Expressing Motivations By Facilitating Other's Inverse Reinforcement Learning

Alexis Jacq[1,2], **Wafa Johal**[1],
Ana Paiva[2], Pierre Dillenbourg[1]
[1]CHILI Lab, École Polytechnique Fédérale de Lausanne, Switzerland,
[2]Instituto Superior Técnico, University of Lisbon, Portugal

Abstract

It is often necessary to understand each other's motivations in order to cooperate. Reaching such a mutual understanding requires two abilities: to build models of other's motivations in order to understand them, and to build a model of "my" motivations *perceived by others* in order to be understood. Having a self-image seen by others requires two recursive orders of modeling, known in psychology as the first and second orders of theory of mind. In this paper, we present a second order-theory of mind cognitive architecture that aims to facilitate mutual understanding in multi-agent scenarios. We study different conditions of empathy and gratitude leading to irrational cooperation in iterated prisoner's dilemma.

1 Introduction

In any interaction that requires cooperation, agents need to understand each other's motivations. A spoken language – or even a gestural language, helps to reach such a mutual understanding [15][12]. Without languages, agents can still understand their respective motivations by observing their goal-directed behaviors. Moreover, they can assume goal-directed behaviors as a *common ground* [2] and adapt their own behavior in order to be more easily understood by others.

Developed by Baron-Cohen and Leslie [1], the *Theory of Mind* (ToM) describes the ability to attribute mental states and knowledge to others. In interactions, humans are constantly collecting and analyzing huge quantity of information to stay aware of emotions, goals, and understandings of their fellows. Different orders of ToM are defined to express how agents can recursively attribute a ToM to other agents: in the first order, an agent only models others as simpler agents, that do not perform ToM, while in the second order, an agent also models how others model others, including itself. It is possible to perpetuate this recursion up to infinite loops of mutual modeling, known as *infinite regress* in epistemic logics [3]. Agents based on second or higher orders ToM show better performances in a range of simple social games [5][6][7][13]. Higher orders, though outperforming the second-order in some cases [6], do not reveal important advantages in general [26]. Hence the second-order seems sufficient to generate rich social behaviors [19].

In multi-agents literature, ToM-based interactions are often presented within *Interactive Partially Observed Markov Decision Problems* (I-POMDPs) framework [9]. I-POMDPs are a generalization of POMDPs [11] where agents have to infer beliefs about their interactive states – that combine physical environment states and other agent motivations – and to learn a policy that optimizes the reception of rewards depending on these interactive states. Solving this problem can be described in two main steps. The belief update step is based on the history of observations of the world and previous actions. For instance, in [18], the Sigma (Σ) cognitive architecture [20] uses a *probabilistic graphical model* [8] in order to infer beliefs about others' intentions given the memory of previous events. Then, the decision step consists in choosing the best action given the present belief state in order to optimize the collection of rewards. This step is commonly approached as a *Markovian Decision Process* (MDP) [16] in the space of beliefs (belief-MDP). I-POMDP framework enables second-order ToM, as performed by PsychSim architecture [19].

As the usual way to model agents solving a (belief-)MDP is *Reinforcement Learning* (RL) [24], the purpose of *Inverse Reinforcement Learning* (IRL) [14] is to infer the motivations of an observed agent, assuming this agent uses a policy that optimizes a RL process. In that way, IRL provides an agent with a first-order ToM, modeling observed agents as optimized RL agents. In [17], they use the Σ architecture to model agents that perform RL in the belief-MDP and use IRL to infer others reward functions. In this paper, we model second-order ToM agents that also use IRL in order to infer their own reward function, as it could be inferred by others. In our approach, we provide each agent with 3 types of models. The first one, the *model of itself*, is the true policy of an agent, learned via RL. The second one, the *model of another*, represents the beliefs of the agent about another's policy, updated via IRL. The third one, the *model of itself seen by others*, represents the beliefs of the agent of how another agent could infer its policy, also updated via IRL.

We propose an algorithm that uses this second-order inference in order to facilitate the mutual understanding of agents rewards through an adaptation of the behavior. We also introduce intrinsic rewards based on models of empathy and gratitude, leading agents to cooperative interactions. Finally, we implement an iterative prisoner's dilemma in order to study the resulting behavior of our approach in a dual-agent system. Through simulations, we explore whether our models, with different conditions, facilitate the mutual understanding of motivations between agents.

2 Mutual modeling

2.1 Model of itself

An agent i makes goal-directed decisions as a RL agent: at 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 $o_i^{t+1} = \mathbf{O}(a_1^t, a_2^t, ..a_n^t)$ (n being the number of agents) and a reward that only depends 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 [22], this framework is simplified as a MDP where the observations are treated as states that just depend on agent's previous observation and action following an unknown probability distribution:

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

Hence, at the beginning, the decision making of the agent is performed by Q-learning [25]. 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
```

2.2 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 IRL. In this setup, the IRL must be performed on-line. In [10] they provide an incremental algorithm for on-line IRL in a MDP framework. 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 child – would infer the motivations of others. Hence we propose the following idea:

If I liked I repeat, otherwise I change: Supposing agent j receives an observation *"light"* and chooses action *"press-the-button"*. Then it receives a new observation *"glass-of-juice"*. If it liked the juice, probably the next time it will observe the signal *"light"* then 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 previous following 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, namely $oprev = O_{i:j}(o_j^t)$. In that case, agent i increments its inferred reward function $\hat{R}_{i:j}(o_j^t)$ for agent j as follow:

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

Where $n_{i:j}(o_j^t)$ 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:

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

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 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:
        $aprev = A_{i:j}(o_j^t)$ previous action of $j$ after $o_j^t$
        $oprev = O_{i:j}(o_j^t)$ previous consequence
        if $a_j^t = aprev$ then
            $\hat{R}_{i:j}(oprev) \leftarrow$
                $(1 - \frac{1}{\sqrt{n(o_j^t)}}).\hat{R}_{i:j}(oprev) + \frac{1}{\sqrt{n(o_j^t)}}$
        else
            $\hat{R}_{i:j}(oprev) \leftarrow$
                $(1 - \frac{1}{\sqrt{n(o_j^t)}}).\hat{R}_{i:j}(oprev) - \frac{1}{\sqrt{n(o_j^t)}}$
        end
    end
    Agent $j$ then observes the new consequence $o_j^{t+1}$
    Update memories:
    $A_{i:j}(o_j^t) = a_j^t$
    $O_{i:j}(o_j^t) = o_j^{t+1}$
    $n_{i:j}(o_j^t) \leftarrow n_{i:j}(o_j^t) + 1$
end
```

2.3 Model of itself seen by others

In order to model itself perceived by other agents, an agent i processes exactly the same way that it would model another agent: 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 before, 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, $R^0_{i:(j:i)}(o_i) = R^0_{j:i}(o_j) = 0 \; \forall i, j, o_i, o_j$), we then have the equality:

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

3 Expressing motivations

Until now, our agents are just behaving in an "egoist" way, trying to maximize their own rewards. But in order to promote cooperation, we provide any agent with a behavior that helps other agents to infer its own reward function. In that purpose, each time it is disappointed by a small reward (or a punishment), an agent can move the next time to another action even if the last one was, in average, the optimal choice. In other words, let's say the agent has learned that, each time it sees an observation *"light"*, the optimal action is *"press-the-button"*. This action leads with probability 0.5 to *"glass-of-wine"* associated with positive reward ($R(\textit{wine}) = 1$) and with probability 0.5 to *"electric-shock"* that is associated with negative reward ($R(\textit{shock}) = -0.9$), while another action *"do-nothing"* always leads to *"nothing"* associated with a null reward ($R(\textit{nothing}) = 0$). 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-shock"* are associated with positive rewards while *"nothing"* is associated with negative (or null) reward. Now, if the agent, each time it receives the electric shock, does nothing the very next time it sees the light, it becomes possible for an observer to guess it does not like the shock 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 previous following consequence $O_{i:(j:i)}(o_i)$:

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

If this reward was acceptable (*e.g.* superior to a fixed threshold) the agent repeats 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 other in inferring their reward functions. Now our agents have the choice between two possible behaviors: the classical Q-learning or this expressing-motivations behavior (described step by step in algo. 3).

Algorithm 3: Expressing-motivations behavior. Agent i helps agent j to infer i's reward function.

```
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
```

4 Empathy and gratitude

We finally provide our agents with intrinsic rewards [23] that depend on how they estimate the rewards of other agents. We define two different intrinsic rewards that agents can feel observing each other's:

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).

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 $\mathtt{r}^t_i$, sum of extrinsic ($R_i(o^t_i)$) and intrinsic (empathy and gratitude) rewards:

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

Where α and β are coefficients of proportionality that are used to try different situations. For example, we can compare agents that only feel empathy ($\alpha > 0$, $\beta = 0$) or only gratitude ($\alpha = 0$, $\beta > 0$). 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-agent system. Each agent has the choice between two actions: *defect* or *cooperate*. If both agents choose to cooperate, they receive a reward $\mathcal{R}$. If they both defect, they receive a smaller reward $\mathcal{P}$. If one agent defects while the other cooperates, 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 agents would systematically tend to the Nash equilibrium [21] 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 1 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 receives the action and the observation of the other agent, a_j^t and o_j^t. But agents are not aware of the payoff matrix that defines the rewards 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:

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

With temperature parameter $\tau = 5$.

6 Results and discussion

6.1 Pure Q-learning

We first tried to let our agents behave without expressing motivations and with no intrinsic rewards for empathy or gratitude ($\alpha = \beta = 0$). As expected, agents quickly tend to the Nash equilibrium and always defect (see figure 1). Each agent learned a wrong reward function for the other. Table 2 shows the average resulting reward functions learned by the agents over 50 IPD game with 1000 iterations. We can see that with variances agents successfully learned that the other has a negative reward $\mathcal{S}$, but, since 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 2).

	$\mathcal{T}$	$\mathcal{R}$	$\mathcal{P}$	$\mathcal{S}$
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: **Pure Q-learning** ($\alpha = 0$, $\beta = 0$). Average learned other's reward function by agents 1 and 2 over 50 IPD games and variances. We can see that with small variances agents successfully learned that the other has negative reward $\mathcal{S}$, but since the other was always defecting at the end, both thought that the other had strong positive reward $\mathcal{P}$ that was, in fact, null (see yellow cells).

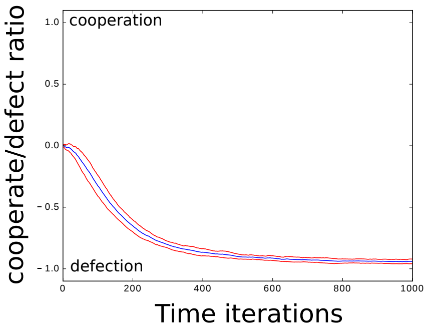

Figure 1: **Pure Q-learning** ($\alpha = 0$, $\beta = 0$). (blue) Average trajectory of defect-cooperate ratio over 50 IPD games and variances. +1 represents a full cooperation (both agents cooperate) and -1 represents a full defection (both agents defect). The trajectory is computed with an exponential moving average of this ratio. (red) +/- standard deviation.

6.2 Q-learning with empathy & gratitude

Gentle vs gentle: Here we look at the behavior of agents where both receive positive intrinsic rewards for empathy and gratitude ($\alpha = 0.9$, $\beta = 0.3$). Paradoxically, it sped up Nash equilibrium's attraction (see figure 2 A). As in pure Q-learning situation, both agents learned a false reward function where $\mathcal{P}$ is high for the other (see column $\mathcal{P}$ of table 3 A). Indeed, they were intrinsically rewarded by empathy while they were defecting. Furthermore, as they also learned that the other is punished while they both cooperate (see column $\mathcal{R}$ of table 3 A), they were intrinsically punished while they cooperated.

Aggressive vs aggressive: This time we looked at the opposite situation, where both agents were intrinsically punished by empathy or gratitude ($\alpha = -0.9$, $\beta = -0.3$). Again paradoxically, it slowed down Nash equilibrium's attraction (see figure 2 B). For the same reason: since both agents learned the other is rewarded by $\mathcal{P}$, they were intrinsically punished when they defected while the other was cooperating (see column $\mathcal{P}$ of table 3 B).

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

		$\mathcal{T}$	$\mathcal{R}$	$\mathcal{P}$	$\mathcal{S}$
B	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 3: **A**: **Gentle vs gentle** ($\alpha = 0.9, \beta = 0.3$). **B**: **Agressive vs agressive** ($\alpha = -0.9, \beta = -0.3$). Average learned other's reward function by agents 1 and 2 over 50 IPD games and variances. We can see that with small variances agents successfully learned that the other has negative reward $\mathcal{S}$, but since the other was always defecting at the end, both thought that the other had strong positive reward $\mathcal{P}$ that was, in fact, null. Furthermore, they also learned that the other was punished while they were both cooperating and receiving reward $\mathcal{R}$ (see yellow cells).

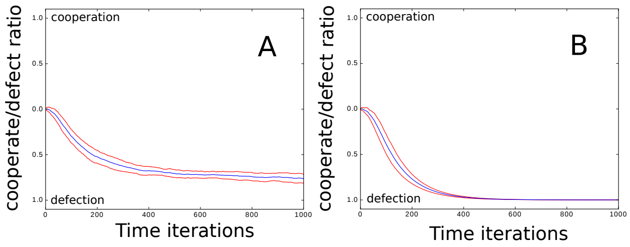

Figure 2: **A**: **Gentle vs gentle** ($\alpha = 0.9, \beta = 0.3$). **B**: **Agressive vs agressive** ($\alpha = -0.9, \beta = -0.3$). (blue) Average trajectory of defect-cooperate ratio over 50 IPD games and variances. +1 represents a full cooperation (both agents cooperate) and -1 represents a full defection (both agents defect). The trajectory is computed with an exponential moving average of this ratio. (red) +/- standard deviation.

6.3 Expressing motivations with empathy & gratitude

Here we implemented the expressing-motivations 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 that 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 way, this stochastic choice represents the hesitation of agents between two temptations: to be content with $\mathcal{R}$ or to focus on maximal reward $\mathcal{T}$.

In our simulations, at the beginning (from $t = 0$ up to $t = 300$) both agents are following Q-learning behavior. Then, during a phase (from $t = 301$ up to $t = 700$) they express their motivations 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 (from $t = 701$ up to $t = 1000$).

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 were attracted by Nash equilibrium. Then, while they were expressing their motivations, in average they defected as much as they cooperated. After this expressing phase, agent could better understand each other's motivations (see table 4 A) and, led by intrinsic reward for empathy, they started to always cooperate (see figure 3 A).

Only gratitude: this time both agents just receive intrinsic reward for gratitude ($\alpha = 0., \beta = 0.9$). As a result, at the beginning agents were attracted by Nash equilibrium. Then, while they were expressing their motivations, they defected as much as they cooperated in average. After this expressing phase, agent could not understand each other's motivations (see table 4, B) and although led by gratitude, they started to always defect (see figure 3 B).

Empathy and gratitude: Finally we look at the resulting behavior when both agents receive intrinsic rewards for both empathy and gratitude ($\alpha = 0.9, \beta = 0.3$). Like in only-empathy condition, agents successfully understood each other's motivations (see table 4 C). But adding the intrinsic reward for gratitude sped up the cooperation after the expressing-motivations phase, increasing the frequency of double cooperation (see figure 3 C).

6.4 Playing with empathy

Regarding results of subsection 6.3 it appears that with expressing-motivations phases, empathy is a necessary and sufficient condition to reach cooperation, while gratitude added to empathy stabilizes 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 motivations phase for each of the 400 resulting combinations. We displayed the average final defect-cooperate ratio (the same measure used for all figure in the previous subsection) on a map reported in figure 3 D. We can see that cooperation only

A

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

B

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

C

	$\mathcal{T}$	$\mathcal{R}$	$\mathcal{P}$	$\mathcal{S}$
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 4: **A**: **Only empathy** ($\alpha = 0.9, \beta = 0$). **B**: **Only gratitude** ($\alpha = 0, \beta = 0.9$). **C**: **Empathy and gratitude** ($\alpha = 0.9, \beta = 0.3$). Average learned other's reward function by agents 1 and 2 over 50 IPD games and variances. Between times $t = 301$ and $t = 700$ agents were following expressing-motivations behavior. Agents could learn each other's motivations 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).

occurs when both agents have a higher enough intrinsic reward for empathy ($\alpha >\sim 0.5$ in this case). Interestingly, at the edge between cooperations and Nash equilibrium's defections, appears a balanced zone, where agents equally defect or cooperate (see green area on figure 3 D).

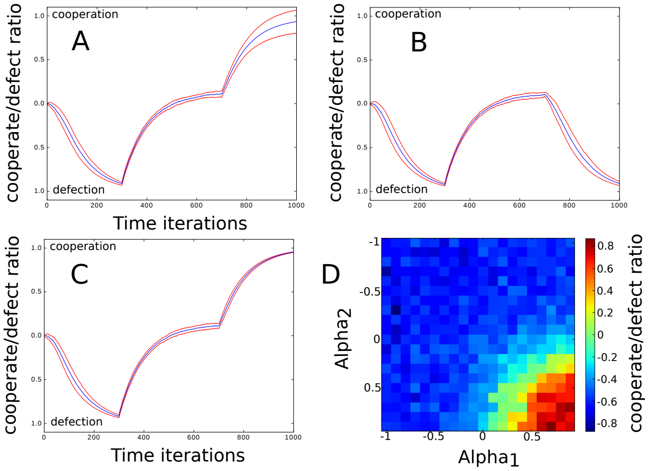

Figure 3: **A**: **Only empathy** ($\alpha = 0.9, \beta = 0$). **B**: **Only gratitude** ($\alpha = 0, \beta = 0.9$). **C**: **Empathy and gratitude** ($\alpha = 0.9, \beta = 0.3$). (blue) Average trajectory of defect-cooperate ratio over 50 IPD games and variances. +1 represents a full cooperation (both agents cooperate) and -1 represents 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-motivations behavior. (red) +/- standard deviation. **D**: 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 adopting expressing-motivations 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 were equally defecting and cooperating.

7 Conclusion

Cooperation in IPD game is said to be irrational. Regarding animals and humans behaviors, we can however observe quantities of such irrational behaviors emerging out of a stressful world, such as friendship or love. Models of natural selection brought global explanations of this emergence, but the detailed causes and mechanisms leading to such complex abilities in animals remain unknown. We strongly believe that no social behaviors would spring up without conditions implying that it was rational to behave that way. With this work, we claim that cooperation is no more irrational when agents can express and understand each other's motivations.

We introduce a cognitive architecture enabling a second order of theory of mind for social agents. This architecture is not recursive in the sense that each agent develop models for itself, others or itself perceived by others and none of these models recursively enable a theory of mind. Agents are modeled as RL-agents and use IRL to model others or themselves seen by others. In this framework, it is possible to design a decision making algorithm aiming to enable agents to express each other's motivations. We add two intrinsic rewards based on empathy and gratitude, empathy being the ability to feel others rewards while gratitude is the ability to feel how others would estimate its own rewards. Through an 2-agent system based on IPD game, we show that when agents can express their motivations, the intrinsic reward for empathy is a necessary and sufficient condition to promote cooperation, while gratitude added to empathy seems to speed up and to stabilize this cooperation.

We make the assumption that the behavior used to express motivation could also be understood by humans and hence improve Human-Robot interactions, especially in cooperative tasks. In further work, we will explore HRI testbeds, using *e.g.* a PD game within a human vs robot context. Our question regards to what extent humans would understand the expressing-motivations behavior, and to what extent the robot's IRL would infer humans motivations.

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.

[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. Theory of mind in the mod game: An agent-based model of strategic reasoning. In *ECSI*, pages 128–136, 2014.

[7] 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.

[8] N. Friedman and D. Koller. Probabilistic graphical models: Principles and techniques, 2009.

[9] P. J. Gmytrasiewicz and P. Doshi. Interactive pomdps: Properties and preliminary results. In *Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems-Volume 3*, pages 1374–1375. IEEE Computer Society, 2004.

[10] 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.

[11] L. P. Kaelbling, M. L. Littman, and A. R. Cassandra. Planning and acting in partially observable stochastic domains. *Artificial intelligence*, 101(1):99–134, 1998.

[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] 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.

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

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

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

[17] D. V. Pynadath, P. S. Rosenbloom, and S. C. Marsella. Reinforcement learning for adaptive theory of mind in the sigma cognitive architecture. In *International Conference on Artificial General Intelligence*, pages 143–154. Springer, 2014.

[18] D. V. Pynadath, P. S. Rosenbloom, S. C. Marsella, and L. Li. Modeling two-player games in the sigma graphical cognitive architecture. In *International Conference on Artificial General Intelligence*, pages 98–108. Springer, 2013.

[19] 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.

[20] P. S. Rosenbloom. The sigma cognitive architecture and system. *AISB Quarterly*, 136:4–13, 2013.

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

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

[23] S. Singh, R. L. Lewis, A. G. Barto, and J. Sorg. Intrinsically motivated reinforcement learning: An evolutionary perspective. *IEEE Transactions on Autonomous Mental Development*, 2(2):70–82, 2010.

[24] R. S. Sutton and A. G. Barto. *Reinforcement learning: An introduction*, volume 1. MIT press Cambridge, 1998.

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

[26] 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.