LEARNING ACTION SELECTION STRATEGIES IN COMPLEX SOCIAL SYSTEMS

Marco Remondino, Anna Maria Bruno and Nicola Miglietta
Management and Administration Department, University of Turin, Italy

Keywords: Management, Action selection, Reinforcement learning.

Abstract: In this work, a new method for cognitive action selection is formally introduced, keeping into consideration an individual bias for the agents: ego biased learning. It allows the agents to adapt their behaviour according to a payoff coming from the action they performed at time t-1, by converting an action pattern into a synthetic value, updated at each time, but keeping into account their individual preferences towards specific actions. In agent based simulations, the many entities involved usually deal with an action selection based on the reactive paradigm: they usually feature embedded strategies to be used according to the stimuli coming from the environment or other entities. The actors involved in real Social Systems have a local vision and usually can only see their own actions or neighbours' ones (bounded rationality) and sometimes they could be biased towards a particular behaviour, even if not optimal for a certain situation. Some simulations are run, in order to show the effects of biases, when dealing with an heterogeneous population of agents.

1 INTRODUCTION

Multi agent models allow to capture the complexity by modeling the system from the bottom, by defining the agents' behavior and the rules of interaction among them and the environment. Agent Based Simulation (ABS), in this field, is not only about understanding the individual behavior of agents, or in optimizing the interaction among them, in order to coordinate their actions to reach a common goal, like in other Multi Agent Systems (MAS), but above all it's about re-creating a real social system (e.g.: a market, an enterprise, a biological system) in order to analyze it as if it were a virtual laboratory for experiments. Reactive agents or cognitive ones can be employed in multi agent systems (Remondino, 2005); while the former model deals with the stimulus-reaction paradigm, the latter provides a "mind" for the agents, that can decide which action to take at the next step, based on their previous actions and the state of the world. When dealing with the problem of action selection, reactive agents simply feature a wired behavior, deriving from some conditional embedded rules that cannot be changed by the circumstances, and must be foreseen and wired into them by the model designer. For example given a set of agent's states $X(x_1, x_2)$, and a set of agent's actions $A(a_1, a_2)$, a deterministic reactive agent could consist of a set of rules like:

$$if\ X = x_1 then\ A = a_1; else\ A = a_2 \quad (1)$$

Or, if we have a wider set of states $X(x_1, \dots, x_n)$, with $k \leq n$, but again just a binary set of actions to be performed $A(a_1, a_2)$, the rule could be as follows:

$$if\ X = range(x_1, x_k)\ then\ A = a_1; else\ A = a_2 \quad (2)$$

Where "range" is used to synthetically indicate the states among *1* and *k*, in the system. This can be applied in the same way with several discrete intervals, instead of just two. Of course, if the actions are more than two, like a set $A(a_1, \dots, a_n)$, the rules can be simply multiplied as follows:

$$\begin{cases} if\ X = x_1 then\ A = a_1 \\ if\ X = x_2 then\ A = a_2 \\ \dots \\ if\ X = x_n then\ A = a_n \end{cases} \quad (3)$$

And linear combinations of both sets. The state of the agent could be function of both a stimulus coming from the environment and of some action performed by other agents or by the agent itself.

Reactive agents could also be stochastic, in the sense that they could have a probabilistic distribution connected to their action selection function. For each action/state combination a probability function

Remondino M., Maria Bruno A. and Miglietta N. (2010).
LEARNING ACTION SELECTION STRATEGIES IN COMPLEX SOCIAL SYSTEMS.
In *Proceedings of the 2nd International Conference on Agents and Artificial Intelligence - Agents*, pages 274-281
DOI: 10.5220/0002706802740281

is defined, so that for a defined $x_k \in X$ and a defined $a_h \in A$ we have that:

$$P(x_k, a_h) = \lambda \text{ where } 0 \leq \lambda \leq 1 \quad (4)$$

And

$$\sum_{i=1}^{n} P(x_k, a_i) = 1 \quad (5)$$

As an example we may think of a simple situation in which we have two possible actions a_1, a_2 so that for a particular state x_k we could have that $P(x_k, a_1) = 0.2$ and that $P(x_k, a_2) = 0.8$, meaning that the agent, when facing the state x_k will perform 20% of the times the action a_1 and 80% of the times action a_2, using a uniform distribution.

Some actions could feature a probability of 0 for certain states, and others could have a probability of 1 for a given state. If all the probabilities for the actions, given a state, are either 0 or 1, we are back at the deterministic situation presented above since, in that particular case, only one action could be performed (probability equal to 1) while the others would never be performed (probability equal to 0).

Reactive agents can be good for simulations, since the results obtained by employing them are usually easily readable and comparable (especially for ceteris paribus analysis). Besides, when the agent's behavior is not the primary focus, reactive agents, if their rules are properly chosen, can give very interesting aggregate results, often letting emergent system properties emerge at a macro level. Though, in situations in which, for example, learning coordination is important, or the focus is on exploring different behaviors in order to dynamically choose the best one for a given state, or simply agent's behavior is the principal topic of the research, cognitive agents could be employed, embedded with some learning technique. Besides, if the rules of a reactive agent are not chosen properly, they could bias the results; these rules, in fact, are chosen by the designer and could thus reflect her own opinions about the modeled system. Since many ABS of social systems can formulated as stage games with simultaneous moves made by the agents, some learning techniques derived from this field can be embedded into them, in order to create more realistic response to the external stimuli, by endowing the agents with a self adapting ability. Though, multi-agent learning is more challenging than single-agent, because of two complementary reasons. Treating the multiple agents as a single agent increases the state and action spaces exponentially and is thus unusable in multi agent simulation, where so many entities act at the same time. On the other hand, treating the other agents as part of the environment makes the environment non-stationary and *non-Markovian* (Mataric, 1997). In particular, ABS are non-Markovian systems if seen from the point of view of the agents (since the new state is not only function of the individual agent's action, but of the aggregate actions of all the agents) and thus traditional Q-learning algorithms (Watkins, 1989; Sutton and Barto, 1998) cannot be used effectively: the actors involved in real Social Systems have a local vision and usually can only see their own actions or neighbours' ones (bounded rationality) and, above all, the resulting state is function of the aggregate behaviours, and not of the individual ones.

While, as discussed in Powers and Shoham (2005), in iterated games learning is derived from facing the same opponent (or another one, with the same goals), in social systems the subjects can be different and the payoff could not be a deterministic or stochastic value coming from a payoff matrix. More realistically, in social systems the payoff could be a value coming from the dynamics of interaction among many entities and the environment, and could have different values, not necessarily within a predefined scale. Besides, social models are not all and only about coordination, like iterated games, and agents could have a bias towards a particular behavior, preferring it even if that's not the best of the possible ones. An example from the real world could be the adoption of a technological innovation in a company: even though it can be good for the enterprise to adopt it, the managerial board could be biased and could have a bad attitude towards technology, perceiving a risk which is higher than the real one. Thus, even by looking at the positive figures coming from market studies and so on, they could decide not to adopt it. This is something which is not taken into consideration by traditional learning methods, but that should be considered in ABS of social systems, where agents are often supposed to mimic some human behavior. Besides, when the agents are connected through a social network, the experience behind a specific action could be shared with others, and factors like the individual reputation of other agents could be an important bias to individual perception. In order to introduce these factors, a formal method is presented in the paper: *Ego Biased Learning (EBL)*. Another paradigm is briefly described as a future development, called *Reputation Based Socially Biased Learning*.

The purpose of this work is not that of supplying an optimized algorithm for reinforcement learning (RL); instead, the presented formalisms mimic as much as possible the real cognitive process taken by human agents involved in a social complex system, when needing to take an individual strategic decision; this is useful to study aggregate results.

2 REINFORCEMENT LEARNING

Learning from reinforcements has received substantial attention as a mechanism for robots and other computer systems to learn tasks without external supervision.

The agent typically receives a positive payoff from the environment after it achieves a particular goal, or, even simpler, when a performed action gives good results. In the same way, it receives a negative (or null) payoff when the action (or set of actions) performed brings to a failure. By performing many actions overtime (trial and error technique), the agents can compute the expected values (EV) for each action. According to Sutton and Barto (1998) this paradigm turns values into behavioral patterns; in fact, each time an action will need to be performed, its EV, will be considered and compared with the EVs of other possible actions, thus determining the agent's behavior, which is not wired into the agent itself, but self adapting to the system in which it operates.

Most RL algorithms are about coordination in multi agents systems, defined as the ability of two or more agents to jointly reach a consensus over which actions to perform in an environment. In these cases, an algorithm derived from the classic Q-Learning technique (Watkins, 1989) can be used. The EV for an action – $EV(a)$ – is simply updated every time the action is performed, according to the following, reported by Kapetanakis and Kundenko (2004):

$$EV(a) \leftarrow EV(a) + \lambda(p - EV(a)) \quad (6)$$

Where $0 < \lambda < 1$ is the learning rate and p is the payoff received every time that action a is performed.

This is particularly suitable for simulating multi stage games (Fudenberg and Levine 1998), in which agents must coordinate to get the highest possible aggregate payoff. For example, given a scenario with two agents (A and B), each of them endowed with two possible actions a_1, a_2 and b_1, b_2 respectively, the agents will get a payoff, based on a payoff matrix, according to the combination of performed actions. For instance, if a_1 and b_1 are performed at the same time, both agents will get a positive payoff, while for all the other combinations they will receive a negative reward.

Modifications of the (6) have been introduced to make the converging process faster and more efficient under these conditions.

2.1 Learning and Social Simulation

ABS applied to social system is not necessarily about coordination among agents and convergence to the optimal behaviour, especially when focusing on the aggregate level; it's often more important to have a realistic behaviour for the agents, in the sense that it should replicate, as much as possible, that of real individuals.

The aforementioned RL algorithm analytically evaluates the best action based on historical data, i.e.: the EV of the action itself, over time. This makes the agent perfectly rational, since it will evaluate, every time he has to perform it, the best possible action found till then. If this is very useful for computational problems where convergence to an optimal behaviour is important, it's not realistic when applied to a simulation of a social system. In this kind of systems, learning should keep into account the human factor, in the shape of perception biases, preferences, prejudice, external influences and so on. When a human (or an organization driven by humans) faces an alternative, the past results, though important for evaluation, are just one of the many components behind the action selection process. As an example we could think of the innovation adoption process; while a technological innovation could provide money savings and improved life style, it often spreads much slower than it should. This is due mainly to the resistance to innovation, typical of many human beings. If the humans worked in the same way as expressed with equation (6), then an innovation bringing even the smallest saving should be adopted immediately.

Another effective example is to be found in social systems; when deciding which action to perform, humans are usually biased by the opinion of their neighbours (e.g.: friends, colleagues, ad so on). This means that their individual experience is important, but not the only driver behind the action to perform, while other variables are considered and should be introduced in the evaluation process, when dealing with a simulation of a social system, in order to improve the realism of the model and to focus on aggregate results.

Traditional learning models can't represent individualities in a social system, or else they represented all of them in the same way – i.e.: as focused and rational agents; since they ignore many other aspects of behaviour that influence how humans make decisions in real life, these models do not accurately represent real users in social contexts.

3 EGO BIASED LEARNING

While discussing the cognitive link among preferences and choices is definitely beyond the purpose of this work, it's important to notice that it's commonly accepted that the mentioned aspects are strictly linked among them. The link is actually bi-directional (Chen, 2008), meaning that human preferences influence choices, but in turn the performed actions (consequent to choices) can change original preferences.

As stated in Sharot et al. (2009): "*…past preferences and present choices determine attitudes of preferring things and making decisions in the future about such pleasurable things as cars, expensive gifts, and vacation spots*".

Even if preferences can be modified according to the outcome of past actions (and this is well represented by the RL algorithms described before), humans can keep an emotional part driving them to prefer a certain action over another one, even when the latter has proven better than the former. Some of these can be simply wired into the DNA, or could have formed in many years and thus being hardly modifiable. A bias is defined as "*a particular tendency or inclination, esp. one that prevents unprejudiced consideration of a question; prejudice*" (www.dictionary.com). That's the point behind learning: human aren't machines, able to analytically evaluate all the aspects of a problem and, above all, the payoff deriving from an action is filtered by their own perceptions. There's more than just a self-updating function for evaluating actions and in the following a formal RL method is presented, keeping into consideration a bias towards a particular action, which, to some extents, make it preferable to another one that would analytically prove. EBL allows to keep this personal factor into consideration, when applying a RL paradigm to agents.

In the first formulation, a dualistic action selection is considered, i.e.: $A(a_1, a_2)$. By applying the formal reinforcement learning technique described in equation (6) an agent is able to have the expected value for the action it performed. Each agent is endowed with the RL technique. At this point, we can imagine two different categories of agents (α_1, α_2): one biased towards action a_1 and the other one biased towards action a_2. For each category, a constant is introduced $(0 < K_1, K_2 < 1)$, defining the propensity for the given action, used to evaluate $\overline{EV(a_1)}$ and $\overline{EV(a_2)}$ which is the expected value of the action, corrected by the bias. For the category of agents biased towards action a_1 we have that:

$$\alpha_1: \begin{cases} \overline{EV(a_1)} = EV(a_1) + (|EV(a_1)| * K_1) \\ \overline{EV(a_2)} = EV(a_2) - (|EV(a_2)| * K_1) \end{cases} \quad (7)$$

In this way, K_1 represents the propensity for the first category of agents towards action a_1 and acts as a percentage increasing the analytically computed $EV(a_1)$ and decreasing $EV(a_2)$. At the same way, K_2 would represent the propensity for the second category of agents towards action a_2 and acts on the expected value of the two possible actions as before:

$$\alpha_2: \begin{cases} \overline{EV(a_1)} = EV(a_1) - (|EV(a_1)| * K_2) \\ \overline{EV(a_2)} = EV(a_2) + (|EV(a_2)| * K_2) \end{cases} \quad (8)$$

The constant K acts like a "friction" for the EV function; after calculating the objective $EV(a_i)$ it increments it of a percentage, if a_i is the action for which the agent has a positive bias, or decrements it, if a_i is the action for which the agent has a negative bias. In this way, the agent α_1 will perform action a_1 (instead of a_2) even if $EV(a_1) < EV(a_2)$, as long as $\overline{EV(a_1)}$ is not less than $\overline{EV(a_2)}$. In particular, by analytically solving the following:

$$EV(a_1) + (|EV(a_1)| * K_1) \geq EV(a_2) - (|EV(a_2)| * K_1) \quad (9)$$

We have that agent α_1(biased towards action a_1) will perform $a_1$as long as:

$$EV(a_1) \geq EV(a_2) * \frac{1 - K_1}{1 + K_1} \quad (10)$$

Equation number 10 applies when both $EV(a_1)$ and $EV(a_2)$ are positive values. If $EV(a_1)$ is positive and $EV(a_2)$ is negative, then a_1 will obviously be performed (being this a sub-case of equation 10), while if $EV(a_2)$ is positive and $EV(a_1)$ is negative, then a_2 will be performed, since even if biased, it wouldn't make any sense for an agent to perform an action that proved even harmful (that's why it went down to a negative value). If $\overline{EV(a_1)} = \overline{EV(a_2)}$, by definition, the performed action will be the favorite one, i.e.: the one towards which the agent has a positive bias.

In order to give a numeric example, if $EV(a_1) = 50$ and $K_1 = 0.2$ then a_1 will be performed by agent α_1 till $EV(a_2) > 75$. This friction gets even stronger for higher K values; for example, with a $K_1 = 0.5$, a_1 will be performed till $EV(a_2) > 150$ and so on.

In figure 1, a chart is shown with the various resulting $\overline{EV(a_1)}$, calculated according to equation 8, for agent α_1, given K_1. When compared to the baseline results ($K_1 = 0$) it's evident that by increasing the value of K_1, the positive values of $EV(a_1)$ turns into higher and higher values of $\overline{EV(a_1)}$. At the same time, a negative value of $EV(a_1)$ gets less

and less negative by increasing K_1, while never turning into a positive value (at most, when K_1, $\overline{EV(a_1)}$ gets equal to 0 for every $EV(a_1) < 0$). For example, with $K_1 = 0.1$, $\overline{EV(a_1)}$ is 10% higher than $EV(a_1)$.

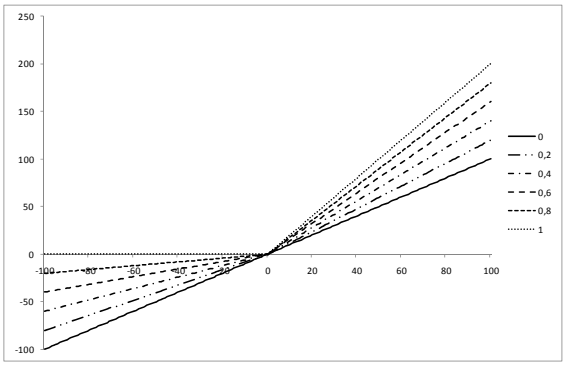

Figure 1: $\overline{EV(a_1)}$ for agent α_1 given $EV(a_1)$, for various K_1.

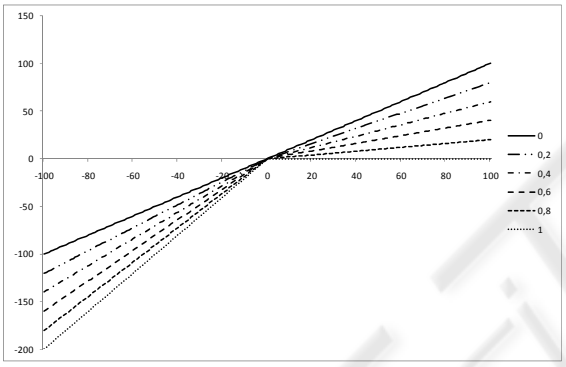

Figure 2: $\overline{EV(a_2)}$ for agent α_1 given $EV(a_2)$, for various K_1

In figure 2, a chart is shown with the various resulting $\overline{EV(a_2)}$, calculated according to equation 8, for agent α_1, given K_1. This time, when compared to the baseline result ($K_1 = 0$), since a_2 is the action towards which the agent α_1 has a negative bias, it's possible to notice that the resulting $\overline{EV(a_2)}$ is always lower (or equal, in case they are both 0) than the original $EV(a_2)$ calculated according to equation 6. In particular, higher K_1 corresponds to more bias (larger distance among the objective expected value), exactly opposite as it was before for action a_2. Note that for a $K_1 = 1$ (i.e.: maximum bias) $\overline{EV(a_2)}$ never gets past zero, so that a_2 is performed if and only if $EV(a_1)$ - and hence $\overline{EV(a_1)}$ - is less than zero.

3.1 General Cases

The first general case (more than two possible actions and more than two categories of agents) is actually a strict super-case of the one formalized in 3.1. Each agent is endowed with an evaluation biased function derived from equations (7). Be $\alpha(\alpha_1, \alpha_2, \dots, \alpha_n)$ the set of agents, and $A(a_1, a_2, \dots, a_m)$ the set of possible actions to be performed, then the specific agent α_k, with a positive bias for action a_h will feature such a biased evaluation function:

$$\alpha_k: \begin{cases} \overline{EV(a_1)} = EV(a_1) - (|EV(a_1)| * K_1) \\ \dots \\ \overline{EV(a_{h-1})} = EV(a_{h-1}) - (|EV(a_{h-1})| * K_1) \\ \overline{EV(a_h)} = EV(a_h) + (|EV(a_h)| * K_1) \\ \overline{EV(a_{h+1})} = EV(a_{h+1}) - (|EV(a_{h+1})| * K_1) \\ \dots \\ \overline{EV(a_m)} = EV(a_m) - (|EV(a_m)| * K_1) \end{cases} \quad (11)$$

This applies to each agent, of course by changing the specific equation corresponding to her specific positive bias. Even more general, an agent could have a positive bias towards more than one action; for example, if agent α_5 has a positive bias for actions a_1 and a_2 and a negative bias for all the others, the resulting equations will be:

$$\alpha_5: \begin{cases} \overline{EV(a_1)} = EV(a_1) + (|EV(a_1)| * K_1) \\ \overline{EV(a_2)} = EV(a_2) + (|EV(a_2)| * K_1) \\ \overline{EV(a_3)} = EV(a_3) - (|EV(a_3)| * K_1) \\ \dots \\ \overline{EV(a_m)} = EV(a_m) - (|EV(a_m)| * K_1) \end{cases} \quad (12)$$

In the most general case, for each $\overline{\mathrm{EV}(a_i)}$:

$$\overline{EV(a_i)} = EV(a_i) \mp (|EV(a_i)| * K_j) \quad (13)$$

In case that two or more $\overline{\mathrm{EV(a)}}$ have the same value, the agent will perform the action towards which it has a positive bias; in the case explored by equation (12), in which the agent has the same positive bias towards more than one action, then the choice among which action to perform, under the same $\overline{\mathrm{EV(a)}}$, could be managed in various ways (e.g.: randomly, stochastically and so on).

As a last general case, the agents could be a different positive/negative propensity towards different actions. In this case, the K variable to be used won't be the same for all the equations regarding an individual agent. For example, given a set of $K(K_1, K_2, \dots, K_n)$ and a set of actions $A(a_1, a_2, \dots, a_m)$, for each agent (α_k) we have:

$$\alpha_k: \begin{cases} \overline{EV(a_1)} = EV(a_1) \mp (|EV(a_1)| * K_1) \\ \dots \\ \overline{EV(a_m)} = EV(a_m) \mp (|EV(a_m)| * K_n) \end{cases} \quad (14)$$

Besides being a fixed parameter, K could be a stochastic value, e.g.: given a mean and a variance.

4 SIMULATED EXPERIMENTS

Some experiments were done in order to text the basic EBL equations introduced in paragraph 3.1. The agents involved in the simulation can perform two possible actions, a_1 and a_2. The agents in the simulation randomly meet at each turn (one to one) and perform an action according to their EV. A payoff matrix is used, in the form of:

Table 1: Example of payoff matrix.

	a_1	a_2
a_1	p_1	p_2
a_2	p_3	p_4

Where p_1 is the payoff originated when both agents perform a_1, p_2 is the payoff given to the agents when one of them performs a_1 and the other one performs a_2 and so on. Usually p_2 and p_3 are set at the same value, for coherency. For each time-step in the simulation, the number of agents performing a_1 and a_2 are sampled and represented on a graph.

Table 2: Payoff matrix for experiments 1 and 2.

	a_1	a_2
a_1	1	-1
a_2	-1	2

The first baseline experiment reproduces the classical RL equation (6), i.e.: with both K_1 and K_2 equals to zero. A total of 100 agents are used, with a learning rate (λ) set to 0.2. The payoff matrix used for the experiment is shown in Table 2. Action a_2 is clearly favored by the matrix, and coordination, in the form of performing the same action, is rewarded, while miscoordination punished.

In the first experiment, 50 agents start performing action a_1 and 50 agents start performing action a_2. The results are depicted in figure 3; convergence is subtle and stable, once the equilibrium is reached.

In the second experiment, a small bias towards action a_1 is introduced for fifty α_1 agents (K_1 = 0.1), while the payoff matrix remains the same as in previous experiment. Agents α_2 do not have a bias, but all start playing action a_2; this will be different in the following experiments, where unbiased agents will start performing a random action. The results are quite interesting, and depicted in figure 4.

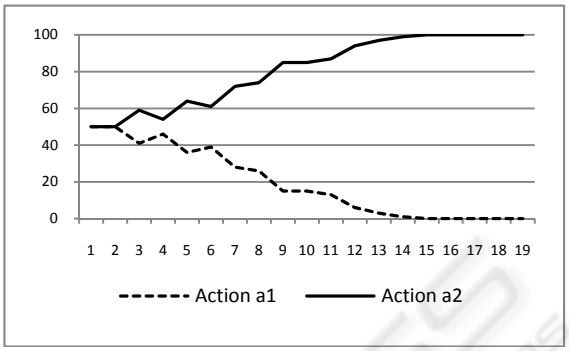

Figure 3: Baseline experiment: no biased agents.

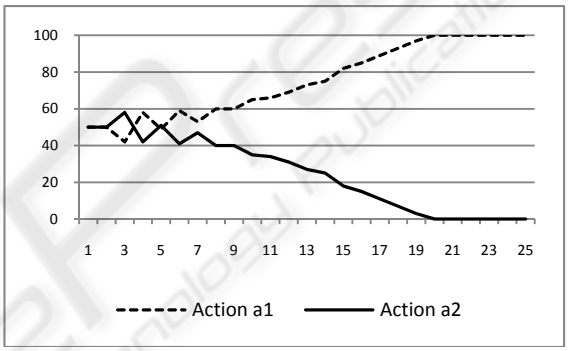

Figure 4: Experiment 1: biased Vs unbiased agents.

Even if action a_2 is clearly favored by the payoff matrix, after taking an initial lead in agents' preferences, all the population moves towards action a_1. This is due to the resilience of biased agents in changing their mind; doing this way, the other 50 non-biased agents find more and more partners performing action a_1, and thus, if they perform a_2 they get a negative payoff. In this way, in order to gain something, since they are not biased, they are forced to move towards the sub-optimal action a_1, preferred by the biased agents. In order to give a social explanation of this, we can think to the fact that often the wiser persons adapt themselves to the more obstinate ones, when they necessarily have to deal with them, even if the outcome is not the optimal one, just not to lose more. This is particular evident when the wiser persons are the minority, or, as in our case, in an equal number. At this point we wonder how many "rotten apples" (i.e.: biased agents) are needed to ruin the entire barrel (i.e.: turn away all the agents from the optimal convergence), given the same payoff matrix. With a series of *ceteris paribus* experiments, we found the critical division to be 20/80; the results are shown in figure 5. The unbiased agents now start by performing a random action, in

order to probe for the best move, and then adapt themselves on the basis of their perceptions. On the other hand, the biased agents start by performing action a_1. In this way the agents performing a_2, both biased and un-biased ones, when they meet an agent performing a_1 get a negative bias. Even if the optimal combination would be $a_2 + a_2$, once again the equilibrium is found on the suboptimal joint action, which is $a_1 + a_1$.

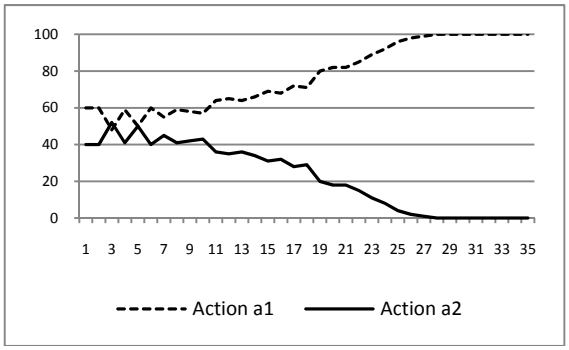

Figure 5: Experiment 2: Critical threshold.

Till now the advantage of performing join action $a_2 + a_2$ over $a_1 + a_1$ was evident (payoff 2 vs 1) but not huge; in the next experiment, a new payoff matrix is used, in the joint action $a_2 + a_2$ is rewarded 3, instead of 2. The purpose is investigating how much the previous threshold would increase under these hypotheses. The empirical finding is 25/75, and the convergence is again extremely fast, and much similar to the previous experiment. Even a bigger advantage for the optimal action is soon nullified by the presence of just 25% biased agents, when penalty for miscoordination exists. This explains why sometimes suboptimal actions (or non-best products) become the most spread and common. In the real world, marketing could be able to bias a part of the population, and a good distribution or other politics for the suboptimal product/service could act as a penalty for unbiased players when interacting with biased ones. The following experiment investigates the case with no penalty for miscoordination.

Table 3: Payoff matrix for experiments 3.

	a_1	a_2
a_1	1	0
a_2	0	2

To explore the differences with experiment 1, twenty biased agents were employed, out of 100. The results are shown in figure 6.

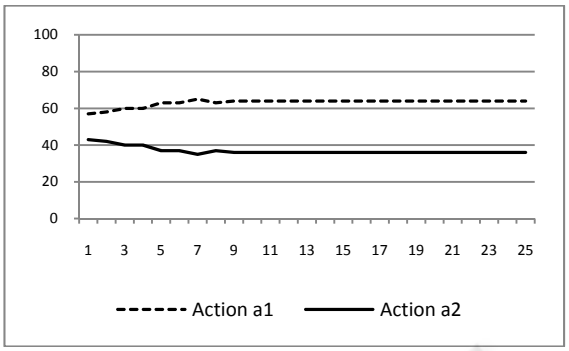

Figure 6: Experiment 3: No penalty for miscoordination.

As it's evident, now the results are less extreme, in the sense that a part of the agents succeed in performing the optimal action; though, many unbiased agents are dragged along by the biased ones, to the suboptimal action. Numerically speaking, about 50% of unbiased agents become supporter of the suboptimal action, even if the biased agents are a small part of the population (20%).

This shows that penalty for miscoordination is important, but not crucial, for averting the majority of the population from the best possible choice.

5 FUTURE DEVELOPMENTS

While individual preferences are very important as a bias factor for learning and action selection, when dealing with social systems, in which many entities operate at the same time and are usually connected over a network, other factors should be kept into consideration. In particular, the preferences of other individuals with which a specific agent is in touch can affect her choices, modifying the perception mechanism described in equation 6. Once again, if the goal is that of representing agents mimicking human behaviour, then it's not realistic to consider perfect perception of the payoffs deriving from past actions. Besides, Fragaszy and Visalberghi (2001) agree that socially-biased learning is widespread in the animal kingdom and important in behavioural biology and evolution. It's important to distinguish between imitation and socially biased learning; while the former is limited to *de facto* imitating the behaviour of another individual (possible with some minor changes), the second is referred to modifying the possessed behaviour after the observation of others' behaviours. While imitation is most passive and mechanical, social learning supposes a form of intelligence in selecting how to modify the past behaviour, taking into account others' experience.

Box (1984) defines socially biased learning as: *a change in behaviour contingent upon a change in cognitive state associated with experience that is aided by exposure to the activities of social companions*. The first part of this is already taken care by RL methods (equation 6) and by the EBL proposed in the previous sections. What is still lacking is the bias coming from social companions, i.e.: other agents coexisting in the same environment. In future works a reputational based approach will be used, to embed a form of social bias into the agents. This will keep into consideration the payoffs deriving from other agents' actions, weighted by agents' individual reputations, acting as a bias for the equation defining the RL strategy, along with EBL.

6 CONCLUSIONS

In this work a formal method for action selection is introduced: it's based on one step QL algorithm (equation 6), but it takes into account individual preference for one or more actions. This method is designed to be used in simulation of social systems employing MAS, where many entities interact in the same environment and must take some actions at each time-step. In particular, traditional methods do not take into account human factor, in the form of personal inclination towards different strategies, and consider the agents as totally rational and able to modify their behaviour based on an analytical payoff function derived from the performed actions.

Ego biased learning is formally presented in the most simple case, in which only two categories of agents are involved, and only two actions are possible. That's to show the basic equations and explore the results, when varying the parameters.

After that, some general cases are faced and equations are supplied, where an arbitrary number of agents' categories is taken into account, along with an equally discretionary number of actions. There can be many sub-cases for this situations, e.g.: one action is preferred, and the others are disadvantaged, or an agent has the same bias towards more actions, or in the most general situation, each action can have a positive or negative bias, for an agent.

Some simulations are run, and the results are studied, showing how, even a small part of the population, with a negligible bias towards a particular action, can affect the convergence of a RL algorithm. In particular, if miscoordination is punished, after few steps all the agents converge on the suboptimal action, which is the one preferred by the biased agents. With no penalty for miscoordination, things are less radical, but once again many non-biased agents (even if not all of them) converge to the suboptimal action. This shows how personal biases are important in social systems, where agents must coordinate or interact.

ACKNOWLEDGEMENTS

The authors wish to gratefully acknowledge as their mentor prof. Gianpiero Bussolin, who applied new technologies and simulation to Management and Economics since 1964. This work is the ideal continuation of his theories.

REFERENCES

Box, H. O., 1984. *Primate Behaviour and Social Ecology*. London: Chapman and Hall.

Chen M. K., 2008. Rationalization and Cognitive Dissonance: do Choices Affect or Reflect Preferences? *Cowles Foundation Discussion Paper No. 1669*

Mataric M. J., 2004. Reward Functions for Accelerated Learning. In *Proceedings of the Eleventh International Conference on Machine Learning.*

Mataric, M. J., 1997. Reinforcement Learning in the Multi-Robot domain. *Autonomous Robots*, 4(1)

Fudenberg, D., and Levine, D. K. 1998. The Theory of Learning in Games. Cambridge, MA: MIT Press

Fragaszy, D. and Visalberghi, E., 2001. Recognizing a swan: Socially-biased learning. *Psychologia, 44*, 82-98.

Powers R. and Shoham Y., 2005. New criteria and a new algorithm for learning in multi-agent systems. *In Proceedings of NIPS.*

Sharot T., De Martino B., Dolan R.J., 2009. How Choice Reveals and Shapes Expected Hedonic Outcome. *The Journal of Neuroscience, 29(12):3760-3765*

Sutton, R. S. and Barto A. G., 1998. Reinforcement Learning: An Introduction. *MIT Press, Cambridge, MA. A Bradford Book*

Watkins, C. J. C. H. 1989. Learning from delayed rewards. *PhD thesis*, Psychology Department, Univ. of Cambridge.