathcal{S}$, but as the other was always defecting at the end, both thought that the other had strong positive reward $\mathcal{P}$ that is, in fact, null. Furthermore, as they also learned that the other is punished while they both cooperate and receive reward $\mathcal{R}$ (see yellow cells).

Agressive vs agressive: This time we looked at the opposite situation, where both agent were intrinsically punished by empathy or gratitude ($\alpha = -0.9$, $\beta = -0.3$). Again paradoxically, it slowed down Nash equilibrium's attraction. For the same reason: as both agent learned the other is rewarded by $\mathcal{P}$, they are intrinsically punished when they defect while the other is cooperating (see column $\mathcal{P}$ of table 6.2).

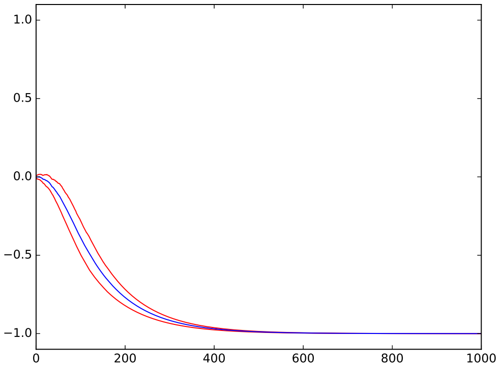

Figure 2: $\alpha = 0.9$, $\beta = 0.3$. Average trajectory of defect-cooperate ratio over 50 IPD games and variances. +1 represent a full cooperation (both agents cooperate) and -1 represent a full defection (both agents defect). the trajectory is computed with an exponential moving average of this ratio.

	$\mathcal{T}$	$\mathcal{R}$	$\mathcal{P}$	$\mathcal{S}$
Truth	1	0.6	0	-1
$\hat{R}_{1:2}$	0.79	-0.23	0.58	-0.41
σ^2	0.073	0.14	0.16	0.089
$\hat{R}_{2:1}$	0.65	-0.16	0.55	-0.37
σ^2	0.14	0.16	0.15	0.083

Table 4: $\alpha = -0.9$, $\beta = -0.3$. Average learned other's rewards functions by agents 1 and 2 over 50 IPD games and variances. We can see that with a small variance agents successfully learned that the other has negative reward $\mathcal{S}$, but as the other was always defecting at the end, both thought that the other had strong positive reward $\mathcal{P}$ that is, in fact, null (see yellow cells).

6.3 Expressing intentions with empathy & gratitude

Here we implemented the expressing-intentions behavior described in section 3. The choice of the threshold θ that determines if the previous reward was worth to repeat the previous action is tricky in the case of IPD. If $\mathcal{P} < \theta \leq \mathcal{R}$, then agents always cooperate. Indeed, as soon as both agents defect, they simultaneously change to cooperation and keep cooperating till the end. For a similar reason, if $\mathcal{P} \geq \theta$, then, if both agents start by cooperation, they always cooperate otherwise they always defect. In both cases, they can not efficiently learn the reward function of the other. This singularity comes from the fact agents just have two possibilities of action. To avoid this problem, we used a random threshold θ that is, with probability $p = \mathcal{R}$, higher than $\mathcal{R}$ and with probability $1 - p$ smaller than $\mathcal{R}$ (which amounts, in our case, to take θ uniformly in [0;1]). In a sens, this stochastic choice represents the hesitation of agents between two temptations: to be content with $\mathcal{R}$ or to focus on maximal reward $\mathcal{T}$.

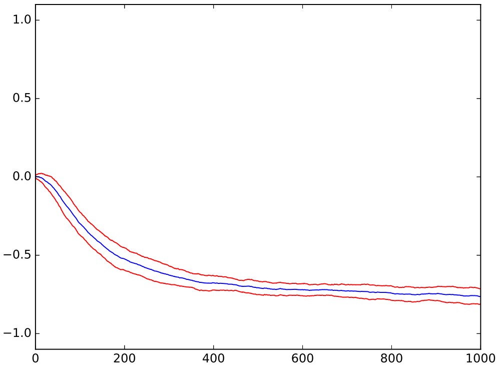

Figure 3: $\alpha = -0.9$, $\beta = -0.3$. Average trajectory of defect-cooperate ratio over 50 IPD games and variances. +1 represent a full cooperation (both agents cooperate) and -1 represent a full defection (both agents defect). the trajectory is computed with an exponential moving average of this ratio.

In our simulations, at the beginning (while $t < 300$) both agents are following Q-learning behavior. Then, during a phase (from $t = 301$ up to $t = 700$) they express their intentions using the algorithm of section 3. Finally, assuming they had time to learn about each other, they move back to Q-learning till the end ($t = 1000$).

Only empathy: we first look at the resulting behavior when both agents just receive intrinsic reward for empathy ($\alpha = 0.9$, $\beta = 0$). As a result, at the beginning agents are attracted by Nash equilibrium. Then, while they were expressing their intentions, they defected as much as they cooperated in average. After this expressing phase, agent could better understand each other's intentions (see table 6.3) and, led by intrinsic reward for empathy, they started to always cooperate (see figure 6.3).

	$\mathcal{T}$	$\mathcal{R}$	$\mathcal{P}$	$\mathcal{S}$
Truth	1	0.6	0	-1
$\hat{R}_{1:2}$	0.56	0.99	-0.92	-0.68
σ^2	7.6e-02	3.0e-09	9.1e-02	1.0e-01
$\hat{R}_{2:1}$	0.54	0.99	-0.88	-0.72
σ^2	9.3e-02	4.6e-10	9.5e-02	9.3e-02

Table 5: $\alpha = 0.9$, $\beta = 0$. Average learned other's rewards functions by agents 1 and 2 over 50 IPD games and variances. Between times $t = 301$ and $t = 700$ agents were following expressing-intention behavior. Agents could learn each other's intentions and understood that $\mathcal{T}$ and $\mathcal{R}$ are positive rewards for the other. As they finally always cooperated (because of empathy), they estimated other's rewards higher for $\mathcal{R}$ than for $\mathcal{T}$(see yellow cells).

Only Gratitude: this time both agents just receive intrinsic reward for gratitude ($\alpha = 0.$, $\beta = 0.9$). As a result, at the beginning agents are attracted by Nash equilibrium. Then, while they were expressing their intentions, they defected as much as they cooperated in average. After this expressing phase, agent could not understand each other's

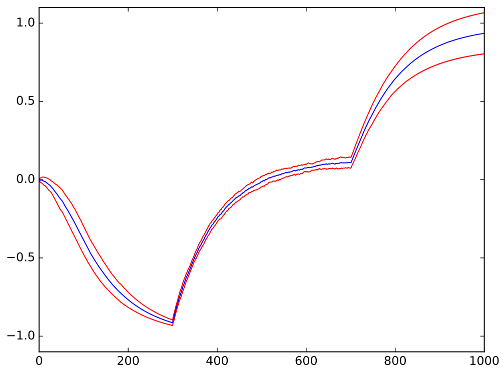

Figure 4: $\alpha = 0.9$, $\beta = 0$. Average trajectory of defect-cooperate ratio over 50 IPD games and variances. +1 represent a full cooperation (both agents cooperate) and -1 represent a full defection (both agents defect). the trajectory is computed with an exponential moving average of this ratio. Between times $t = 301$ and $t = 700$ agents were following expressing-intention behavior.

intentions (see table 6.3) and although led by gratitude, they started to always defect (see figure 6.3).

	$\mathcal{T}$	$\mathcal{R}$	$\mathcal{P}$	$\mathcal{S}$
Truth	1	0.6	0	-1
$\hat{R}_{1:2}$	0.58	-0.068	0.99	-0.68
σ^2	5.3e-02	6.9e-02	3.9e-07	3.9e-02
$\hat{R}_{2:1}$	0.58	-0.064	0.99	-0.66
σ^2	0.034	0.095	8.5e-4	0.034

Table 6: $\alpha = 0$, $\beta = 0.9$. Average learned other's rewards functions by agents 1 and 2 over 50 IPD games and variances. Between times $t = 301$ and $t = 700$ agents were following expressing-intention behavior. We can see that with a small variance agents successfully learned that the other has negative reward $\mathcal{S}$, but as the other was always defecting at the end, both thought that the other had strong positive reward $\mathcal{P}$ that is, in fact, null (see yellow cells)

Empathy and gratitude: Finally we look at the resulting behavior when both agents receive intrinsic rewards for both empathy and gratitude ($\alpha = 0.9$, $\beta = 0.3$). As in only-empathy condition, agents successfully understood each other's intentions (see table 6.3). But adding the intrinsic reward for gratitude sped up the cooperation after the expressing-intention phase, increasing the probability of double cooperation (see figure 6.3).

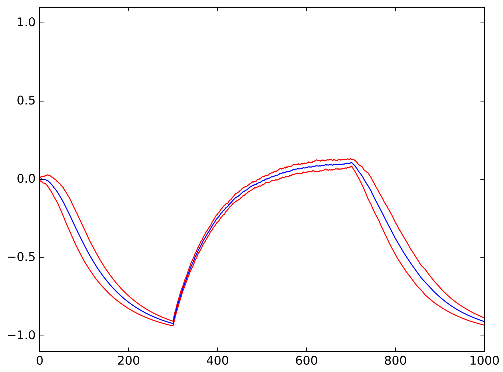

Figure 5: $\alpha = 0$, $\beta = 0.9$. Average trajectory of defect-cooperate ratio over 50 IPD games and variances. +1 represent a full cooperation (both agents cooperate) and -1 represent a full defection (both agents defect). the trajectory is computed with an exponential moving average of this ratio. Between times $t = 301$ and $t = 700$ agents were following expressing-intention behavior.

	$\mathcal{T}$	$\mathcal{R}$	$\mathcal{P}$	$\mathcal{S}$
Truth	1	0.6	0	-1
$\hat{R}_{1:2}$	0.62	1.	-0.95	-0.7
σ^2	7.0e-02	1.8e-17	1.4e-02	8.6e-02
$\hat{R}_{2:1}$	0.55	1.	-0.933	-0.77
σ^2	7.7e-02	1.9e-17	2.2e-02	7.4e-02

Table 7: $\alpha = 0.9$, $\beta = 0.3$. Average learned other's rewards functions by agents 1 and 2 over 50 IPD games and variances. Between times $t = 301$ and $t = 700$ agents were following expressing-intention behavior. Agents could learn each other's intentions and understood that $\mathcal{T}$ and $\mathcal{R}$ are positive rewards for the other. As they finally always cooperated (because of empathy), they estimated other's rewards higher for $\mathcal{R}$ than for $\mathcal{T}$(see yellow cells).

6.4 Playing with empathy

Regarding results of subsection 6.3 it appears that with expressing-intention phase, the empathy is a necessary and sufficient condition to reach cooperation, while gratitude added to empathy stabilize of this cooperation. This is why we finally focused just on empathy in order to explore all possible combination of the α parameters of both agents (α_1 for agent 1, α_2 for agent 2). For that, we divided the area of possible values in a grid of 20 values between -1 and 1 for both parameters α. We simulated 10 IDP games with an expressing intention phase for each of the 400 resulting combinations. We display the average final defect-cooperate ratio (the same measure used for all figure in the previous subsection) on a map reported in figure 6.4.

7. CONCLUSION

NEED CONCLUSION

REFERENCES

[1] S. Baron-Cohen, A. Leslie, and U. Frith. Does the autistic child have a "theory of mind" ? *Cognition*, 1985.

[2] H. H. Clark. *Using language*. Cambridge university press, 1996.

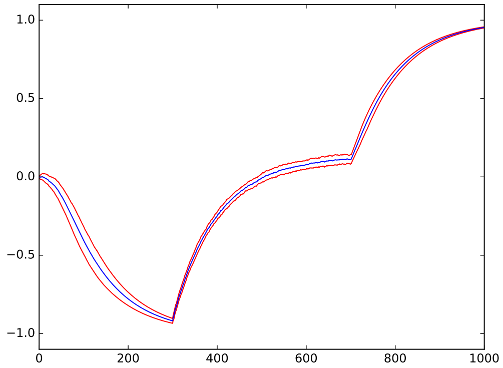

Figure 6: $\alpha = 0.9$, $\beta = 0.3$. Average trajectory of defect-cooperate ratio over 50 IPD games and variances. +1 represent a full cooperation (both agents cooperate) and -1 represent a full defection (both agents defect). the trajectory is computed with an exponential moving average of this ratio. Between times $t = 301$ and $t = 700$ agents were following expressing-intention behavior.

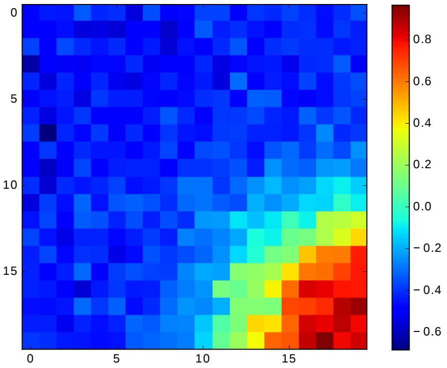

Figure 7: Average final defect-cooperate ratio over 10 IDP games for a grid of 400 possible (α_1, α_2) combinations. In each game, agents were following expressing-intention behavior between time $t = 301$ and $t = 700$. Red areas correspond to combinations that led to cooperation while blue areas correspond to combinations that led to Nash equilibrium. In green areas, agents where equally defecting and co-operating.

[3] H. H. Clark and S. E. Brennan. Grounding in communication. *Perspectives on socially shared cognition*, 13(1991):127–149, 1991.

[4] F. B. De Waal. Putting the altruism back into altruism: the evolution of empathy. *Annu. Rev. Psychol.*, 59:279–300, 2008.

[5] H. de Weerd, E. Broers, and R. Verbrugge. Savvy software agents can encourage the use of second-order theory of mind by negotiators. In *Proceedings of the 37th Annual Meeting of the Cognitive Science Society*, pages 542–547, 2015.

[6] H. de Weerd, R. Verbrugge, and B. Verheij. Agent-based models for higher-order theory of mind. In *Advances in Social Simulation*, pages 213–224. Springer, 2014.

[7] H. De Weerd, R. Verbrugge, and B. Verheij. Theory of mind in the mod game: An agent-based model of strategic reasoning. In *ECSI*, pages 128–136, 2014.

[8] H. De Weerd and B. Verheij. The advantage of higher-order theory of mind in the game of limited bidding. In *Proc. workshop reason. about other minds, ceur workshop proceedings*, volume 751, pages 149–164. Citeseer, 2011.

[9] P. J. Gmytrasiewicz and E. H. Durfee. A rigorous, operational formalization of recursive modeling. In *ICMAS*, pages 125–132, 1995.

[10] K. Haroush and Z. M. Williams. Neuronal prediction of opponents behavior during cooperative social interchange in primates. *Cell*, 160(6):1233–1245, 2015.

[11] Z.-j. Jin, H. Qian, S.-y. Chen, and M.-l. Zhu. Convergence analysis of an incremental approach to online inverse reinforcement learning. *Journal of Zhejiang University SCIENCE C*, 12(1):17–24, 2011.

[12] G. Knoblich and N. Sebanz. Evolving intentions for social interaction: from entrainment to joint action. *Philosophical Transactions of the Royal Society of London B: Biological Sciences*, 363(1499):2021–2031, 2008.

[13] S. Lemaignan and P. Dillenbourg. Mutual modelling in robotics: Inspirations for the next steps. In *Proceedings of the 2015 ACM/IEEE Human-Robot Interaction Conference*, 2015.

[14] B. Meijering, H. Van Rijn, N. Taatgen, and R. Verbrugge. I do know what you think i think: Second-order theory of mind in strategic games is not that difficult. In *Proceedings of the 33rd annual conference of the cognitive science society*, pages 2486–2491, 2011.

[15] A. Y. Ng, S. J. Russell, et al. Algorithms for inverse reinforcement learning. In *Icml*, pages 663–670, 2000.

[16] E. Palagi. Sharing the motivation to play: the use of signals in adult bonobos. *Animal Behaviour*, 75(3):887–896, 2008.

[17] M. Puterman. Markov decision processes. 1994.

[18] D. V. Pynadath, M. Si, and S. C. Marsella. Modeling theory of mind and cognitive appraisal with decision-theoretic agents. *Social emotions in nature and artifact: emotions in human and human-computer interaction*, pages 70–87, 2011.

[19] T. W. Sandholm and R. H. Crites. Multiagent reinforcement learning in the iterated prisoner's dilemma. *Biosystems*, 37(1):147–166, 1996.

[20] P. Sequeira, F. S. Melo, and A. Paiva. The influence of social display in competitive multiagent learning. pages 64–69, 2014.

[21] C. J. Watkins and P. Dayan. Q-learning. *Machine learning*, 8(3-4):279–292, 1992.

[22] H. Weerd, R. Verbrugge, and B. Verheij. Negotiating with other minds: the role of recursive theory of mind in negotiation with incomplete information. *Autonomous Agents and Multi-Agent Systems*, pages

1–38, 2015.