department of economics
"S. Cognetti de Martiis"

Via Po, 53 – 10124 Torino (Italy)
Tel. (+39) 011 6702704 - Fax (+39) 011 6702762
URL: http//www.de.unito.it

WORKING PAPER SERIES

MANAGING KNOWLEDGE IN AGENT-BASED MODELS: THEORETICAL AND METHODOLOGICAL ISSUES

Gianluigi Ferraris e Magda Fontana

Dipartimento di Economia "S. Cognetti de Martiis"

Working paper No. 03/2006

Università di Torino

Managing Knowledge in Agent-based Models: Theoretical and Methodological Issues

G. Ferraris – M. Fontana

*<< A wide range of computer-based adaptive algorithms exits for exploring artificial adaptive systems including **classifier systems**, **genetic algorithms**, **neural networks**, and **reinforcement learning mechanisms**. The multiplicity of techniques presents a problem for analysis. How sensitive are the results to a particular incarnation of the adaptive agent? This problem, of course, confronts any attempt to lessen the rationality postulates traditionally used in economic theory. Usually, there is only one way to be fully rational, but there are many ways to be less rational. It is important in building a theory of artificial adaptive agent to construct agents that exhibit robust behaviour across algorithmic choices>>*

Holland and Miller (1991), "Artificial Adaptive Agents in Economic Theory", *AER,* p. 365.

ABSTRACT

The paper proposes an experimental setup to compare different representations of artificial adaptive agents (genetic algorithms, artificial neural networks, and classifier systems) and suggests some criteria to assess equivalence and robustness of performance. In economic theory, the use of artificial adaptive agents as substitutes for the homo oeconomicus raises important methodological issues. While the reductionist approach grounded on Olympic rationality offers full rationality as the unique reference point for problem solving, weaker notions of rationality generate a variety of processes and outcomes of decision-making. The paper gives some suggestions on sensitivity of the behaviour of agents to the algorithmic choice and to the codification of knowledge. Preliminary results show that in an iterated prisoner's dilemma interesting patterns of behaviour (such as strategies that perform better than the tit-for-tat) emerge.

Keywords: rationality postulate, artificial adaptive agents, agent-based modelling.

0. Introduction

The diffusion of the theory of complexity in economics has lead to a reassessment of the way in which agents are conceived and modelled. The underpinnings of complexity are quite irreconcilable with the typical reductionism of mathematical tools (Dalmazzone – Fontana 2006). Leaving aside the rationality postulate (read abandoning the maximisation techniques) opens up a crucial issue: while there is only one way of being fully rational, there are many ways of being less rational (Holland –Miller 1991), so how to model agents' behaviour? The question is relevant both in the theoretical - that is to say how to choose the degree of rationality[1] - and in methodological sense -how to describe the decision making process without equations- .

In order to tackle complexity, economists have turned to the concept of adaptive agent and to the armoury of artificial intelligence. The hallmark of an adaptive agent is her ability to modify the patterns of action as to improve her probability of persisting in highly changeable environments; while artificial intelligence tools are roughly computer programs which search the problem space in order to find suitable (possibly optimal) solutions by using limited informational sets and bounded 'cognitive' skills.

In the last years, economics has witnessed an increasing diffusion of learning algorithms[2]. They are used mainly to replicate some observed regularity to conduct investigations of the "what if" kind and to find plausible individual motivations to sustain a given macro phenomena. However, we feel that such spread is taking place without a parallel development of the methodological issues related to the algorithmic choice: economic literature misses a frame to systematise the use of such instruments.

As it appears from the opening citation, a crucial point is to fully understand the functioning of the algorithms and then to compare their behaviours to find out whether their outputs correspond or diverge. However, trying to test such robustness is not an easy task because those algorithms are different under very many respects. First, they rely on different, and somehow conflicting, theories of learning. Artificial Neural networks (hereafter NN) are based upon connectionism and the Artificial Intelligence Paradigm, while genetic algorithms (hereafter GA) and classifier systems (hereafter CS) are born within the Complex Adaptive System Paradigm and lean on an evolutionary idea of learning. Those differences result in algorithmic forms which are utterly dissimilar. A preliminary issue to deal with in order to shed some light on this topic is therefore that of the (technical) possibility and the (theoretical) opportunity of such a comparison.

---

[1] Here rationality is defined as the appropriateness of means to pursue given ends.

[2] For an analysis of data concerning the diffusion of learning algorithms across the sub-disciplines of economics, see Fontana (2006).

Second, having answered affirmatively to the previous questions, it comes to the matter of how to judge robustness. We are in need of a definition of equivalence of results that is able to allow comparison among the models. In addition to the notions of numerical, relational and distributional equivalence that have already been developed in literature (Axtell 1996), we suggest a two-stage concept of equivalence. The first stage, *numerical identity*, implies convergence on the (pretty) same solution. The second stage, *procedural equivalence*, requires similarity in the process of convergence, that is to say that the way in which the solution is found and the required time (in terms of run or steps) must be reasonably alike. We strongly believe that equivalence must be judged according to both outcome and process likeness since in the interpretation of economic phenomena from which learning algorithms stemmed puts a great emphasis on the paths that lead to given states of the world. To put it bluntly, who cares to know that an agent will make the best decision of all if it will take a billion of years to be computed? Or else, as long as we can think of the end of the learning process as an equilibrium (in the sense of a state of rest), we feel that the out of equilibrium behaviour is indeed an important part of the story.

Third, we also appreciate algorithms according to the goodness of the result of their learning. In addition to the already developed criteria of effectiveness and efficiency, we propose to evaluate learning by taking into account graceful degradation, stability, and rapidity in order to increase knowledge of the functioning of the algorithms and improve the assessments of the results of their application.

The present work also constitutes a foray in one the most debated issue concerning agent-based simulation, that of the difficulty in comparing models and replicating results. It is felt that "without such a process of close comparison, computational modelling will never provide the clear sense of "domain validity" that typically can be obtained for mathematised theories […] alignment is essential to support two hallmarks of cumulative disciplinary research: critical experiment and subsumption. If we cannot determine whether or not two models produce equivalent results in equivalent conditions, we cannot reject one model in favour of another that fits data better; nor we are able to say that one model is a special case of another more general one" (Axtell et al. 1996, p. 124).

In what follows we describe an experimental setup in silico which allows for meaningful comparison of artificial agents and discusses some issues concerning the relevance of the codification of information in such algorithms. As we will show, the setup regards a simple problem with a known and certain result. The problem must be simple in order to facilitate the reading of results and the comprehension of the path to problem solving. The solution must be known in order to judge the goodness of the learning process without discussing the goodness of its outcome. In addition, we want the entire framework to be as simple as possible in obeisance to the principle that complexity arises from simple behaviour when interaction among agents is autonomous.

In the following paragraphs the framework for simulations and the model are introduced, we then discuss the issues of comparability and propose a methodological frame to test for robustness of behaviour. Finally, some illustrative results are presented.

1. The framework

The simulation implements the traditional iterated Prisoner's dilemma (hereafter IPD) by means of the agent-based modelling paradigm. It is, in spirit, very similar to the one proposed by Robert Axelrod in his 1997's book *The Complexity of Cooperation.* As it is well known, the main result obtained by Axelrod in 1984 was the emergence of cooperation in two-person iterated prisoner's dilemma under the condition of a sufficiently long horizon for repetition. He reached such result in an experimental setup in which he asked scientists and amateurs to submit strategies for a tournament whose rules where those of the PD. In 1997, Axelrod decided to extend his previous work by using artificial adaptive agents (genetic algorithm) instead of human agents. He turned to simulation in order to explore the nonlinear effects deriving from interaction and to test robustness of his 1984's results[3]. Our simulations take Axelrod experimental setting as a reference point and amends it order to observe and compare the performances of the algorithms.

[3] In his words: *"[...] having done two rounds of the tournament, I wondered whether the amount of cooperation I observed was due to prior expectations of the people who submitted the rules*" (Axelrod 1997, p. 6). This statement is particularly interesting on the methodological stance. In fact, it represents an interesting route to bridge the gap between experiments conducted with human and artificial agents.

In our opinion, the IPD meets the requirements of simplicity and certainty of outcomes set in the introduction. In fact, the only constraints on agents' problem solving are the payoffs matrix and the rules of interaction. As for results, we expect the algorithms to discover patterns of mutual cooperation that perform at least as well as tit-for-tat to escape the sub-optimality of defection typical of one-shot PD. However, we also feel that the framework retains elements of richness that can sustain a wide span for the analysis. While the set of plausible outcomes is closed, (agents can only choose between cooperation or defection, the scope for composite strategies is extremely wide). For instance, at the macro level, the basic strategies such as tit-for-tat, tit-for-2tat, cooperate or defect can emerge and endure or appear as temporary T-equilibria or follow each other in waves or cycles. Or else, at the micro level, in the history of a single agent they can show up in various combinations that change as adaptation take place. In addition, the game also present local optima in which the learning process can be locked in.

Finally, the tournament design encompasses another argument stressed by complexity theorists: the profitability of a strategy cannot be computed in isolation. As pointed out by Arthur (1994) in the El Farol Bar problem, there is not an a priori best strategy (a true model of the world) but the fitness of strategy depends on what the others are doing. The agent (and her problem-solving device) is asked to elaborate a strategy that can do well in the environment provided all the other strategies. For example, 'defect' can be the better response in world of pure cooperators while can lead to a poor performance in a world of tit-for-tat players.

1.1 The model

The payoffs bi-matrix, which shows the same values as in Axelrod's study, is the following:

Player Row / ***Player Column***	***Defect***	***Cooperate***
Defect	1,1	5,0
Cooperate	0,5	3,3

The interaction is organised by means of a double round robin tournament in which each player is paired with all the other members of the population. The latter is composed of agents with predetermined rules and of agents exploiting evolving rules. This allows for two kinds of tests. Firstly, to explore their properties learning algorithms are run in an environment in which variability is 'controlled' by having only one kind of learning process going on while all the other agents stick to a given rule. Secondly, when the learning process has been validated, evolutionary agents can play the one against the other.

Agents with predetermined rules have an inborn strategy that is applied deterministically, irrespective of the opponent they are facing. In more details, they are labelled as:

i) Perpetual cooperator: which always cooperates;

ii) Perpetual defector: which always defects;

iii) Noise player: which randomly picks a strategy;

iv) Tit-for-tat (hereafter TfT): which starts with a cooperation and than always plays by reproducing the strategy encountered in the last interaction.

v) Tit-for-2tat (hereafter TF2T): is a more tolerant player than TfT. It forgives the first defection before turning itself into a defector.

Before describing the used kinds of adaptive agents few worlds must be spent in order to delimit our analysis. Adaptive agents are able to devise strategies that change in response to environmental variability. An important feature of artificial agents is that they have no a priori knowledge of the system, nor the researcher has to pre-constitute rules for them to work with. They develop behavioural patterns by learning out of their experience often starting from a set of random rules. The strength of such agents is not in their perfect knowledge or complete information rather it resides in that they can make meaningful decision independently of the degree of knowledge (even on the researcher side) of the problem to be solved. Moreover, they are able to generate novel behaviour. In practice, such agents are nothing but computational devices that perform in ways that we would be inclined to call intelligent[4] and whose way of exploring the

[4] These definitions rely on Berkeley (1997).

solution space recalls what, when referring to living beings, we call learning. They are the object of a field of research known as Artificial Intelligence (hereafter AI). Roughly, AI research has generated two streams: i) the strong AI paradigm which refers to the attempt of reproducing the actual functioning of the mind and the brain; ii) the weak AI paradigm that focuses on obtaining results that may - on an intuitive basis – be similar to those produced by human problem solving. Our work, as most of applications of AI to economics, is grounded in the weak paradigm: the emphasis is on modelling agents that are able to make decisions autonomously and creatively, leaving aside the actual resemblance between the algorithms and the mechanisms, that are largely unknown, ruling the formation and evolution of ideas in biological brains and in human minds. Rather, we share the fascination of the colleagues economists with artificial agents[5] that has to do with the spread of the interpretation of the economic process in terms of complex adaptive systems. As sketched above, among the other implications, adhering to the complexity perspective amounts to renounce not only to the rationality postulate but, most importantly, to the mathematical tools of (dynamic) optimisation (Foster 2005). In fact, such a technique only works when there is a known set of possible outcomes and known probabilities associated with each event. In economic systems such condition is not easily fulfilled: the future outcomes of action are partially foreseeable due to their inherent path-dependent character but, nevertheless they retain a high degree of uncertainty due to continuous adjustments of external (structure, connections, innovations in other agent's behaviour) and internal (experience, imitation, creativity) environments. Artificial agents tend to substitute constrained optimisation techniques in complex environments and therefore they can be seen as general decision-making devices (Markose 2005). In what follows, terms as 'learning' and 'knowledge' are used metaphorically. However, as it will be shown in the following discussion there is an important link between the functioning of the algorithm and the mind of the researcher. In fact, even the simplest situation can be framed and codified in a variety of ways that are likely to affect the algorithm performance.

The analysis will concern the following artificial adaptive agents:

---

5 See for instance Axelrod (1997).

i) GAs (Holland 1975) are modelled on the processes of biological evolution. A genetic algorithm manipulates a set of structures called population. Structures are coded as strings of symbols drawn from some finite alphabet (often binary). For example, in our context a string is the list of strategies that the agent has played with each of the agents' type. Each string his assigned value (fitness) on the basis of the result of its mediated or direct[6] interaction with the environment. Genetic algorithm operates on the population by making copies of strings in proportion to the observed fitness. That is the fittest strings have a higher chance to be reproduced. While reproducing the strings the genetic algorithms trigger some genetic operator such as crossover that switches some of the characters of the strings and mutation that randomly varies some symbols of the string. It appears the artificial mind reproduced by a GA works with a selection mechanism that tends to adopt the action that has proven well in the past, however has a drift of creativity (introduced by the genetic operators) that generates new rules[7]. The result is a population of rules that adapt to the environment in an increasingly fit way. Adaptation operates at the group level, each evaluation regards the entire set of strings and therefore it permits to exploit the mutual information inherent in the population rather than simply exploit the best individual in it.

A CS is an adaptive, rule-based system that interacts with the environment by activating the appropriate set of rules. Each rule is in condition/action form and many rules can be active simultaneously. The condition part must be met by the environment for the action to be executed. However, the rule is not automatically acted. Rather, it enters a competition with other rules and is chosen according to its strength. A rule's strength measures its past usefulness and is modified according to the system learning mechanism. A rule's strength is determined by means of a procedure of reward sharing known as *bucket-brigade algorithm*. The creation of new rules is done by discovery

[6] This distinction refers to the specific use of the genetic algorithm. Interaction with the environment is direct when the GA embodies the agent, whereas is mediated when, as in our case, it represents the mind (read learning routine) of the agent. If this is the case, the GA proposes an action and the agent executes it in the environment.

[7] This often has been seen as metaphor for the innovation activity. While to some extent it may be so, the genetic operators have a precise technical role that of preventing the GA to be locked in sub optimal solutions.

algorithm (often a genetic algorithm[8]) that tests new approaches in the environment. These two elements allow the CS to generate categories to classify the environment and progressively refine them according to experience.

NNs aim at simulating on a very simplified scale the functioning of the brain: the elaboration units mimic the behaviour of neurons, while the connections among neuron symbolise dendrites, the weight of each connection determines the synapses behaviour. Technically a NN is composed by:

1. units which are in communication with the external environment that are devoted to receive information from the outside (input nodes);
2. units devoted to provide the behaviour of the agent (output nodes);
3. units located in the so called hidden layers: their task is to mediate the transformation of information through non-linear elaboration.

The reaction of a NN to the information received through input nodes depends on the weight of each of the connections: in modifying their weight, the network adapts its response and tunes the agent’s behaviour. Usually, in order to correctly calibrate such behaviour network must be trained: the weighting of connection takes place prior to the proposition of a given conduct. The modification of weight is made through a mechanism that aims at reducing the error: the so called *back propagation technique.*

In the model, agents with artificial minds are paired with agent endowed with fixed rules whose features recall those emerged from Axelrod experiments. The objective of the agent, pursued through the various types of artificial minds, is to extract the maximum payoff from playing the IPD. Agents have limited information about the environment (they do not know the matrix of payoffs) and about the other members the population (they recognise the type of agent but cannot remember past interactions)[9] and the length of the game. Moreover they exhibit a sort of procedural rationality à la Simon in that their decision making process develops through rules of thumb that are tested against the environment.

---

[8] The GA, in this case does not work as to converge on a optimal solution, it simply manipulates the population of rules.

[9] When there is only one instance of each kind of agent the information is more accurate.

As for the observed variables, our focus is on cumulative payoff (computed after having played with all the agents in the tournament), path to the outcome (which is observed both graphically and numerically as number of required step to reach the outcome), and characteristics of the rules which are stored in the agent's mind.

2. Making the comparison

As stressed in the introduction a preliminary issue to deal with is the theoretical opportunity and the technical possibility of such a comparison. On the theoretical ground, literature (Norvig etc) emphasises a number of important the differences. On the one side, the learning operated by GA and CC is known as reinforced learning as to stress the procedure of linking the probability of executing an action to its strength which in turn depends on past performances. While NN is said to use a form of supervised learning that is to say that it preliminarily needs some instances of phenomena to learn on without a proper return of rules in terms of fitness. However, as some authors report it is still possible to think of the process of adjusting weights as a reinforcement procedure. Moreover, as long as we do not delve into the discussion regarding the extent to which artificial minds resemble the actual one we can neglect the difference in the models of learning underlying each adaptive agent.

The technical opportunity of such a comparison depends on the possibility of changing the algorithms in order to smooth differences while preserving their original nature. The main structural differences are: the need of preliminary training for NNs, dependence on different inputs, sensitivity to the choice of a given fitness function.

The above-cited distinctions between reinforcement and supervised learning, technically results in the fact that the GA and CS do not need a preliminary training: the selection of the structures based on their assessment is sufficient to promote the spread of the better rules. NNs instead require a training activity to tune their performance on the problem. This difficulty can be overcome by adopting the cross target method (Terna) that allows the network to assign a value to its single performance. This name comes from "the technique used to figure out targets in a class of models founded upon artificial adaptive agents whose main characteristic is developing some kind of internal consistency"[10]. It is a learning mechanism that produces guesses about both actions and their effects, on the basis of an information set. Actual effects are estimated through the guessed action and results are used to train the mechanism that guesses the effects. Actions that are necessary to match the guessed effects are, on the contrary, employed to train decision mechanism about actions.

As for input data, NN and CS require an informational flow to describe the current condition under which they are asked to generate an action while GA does not need this kind of input: it generates rules internally and proposes them independently of any conditions. To nuance this difference, the evaluation of a single rule takes place at the end of each cycle of simulations, while in the other methods happens after each interaction. In GA the input condition does not trigger an action but simply determines the part of the rule (namely, which part of the string) that will be activated to induce action.

For what concerns sensitivity, GAs and NNs are less affected by the way in which performance is measured. In fact, a GA evaluates all its structures in a non-cumulative way and prior to the learning and therefore changes in performance are likely to be smooth; NN operates as to reduce the distance between the target and the actual performance and therefore tends to restrict the span of the dominium of the objective function. On the other hand, CS evolves rules on the basis of the accumulation of rewards obtained by each rule and thus rules that are less fit but have already a relatively high strength can persist in spite of the presence of more suitable rules that have a comparatively lower fitness value. For instance, this can happen for newborn rules of for rules they are generated from 'poor' parents and have inherited a low index of performance regardless of their effectiveness. This characteristic of CS is critical because may prevent it from fully exploiting the potential of the set rules and could lock it onto a local optimum.

In its implementation, our framework is inspired to the strict separation between agents, cognitive endowment and rules to exploit it. This way of modelling allow for rigorous

[10] Beltratti et al. (1996), p. 110.

and controlled comparison between artificial minds since, it easy to 'plug' the agent with different learning algorithms (made of cognitive endowment and rules for its manipulation) leaving the rest of the simulation unchanged.

In this first experiment, evaluation of goodness revolves on the numerical identity only. The performance is expressed as a numerical values obtained as the accumulation of the payoff obtained in the tournament by the rule suggested by the learning device. As we will discuss below, this kind of criterion can be used in very simple situation only. In future setting as the degree of complexity of the simulation increases, we will adopt different criteria.

The agenda for simulation is the following.

- Individual IAAs that play against opponents with fixed rules;
- Competition among learning algorithms;
- Mixed scenario.

The agenda aims at observing the behaviour and the properties of the learning device under a closed and very easily inspecting result as to stabilise and improve knowledge of their functioning and then apply then to more complex open-ended situations.
This is an important part of the story, in fact in complex situations we cannot say whether the configuration that system has assumed is 'correct' in the same way we have computed that maximum payoff that an agent can extract from its fellow players since inherent unpredictability of non-linear interaction can produce configuration that we cannot compute or expect in advance. Our judgement of a system's state and its being a plausible outcome of theoretical statements embedded into the simulation (and not a drift generated by a mistake in the program code or not an effect resulting from the application of a given learning algorithm to a problem that it is not very well equipped to solve) depend mainly of the degree of knowledge that we can master about the behaviour of the system units: the agent and its rules of adaptation.

Criteria of Comparison

In the introduction, we have pointed out that economics has not yet developed a literature on comparing and testing the behaviour of artificial minds. It therefore misses an established and shared set of criteria to draw on. In this paragraph, we list the criteria adopted in our simulation. Some of them are our own doings (procedural equivalence and its coupling with numerical identity) while other derive from pioneering works of fellows economists (Axtell et al. 1996) and from neighbouring endeavours, mainly computer science.

Robustness and Equivalence

Obviously, the concept of robustness is related to the notion of equivalence. In fact the possibility to state that a result is (or is not) robust with respect to the algorithmic choice is subject to some underlying idea of how to define likeness of results, that is to say a notion of equivalence. The question of determining whether equivalent results are produced in equivalent condition is by no means trivial as discussed in the outstanding research of Axtell and other (1996). In our research, we will adopt the following criteria:

- **Numerical identity** that implies convergence of different runs on the same numerical value. This equivalence can be expected to hold only for very simple settings. Moreover, it is less likely to show in models containing stochastic and random elements.

- **Statistical (or distributional) equivalence:** the models are said to be statistically equivalent when they produce several distributions of measurements that are statistically indistinguishable the ones from the others (Axelrod 1997, p. 192). The statistical testing takes place through conventional non-parametric statistics (for instance, Mann-Whitney U statistics and Kolmogorov-Smirnov test). The problem is formulated as rejection of the null hypothesis of distributional identity at conventional confidence probabilities[11].

- **Relational Equivalence,** which embodies a looser standard with respect to statistical equivalence, implies that models produce the same internal

[11] "The unsatisfactory part of this approach is that it creates an incentive for researcher to test equivalence with small sample sizes. The smaller the sample, the greater the chance of establishing equivalence" Axelrod (1997) p.194.

relationship among their results (Axelrod 1991, p. 194). For instance, the model could show that the level of cooperation is a given – say quadratic – function of the number of players.

- **Procedural Equivalence,** which implies similarities in the path to outcome. For instance, the solution is found following a linear or cyclical path. Procedural equivalence is a proxy of the way in which the algorithm explore the solution space.
- **Two stages-equivalence,** which pairs numerical identity with procedural equivalence in order to test similarities in the path to outcome. This can be appreciated through graphical inspection. In our case, when two algorithms provide the same amount of payoffs comparison extends to their trend and to the dynamics, that has generated it. It can be appreciated graphically by superimposing the plots of payoff in time and statistically by comparing the values of payoffs in time.

Goodness

The use of the algorithms as to represent the problem solving activity of an agent in an economic model requires the assessment of the performance of the adopted methods. In complex systems where autonomous agent interact, the outcome of the process under study is often unknown and therefore its reliability must be grounded in the detailed knowledge of the learning processes rather than it the goodness of the outcome per se.

We therefore take into account the following attributes of the learning process:

1. Stability of strategies;
2. Memory span, that is the ability to operate with a high number of strategies.
3. Segregation of strategies, that is the ability to isolate strategies that are not currently applied. The ability to create such kind of speciation implies that an algorithm can satisfactorily face a given number of different events.
4. Rielaboration, of unfit rules, that is the ability to quickly modify rules that become unfit due to changes occurred in the environment.
5. Rapidity in recalling strategies that have not been used for a long time due to their

ineffectiveness.

6. Graceful degradation[12] ability to cope with imperfect information.

As we will show each of this features is present in the different algorithms in various degrees. This, as a first consideration, suggests that each methods has a given context of application. Moreover, knowing the performance of algorithms improve the understanding of the outcome of a simulation in which they are included. The simulation setup has been thought as to encompass all these criteria in order to extend comparison to effectiveness and efficiency of learning.

The first scenario allows for testing the effectiveness of methods measured as a ratio of the better stable performance to the maximum performance - which in this closed formulation of the problem can be easily calculated[13] due to the presence of agents with fixed rules. To avoid spurious data we are not using the Noise player and we will generate a single instance for each kind of agent.

The predominance of a strategy would account for the ability to stick to the best strategy once found (criterion sub 1). While the ability to successfully face the agent with fixed rules will account for the memory span (criterion sub 2)

In further scenarios, in addition to the Noise player, we will use more than one instance of each non-learning agents as to differentiate the number of predetermined behaviours in order to check for the ability to create species of strategies (criterion sub 3) to develop the ability to cope with various scenarios.

Moreover, we will introduce the possibility for the agents to disguise their identity. That is to say that they switch their identity in order to confuse (read increase environmental changeability) the artificial mind that has to revise its 'categorisation' by elaborating the strategies that have suddenly turned to be inefficient (criteria sub 4). Furthermore,

---

[12] Other criteria which are relevant from the technical viewpoint are simplicity in the coding of input data and output data; facilità in interpreting and evaluating results.

[13] See appendix 1.

its ability to correctly reconstruct its vision of the world in addition is a test for graceful degradation (criteria sub 6).

By controlling the changes in the identity of agents, the artificial mind will be periodically faced with already experimented environmental conditions. The ability to rapidly recall strategies that have been effective in the past will account for criteria sub 5.

Besides the performances of each algorithm taken singularly, comparison across algorithms in the light of these criteria will allow for an assessment of their relative effectiveness. In addition, comparison of time necessary to perform tasks under the different settings will reveal their efficiency.

Preliminary results

As a first step, we have conducted a set of runs to validate the framework. In details:
1. separate introduction of agent with fixed rules in order to check the correspondence with the theoretical framework. This has implied calculation of payoff and observation of played strategies;
2. functioning of the round robin tournament and of the random pairing of agents;
3. check for the statistical neutrality of the introduction of the Noise agent, which action are determined by a pseudo random uniform distribution.

The first set of runs has concerned a genetic algorithm. This kind of simulation has permitted to highlight the sensitivity of optimal results to the codification of knowledge. This raises a very important question: independently of the internal functioning of the algorithm there is an eliminable ‘subjective’ element (the way in which the investigator choose to convey information into the algorithm) that, ceteris paribus, can produce results that diverge sensibly. In our setting, under some codifications of knowledge the GA was unable to discover the optimal strategy.

Three kinds of codifications have been compared: technically, they consist in different algorithms charged with the interpretation of the strings of value that in turn represent the rules of the GA. We have also modified the length of strings.

Common elements in all the experiments have been:

- Alphabet: each symbol in the string represents an action to execute, 1 or 0 to represent cooperate or defect respectively.
- Fitness: of each rule is given by the sum of the payoffs earned by the agent in a double round robin tournament.
- Agents' type revealed by a characteristic number[14]

Whereas codification has been changed in the following ways:

- Strings of 16 values. The action to activate (correspondent with a position of the string) was determined according to the number passed by the opponent. This consisted in the history of the previous four encounters with the same agent[15].
- Pairing of strategies: instead of considering the strategies adopted in the past by the opponent the second experiment has taken into account two pairs of strategies composed by the action adopted in the last two encounters by the GA and the opponent.
- Last shot: the GA could memorise only the last action it has executed with a given opponent. This implied a string of 8 positions (four agents excluding the GA and two possible strategies for each kind of agent).

Only in the third case, the agent has been able to meet the above quoted requirements. In the other two optimal strategies where discovered only with some type of agent and were not stable in the presence of a variety of types. This amounts to say that the GA was unable to develop strategy apt to cope with different situations. In the first codification, the GA was able to discover the best strategy only when he faced a kind of agent at the time. The ineffectiveness of the learning has to be ascribed to the ambiguity

[14] 0 means Perpetual Cooperator, 1 means Perpetual Defector, 2 tit- for- tat, 3 means tit- for- two -tat, 4 Noise Player (not used here) and 5 means GA. When the simulation is run with a single instance of each kind of agent the type coincides with the identity of the agent.

[15] Disposition with repetition of two elements (cooperate or defect) gives 16.

emerging from the codification of information: when paired with a Tf2t information regarding the two last actions made impossible for the GA to distinguish it from a Tft. Moreover, when the GA learned to cooperate with a Tft the history of their interactions was equal to the that of the Perpetual cooperator. This could cause the GA to:

- Systematically defect the Tft and the Perpetual cooperator with a payoff di 12[16]
- Systematically cooperate with the Tft and the Perpetual cooperator with a payoff di 12[17]

Since the payoff obtained from the two strategies is identical, it was impossible for the GA to distinguish the two strategies preserving their plurality. Since the history was the same for the two kinds of agent, the same position of the string was used to determine the strategy to employ to interact with both a Perpetual Cooperator and a Tft. When by mistake the GA converged on defect with a Tft, a similar ambiguity aroused with the Perpetual defector since both exhibited a story of defections.

The second solution, in our intention, should have solved the problem of distinguishing Tft from Tf2t however it still admitted several ambiguities (for instance, in an history of the kind GA cooperates, Tf2t cooperates, GA defects, Tf2t cooperate will still be not distinguishable from the sequence with a Perpetual Cooperator). In both cases, the algorithm had difficulties in reaching stable patterns.

The effective codification has been found by making the agents declare their type. This was the only way for it to develop univocal strategies. It is worth stressing that the 'mistake' was not in the functioning of the algorithm but in the way it was asked to process information. It follows that the criterion sub 6 is only partially fulfilled by the GA.

A further consideration concerns the evaluation of the rules of the GA's population. The presence of different rules with the same evaluation can significantly reduce the

[16] 1 + 1 + 5 + 5 = 12

[17] 3 + 3 + 3 + 3 = 12

method's performance. Another interesting point is that, at least in this setting, the adaptive strategy performs better than the Tft the traditional winner of the Axelrod's tournaments.

Moreover, we have run computation in order to appreciate:

- Robustness of methods
- Stability: measured as the average payoff over 10000 tournaments after having found the best strategy.
- Effectiveness measured as the amount of cumulative payoff gained in 100 tournaments.
- Memory span measured has the ability to discover the best possible strategy to play with each kind of agent.
- Efficiency measured as the number of tournaments necessary to reach the optimal solution.

In what follows, we report the results of a series of experiments of 10000 tournaments with a GA and one instance of each agent except for the Noise Player with a codification of knowledge 'last shot'.

For the given rules of interaction and the matrix of payoffs, the best possible performance for the GA was a payoff of 26 for each tournament (See appendix 1). From the point of view of game theory, it might be worth noting that such payoff implies a free ride on the cooperation of the other agents. Moreover, this payoff is higher than that obtained from the tit-for-tat strategy.

The following table reports the values referred to the performance for the GA. The first column show the number of tournaments necessary to reach the maximum payoff, the second one the average (per step) performance after having reached the optimum. In the distribution, we have not included two experiments in which the GA had failed to reach the optimum in order to compute the average. The table with detailed data is available in appendix 2.

Number of Tournaments	*Data type*	*Steps to optimum*	*Average Performance*
52	**Average**	3450	25,58
52	**Variance**	360000	0,01

From the table we can draw some tentative conclusions. In details:

1. General effectiveness of the GA is confirmed by its ability to locate the optimal solution in almost all the simulations.
2. The variance of the distribution of the number of steps necessary to reach the equilibrium suggests a link between efficiency in computation and distribution of the pseudo random numbers.
3. The variance of the average results after having located the optimum witnesses for a substantial stability of strategies. Stability is attained in spite of the fact that the learning mechanism remains in action as shown by the small variation due to the genetic operator Mutation.
4. The GA has been able to face four types of different agents also exhibiting articulated behaviours such as Tf2t. It therefore can be said to have passed a first memory test (criteria sub 2).

The experimental setup will also include run with the Noise player which has the role of perturbing the learning process. The expected result is that the GA would express towards the Noise player a strategy of continuous defection as to limit the losses. In fact the Noise player is not as revengeful as the Tft and defecting with him would grant a payoff of 5 in about the half of cases in the rest of the cases it would have earned 1 (an average payoff of 3 per interaction which outperform all mixed strategies.

Concluding remarks

In the present work, introduce an experimental setup to compare different artificial minds. As the AI tools find a steady employment in economics, the necessity of this kind of analyses becomes compelling. On the one hand, we have shown that in spite of the simplicity of the problem that the GA was required to solve we have remarked a high sensitivity of results to the way the relevant information was codified by the researcher.

This should warn us on different aspects of the attempts to deal with complexity. Intelligent artificial agents constitute an advance in exploring complex adaptive systems when the knowledge of the system is only partial and when the researcher cannot – through a traditional mental experiment – acquire it in more details, or when the traditional deductive methods fail to help because of their inapplicability.

However, their sensitivity to codification replicates and somehow magnifies the issue of subjectivity in human perception. The modeller codify its vision of world in the artificial agent without being able to foresee how it will developed and amend in the simulation. The issue is not trivial. When the final state of a complex system is not known (as usually is) to what extent can we trust the solution found by an artificial mind? Would another algorithm embedding a different codification or a different set of focal points converge on the same solution?

More generally how sensitive are the results of economic models to the choice of a given algorithm and to its application to a given instance of problem solving?

In the light of this question, we feel that our analysis can be helpful in clarifying which are the crucial points to take into consideration while engaging in such exercise and the criteria to be use when trying to evaluate our models.

Appendix 1: arithmetical derivation of the maximum payoff (26)

In a run without Noise player in a double round robin tournament the best set of strategies is :

- Always defecting with Perpetual Cooperator gives 5 +5
- Always defecting with a Perpetual Defector gives 1+1
- Always cooperating with Tft gives 3+3
- Alternate defect and cooperate with Tf2 gives 5+3

Appendix 2: Sensitivity analysis to random seed generator for average payoffs

The tables show in details the data summarised in the text. The analysis has been conducted by changing the seed of the random number generator while keeping all the rest of the setup unvaried.

Steps	*Average Payoff*
3002	25,631556
4502	25,517191
3702	25,565507
3502	25,584731
4202	25,552182
4302	23,192557
3202	25,601295
3602	25,597311
3602	25,595435
2702	25,58065
3802	25,586574
2402	25,614058
3802	25,535259
3802	25,535582
3302	25,608183
2802	25,637349
2502	25,650527
4202	25,547697
2602	25,597134
3902	25,556175
3702	25,592822
3202	25,543916
5002	25,485291
4102	25,520773
4202	25,555977
2702	25,644237
3102	25,562128
3702	25,591869
3602	25,598874
3202	25,578049
4302	25,541162
4302	25,480428
3102	25,626794
3602	25,59481
3802	25,58722
3002	25,632128
3002	25,628412
2602	25,650128
2202	25,67013
3302	25,512618

3102	25,623894
3402	25,608004
3402	25,571927
3502	25,558258
3402	25,579203
4502	25,522285
3302	25,595938
2702	25,648075
2302	25,663765
2902	21,939411
3502	25,582269
4202	25,528722
Average 3450	**25,580139**
Variance 360000	0,002643708

References

Arthur B. (1994), “Inductive Reasoning and Bounded Rationality”, *American Economic Review* (Papers and Proceedings), 84,406-411.

Axelrod R. (1997), *The Complexity of Cooperation,* Princeton University Press, Princeton NJ.

Axtell R. – Axelrod R. – Epstein J. – Cohen M. (1996), “Aligning Simulation Models: A Case Study and Results”, *Computational and Mathematical Organization Theory,* 1, 123-41.

Beltratti A., Margarita S, Terna P. (1996), *Neural Networks foe Economic and Financial Modelling*, London, International Thompson Computer Press.

Berkeley I. (1997), “What is Artificial Intelligence”

Conte R. - Hegselmann R. - Terna P. (eds.) (1997), *Simulating Social Phenomena*, Springer, Heidelberg.

Dalmazzone S. – Fontana M. (2006), “Some Reflections on Complexity in Economic Theory”,

Holland J.H. (1975), *Adaptation in Natural and Artificial Systems*, Ann Arbor, University of Michigan Press.

Holland J. - Miller J. (1991), “Artificial Adaptive Agents in Economic Theory”, *American Economic Review*, vol. 81 (2), pp. 365-370.

Markose S. (2005), “Computability and Evolutionary Complexity: Markets as Complex Adaptive Systems”, *The Economic Journal,* 115, pp.159-192.

Fontana M. (2006), “The Use of Simulation in Economics: Evidence on Diffusion and Communication”, forthcoming in Journal of Artificial Societies and Social Simulation.

Foster J. (2005), “From Simplistic to Complex System In Economics”, *Cambridge Journal of Economics,* 29, 873-892.

Terna P. (1997), “A Laboratory for Agent Based Computational Economics: The Self-Development of Consistency in Agents’ Behaviour”, in Conte et al. (1997).