Informatica e diritto, XXXIX annata, Vol. XXII, 2013, n. 1, pp. 115-129

Learning Agents and Decisions: New Perspectives

PIETRO TERNA*

"Away! let's go learn the truth of it"
W. Shakespeare, "Measure for Measure"

1. PREMISE

Is it possible to help different form of knowledge to be closer to hard sciences, providing them of experiment capabilities and theory refusal?

One potential reply is that of widening the field of application of agent-based simulation to new disciplines. With agents in a computer, we recreate actual world on an artificial basis, to see the effects of the action and interaction of entities (our agents) built on rules that we give them in an organized arena.

The roots of this proposal are quite old, referring to the cybernetics dream. With Rosenblueth and Wiener[1], being Wiener the actual founder of cybernetics[2]:

> A material model is the representation of a complex system by a system which is assumed simpler and which is also assumed to have some properties similar to those selected for study in the original complex system. A formal model is a symbolic assertion in logical terms of an idealized relatively simple situation sharing the structural properties of the original factual system.
>
> Material models are useful in the following cases. a) They may assist the scientist in replacing a phenomenon in an unfamiliar field by one in a field in

* The Author is full professor of Economics, University of Turin (Italy). The financial support of Regione Piemonte (ICT4LAW project) is gratefully acknowledged.

[1] A. ROSENBLUETH, N. WIENER, *The Role of Models in Science*, in "Philosophy of Science", Vol. 12, 1945, n. 4, pp. 316-321.

[2] The word is inspired from the Ancient Greek term κυβερνήτης, meaning steersman, governor, pilot.

> which he is more at home. (...) b) A material model may enable the carrying out of experiments under more favorable conditions than would be available in the original system.

Substitute to the "material model" idea (the actual artifact) an agent-based model (the synthetic artifact), and you have exactly the capacity of making "experiments under more favorable conditions than would be available in the original system".

Doing that, we have to be aware that behind the corner, the trap of the complexity is trying to catch us. With Anderson[3]:

> The reductionist hypothesis may still be a topic for controversy among philosophers, but among the great majority of active scientists I think it is accepted without questions. The workings of our minds and bodies, and of all the animate or inanimate matter of which we have any detailed knowledge, are assumed to be controlled by the same set of fundamental laws, which except under certain extreme conditions we feel we know pretty well.
>
> (...) The main fallacy in this kind of thinking is that the reductionist hypothesis does not by any means imply a "constructionist" one: The ability to reduce everything to simple fundamental laws does not imply the ability to start from those laws and reconstruct the universe.
>
> The constructionist hypothesis breaks down when confronted with the twin difficulties of scale and complexity. The behavior of large and complex aggregates of elementary particles, it turns out, is not to be understood in terms of a simple extrapolation of the properties of a few particles. Instead, at each level of complexity entirely new properties appear, and the understanding of the new behaviors requires research which I think is as fundamental in its nature as any other.

A remark: we have a wonderful tool in our hands, to be handled with care. Let now have a look to an agent model running on line, to verify if the explanations meet the self-communicating capabilities of an example.

2. An Example, to Start

At http://eco83.econ.unito.it/terna/chameleons/chameleons.html we have a simple but provocative example: our agents – one hundred as default, but we can increase or decrease their number with the slider *num*, are chameleons, changing their color in given situations[4].

[3] P.W. Anderson, *More Is Different*, in "Science", Vol. 177, 1972, n. 4047, pp. 393-396.

[4] I have to thank Riccardo Taormina, a former undergraduate student of mine, for developing this kind of application with great involvement and creativity. Many thanks also to Marco Lamieri, a former PhD student of mine, for introducing the powerful chameleon idea.

We are using here SLAPP - *Swarm-like Agent Protocol in Python*[5], to develop this kind of model. The basic code demonstrates that we can implement a rigorous protocol like that of Swarm[6] with a simple coding system[7], consistently with the goals exposed in this premise. At the same SLAPP web address, the Chameleons application may also be found, in both SLAPP and NetLogo[8] versions.

In the starting phase, we have chameleons of three colors: red, green and blue. When two chameleons of different colors meet, they both change their color, assuming the third one. If all chameleons turn the same color, we have a steady-state situation. This case is possible, although rare.

But what if the chameleons of a given color want to preserve their identity? On the other hand, what if they strongly want to change it?

The NetLogo version of the chameleon model has been built just to have the experiment running on-line in an easy way: go to the link above and hit *setup* to create the chameleons and hit *go* to run the code. We can so see the chameleons moving randomly in their space. If we tell a specific type of chameleons (i.e., the red ones, choosing R, for runner, in the *smart_red* "chooser" widget) to be conservative, adopting the mind created via the reinforcement learning technique to avoid contacts, they become capable of increasing their number, with the strategy of decreasing their movement, to remain in zones free from chameleons of other colors, and getting close to subjects with their color.

To counter-verify this interpretation, we substitute the minded moves with the simple imperative actions (i) do not move, with numberStep=0 for the red subjects, or (ii) go closer to subjects similar to you, with closeness that can be defined as near closeness or as absolute, with the switches red_close_to_red and not_too_close.

If the red chameleons do not move, they gain some advantage, although quite a limited one; also, staying close without intelligence does not give any advantage to the conservative group, which is in any case moving around for the internal interactions; the advantage arises only from absolute closeness, which leads to the by-product of immobility.

We can underline the difference between the more sophisticated reinforcement learning behavioral rules and the fixed rules machineries guiding

[5] It can be found at http://eco83.econ.unito.it/terna/slapp.

[6] See http://www.swarm.org.

[7] See http://python.org.

[8] NetLogo is a significant simulation tool with an equilibrate compromise between easiness of use and power; we can find it at http://ccl.northwestern.edu/netlogo/index.shtml.

our agents. We also verify that the offensive and defensive behavioral mechanisms emerging from the simulation do not correspond to simple rules *à la* Schelling[9], but come from actions based on more sophisticated rules.

In which way they learned? With self-reinforcement trial and error cycles the learning scheme generates a sequence of successful and unsuccessful choices; via a neural network mechanism, i.e. managed with the quick and powerful *nnet* function of R[10], we memorize both situations; using the accumulated experience. The sequence is represented in Fig. 1.

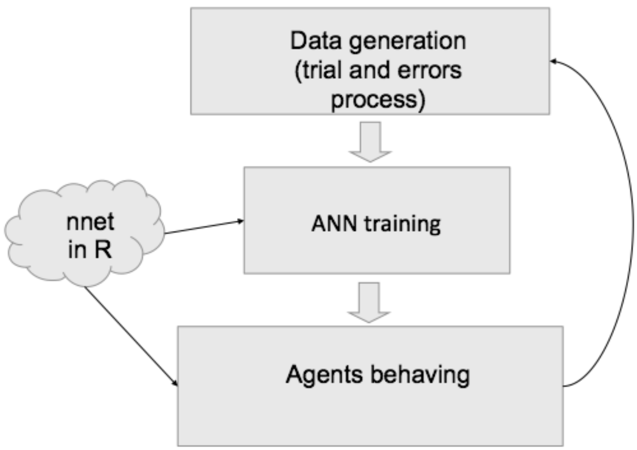

Fig. 1 – The never ending data generation process and the neural network training.

3. Background in Agent-based Models

Following Ostrom[11], and to some extent, Gilbert and Terna[12], in social science, we traditionally build models as simplified representations of reality in two ways: (i) verbal argumentation and (ii) mathematical equations, typically with statistics and econometrics. The first way (i) is absolutely flexible

[9] T. Schelling, *Micromotives and Macrobehavior*, New York, Norton, 1978.

[10] See http://www.r-project.org.

[11] T.M. Ostrom, *Computer Simulation: The Third Symbol System*, in "Journal of Experimental Social Psychology", Vol. 24, 1988, pp. 381-392.

[12] N. Gilbert, P. Terna, *How To Build and Use Agent-based Models in Social Science*, in "Mind & Society", Vol. 1, 2000, n. 1, pp. 57-72.

and adaptable, as in the case of a historical book reporting an analysis of past events, but mere descriptions and discussion, by their nature, preclude tests and verifications of hypotheses. In contrast, the second way (ii) allows for computations and verifications, but suffers from severe limitations in flexibility and adaptation, especially with respect to how agents are expected to operate in the model and when accounting for their heterogeneity and interactions.

There is a third way to build models, (iii) computer simulation, mainly if agent-based. Computer simulation can combine the extreme flexibility of a computer code where we can create agents who act, make choices, and react to the choices of other agents and to modification of their environment – and its intrinsic computability. This allows us to use the descriptive capabilities of verbal argumentation and the ability to calculate the effects of different situations and hypotheses together. From this perspective, the computer program is a form of mathematics. In addition, we can generate time series from our models and analyze them employing statistics and econometrics.

However, reality is intrinsically agent-based, not equation-based[13]. At first glance, this is a strong criticism. Why reproduce social structures in an agent-based way, following (iii), when science applies (ii) to describe, explain, and forecast reality, which is, *per se*, too complicated to be understood?

The first reply is that we can, with agent-based models and simulation, produce artifacts of actual systems and "play" with them, i.e., showing consequences of perfectly known *ex-ante* hypotheses and agent behavioral designs and interactions. Then we can apply statistics and econometrics to the outcomes of the simulation and compare the results with those obtained by applying the same tests to actual data. In this view, simulation models act as a sort of magnifying glass that may be used to understand reality in a better way.

Considering the analysis of an *agent-based simulation model* - ABM as a source of knowledge, there is another "third way view" of these kinds of tools. Referring to Axelrod and Tesfatsion[14]:

[13] For a short, but illuminating discussion of this consideration, see S. WEINBERG, *Is the Universe a Computer?*, in "The New York Review of Books", Vol. 49, 2000, n. 16, http://www.nybooks.com/articles/15762, in his review of Wolfram's book, *A New Kind of Science*.

[14] R. AXELROD, L. TESFATSION, *A Guide for Newcomers to Agent-based Modeling in the Social Sciences*, in Judd K.L., Tesfatsion L. (eds.), "Handbook of Computational Economics.

> Simulation in general, and ABM in particular, is a third way of doing science in addition to deduction and induction. Scientists use deduction to derive theorems from assumptions, and induction to find patterns in empirical data. Simulation, like deduction, starts with a set of explicit assumptions. But unlike deduction, simulation does not prove theorems with generality. Instead, simulation generates data suitable for analysis by induction. Nevertheless, unlike typical induction, the simulated data come from a rigorously specified set of assumptions regarding an actual or proposed system of interest rather than direct measurements of the real world. Consequently, simulation differs from standard deduction and induction in both its implementation and its goals. Simulation permits increased understanding of systems through controlled computational experiments.

The considerations above act in a way similar to abduction, or inference to the best explanation, where one chooses the hypotheses that, if true, give the best explanation of the actual evidence. Note that in the ABM perspective, the hypotheses are also related to the rule that determines the behavior of the agents.

4. More on Learning

Things become more and more complicated when we put learning capabilities into our artificial agents. The line of work proposed in the chameleon example is that of generating data via a trial and errors process and then to take note of the successful and unsuccessful replies to the different situations, together with the action done and the related degree of achievement or failure. How to store a huge quantity of information of this kind? In other terms, how to apply the reinforcement learning strategy in a wide sense? We are here referring to the field of machine learning, as perfectly engineered in packages like *mlpy*[15].

A sub set of this wide field of techniques is that of the artificial neural networks; specifically, feed forward ones[16]. The best application of this kind of tools is in classification, mapping an input vector to an output one. A Neural Network (NN) function contains parameters that have to be estimated, but based on which data? This is the key point, because, only if we have data, the estimation of a function like (1) is possible. x is a vector of dimension n

Vol. 2: Agent-Based Computational Economics", Amsterdam, North-Holland, 2005, pp. 1647-1658, http://www.econ.iastate.edu/tesfatsi/GuidetoABM.pdf.

[15] See http://mlpy.sourceforge.net.

[16] C.M. Bishop, *Neural Networks for Pattern Recognition*, Oxford, Clarendon Press, 1995.

containing information and z is a vector of dimension m containing actions. Omitting — in the notation only — the constant input needed to evaluate the so-called bias (same role of the intercept in linear estimations), A matrix has dimension h, $n+m$ where h is the intermediary dimension of the function (a vectorial function of vectors) or the so-called number of hidden nodes; B has dimension k, h where k is the number of possible effects.

(1) y = g(x,z) = f(B f(A (x',z')'))

If z = $z_i, \ldots, z_m$ and for each case in the set, we want to evaluate a separate effect, we can also have m eq. of the (2) type, where the index i is 1,...,m, so the action is implicitly inside its related equation.

(2) y_i = g(x,z) = f(B f(A (x))

f is an S shaped function, such as $1/(1 + \exp(-x))$ or similar.

If we have data describing the behavior of each artificial agent, we can estimate A and B in a direct way. The evaluation of the parameter values of NNs is complicated, but technically possible.

We use both techniques close to that of non linear multiple regression or the iterated back correction (so-called back propagation of the errors). In application here, we use the *nnet* function of R prepared by prof. Ripley; the function is described in Fig. 2.

nnet {nnet}

R Documentation

Fit Neural Networks

Description

Fit single-hidden-layer neural network, possibly with skip-layer connections.

Usage

`nnet(x, ...)`

Optimization is done via the BFGS method of optim.

Method `"BFGS"` is a quasi-Newton method (also known as a variable metric algorithm), specifically that published simultaneously in 1970 by Broyden, Fletcher, Goldfarb and Shanno. This uses function values and gradients to build up a picture of the surface to be optimized.

References

Ripley, B. D. (1996) *Pattern Recognition and Neural Networks.* Cambridge.

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Fourth edition. Springer.

Fig. 2 – The nnet function in R and its characteristics.

To evaluate the parameters of the function, which are contained in A and B matrixes, we need data (observations). A rare case is that of having observed the behavior of the agents. As an example, I am building a simulation framework about the behavior of pupils of the primary school. Two young scholars of educational science followed four classes for several weeks, recording movies with a hidden camera; so, we have observations linking *ex-ante* situations and actions done, x and z, to the effects y. Consider two observed subjects, α and β: their data form two tables as in Fig. 3.

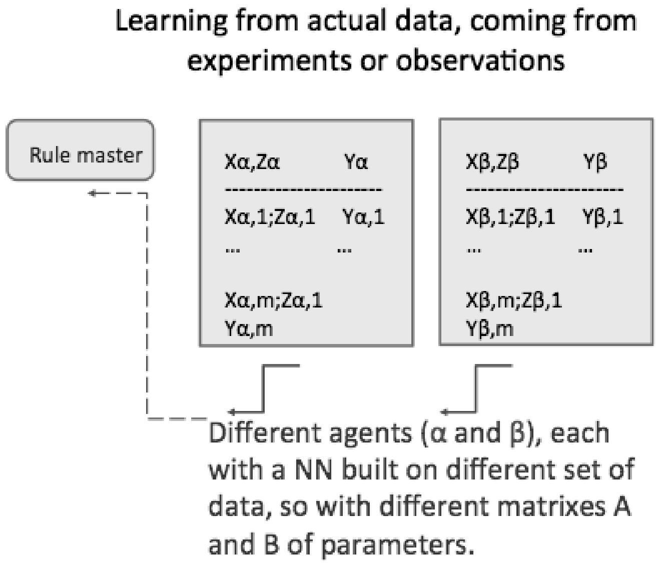

Fig. 3 – Learning from actual data (observation).

Unfortunately, using agent-based models, the commonest situation is that of the absence of observed data at the level of the agents. We have to generate them, as described in Fig. 1. Agents behave, doing actions z, different for each situation x, regardless of the upcoming results; these are evaluated by the simulation environment which we are running, taking into account the actions of each agent and all the consequent interactions.

This way of acting and learning is recapitulated in Fig. 4, where we only better display the analysis reported above. Contemporary, we explicate the way by which the agents chose: executing in each x case, the z action (or actions) giving the y that has the higher effect both in an individual U function (that we can interpret as utility function, if we prefer to reason in this way) of in some societal wellbeing evaluation. In Fig. 4, we underline that laws modify the effects y and so the actions z, while we can interpret that social norms modify U.

The use of NNs in the above construction is a generalization of the usual reinforcement learning schemes[17], where from the experience, we modify some parameter or some possible/forbidden action. With NNs we have to cope with a quite complicated tool, but the effort is paid back by the fact the NN can interpolate a never seen *ex-ante* case x among already known situations (generalization capability), and mainly it is implicitly capable of fuzzy realistic replies about the rewards y.

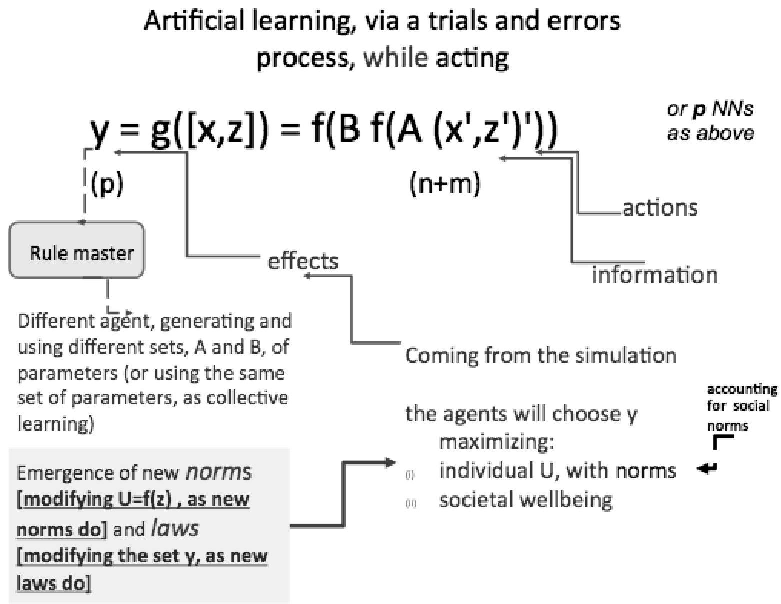

Fig. 4 – Acting, generating effects and learning.

[17] R.S. SUTTON, A.G. BARTO, *Reinforcement Learning: An Introduction*, Cambridge, MIT Press, 1998.

5. An Example of Learning and a Crucial Question

We are currently developing a SLAPP extension for learning[18], based on R function nnet, described in Fig. 2, in which agent act, with errors, and learn from their errors.

As a quick example, we start from a situation of randomly distributed agents in a space, and then we order them to try to learn, via initially random moves and after that with more and more wisely directed actions, to form groups (Fig. 5) or to stay alone (Fig. 6).

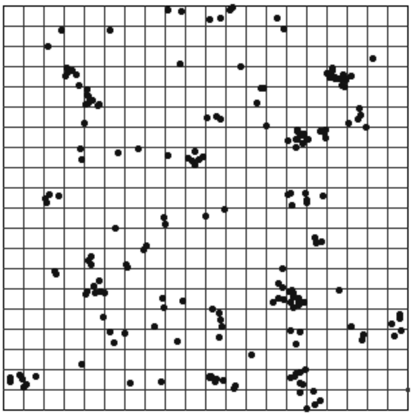

Fig. 5 – Agents forming groups

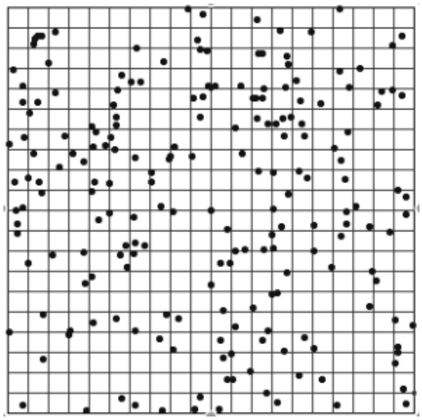

Fig. 6 – Agents staying alone

The crucial question now is: *why* they do that?

Apparently, it is an irrelevant question: they do that because we asked them to learn how to behave to accomplish that action, but we are considering a tiny problem. In a highly complex one, with different types of agents, acting in very distinctive ways, to have the capability of tracing, in our simulator, with precision the kind of behavior that the agents are following and the explanation of their choice is extremely important.

We have to add, in our model, a layer dealing with the so-called *Beliefs Desires Intentions* - BDI agent definition. In SLAPP that layer presently does not exist. We can quite easily refer to an extension of NetLogo, adding

[18] Named z_learningAgents_v.?.?.zip, at http://eco83.econ.unito.it/terna/simoec12/Python_examples/ (November 2012) and in perspective to be included in SLAPP site at http://eco83.econ.unito.it/terna/slapp/.

BDI capabilities, with a few simplifications, as a project of the University of Macedonia, in Greece[19].

This excellent tool[20] uses both the BDI scheme, focusing on Beliefs and Intentions, and a sophisticated formalism to develop messages between agents, based on the FIPA - *Foundation for Intelligent Physical Agents* specifications[21].

Is this construction a not useful added complication? At a first look, it seems to be, but if we use a model of this class, with non trivial agents doing sophisticated action, immediately we understand that we receive a significant added value by having the possibility of being formally informed, by the simulator, of why and in which way agents act. From Sakellariou and colleagues[22], we fully understand that:

> (...) Agent planning, commitment strategies, agent architectures, message passing, cooperation protocol design and evaluation, issues on functional and spatial decomposition of problems, and even team formation and disbanding can be addressed given an appropriate scenario.

We are now in the presence of a very complicated crossroad, with: (i) learning in agents as first element, to be able to understand how agents modify their behavior, (ii) BDI definition to clarify the motivation of that behavior. A very few works exist in that direction[23] and presently no one is used a generalized learning scheme as proposed here. To the cross-road we have to connect two open directions: (a) that of the micro-macro link, which is a key step in understanding the world we are immersed in; a significant reading on that is Chapter 1 "What is agent-based computational sociology all about?"

[19] See http://users.uom.gr/ ˜ iliass/projects/NetLogo/.

[20] I. SAKELLARIOU, P. KEFALAS, I. STAMATOPOULOU, *Enhancing NetLogo to Simulate BDI Communicating Agents*, in "Lecture Notes in Computer Science", 5138, 2008, pp. 263-275, http://users.uom.gr/ ˜ iliass/projects/NetLogo/Papers/Extending_NetLogo_SETN08_SVerlag_Camera_ready.pdf.

[21] See http://www.fipa.org.

[22] I. SAKELLARIOU, P. KEFALAS, I. STAMATOPOULOU, *MAS Coursework Design in NetLogo*, in "Proceedings of the Educational Uses of Multi Agent Systems (EduMAS 09)", Budapest, 2009, http://users.uom.gr/ ˜ iliass/projects/NetLogo/Papers/NetLogoCoursework_EDUMAS.pdf.

[23] A. GUERRA-HERNÁNDEZ, G. ORTÍZ-HERNÁNDEZ, *Toward BDI Sapient Agents: Learning Intentionally*, in Mayorga R.V., Perlovsky L.I. (eds.), "Toward Artificial Sapience: Principles and Methods for Wise Systems", London, Springer, 2008, pp. 77–91, http://www.uv.mx/aguerra/documents/2008a-guerra.pdf.

in Squazzoni[24]; (b) the interaction between our agents, mainly if considered as robots (think about algorithmic robots acting in high-frequency trade in stock-markets) and the huge arena of responsibility and law[25].

The research field is huge.

6. Decisions and Actions in Policy and Law: Could Agent-based Simulation Help?

Starting from the field described hereto, the question is what is passing around in the field of policy and collective action, related to simulation and mainly agent-based one. The key idea is now that of defining policy in a participative way, with citizens, and from there to evolve laws and improve social norms in a better understood and shared approach.

Is that a dream or a research field, huger than the previous one? Let us reply with a few existing examples.

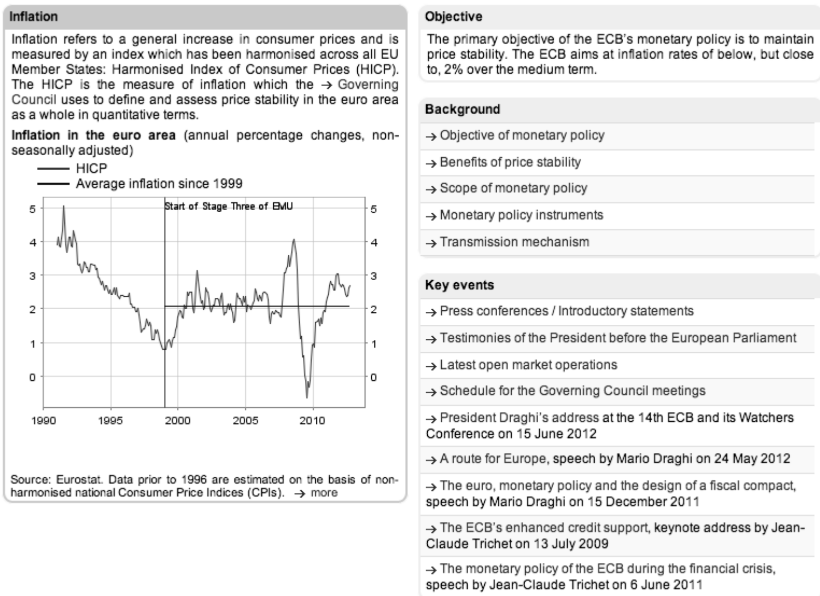

Fig. 7 – The background of the ECB game.

[24] F. Squazzoni, *Agent-based Computational Sociology*, Chichester, Wiley, 2012.

[25] U. Pagallo, *Robotica*, in Durante M., Pagallo U. (a cura di), "Manuale di informatica giuridica e diritto delle nuove tecnologie", Torino, UTET, 2012, pp. 141-158.

6.1. European Central Bank

With €CONOMIA - *The Monetary Policy Game* - we can play to be the president of the European Central Bank. From the web site[26]:

> Ever wondered what monetary policy is? Or how the key interest rate affects inflation? Play €CONOMIA and find out how it works. Your goal: Keep inflation low and stable at just under 2%. Your tool: the key interest rate.

As seen in Fig. 7, the background of the game is extremely well justified and thoughtful.

6.2. Edgeryders, Transition to the Future

Things seem to become more and more complicated in Edgeryders[27], where:

> You are riding the edge to the future, and that makes you an expert on the transition. Inspire European decision makers to take action so that your and others' journey to the future are made a little smoother and faster, and no one gets lost along the way.
>
> Edgeryders is a social game aimed at building a nurturing environment where we get help, inspire one another and make sense of it all. You may wonder what the big deal with the Reputation score is. To make a long story short, you get Rep when you participate, you play Missions, you share what you do with your friends on Facebook and Twitter, you talk to other Edgeryders and comment their posts. Everything that keeps the community healthy should give Rep.
>
> The whole idea about Rep is to make easy for newcomers and expert Edgeryders alike to connect with people who are ready, willing and able to help and play together. This is not a competition, we're all in this together.
>
> Young people everywhere (and quite a few not-so-young people, too) are busy building their lives – and, as they do so, they build our common future, piece by piece.

6.3. eGovernment Meets the eSociety

A further step, close to be tangible, at wegov (where eGovernment meets the eSociety)[28], where we read that:

[26] See http://www.ecb.int/ecb/educational/economia/html/index.en.html.
[27] See http://edgeryders.ppa.coe.int.
[28] See http://wegov-project.eu.

> Social networking technology provides major new opportunities for policy makers (eGovernment) to engage with the community (eSociety).
>
> We will develop a toolset that allows full advantage to be taken of a wide range of existing and well established social networking sites (Facebook, Twitter, Bebo, WordPress etc.) to engage citizens in two-way dialogs as part of governance and policymaking processes. The tools will make it possible to detect, track and mine opinions and discussions on policy oriented topics.
>
> The tools will allow discussions to be seeded and stimulated through injection of policy discussion points into relevant communities in a secure and managed way. The tools will allow the origins, bias and evolution of opinions to be tracked to provide auditable records of provenance, guard against misuse, and ensure trust and privacy for all involved.

6.4. Laws in a Bottom-up Process

Finally, at http://gigaom.com/europe/online-crowdsourcing-can-now-help-build-new-laws-in-finland, we read about an actual living agent application of bottom-up true democracy:

> Who makes laws? In most of the democratic world, that's the sole preserve of elected governments. But in Finland, technology is about to make democracy significantly more direct.
>
> Earlier this year, the Finnish government enabled something called a "citizens' initiative", through which registered voters can come up with new laws – if they can get 50,000 of their fellow citizens to back them up within six months, then the Eduskunta (the Finnish Parliament) is forced to vote on the proposal.

6.5. Barabási's Considerations

Is all that a sort of science-fiction literature? Is all that only positive or contains dangerous elements? A conversation of Barabási[29] opens new perspectives, in a concrete way:

> One question that fascinated me in the last two years is, can we ever use data to control systems? Could we go as far as, not only describe and quantify and mathematically formulate and perhaps predict the behavior of a system, but could you use this knowledge to be able to control a complex system, to control a social system, to control an economic system?

[29] A.-L. BARABÁSI, *Thinking in Network Terms. A Conversation with Albert-László Barabási*, http://www.edge.org/conversation/thinking-in-network-terms, 2012.

7. A Temporary Conclusion

We are back to the crossroad presented above, and to the question if agent-based simulation could help in this perspective of policy management and law creation or norm emergence. Following the assessments introduced here, adding the network framework, in the Barabási's sense, to agents is a big significant open issue, still to be improved in spite of not being a novelty[30].

A program of future work is: the integration of learning capabilities in SLAPP; the creation of a BDI layer for SLAPP; the creation of a learning BDI version; the use of networks to connect the agents in simulation frameworks.

[30] E. Bonabeau, *Agent-based Modeling: Methods and Techniques for Simulating Human Systems*, in "Proceedings of the National Academy of Sciences of the United States of America", Vol. 99, 2002, n. 3, pp. 7280-7287.