Modeling the 'El Farol Bar Problem' in NetLogo

PRELIMINARY DRAFT*

Mark Garofalo†

Dexia Bank Belgium
Boulevard Pachéco 44
B-1000 Brussels, Belgium
http://www.markgarofalo.com

Abstract

Arthur's 'El Farol Bar Problem' (Arthur (1994)) is a metaphor of a complex economic systems' model based on interaction and it is perfectly suited for the multi-agent simulation framework. It is a limited resource model in which agents are boundedly rational and where no solution can be deduced *a priori*. In this paper we discuss in particular the implementation of the model in NetLogo, a multi-agent platform introduced by Wilensky (1999). Our aim is to translate Arthur's model described in natural language into a workable computational agent-based tool. Throughout the process of setting up the tool we discuss the necessary assumptions made for it's implementation. We also conduct simulations to illustrate the model and to show that the design of the tool is appropriate.
Keywords: Complex system, El Farol Bar Problem, model, multi-agent, NetLogo, simulation.

1 Introduction

The problem set up by Arthur in the 'El Farol Bar Problem', here after EFBP, is a popular paradigm of complex economic systems in which agents self-organise themselves while they are in competition for a limited resource, without possibility of communication, thus there is no solution deductible *a priori*. In this problem, every agent has to choose *to go* or *not to go* to the 'El Farol' bar each week using a predictor of the next attendance. It is given that the agents try to avoid crowd and they will not show up at the bar if they forecast that more than sixty percent of the agents go. But since there is no single predictor that can work for everybody at the same time, there is no deductively rational solution. In fact the agents would have to know which predictors the others will use, which they don't, so from the point of view of the agent, the problem is ill-defined. They can only observe the attendance, thus their reasoning, as pointed out by Arthur, is *inductive*. The consequence is a self-defeating prophecy feature because at each attendance forecast, 'if all believe few will go, all will go (...) and if all believe most will go, nobody will go' (Arthur (1994)).

Resource limitation is characteristic to many other areas such as road traffic, computer networks like internet or financial markets. These systems are complex in the sense that they composed by a large variety of elements and that the interaction in between these elements are non-linear and not completely captured (LeMoigne (1990), Kauffman (1993)). Paraphrasing Casti (1996), in the EFBP the agents are in a universe in which their 'forecasts (...) act to create the world they are trying to forecast'[1]. Arthur's simple scenario provides useful insights into complex systems and this is why it is viewed as paradigmatic.

Modeling in mainstream economics is generally equation based. However to study the EFBP, rather than considering a particular stochastic process, a computational agent based approach seems to be an appropriate alternative[2] (Gilbert and Troitzsch (1999)). A computational agent-based model includes interacting agents who rely on their experience, rules and information to determine their actions. Agents can interact in both space and time, creating emerging dynamic patterns, and potentially new behaviours not introduced by the modeler (see Reynolds (1987)). Computational models are not constrained by the limits imposed by the desire of elegance in mathematics. The models also permit heterogeneity in not only agent preferences but also their behavior, offering flexibility. Several references discuss multi-agent systems; Weiss (2000) and Ferber (1995) are good ones.

The agent-based approach seems therefore to be a very natural and flexible way to model the EFBP, and that is what we have decided to apply here. The EFBP has already been studied by simulation in Arthur's paper and by Fogel, Chellapilla, and Angeline (1999). Mathematical formalisation has also been used by Challet, Marsili, and Ottino (2004) based on the findings from the *minority games* (Challet (2000)), however we propose here to rewrite step by step an agent-based version of the model and we perform simulations of the collective behaviour to align them with Arthur's findings. Rewriting models that others have described is common practice in the multi-agent based simulation community in order to understand them more deeply and reproduce the stated results (Hales, Rouchier, and Edmonds (2003), Axelrod (1997)).

The rest of this paper is organised as follows. Section 2 presents the EFBP and discusses Arthur's modeling assumptions. Section 3 presents NetLogo, the multi-agent

†Any opinions expressed here reflect the current judgment of the author, and do not necessarily reflect the opinion of Dexia Bank Belgium or any of its subsidiaries and affiliates. Neither Dexia Bank Belgium nor its subsidiaries/affiliates accept any responsibility for liabilities arising from use of this document or its contents.

[1] I took this quote from Zambrano (2004).

[2] For a deeper discussion on equation based modeling compared to agent based modeling, see Parunak, Savit, and Riolo (1998).

platform we use to perform the simulations and presents the set up of the user interface. Section 4 briefly discusses the implementation of the EFBP. In order to validate the computational tool, we perform simulations on the collective bahaviour and discuss the results in Section 5 and finally Section 6 concludes.

2 Brian Arthur's Model

El Farol is a bar at Santa Fe, New Mexico, where a band plays Irish music on Thursday nights, and, like in many places, the bar is enjoyable only if it is not too crowded. The N (indexed by i) inhabitants of Santa Fe have the same fixed preferences L (the open level), or more generally l, with $l = L/N$; the question asked by Arthur is to find out how these N agents will select the following binary action $a^i \in \{0, 1\}$, where 1 is *go* to the bar and try to avoid crowd to listen to good music, and 0 is *not to go*, and this without any communication or exogeneous information. Thus the total attendance is $A = \sum a^i$ and if $A > L$, the bar does not open, otherwise the bar opens and they have a great evening ($A \leq L$).

In order to take their decision, the only available piece of information to predict the opening at the week t, $A(t)$, that the inhabitants have is the attendance from the d previous weeks[3],

$$\eta_t = \{A(t-d), ..., A(t-2), A(t-1)\}, \qquad (1)$$

published each Friday morning in the local paper.

On this basis, they formulate forecasts of $A(t)$ using models called *predictors*. Arthur supposed that there exists a large amount (several dozen), n, of variations of predictors, called the 'alphabetic soup', S^j, with $S^j = \left\{s^1, s^2, ..., s^n\right\}$ from which each agent N selects randomly a portfolio k from S, with $k < n$. A predictor forecasting for example an attendance lower than L implies that the agents having that predictor in their portfolio, and using that one to forecast, will go to the bar because he expects that it will not be crowded.

Arthur gave some information on the predictors by giving examples; seven to be precise.

- The same attendance as the previous week $A(\eta_t) = A(t-1)$, as two weeks ago $A(\eta_t) = A(t-2)$ or as five weeks ago $A(\eta_t) = A(t-5)$, which he defines as n-period cycle detectors, with $n \in \{1, 2, 5\}$,
- a mirror image around 50% of the number of inhabitants ($N/2$) of the attendance of the last week $A(\eta_t) = N - A(t-1)$,
- a constant forecast $A(\eta_t) = 67\%$
- the four week rounded moving average, that is the average of the last four weeks $A(\eta_t) = 1/4 \sum_{\tau=1}^{4} A(t-\tau)$,
- the trend over the last eight weeks, rounded and bounded by zero and the maximum attendance, N,

  $A(\eta_t) = min([trend\{A_8\}]^+, N)$ with $x^+ \equiv max(x, 0)$ and $trend\{x\} \equiv$ the trend computed using the least squares method.

[3]The mathematical formalisation of the history set and of the predictors are inspired/taken from Challet, Marsili, and Ottino (2004).

At each time step, week after week, the agent uses the most performing predictor - currently the most accurate - over the past d weeks, to decide to go or not to the bar, called the *active predictor*. Thus if the attendance prediction for period t given history η_t of the active predictor $s^{ap}(A(\eta_t)$ is greater than L the action for the i^{th} agent based on that information, $a^i\left[s^{ap}(A(\eta_t))\right]$, is 0 and 1 otherwise. The way in which Arthur attributes the performance to the predictors in order to classify them to determine the active predictor is not specified, however reviewing literature, we propose three interpretations.

We start with what we call the *absolute precision*. This interpretation establishes a function rewarding the predictor indicating the best the opening or not of the bar, that means the occurrence of the event. Say for example, with an attendance of $A(t) = 55$, a predictor giving 10 or 50 is equivalent. Parkes and Steinig (1997) use this interpretation for the precision of the predictors of the inhabitants, and is measured by associating a *confidence* level. They compute it with the 'percentage of times that the predictor has been on the correct side of sixty percent'. The inhabitant, they continue, will simply use at each time step the predictor with the highest confidence level.

The second interpretation is one we call the *relative precision*, and has been used by Brandouy (2003), who gives an interpretation of the efficiency of the predictors by its 'difference between the attendance it has just predicted and the realized attendance once every agent has decided to go or not to the bar', (implicitly supposed in absolute value). The quality of this evaluation function is to be an attendance measure, however, on the contrary to the previous one, the past performance is not taken into account.

A paper from Zambrano (2004) seems however to take away all doubts of this discussion by stating that his evaluation function has been given to him in Arthur's original code thanks to Bruce Edmonds, which we call the *'original' precision*, and is given as follows

$$U_t(s^j) = \lambda U_{t-1}(s^j) + (1-\lambda) \mid s^j(A(\eta_t)) - A(t) \mid, \qquad (2)$$

where η_t (equation 1) is the history of attendance up to period $t-1$ of the d previous weeks, $s^j(A(\eta_t))$ is the attendance prediction of predictor s^j for period t given history η_t, $A(t)$ is the actual attendance for period t, and λ is a number strictly between zero and one.

The absolute and relative precision evaluation functions do remain very interesting and should be considered for further investigation. In this paper however, we only consider equation 2, the 'original precision', for our implementation and the simulations we perform.

3 Agent-Based Modeling in NetLogo

In order to perform our simulations, we designed the EFBP model using NetLogo[4] and it is available on the community website[5] since May 2004. We invite the interested reader to test the simulation (source code included), with the aid of a Java-enabled Internet browser. NetLogo is an agent-based parallel modeling and simulation environment produced by the Center for Connected Learning and Computer-Based

[4]http://ccl.northwestern.edu/NetLogo/

[5]http://ccl.northwestern.edu/NetLogo/models/community/ElFarolBarProblem

Modeling at Northwestern University (Wilensky (1999)). It is particularly well suited for modeling complex systems evolving over time. Modelers can give instructions to hundreds or thousands of independent 'turtles' (the agents) all operating concurrently. This makes it possible to explore connections between micro-level behaviors of individuals and macro-level patterns that emerge from their interactions. NetLogo is easy to get started, with a 'low threshold', and no limitations for advanced users, called 'no ceiling' (Tisue and Wilensky (2004b)).

We chose the NetLogo modeling and simulation environment as a platform as it is well adapted to all of our needs. There exists several dozens or more programming environments of which NetLogo is but one. However we have found this a good choice for our work, and here are some of the reasons.

In order to avoid biased interpretation of the results or other errors, we required a simple language for our implementation, which is the case of NetLogo's declarative language (as BASIC). It is freeware, it has excellent support, and a large and active user community. It comes packaged with extensive documentation and tutorials and a large collection of sample models. It is easy to read and execute a text file of commands that setup, run, and record the results of an experiment. Moreover, its built-in visualization allows dynamic, flexible, and customizable views of model results. NetLogo has been under development since 1999. More on it's origins, a tour of the interface, an introduction to the coding language and other details is presented by Tisue and Wilensky (2004a). Finally NetLogo (version 2.1.0) has a friendly graphical user interface (see Fig. 1).

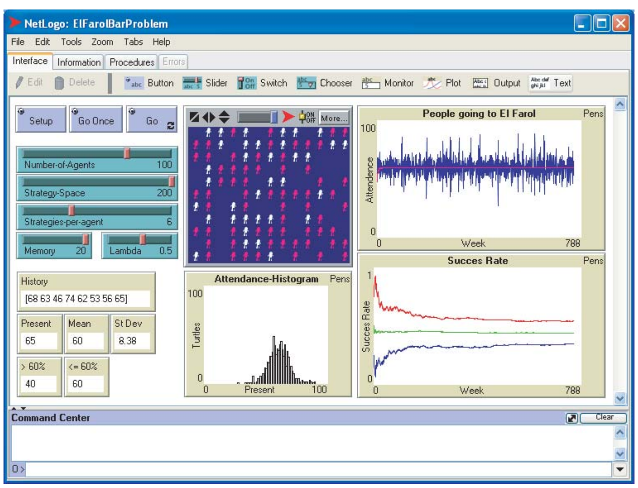

Figure 1: NetLogo's *graphical user interface (version 2.1.0), with the El Farol Bar Problem running on the simulation platform.*

Figure 1 shows the composition of simulation platform, with the controls (sliders and buttons), the world (where the turtles interact), and the display of information (monitors and plots).

The `Number-of-Agents` slider is straight-forward. It determines the number of agents in the game, set at 100 in the standard game, (from 0 to 143). The `Strategy-Space` will determine the size of the 'alphabetic soup' (S = 6 to 200). It allows to user to simulate the impact (if any) of the total amount of strategies before distributing them to the agents. The `Strategies-per-agent` distributes randomly the k strategies (3, 4, ..., or 12 out of S) to each agent. The `Memory` slider will determine the length of the past data (equation 1) the agents use (5 to 20). Since the past attendance is the only available information, patrons having a long memory are more informed. Finally the `Lambda` slider is used in equation 2 to compute the performance of each strategy. A low λ gives more importance to current performance while a high λ uses the past (0 to 0.9). Note that a if λ is set to 0, the predictor performance evaluation only uses current performance thus reduces to the 'relative precision' described in Section 2. We also excluded to consider only the past without adding any new information on the performance ($\lambda = 1$), which depends in this case only on the first attendance.

As in almost any NetLogo simulation, there is a `SETUP` and a `GO` button, which control the execution of the model. The `SETUP` button resets the system and prepares the model to be run. The agents and the strategies are created, and then the strategies are distributed randomly. These are determined by the data from the sliders. The `GO` button, a forever button, runs the model and the `GO ONCE` button is the same except the agents only take one step.

Once the set up finished, the simulation can be launched, and the agents start taking their decisions. The agents have a binary choice, to go or to stay, respectively colored in pink and white and shown in the world[6]. On the top-right part of the simulation screen the attendance at the bar (with the average) is plotted and the histogram is plotted next to it. The best, worst and average score vs time are plotted in the bottom right part of the screen. There are also a few monitors (attendance, mean, standard deviation, number of predictors choosing to go and to stay). There are many displays of information, however in order to understand the results, the implementation of the model has to be discussed, which we do in the next Section (Section 4).

4 Architecture and Implementation

In order to analyse thus understand the results of the simulations of the EFBP in NetLogo we must discuss it's implementation. The schematic representation of the model (Figure 2) puts in evidence the feed-back effects in between the levels (Langton (1989)); the agents' actions (microscopic level) and their attendance at the bar (macroscopic level).

The logic behind the simulation is as follows. The first step consists in the setting up of the environment/bar, the specifications of the agents and the strategies creation and distribution, according to the memory. This is done according to the data retrieved from the sliders on the user interface. The simulations may then begin and all of the agents compute a forecast for the next week attendance using each one of their predictors. They will go if this prediction is equal or less than the open-level and not if it is more, based on their active predictor.

```
to select-action
    ifelse (active-predictor <= open-level)
      [set choice 1]
      [set choice 0]
end
```

[6] I used the same world representation as Nigel Gilbert for SITSIM, available at `http://ccl.northwestern.edu/NetLogo/models/community/Sitsim`

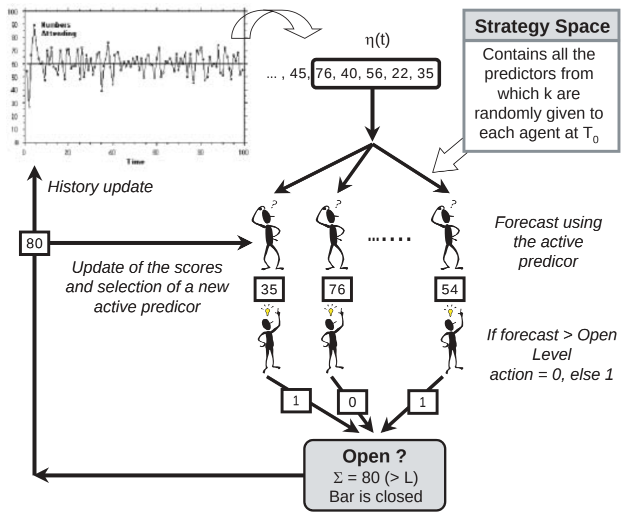

Figure 2: *A schematic representation of our approach to the El Farol Bar Problem.*

Then the number of agents having gone to the bar are counted and the bar opens or not. The colors in the world are refreshed in the GUI, and the attendance is added to the history list.

```
to turtles-setup
  ask turtles
     [ifelse choice = 1
        [set color pink]
        [set color white]]
     set count-list lput count turtles with
        [choice = 1] count-list
     set count-list-go lput ((count turtles
        with [active-predictor > open-level])
        * 100 / count turtles) count-list-go
     ifelse (last count-list > open-level)
        [set history 0]
        [set history 1]
end
```

Since a new opening information is available, the agents update their succes rate, which is the value based on the agents' portfolio of predictors (used to compare individual performance). The scores that agents get to evaluate their success versus time is 1 if they predict the correct outcome (to go or to stay when the bar is open or closed respectively) and 0 if they don't figure out the correct outcome.

```
to update-agent-score
    ask turtles [
        ifelse (choice = history)
            [set agent-score agent-score + 1]
            [set agent-score agent-score + 0]
    ]
end
```

At this step each agent also updates the accuracies of all of their predictors using equation 2 in order to determine a new active predictor for the next forecast.

```
to update-scores-agent-list
ask turtles [
  set step 0
  repeat (strategies-per-agent)[
    set strategies-scores replace-item step
     strategies-scores
      ((Lambda * (item step strategies-scores))
      + ((1 - Lambda) * (abs (last count-list
      - (item step individual-choices-list)))))
     set step (step + 1)]
  ]
end
```

In our simulation the strategy space contains 200 predictors, which are composed mainly by trends, moving averages, fixed rules and tit-for-tat rules. For the special case where the history is not long enough yet, the predictors choose randomly. For a 20 day tit-for-tat on week 10 for example, the prediction will be random[7] and the code looks like this.

```
to TitForTat
ifelse
(length count-list < (item step individual-strategy-list) + 1)
   [set forecast (random count turtles)]
   [set forecast (item ((length count-list)
      - (item step individual-strategy-list)- 1) count-list)]
end
```

The predictors we implemented are simple, limited in number and are often similar in the sens of forecast (the nine and the ten day moving average will forecast a similar attendance). Therefore our work may be viewed as a starting point that could be improved by adding more diversified predictors, but it is however robust enough to perform simulations of the EFBP and we believe that it should be more viewed as a 'skeleton' of the model instead.

5 Simulation Results

In order to evaluate the quality and measure the efficiency of our implementation, we compare our results with the ones from Arthur's paper and since the only results are concerning the collective behaviour, we use that as a benchmark. The domain of validating computational models is in fact less straight forward as in mathematical theory for example and one of the most common technique of validation is comparison of results (Axtell, Axelrod, Epstien, and Cohen (1996)).

The simulations we perform are based on the following parameters; The number of agents, N, is set to 100, the simulations are done over 150 weeks, each agent chooses randomly k, equal to 6, predictors out of 200, d, the memory, on which the predictors are computed, is set to 20 weeks and the weighting factor λ is set to 50%.

Figure 3 depicts an output of a simulation whose emerging result is similar to the one reached by Arthur (1994), with the attendance at the bar (full line) oscillating around 60, the opening level, L. It is interesting to notice that the average (dashed line) hardly moves and seems to stabilize around this level. The dynamics seem random, despite that no random component determines the dynamics of how many people go (Casti (1996)). In fact, although at a microscopic level each agent is applying a different predictor at any one time, with varying degrees of success, when viewed globally, they seem pretty indistinguishable. The only random part of Arthur's model is the setting-up. Therefore once the game is set the output is completely deterministic since the rule update is stable.

[7] We believe that and this proposition seems reasonable (without any mathematical proof) and Arthur did not give any indication on this question.

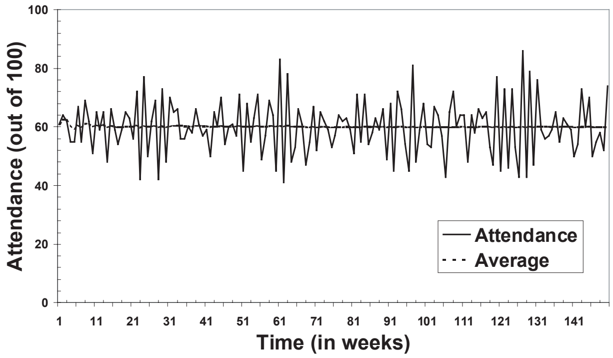

Figure 3: *Standard simulation of the El Farol Bar Problem. The attendance at the bar* (full line) *and the average attendance* (dashed line) *are plotted over 150 weeks. The simulation is done with* $N = 100$, $k = 6$, $S = 200$, $d = 20$ *and* $\lambda = 50\%$.

The average of the attendance (Fig. 3) is stable in time (60.34, 60.01 and 59.99 respectively on week 50, 100 and 150) and so is the standard deviation (7.61, 8.73 and 9.22). However, in order to smooth the output and to validate our computational tool, we re-launched the simulation 100 times. We found that the average of the averages tends toward the opening level (Fig. 4, top left), however lower 59.7073 at the 150^{th} week (< 60). We also notice differences in between the minimum and the maximum average (59.0611 and 60.2214 respectively). Further, we found that the average standard deviation is stable on average (Fig. 4, top right), at 8.8247 on the 20^{th} week and at 8.7546 on the 150^{th}. But the data exhibits some differences in between the minimum (6.8198) and the maximum (10, 8981) average standard deviation.

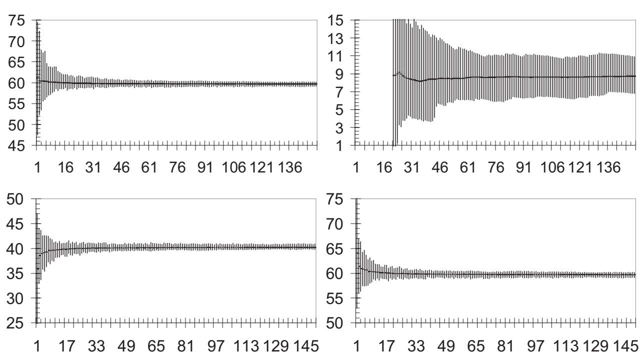

Figure 4: *Bar chart of the minimum, the maximum and the average of the average of the attendances* (top left) *and of the standard deviations* (top right). *Also for the active predictors expecting an attendance greater than the open level, and equal or lower* (respectively bottom left and right). *Simulations are done over 150 weeks for 100 simulations, with* $N = 100$, $k = 6$, $S = 200$, $d = 20$ *and* $\lambda = 50\%$.

Not only does the number of people who attend the bar fluctuate around the open-level attractor at an average level of l, close to 60% in our simulations, there is also a $(L, N-L)$ internal structure of people going and not going (Fig. 4, bottom left and right), which constitutes the well known Nash Equilibria (Arthur (1994), Zambrano (2004)), and which is selected through the computer simulations. We belive that it is a well diversified predictor portfolio that allows the bounded-rational agents to select this equilibrium strategy. This is the same solution as choosing randomly a number in between zero and N (the number of agents) and staying at home if it is strictly greater than L, and to stay home otherwise.

Our results confirm Arthur's observations concerning the attendance ($< A > \approx L$, with a $(L, N-L)$ internal structure), even if the average attendance we measured is slightly lower than the resource level. In fact Challet, Marsili, and Ottino (2004) find that piking randomly and uniformly from the total predictor space, the agents go to the bar with probability $L/(N+1)$ and thus the average attendance settles at that level (59.406 with $N = 100$ and $L = 60$). We consider therefore that we are in presence of a 'steady state' and we believe that our computer tool is workable. However adding a more complete strategy space to our setup, in the sense of Challet, Marsili, and Ottino (2004), could maybe reduce the noise of the output results.

6 Conclusions

We have described throughout this paper the implementation and the simulation in NetLogo of the El Farol Bar Problem. We discussed among others the modeling assumptions made in order to translate Arthur's model described in natural language into a workable computer tool. We then used it and we presented the results of the simulations in order to validate the model by comparison to other results available in the literature. We also discussed briefly the output data and the apparently random patterns even if the set up is completely deterministic.

We have also shown NetLogo's 'low threshold, no ceiling' strength (Tisue and Wilensky (2004b)). Low threshold because our implementation is short, simple and the results we found match Arthur's complex simulations. And no ceiling because the 'El Farol Bar Problem' in NetLogo can now be used for broader simulations; extending the behaviour of the agents (Edmonds (1999)), testing Darwinism on the players' portfolios or testing the state of their information (as in Challet (2000)) or using genetic algorithms (Fogel, Chellapilla, and Angeline (1999)) for example.

The tool we have discussed here is therefore a 'skeleton' for further developments and simulations. The next step in this work could be to investigate further the usage and the performance of the predictors or for example the adaptability of the EFBP metaphor to other domains.

Acknowledgment

I would like to thank Nathalie Sechi and Yves Tomballe for their precious comments. All the remaining mistakes are of course mine.

References

Arthur, B. (1994): "Inductive Resoning and Bounded Rationality," *American Economic Review*, 84(2), 406–411.

Axelrod, R. (1997): "Advacing the Art of Simulation in the Social Sciences," in *Simulating Social Phenomena*, ed. by R. Conte, R. Hegselmann, and P. Terna, pp. 21–40. Springer-Verlag, Berlin, Germany.

Axtell, R., R. Axelrod, J. M. Epstien, and M. D. Cohen (1996): "Aligning Simulation Models: A Case Study and Results," *Computational and Mathematical Organization Theory*, 1, 123–141.

Brandouy, O. (2003): "Stock Markets as Minority Games: Cognitive Heteroginity and Equilibrium Emergence," Discussion Paper 13/2003, CLAREE, Lille, France.

Casti, J. (1996): "Seeing the Light at El Farol," *Complexity*, 1(7), 7–10.

Challet, D. (2000): "Modelling markets dynamics: Minority games and beyond," Thèse de sciences, Université de Fribourg (Suisse).

Challet, D., M. Marsili, and G. Ottino (2004): "Shedding light on El Farol," *Physica A: Statistical Mechanics and its Applications*, 332(1), 469–482, preprint cond-mat/0306445.

Edmonds, B. (1999): "Gossip, Sexual Recombination and the El Farol bar: modelling the emergence of heterogeneity.," *J. Artificial Societies and Social Simulation*, 2(4).

Ferber, J. (1995): "Les systemes multi-agents, vers une intelligence collective," InterEditions, Paris.

Fogel, D. B., K. Chellapilla, and P. J. Angeline (1999): "Inductive Resoning and Bounded Rationality Reconsidered," *IEEE Transactions on Evolutionary Computation*, 3(2), 142–146.

Gilbert, G. N., and K. G. Troitzsch (1999): *Simulation for the Social Scientist*. Taylor & Francis, Inc., Bristol, PA, USA.

Hales, D., J. Rouchier, and B. Edmonds (2003): "Model-to-Model Analysis," *Journal of Artificial Societies and Social Simulation*, 6(4).

Kauffman, S. A. (1993): "The origins of order: self-organization and selection in evolution," Oxford University Press, New York.

Langton, C. (1989): "Artificial Life," In Langton, ed., Santa Fe Institute Studies in the Sciences of Complexity, pages 1-44, Addison Wesley, Redwood City.

LeMoigne, J.-L. (1990): "La modélisation des systèmes complexes," Dunod, Paris.

Parkes, D. C., and D. Steinig (1997): "The Santa Fe Bar Problem: Achieving Robust Coordination with Randomization and Simple Myopic Learning," Discussion Paper of University of Pennsylvenia, Philadelphia, USA.

Parunak, H. V. D., R. Savit, and R. L. Riolo (1998): "Agent-Based Modeling vs. Equation-Based Modeling: A Case Study and Users Guide," in *Proceedings of Multiagent systems and Agent-based Simulation (MABS'98), Springer*, pp. 10–25.

Reynolds, C. W. (1987): "Flocks, Herds, and Schools: a Distributed Behavioral Model," in *Computer Graphics*, pp. 25–34. SIGGRAPH '87 Conference Proceedings, 21(4), `http://www.red3d.com/cwr/boids/`.

Tisue, S., and U. Wilensky (2004a): "NetLogo: A Simple Environment for Modeling Complexity," in *International Conference on Complex Systems*, pp. 16–21.

——— (2004b): "NetLogo: Design and Implementation of a Multi-Agent Modeling Environment," in *SwarmFest 2004*, pp. 9–11.

Weiss, G. (2000): "Multiagent systems: A modern approach to distributed artificial intelligence," MIT Press.

Wilensky, U. (1999): "NetLogo: Center for Connected Learning and Computer-Based Modeling, Northwestern University. Evanston, IL," `http://ccl.northwestern.edu/netlogo/`.

Zambrano, E. (2004): "The Interplay Between Analytics and Computation in the Study of Congestion Externalities: The Case of the El Farol Problem," *Journal of Public Economic Theory*, 6(2).